



# **UNIVERSIDAD TÉCNICA DEL NORTE**

## **FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

### **TEMA:**

**SISTEMA ACADÉMICO WEB BILINGÜE KICHWA-ESPAÑOL PARA LAS UNIDADES EDUCATIVAS DE LAS COMUNIDADES DEL GAD MUNICIPAL DE OTAVALO UTILIZANDO LOS FRAMEWORKS CAKEPHP Y ZURB FOUNDATION. UNIDAD EDUCATIVA PILOTO: UNIDAD EDUCATIVA INTERCULTURAL BILINGÜE “MANUEL J. CALLE”.**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas  
Computacionales

Autor:

Cabascango Chiza Boonter Segundo

Director:

Msc. Daisy Elizabeth Imbaquingo Esparza

Ibarra, 2018



# UNIVERSIDAD TÉCNICA DEL NORTE

## BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información

<b>DATOS DE CONTACTO</b>	
<b>CÉDULA DE IDENTIDAD:</b>	1003309158
<b>APELLIDOS Y NOMBRES:</b>	CABASCANGO CHIZA BOONTER SEGUNDO
<b>DIRECCIÓN:</b>	OTAVALO
<b>EMAIL:</b>	booncabas@gmail.com bscabascangoc@utn.edu.ec
<b>CELULAR:</b>	0981503912

<b>DATOS DE LA OBRA</b>	
<b>TÍTULO:</b>	“SISTEMA ACADÉMICO WEB BILINGÜE KICHWA-ESPAÑOL PARA LAS UNIDADES EDUCATIVAS DE LAS COMUNIDADES DEL GAD MUNICIPAL DE OTAVALO UTILIZANDO LOS FRAMEWORKS CAKEPHP Y ZURB FOUNDATION. UNIDAD EDUCATIVA PILOTO: UNIDAD EDUCATIVA INTERCULTURAL BILINGÜE MANUEL J. CALLE”.
<b>AUTOR:</b>	CABASCANGO CHIZA BOONTER SEGUNDO
<b>FECHA:</b>	18/07/2018
<b>TÍTULO POR EL QUE OPTA:</b>	INGENIERÍA EN SISTEMAS COMPUTACIONALES
<b>DIRECTOR:</b>	MSC. DAISY IMBAQUINGO

## 2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 18 días del mes de julio de 2018

**EL AUTOR:**

.....  
Nombre: Boonter Cabascango



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

Ibarra, 18 de julio del 2018

**CERTIFICACIÓN DIRECTOR DE TESIS**

Por medio del presente yo Msc. Daisy Imbaquingo, certifico que el Sr. Boonter Segundo Cabascango Chiza portador de la cédula de identidad Nro. 100330915-8. Ha trabajado en el desarrollo del proyecto de tesis **SISTEMA ACADÉMICO WEB BILINGÜE KICHWA-ESPAÑOL PARA LAS UNIDADES EDUCATIVAS DE LAS COMUNIDADES DEL GAD MUNICIPAL DE OTAVALO UTILIZANDO LOS FRAMEWORKS CAKEPHP Y ZURB FOUNDATION. UNIDAD EDUCATIVA PILOTO: UNIDAD EDUCATIVA INTERCULTURAL BILINGÜE "MANUEL J. CALLE"**, previo a la obtención del título de ingeniería en sistemas computacionales, lo cual ha realizado en su totalidad.

Es todo cuanto puedo certificar en honor a la verdad.

---

Msc. Daisy Imbaquingo  
**DIRETOR DE TESIS**



GOBIERNO AUTÓNOMO  
DESCENTRALIZADO MUNICIPAL  
DEL CANTÓN OTAVALO

Otavaló, 23 de julio de 2018

## CERTIFICADO

Yo; Carla Echeverría en calidad de Directora del Departamento de Gestión Social Intercultural del Gobierno Autónomo Descentralizado Municipal del Cantón Otavalo CERTIFICO; que el sistema académico web bilingüe kichwa-español para ser utilizadas en las unidades educativas del cantón Otavalo desarrollado por el señor Boonter Segundo Cabascango Chiza, portador de la cédula de identidad 1003309158 estudiante del Universidad Técnica del Norte de la FICA del 10mo semestre está terminado y ha pasado las pruebas de implementación y está listo utilización para el año lectivo 2018-2019.

Atentamente:

Ing. Carla Echeverría M.

**DIRECTORA DE GESTIÓN SOCIAL INTERCULTURAL GADMCO**

**Dirección:** García Moreno # 505 / **Telf:** 06 2 920 - 460 / 06 2 924 - 566  
**Fax:** 06 2 920 - 404  
**OTAVALO - ECUADOR**



**UNIDAD EDUCATIVA INTERCULTURAL BILINGÜE  
MANUEL J. CALLE GUARDIANA DE LA LENGUA**

Eugenio Espejo, Calle José Puente y Panamericana Teléfono: (06) 2635070  
E-mail: cecibmanueljcalles@yahoo.com Página Web: www.ueib.manueljcalles.com  
Otavalo – Imbabura - Ecuador



MINISTERIO  
DE EDUCACIÓN

## CERTIFICADO

En mi calidad de rectora de la Unidad Educativa Intercultural Bilingüe "MANUEL J. CALLE" de la parroquia Eugenio Espejo, cantón Otavalo, provincia de Imbabura, luego de que los docentes realizaron las pruebas de funcionamiento, tengo a bien **CERTIFICAR QUE:**

El Sr. **BOONTER SEGUNDO CABASCANGO CHIZA** con cédula 1003309158, ha terminado el desarrollo del sistema académico Web Bilingüe Kichwa- Español adaptativo a dispositivos móviles, mismo que contribuirá en el mantenimiento actualizado del archivo de información, envío y recepción de documentos y comunicados, la generación de reportes de nuestra institución educativa.

Es cuanto puedo certificar en honor a la verdad, dando facultad al interesado hacer uso de la presente en lo que estime conveniente.

Eugenio Espejo a 03 de julio del 2018

Atentamente:

Lcda. Soraya Chicaiza

RECTORA -E



## **Dedicatoria**

A Dios que es la razón y la alegría para vivir y a mis seres amados que son una bendición en mi vida.

Boonter Cabascango

## **Agradecimientos**

A mi directora de tesis Msc. Daisy Imbaquingo, a mi asesora Msc. Silvia Arciniega, al personal del Municipio de Otavalo y de la unidad educativa “Manuel J. Calle” que colaboraron en el desarrollo de este proyecto, a mis docentes y a mis compañeros.

Boonter Cabascango



## Tabla de contenido

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD .....	ii
Dedicatoria .....	vii
Agradecimientos.....	viii
Índice de figuras .....	xiv
Índice de tablas.....	xvii
RESUMEN.....	xx
ABSTRACT .....	xxi
INTRODUCCIÓN .....	1
Antecedentes .....	1
Prospectiva.....	2
Definición del problema.....	2
Objetivo general .....	3
Objetivos específicos.....	3
Alcance .....	4
Justificación.....	4
CAPÍTULO 1 .....	7
MARCO TEÓRICO .....	7
1.1 Metodología de desarrollo de software XP.....	7
1.1.1 Definición .....	7
1.1.2 Historia y evolución.....	7
1.1.3 Valores de XP .....	8
1.1.4 Prácticas técnicas de XP .....	8
1.1.5 Roles XP .....	10
1.2 Historia y relevancia del idioma Kichwa.....	11
1.2.1 Comunicación entre indígenas y el origen del kichwa .....	11
1.2.2 Diferencia entre quechua y kichwa (quichua).....	12
1.2.3 Pérdida de la lengua kichwa.....	13
1.2.4 El kichwa unificado.....	13
1.2.5 El uso de las TIC en la preservación de las lenguas originarias .....	14
1.3 Framework CakePHP .....	15
1.3.1 Definición .....	15
1.3.2 Características.....	16
1.3.3 Ciclo de una petición CakePHP .....	16
1.4 Framework ZURB Foundation.....	17
1.4.1 Definición .....	17

1.4.2 Características.....	18
1.5 Lenguaje de programación PHP .....	19
1.5.1 Historia y evolución.....	19
1.5.2 Características.....	20
1.6 Base de Datos MySQL .....	21
1.6.1 Definición de base de datos .....	21
1.6.2 Sistema de Base de Datos(SBD) .....	21
1.6.3 Historia y evolución de MySQL.....	21
1.6.4 Características.....	22
1.7 Patrón de arquitectura de software MVC.....	22
1.7.1 Historia y evolución.....	22
1.7.2 MVC en aplicaciones PHP .....	23
1.8 Servidor Apache .....	24
1.8.1 Definición.....	24
1.8.2 Características.....	24
1.9 Instructivo para la aplicación de la evaluación estudiantil.....	25
1.9.1 Niveles y subniveles educativos .....	25
1.9.2 Evaluación en el nivel de Educación Inicial y el subnivel de Preparatoria de EGB... 26	
1.9.3 Calificación de los aprendizajes.....	26
1.9.4 Exámenes.....	27
1.9.5 Procedimientos institucionales de la calificación de los aprendizajes .....	28
y del refuerzo académico.....	28
1.9.6 Evaluación del comportamiento.....	28
1.9.7 Evaluación de proyectos educativos.....	29
1.10 Herramientas de desarrollo .....	30
1.10.1 NetBeans IDE 8.2 .....	30
1.10.2 Xampp 5.6.12 .....	30
1.10.3 MySQL Workbench 6.1.7.....	30
1.10.4 FPDF.....	30
CAPITULO 2.....	32
DESARROLLO DE LA SOLUCIÓN.....	32
2.1 Arquitectura de la solución.....	32
2.2 Planificación .....	33
2.2.1 Personal involucrado .....	33
2.2.2 Historias de usuario .....	34
2.2.3 Características de los usuarios.....	41

2.2.4 Restricciones.....	41
2.2.5 Requisitos Funcionales .....	42
2.2.6 Requisitos no funcionales .....	44
2.3 Parámetros configurables para el sistema multi institución.....	46
2.3.1 Información de presentación de la institución.....	47
2.3.2 Logos.....	47
2.3.3 Tipos de calificaciones cualitativas de asignaturas y sus equivalencias.....	48
2.3.4 Rangos y equivalencias de calificaciones cuantitativas de asignaturas.....	49
2.3.5 Tipos de calificaciones cualitativas de proyectos educativos y sus equivalencias....	52
2.3.6 Tipos de calificaciones cualitativas de comportamiento y sus equivalencias.....	53
2.3.7 Parámetros para evaluar en proyectos educativos.....	55
2.3.8 Parámetros para evaluar en comportamiento .....	55
2.3.9 Parámetros para evaluar en asistencia .....	56
2.3.10 Nombres de ítems académicos .....	56
2.3.11 Reportes de matrícula .....	58
2.3.12 Reportes de promoción.....	58
2.3.13 Reportes de calificaciones de estudiantes.....	59
2.3.14 Reportes de resúmenes de calificaciones de docentes .....	60
2.4 Diseño .....	61
2.4.1 Modelo Conceptual .....	61
2.4.2 Diagrama de clases .....	62
2.4.3 Modelo Entidad Relación .....	63
2.5 Casos de uso .....	65
2.5.1 Caso de uso administrador.....	65
2.5.2 Caso de uso docente.....	66
2.5.3 Caso de uso estudiante.....	67
2.5.4 Caso de uso representante .....	68
2.6 Integración de Zurb Foundation con CakePHP .....	69
2.6.1 Estructura del proyecto .....	69
2.6.2 Propuesta de diseño adaptativo .....	71
2.7 Propuesta de desarrollo bilingüe.....	72
2.8 Plan de desarrollo del proyecto.....	74
2.9 Primera iteración.....	74
2.9.1 Historia de usuario – Registrar usuarios .....	75
2.9.2 Historia de usuario - Ingreso al sistema .....	76
2.9.3 Historia de usuario - Perfil de usuario.....	77

2.9.4 Historia de usuario - Acceso a la información .....	79
2.9.5 Historia de usuario – Registrar docentes .....	80
2.9.6 Historia de usuario – Registrar estudiantes .....	81
2.9.7 Historia de usuario - Registrar representantes.....	82
2.9.8 Historia de usuario - Registrar administradores.....	83
2.9.9 Historia de usuario - Registrar cursos.....	84
2.9.10 Historia de usuario - Registrar asignaturas .....	85
2.9.11 Pruebas de aceptación .....	86
2.10 Segunda iteración .....	90
2.10.1 Historia de usuario – Matricular estudiante.....	91
2.10.2 Historia de usuario – Determinar asignaturas de un curso .....	92
2.10.3 Historia de usuario – Asignar carga horaria a docentes.....	93
2.10.4 Historia de usuario – Asignar tutores a cursos.....	95
2.10.5 Historia de usuario – Registrar notas parciales .....	96
2.10.6 Historia de usuario – Registrar notas de exámenes.....	97
2.10.7 Historia de usuario – Registrar asistencias.....	98
2.10.8 Historia de usuario – Registrar comportamientos.....	99
2.10.9 Historia de usuario – Registrar proyectos educativos.....	100
2.10.10 Pruebas de aceptación .....	101
2.11 Tercera iteración.....	105
2.11.1 Historia de usuario – Administración de documentos .....	105
2.11.2 Historia de usuario – Publicación de notificaciones .....	107
2.11.3 Historia de usuario – Generación de reportes.....	108
2.11.4 Pruebas de aceptación .....	110
CAPÍTULO 3 .....	113
RESULTADOS .....	113
3.1 Instalación.....	113
3.1.1 Requerimientos.....	113
3.1.2 Alojamiento.....	113
3.1.3 Conexión con la base de datos.....	114
3.2 Despliegue.....	115
3.2.1 Funciones del sistema para el tipo de usuario administrador .....	116
3.2.2 Funciones del sistema para el tipo de usuario docente .....	119
3.2.3 Funciones del sistema para el tipo de usuario estudiante.....	122
3.2.4 Funciones del sistema para el tipo de usuario representante.....	124
CONCLUSIONES.....	127

RECOMENDACIONES .....	128
Referencias .....	129

## Índice de figuras

Figura 1: Valores y prácticas de XP .....	10
Figura 2. Ciclo de una petición CakePHP (Fuente: CakePHP 2017).....	17
Figura 3: Funcionamiento de MVC (Fuente: <a href="https://helloacm.com/model-view-controller-explained-in-c/">https://helloacm.com/model-view-controller-explained-in-c/</a> ) .....	23
Figura 4: Patrón MVC en aplicaciones web (Fuente: Sánchez 2012).....	24
Figura 5: Arquitectura de la solución .....	33
Figura 6: Modelo conceptual .....	62
Figura 7: Diagrama de clases .....	63
Figura 8: Modelo Entidad Relación .....	64
Figura 9: Caso de uso general para el tipo de usuario “Administrador” .....	65
Figura 10: Caso de uso general para el tipo de usuario “Docente” .....	66
Figura 11: Caso de uso general para el tipo de usuario “Estudiante” .....	67
Figura 12: Caso de uso general para el tipo de usuario “Representante” .....	68
Figura 13: Estructura del proyecto con CakePHP y Foundation.....	70
Figura 14: Referencia de Foundation en un archivo plantilla de CakePHP.....	70
Figura 15: Propuesta de diseño anti desborde horizontal.....	71
Figura 16: Ingreso de notas en dispositivo móvil pequeño .....	71
Figura 17: Ingreso de notas en dispositivo móvil mediano .....	72
Figura 18: Ingreso de notas en pantalla de PC .....	72
Figura 19: Formulario en kichwa-español.....	72
Figura 20: Botón en kichwa-español .....	73
Figura 21: Barra de navegación en kichwa-español.....	73
Figura 22: Menú en kichwa-español.....	73
Figura 23: Encabezado de tablas en kichwa-español.....	73
Figura 24: Opciones en kichwa-español.....	73
Figura 25: Mensaje de confirmación en kichwa-español .....	74
Figura 26: Interfaz de administración de usuarios .....	75
Figura 27: Interfaz de acceso al sistema.....	77
Figura 28: Interfaz de tipo de usuario Administrador .....	78
Figura 29: Interfaz de usuario Docente .....	78
Figura 30: Interfaz de usuario Estudiante.....	79
Figura 31: Interfaz de usuario Representante .....	79
Figura 32: Interfaz principal de administración de docentes .....	80
Figura 33: Interfaz principal de administración de estudiantes .....	81
Figura 34: Interfaz principal de administración de representantes.....	82
Figura 35: Interfaz principal de administración de administradores .....	83

Figura 36: Interfaz principal de administración de cursos.....	84
Figura 37: Interfaz principal de administración de asignaturas .....	86
Figura 38: Interfaz de registrar matrículas.....	91
Figura 39: Interfaz de asignación de asignaturas a curso .....	93
Figura 40: Interfaz de asignación de carga horaria a docentes .....	94
Figura 41: Interfaz de asignación de tutores a cursos .....	95
Figura 42: Interfaz el ingreso de notas parciales .....	96
Figura 43: Interfaz de ingreso de notas de exámenes.....	97
Figura 44: Interfaz de ingreso de asistencias .....	98
Figura 45: Interfaz de ingreso de comportamientos .....	99
Figura 46: Interfaz de ingreso de proyectos educativos .....	100
Figura 47: Interfaz principal de administración de documentos .....	105
Figura 48: Interfaz para subir documentos.....	106
Figura 49: Interfaz principal para administración de comunicados .....	107
Figura 50: Instalación de setassign/fpdf.....	108
Figura 51: Estructura de setsign/fpdf en el proyecto .....	108
Figura 52: Llamado y utilización de fpdf .....	109
Figura 53: Estructura de dakota/cake-excel en el proyecto .....	110
Figura 54: Alojamiento del Proyecto (Localhost y host remoto).....	114
Figura 55: Conexión con la base de datos .....	115
Figura 56: Administración de unidades educativas .....	115
Figura 57: Opciones del usuario Administrador (Vista computador y dispositivo móvil).....	116
Figura 58: Opciones de la función Información (vista computador y dispositivo móvil).....	116
Figura 59: Opciones de la función Asignaciones (Vista computador y dispositivo móvil)...	116
Figura 60: Opciones de la función Reportes de docente (Vista computador y dispositivo móvil) .....	117
Figura 61: Opciones de la función Reportes de tutor (Vista computador y dispositivo móvil) .....	117
Figura 62: Opciones de la función Reportes de estudiante (Vista computador y dispositivo móvil) .....	118
Figura 63: Opciones de la función Calificaciones (Vista computador y dispositivo móvil) ..	118
Figura 64: Opciones de la función Configuración (Vista computador y dispositivo móvil) ..	119
Figura 65: Opciones del usuario Docente (Vista computador y dispositivo móvil) .....	119
Figura 66: Opciones de la función Calificaciones (Vista computador y dispositivo móvil) ..	119
Figura 67: Opciones de la función Tutor (Vista computador y dispositivo móvil) .....	120
Figura 68: Opciones de la función Reportes de docente (Vista computador y dispositivo móvil) .....	120

Figura 69: Opciones de la función Reportes de tutor (Vista computador y dispositivo móvil)	121
Figura 70: Opciones de la función Reportes de estudiante (Vista computador).....	121
Figura 71: Opciones de la función Documentos (Vista computador y dispositivo móvil) ....	121
Figura 72: Opciones de la función Comunicados (Vista computador y dispositivo móvil) ..	122
Figura 73: Opciones del usuario Estudiante (Vista computador y dispositivo móvil).....	122
Figura 74: Opciones de la función Calificaciones (Vista computador y dispositivo móvil) ..	122
Figura 75: Opciones de la función Docentes (Vista computador y dispositivo móvil).....	123
Figura 76: Opciones de la función Datos personales (Vista computador y dispositivo móvil)	123
Figura 77: Opciones de la función Representante (Vista computador y dispositivo móvil)	123
Figura 78: Opciones del usuario Representante (Vista computador y dispositivo móvil) ...	124
Figura 79: Opciones de la función Calificaciones (Vista computador y dispositivo móvil) ..	124
Figura 80: Opciones de la función Docentes (Vista computador y dispositivo móvil).....	124
Figura 81: Opciones de la función Datos personales (Vista computador y dispositivo móvil)	125
Figura 82: Opciones de la función Estudiantes (Vista computador y dispositivo móvil) .....	125



## Índice de tablas

TABLA 1.1: DIFERENCIAS ENTRE QUECHUA Y KICHWA.....	12
TABLA 1.2: HISTORIA DEL ALFABETO KICHWA .....	14
TABLA 1.3: CARACTERÍSTICAS DEL FRAMEWORK CAKEPHP .....	16
TABLA 1.4: CARACTERÍSTICAS DE ZURB FOUNDATION.....	18
TABLA 1.5: CARACTERÍSTICAS DE PHP .....	20
TABLA 1.6: CARACTERÍSTICAS DE MySQL.....	22
TABLA 1.7: COMPARACIÓN ENTRE EL SNE Y EL SEIB.....	26
TABLA 1.8: ESCALA DE EVALUACIÓN PARA EDUCACIÓN INICIAL Y PREPARATORIA.....	26
TABLA 1.9: ESCALA DE CALIFICACIONES .....	26
TABLA 1.10: CÁLCULO DE NOTA PARCIAL CON REFUERZO ACADÉMICO.....	28
TABLA 1.11: ESCALA DE EVALUACIÓN DEL COMPORTAMIENTO .....	29
TABLA 1.12: ESCALA CUALITATIVA PARA EVALUAR LOS PROYECTOS EDUCATIVOS .....	29
TABLA 2.1: EQUIPO DE DESARROLLO DEL PROYECTO .....	34
TABLA 2.2: HISTORIA DE USUARIO – REGISTRAR USUARIOS.....	34
TABLA 2.3: HISTORIA DE USUARIO - INGRESO AL SISTEMA.....	35
TABLA 2.4: HISTORIA DE USUARIO – PERFIL DE USUARIO.....	35
TABLA 2.5: HISTORIA DE USUARIO – ACCESO A LA INFORMACIÓN .....	35
TABLA 2.6: HISTORIA DE USUARIO - REGISTRAR DOCENTES.....	36
TABLA 2.7: HISTORIA DE USUARIO - REGISTRAR ESTUDIANTES .....	36
TABLA 2.8: HISTORIA DE USUARIO - REGISTRAR REPRESENTANTES.....	36
TABLA 2.9: HISTORIA DE USUARIO - REGISTRAR ADMINISTRADORES .....	37
TABLA 2.10: HISTORIA DE USUARIO – REGISTRAR CURSOS .....	37
TABLA 2.11: HISTORIA DE USUARIO – REGISTRAR ASIGNATURAS .....	37
TABLA 2.12: HISTORIA DE USUARIO - MATRICULAR ESTUDIANTE.....	37
TABLA 2.13: HISTORIA DE USUARIO – ASIGNAR ASIGNATURAS A CURSOS .....	38
TABLA 2.14: HISTORIA DE USUARIO – ASIGNAR CARGA HORARIA A DOCENTES.....	38
TABLA 2.15: HISTORIA DE USUARIO – ASIGNAR TUTORES A CURSOS.....	38
TABLA 2.16: HISTORIA DE USUARIO - REGISTRAR NOTAS PARCIALES .....	38
TABLA 2.17: HISTORIA DE USUARIO - REGISTRAR NOTAS DE EXÁMENES.....	39
TABLA 2.18: HISTORIA DE USUARIO - REGISTRAR ASISTENCIA A CLASES .....	39
TABLA 2.19: HISTORIA DE USUARIO - REGISTRAR COMPORTAMIENTO .....	39
TABLA 2.20: HISTORIA DE USUARIO - REGISTRAR NOTAS DE PROYECTOS EDUCATIVOS.....	39
TABLA 2.21: HISTORIA DE USUARIO – ADMINISTRACIÓN DE DOCUMENTOS .....	40
TABLA 2.22: HISTORIA DE USUARIO - PUBLICACIÓN DE NOTIFICACIONES .....	40

TABLA 2.23: HISTORIA DE USUARIO - GENERACIÓN DE REPORTE	40
TABLA 2.24: CARACTERÍSTICAS DE LOS USUARIOS	41
TABLA 2.25: REQUISITOS FUNCIONALES: APLICACIÓN	42
TABLA 2.26: REQUISITOS FUNCIONALES: ENTRADA DE DATOS	43
TABLA 2.27: REQUISITOS NO FUNCIONALES - ARQUITECTURA	44
TABLA 2.28: REQUISITOS NO FUNCIONALES: ARQUITECTURA	44
TABLA 2.29: REQUISITOS NO FUNCIONALES: SEGURIDAD	45
TABLA 2.30: REQUISITOS NO FUNCIONALES: MANTENIBILIDAD	45
TABLA 2.31: REQUISITOS NO FUNCIONALES: PORTABILIDAD	46
TABLA 2.32: REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE ASIGNATURAS	49
TABLA 2.33: REGISTRO CONFIGURABLE DE CALIFICACIONES CUANTITATIVAS DE ASIGNATURAS	51
TABLA 2.34: REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE PROYECTOS EDUCATIVOS	53
TABLA 2.35: REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE COMPORTAMIENTO	54
TABLA 2.36: PLAN DE DESARROLLO DEL PROYECTO	74
TABLA 2.37: PLAN DE DESARROLLO DE LA PRIMERA ITERACIÓN	75
TABLA 2.38: CASO DE PRUEBA – REGISTRAR USUARIOS	87
TABLA 2.39: CASO DE PRUEBA – INGRESO AL SISTEMA	87
TABLA 2.40: CASO DE PRUEBA – PERFIL DE USUARIO	87
TABLA 2.41: CASO DE PRUEBA – ACCESO A LA INFORMACIÓN	88
TABLA 2.42: CASO DE PRUEBA – REGISTRAR DOCENTES	88
TABLA 2.43: CASO DE PRUEBA – REGISTRAR ESTUDIANTES	88
TABLA 2.44: CASO DE PRUEBA – REGISTRAR REPRESENTANTES	89
TABLA 2.45: CASO DE PRUEBA – REGISTRAR ADMINISTRADORES	89
TABLA 2.46: CASO DE PRUEBA – REGISTRAR CURSOS	90
TABLA 2.47: CASO DE PRUEBA – REGISTRAR ASIGNATURAS	90
TABLA 2.48: PLAN DE DESARROLLO DE LA SEGUNDA ITERACIÓN	91
TABLA 2.49: CASO DE PRUEBA – MATRICULAR ESTUDIANTE	101
TABLA 2.50: CASO DE PRUEBA – ASIGNAR ASIGNATURAS A CURSOS	102
TABLA 2.51: CASO DE PRUEBA – ASIGNAR CARGA HORARIA A DOCENTES	102
TABLA 2.52: CASO DE PRUEBA – ASIGNAR TUTORES A CURSOS	102
TABLA 2.53: CASO DE PRUEBA – REGISTRAR NOTAS PARCIALES	103
TABLA 2.54: CASO DE PRUEBA – REGISTRAR NOTAS DE EXAMENES	103
TABLA 2.55: CASO DE PRUEBA – REGISTRAR ASISTENCIAS	104

TABLA 2.56: CASO DE PRUEBA – REGISTRAR COMPORTAMIENTOS .....	104
TABLA 2.57: CASO DE PRUEBA – REGISTRAR PROYECTOS EDUCATIVOS.....	104
TABLA 2.58: PLAN DE DESARROLLO DE LA TERCERA ITERACIÓN .....	105
TABLA 2.59: CASO DE PRUEBA – ADMINISTRACIÓN DE DOCUMENTOS .....	111
TABLA 2.60: CASO DE PRUEBA – PUBLICACIÓN DE NOTIFICACIONES .....	111
TABLA 2.61: CASO DE PRUEBA – GENERACIÓN DE REPORTES .....	112
TABLA 3.1: REQUERIMIENTOS DE HARDWARE Y SOFTWARE.....	113

## RESUMEN

El idioma kichwa es uno de los lenguajes de relación intercultural del Estado Ecuatoriano y forma parte importante de la identidad y de la riqueza cultural de los habitantes del cantón Otavalo. El uso del idioma kichwa en el desarrollo de software es mínimo y el incremento del uso de aplicaciones web en la vida cotidiana brinda la oportunidad de fortalecer el uso del kichwa al desarrollar aplicaciones que incluyan este lenguaje. El sistema académico web bilingüe kichwa-español aporta a la preservación de esta lengua ancestral del pueblo indígena otavaleño.

El ciclo de vida del sistema fue definido por la metodología ágil de desarrollo de software “Programación Extrema” que hace énfasis en entregas cortas y trabajo colaborativo basado en historias de usuario que reemplazan a los tradicionales casos de uso con el fin de reducir el tiempo dedicado a la elaboración de documentación y dedicarlo a la programación.

El software libre utilizado para el desarrollo del sistema académico web bilingüe es: el lenguaje de programación PHP, una base de datos MySQL, el servidor de aplicaciones Apache, el framework CakePHP para obtener un sistema modular basado en el patrón de arquitectura modelo-vista-controlador y el framework ZURB Foundation para diseñar interfaces adaptables a dispositivos móviles. Con el propósito de obtener un software académico multiinstitucional se tomó como base de desarrollo que el sistema pueda ser configurado a las necesidades de cada institución y que cumpla con las normativas del ministerio de educación para la aplicación de la evaluación estudiantil, contemplando la adaptación a futuros cambios en sus normativas.

**PALABRAS CLAVES:** Kichwa, bilingüe, sistema académico, programación extrema, CakePHP, Foundation, adaptable.

## ABSTRACT

Kichwa language is one of the idioms of intercultural relationship from Ecuadorian State and it is an important part of the identity and cultural wealth of inhabitants from Otavalo canton. The use of the Kichwa language in the software development is minimal and the increase of the use of web applications in daily life, provides the opportunity to strengthen the use of Kichwa in developing applications that include this language. The Kichwa-Spanish bilingual web academic system contributes to the preservation of this ancestral tongue of the indigenous people from Otavalo.

The life cycle of this program, was defined by the agile software methodology development "Extreme Programming" that emphasizes short deliveries and collaborative work based on user stories that replace the use of traditional cases in order to reduce the spent time during elaboration of documentation and dedicate it to programming.

The free softwares used for the development of the bilingual web academic system are: the PHP programming language, a MySQL database, the Apache application server, the CakePHP framework to get a modular system based on the model-view-controller architecture pattern and the ZURB Foundation framework to design adaptive interfaces to mobile devices. In order to get a multi-institutional academic software, it was taken as a development base where the system can be adapted to the each institution's needs and that it also needs to fulfill the regulations of the Ministry of Education to the application of the student evaluation, contemplating the adaptation to future changes in its regulations.

**KEY WORDS:** Kichwa, bilingual, academic system, extreme programming, CakePHP, Foundation, responsive.

# INTRODUCCIÓN

## Antecedentes

Las aplicaciones web se han convertido en herramientas que las personas usan día a día. Cuando una persona ingresa a internet, la mayoría de las veces es una aplicación web que se está ejecutando. Puede considerarse una aplicación web, a los recursos o herramientas que son accesibles a través de un navegador web utilizando una red intranet o de Internet. (Ramos & Ramos, 2014)

El uso del idioma kichwa en el desarrollo de aplicaciones web es una iniciativa encaminada al refuerzo del uso de la lengua materna de los pueblos indígenas kichwahablantes. El aumento del uso de las tecnologías de la información y comunicación (TIC), ha dado lugar a la inclusión de los pueblos y nacionalidades en el desarrollo de estas tecnologías. Un buen ejemplo de este tipo de desarrollo es la página oficial de la Asamblea Nacional del Ecuador<sup>1</sup>, que está disponible en español y kichwa.

Los sistemas informáticos de entorno web en las instituciones de educación básica se han convertido en herramientas indispensables para administrar la información que es vital para el correcto funcionamiento de la institución. El beneficio de alojar la información en un lugar que brinde mejores opciones de seguridad que en una máquina de escritorio, la posibilidad de acceder al sistema desde un lugar remoto a través desde cualquier dispositivo que tenga un navegador web con acceso a Internet, el ahorro de tiempo en el seguimiento académico de los alumnos, la facilidad de generar reportes para entrega de notas de manera automática, entre otras funciones, demuestran la importancia de los sistemas académicos web en estas instituciones de educación.

El sistema académico de gobierno que utilizan las unidades educativas públicas para el ingreso de notas y generación de reportes presenta falencias tanto en disponibilidad como en usabilidad, lo que produce retrasos en la entrega de notas y pérdida de información. Además, el sistema no permite mantener un registro histórico de las calificaciones porque los docentes o autoridades no pueden obtener información de años lectivos anteriores. La mayoría de unidades educativas optan por adquirir un sistema propio para el seguimiento académico de los estudiantes y la generación de reportes de calificaciones. En las instituciones que no cuentan con los recursos económicos para la adquisición de un sistema hecho a la medida,

---

<sup>1</sup> URL: <http://www.asambleanacional.gob.ec>

los docentes llevan sus registros de calificaciones de forma manual en agendas de papel y para la generación de reportes necesitan usar plantillas de Excel poco funcionales y amigables. Además, los tutores encargados de cada curso deben solicitar los registros de calificaciones de otros docentes de manera personal, lo que complica sus tareas.

En el cantón Otavalo de la provincia de Imbabura, se encuentran varias comunidades de población indígena Kichwa-Otavalo, reconocidos nacional e internacionalmente por su habilidad comerciante y artesanal, pero, la influencia de la cultura mestiza ha dado como resultado la pérdida de identidad y de su lengua materna. El Gobierno Autónomo Descentralizado (GAD) del cantón Otavalo, contribuye al desarrollo de las comunidades y al rescate de su cultura, su identidad y su idioma. A través de este proyecto de desarrollo de un sistema académico bilingüe, busca el refuerzo del uso del idioma kichwa.

### **Prospectiva**

El sistema académico web bilingüe kichwa-español es la solución para el seguimiento académico de los estudiantes y para el manejo de la información de forma segura y ágil, facilitando así el trabajo de los administradores y docentes de las unidades educativas, y fortaleciendo el uso del idioma kichwa como lengua ancestral de los habitantes de las comunidades del cantón Otavalo.

La integración de un framework de back-end (CakePHP) y un framework de front-end (Zurb Foundation) en el desarrollo del sistema web facilita la construcción de una aplicación modular, estandarizada, más segura y con un mejor desempeño adaptativo de manera rápida y con menor esfuerzo.

### **Definición del problema**

Desde tiempos de la colonia, se produjo una marcada separación de los pueblos indígenas y hasta el día de hoy persisten muchas formas de discriminación, racismo e inequidad social. El Gobierno Nacional a través del Plan Nacional para el Buen Vivir<sup>2</sup> contribuye para que la sociedad ecuatoriana inicie un proceso de cambio que dé lugar a una sociedad intercultural y un estado plurinacional. Gracias a estos esfuerzos las nacionalidades y pueblos del Ecuador han realizado un retorno protagónico que ha marcado una nueva etapa en la historia. El sistema de Educación Intercultural Bilingüe es una muestra del avance de las políticas

---

<sup>2</sup> Plan presentado por el gobierno nacional desde el año 2009 basado en el “Sumak Kawsay” que tiene la visión centrada en el ser humano.

sociales y educativas que promueven una educación de calidad en los idiomas propios de los pueblos indígenas.

Es notable que el acceso a la tecnología e internet ha llegado a todos los sectores facilitando la comunicación y el desarrollo de actividades laborales y educativas, pero hay que resaltar también la falta de las TIC en los lenguajes oficiales de relación intercultural. En el cantón Otavalo, con el propósito de influir en la integración efectiva de las TIC en el desarrollo de los pueblos indígenas y en la preservación de su lengua materna, se ha llevado a cabo varios proyectos de enseñanza del idioma kichwa y de su traducción al español, pero se nota la falta de estas tecnologías bilingües en sistemas informáticos para restablecer y revitalizar esta lengua original.

La falta de un sistema académico estandarizado a los requerimientos del Ministerio de Educación, que incluya su lengua original, aumenta la brecha digital o tecnológica que tienen los docentes, estudiantes y sus representantes en las unidades educativas de las comunidades indígenas del cantón Otavalo.

### **Objetivo general**

Desarrollar un sistema académico web bilingüe kichwa-español para las unidades educativas de las comunidades del GAD Municipal de Otavalo utilizando los frameworks CakePHP y ZURB Foundation.

### **Objetivos específicos**

- Investigar la metodología Extreme Programming (XP), los framework CakePHP y ZURB Foundation, las herramientas de desarrollo, la arquitectura e ingeniería de sistemas, el instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación y la traducción del idioma español a kichwa del Ecuador.
- Aplicar la metodología XP al diseño, arquitectura e ingeniería de software de la solución informática propuesta.
- Desarrollar el sistema de acuerdo con las normativas de evaluación estudiantil del Ministerio de Educación del Ecuador en español y kichwa.
- Desarrollar funciones del sistema que permitan la personalización del mismo a las necesidades de cualquier unidad educativa y a los cambios en las normativas de evaluación.



## **Alcance**

El desarrollo del sistema académico web bilingüe kichwa-español para las unidades educativas de las comunidades del GAD Municipal de Otavalo, se lo realizará utilizando los frameworks CakePHP y ZURB Foundation, y será implementado en los servidores del GAD municipal de este cantón.

El sistema se realizará con el patrón de arquitectura modelo-vista-controlador (MVC) que separa el código de funcionamiento de una aplicación y la interfaz gráfica. Las fases del desarrollo serán definidas por la metodología de programación XP y los módulos con los que contará el sistema son los siguientes: usuarios, docentes, estudiantes, matrículas, cursos, tutores, notas, asistencias, comportamientos, monitoreo de ingreso de notas y publicación de notificaciones. Todas las vistas del sistema y los reportes que generará estarán en kichwa y español.

Los docentes, estudiantes y representantes accederán al sistema académico bilingüe mediante un navegador web estándar con conexión a Internet.

## **Justificación**

Ecuador es el escenario de una gran diversidad étnica y cultural. Pero este hecho no ha sido reconocido y en parte se lo ha ocultado, sin que se diera un serio esfuerzo por conocer a los pueblos indígenas, por apreciar sus valores, sus tradiciones y costumbres. En los últimos tiempos las cosas han ido cambiando y los pueblos indígenas se han convertido en actores importantes en el desarrollo de un país diverso. Para que la aceptación de la interculturalidad vaya más allá que simples palabras, se deben realizar los cambios que permitan una relación de equidad entre los grupos que lo componen. (Ayala, 2014)

La Constitución 2008 del Ecuador en su artículo primero hace referencia a la diversidad de la nación y a un estado intercultural y plurinacional, y en su segundo artículo resalta el idioma kichwa junto con el castellano y el shuar como idiomas oficiales de relación intercultural, que el estado respetará, y estimulará su conservación y su uso. (Constitución, 2008) El Gobierno Autónomo Descentralizado del cantón Otavalo acorde con las políticas de estado, promueve y fortalece el uso del idioma kichwa como uno de sus objetivos fundamentales a través de la Dirección de Gestión Social Intercultural y una manera efectiva de alcanzar este objetivo es a través del uso de la tecnología.

El desarrollo del sistema académico bilingüe kichwa-español nace ante la falta de un sistema que cumpla con las regulaciones establecidas por el Ministerio de Educación para la aplicación de la evaluación estudiantil en la Unidad Educativa intercultural bilingüe” Manuel J. Calle” de la parroquia Eugenio Espejo. Un software que cubra estas necesidades es fundamental para que los docentes y autoridades de la institución automaticen las tareas de seguimiento estudiantil.



# CAPÍTULO 1

## MARCO TEÓRICO

### 1.1 Metodología de desarrollo de software XP

#### 1.1.1 Definición

XP o Programación Extrema, es una metodología ágil de desarrollo de software diseñada para proyectos de tamaño pequeño o mediano que requiere la participación frecuente entre desarrolladores y el cliente. XP detalla las prácticas y valores que serán usados en el desarrollo del software. (Laínez, 2015).

#### 1.1.2 Historia y evolución

Desde mediados de los 90s, ya se consideraba importante el conjunto de prácticas de desarrollo de software, pero es el 6 de marzo de 1996, que, impulsado por Kent Beck, Ron Jeffries y Ward Cunningham, adopta el nombre de método, proceso o metodología de desarrollo. Desde entonces, ha demostrado su efectividad en compañías de diferentes tamaños en el mundo entero. En un principio, la metodología XP causó un poco de polémica en el mercado, pero luego de la aparición de la metodología Scrum en 2007 que se centra más en la gestión de proyectos, XP ha tenido gran aceptación porque se centra más en las prácticas de ingeniería. Inclusive se reconoce los beneficios de XP junto con Scrum porque un software no puede existir solo con prácticas de gestión sin la parte técnica. (Laínez, 2015)

Una de las principales razones por las que XP ha tenido mucho éxito, es su enfoque en la satisfacción del cliente, puesto que, en lugar de ofrecer un gran producto en una fecha lejana, ofrece el software a medida que lo necesita. Y es esta característica la que permite a los desarrolladores de software cumplir con los requisitos cambiantes de los clientes, aun cuando el ciclo de vida está en su etapa final.

La programación extrema hace énfasis en un equipo de trabajo colaborativo, que incluye a gerentes, clientes y desarrolladores como socios iguales, propone un entorno de trabajo simple pero eficaz y un equipo organizado para buscar la mejor solución posible a un problema lo que permite ser altamente productivo. (Wells, 2013)

### 1.1.3 Valores de XP

La metodología XP se apoya en cinco valores fundamentales que enfatizan el trabajo colaborativo en equipo. Estos son:

- **Comunicación:** La comunicación promueve la colaboración y el aprendizaje en equipo, es la clave para alcanzar un objetivo.
- **Simplicidad:** Se busca mantener la sencillez, sobre todo en el diseño y codificación, se basa en la experiencia adquirida para no repetir los fracasos anteriores.
- **Retroalimentación:** Se necesita conocer las necesidades del cliente para realizar cambios o correcciones a lo ya realizado.
- **Coraje:** Se promueve la toma de decisiones ante la inacción, esto fomenta la actitud positiva y el valor para enfrentar los problemas que surgen durante el ciclo de vida de un proyecto.
- **Respeto:** Busca el respeto hacia los demás para llegar a acuerdos, lo que produce una mayor efectividad en el equipo de trabajo. (Pantaleo & Rinaudo, 2015)

### 1.1.4 Prácticas técnicas de XP

XP propone doce prácticas técnicas simples para garantizar un mejor resultado en un proyecto. Estas prácticas, que deben aplicarse de forma conjunta, son:

- **Cliente presente:** XP busca acabar con el paradigma de que la presencia del cliente no es necesaria, al contrario, la presencia del cliente es de vital importancia en cada iteración del proyecto. Esto ayudará a despejar dudas que surjan en el transcurso del desarrollo. También permitirá establecer prioridades y ayudará a resolver conflictos.
- **Ritmo sustentable:** El equipo de trabajo no debe asumir más trabajo del que puede realizar. Si se demanda un mayor esfuerzo al equipo, la calidad de los resultados puede disminuir. Esta práctica hace que XP sea muy diferente de otras metodologías que exigen un mayor esfuerzo al equipo cuando hay estimaciones deficientes. XP recomienda no sobrepasar las 8 horas diarias y 40 semanales.
- **Metáfora:** El uso de metáforas ayuda a solucionar problemas de comunicación entre los técnicos y los usuarios, utilizando conceptos cotidianos de la vida real para explicar conceptos técnicos. La metáfora cumple su función cuando es comprendida de inmediato.

- **Refactoring (refactorización):** Esta técnica consiste en reorganizar el código de un software sin afectar su comportamiento. Esto ayuda a eliminar redundancias cuando se desarrolla una funcionalidad nueva, lo que permite que el código evolucione con facilidad con cada iteración.
- **Programación en pares:** Esta práctica se refiere a dos programadores trabajando en una misma computadora. Permite que uno escriba el código mientras el otro lo revisa, o que el programador más experimentado explique el desarrollo al menos experto, o que los dos piensen cómo resolver el código mientras uno lo escribe.
- **Entregas cortas:** Cada vez que se agreguen pequeñas funcionalidades al software, se las debe entregar al cliente para que lo que tenga que probar sea poco. Esto mejora su experiencia con el software.
- **Código estándar:** El uso de estándares en la escritura del código permite que éste sea más limpio y facilita su comprensión a otros programadores. Cuando hay lenguajes que no tienen reglas de estilo o estándares definidos, el equipo debe ponerse de acuerdo para definir y documentar sus propias reglas.
- **Código colectivo:** En XP los desarrolladores tienen acceso a todo el código del sistema. El objetivo es que todos conozcan lo que se está haciendo y como se lo hace; así, cuando surge un problema en cierta parte del código, cualquier miembro del equipo puede resolverlo y no solo el que lo programó.
- **Integración continua:** XP propone que todo el código desarrollado por los programadores en actividades individuales debe ser guardado diariamente antes de realizar las pruebas de integración, así, se evitara que el nuevo código dañe el anterior.
- **Juego de planificación:** En XP la dinámica de planificación se lleva a cabo al inicio de la iteración. Por lo general suele ser la siguiente:
  - *El cliente* entrega una lista de funcionalidades que desea para el sistema, esta lista tiene el formato de "Historias de Usuario", donde se define su comportamiento y sus criterios de aceptación.
  - *El equipo* de desarrollo realiza la estimación de esfuerzo para su desarrollo. En XP las iteraciones suelen durar entre una y cuatro semanas.
  - *El cliente* decide qué "Historias de Usuario" se desarrollarán y el orden en que lo harán.
- **Diseño simple:** Esta práctica permite mantener un diseño sencillo y estandarizado para facilitar su comprensión y refactorización. Se refiere a hacer lo mínimo indispensable, lo más legible posible.

- **Desarrollo orientado a pruebas:** Las pruebas de aceptación se desarrollan en base a las historias de usuario, aquí se define qué criterios debe cumplir el desarrollo para satisfacer los objetivos esperados. (Ramos, Noriega, Laínez, & Durango, 2017)

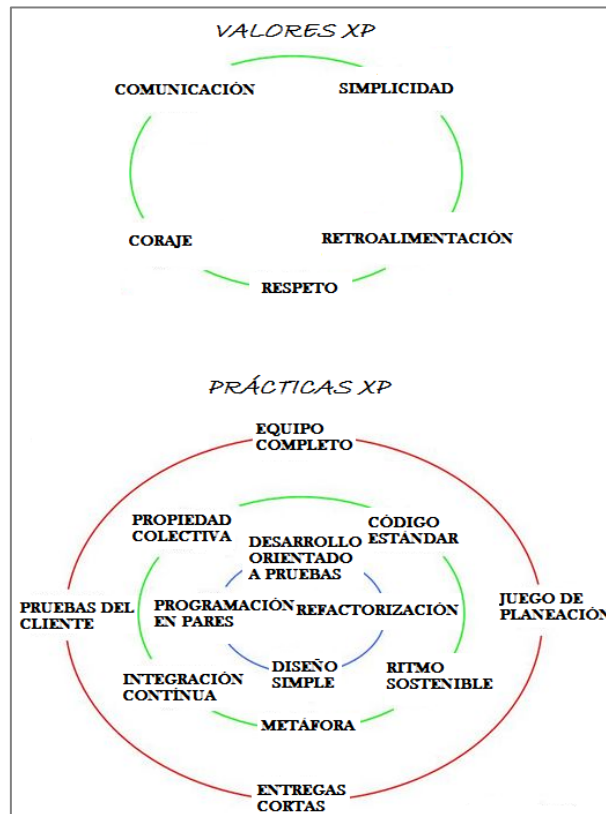


Figura 1: Valores y prácticas de XP

### 1.1.5 Roles XP

De acuerdo con XP, los roles del equipo de trabajo son:

- **Ciente:** Es el responsable de dirigir el proyecto. Define las historias de usuario y su prioridad en el desarrollo. Además, es el encargado de definir las pruebas funcionales para validar su implementación.
- **Programador:** Es el delegado de estimar el tiempo necesario y desarrollar el código de las historias de usuario.
- **Encargado de pruebas (Tester):** Tiene la función de ejecutar las pruebas regularmente y difundir los resultados. Puede ayudar también al cliente a escribir las pruebas funcionales.

- **Encargado de seguimiento (Traker):** Su función es la de verificar si el avance del proyecto va de acuerdo con las estimaciones realizadas y determinar si es necesario realizar cambios para lograr los objetivos.
- **Entrenador (Coach):** Este rol no se encuentra en todos los equipos XP. Es aquel que dirige al equipo de desarrollo para que las prácticas de XP se apliquen correctamente.
- **Gestor (Big Boss):** Es la persona que tiene la idea general del proyecto. Su tarea principal es supervisar el avance del proyecto. Este rol lo puede asumir el cliente cuando no hay un gerente a cargo. (Díaz & Collazo, 2013)

## 1.2 Historia y relevancia del idioma Kichwa.

### 1.2.1 Comunicación entre indígenas y el origen del kichwa

En los tiempos antes de la colonia, los indígenas del gran imperio del Tahuantinsuyo compuesto por Chinchaysuyo, Collasuyo, Antisuyo y Contisuyo, utilizaban un sistema de redacción denominado los Quipus<sup>3</sup>. Este sistema gráfico de cálculo consistía en una serie de cordeles de varios colores anudados de diferentes maneras.

Gracias al conjunto de principios de los Quipus, los indígenas tenían la posibilidad de estructurar documentos para transmitir informes, mandatos, peticiones, contabilidad, administración, etc. Este sistema era único y para todos, es decir, el Quipu confeccionado por una persona era comprensible para cualquier otra persona, existían leyes que enseñaban su correcta elaboración e interpretación de acuerdo con sus nudos y a sus colores. Este tipo de comunicación les permitió expresar conceptos sobre historias, leyes, ceremonias, negocios, etc.; y sin ellos, los mensajes hubiesen sido orales y su transmisión hubiese sufrido distorsiones o adulteraciones. El sistema de los Quipus evidencia que el kichwa sí tenía un tipo de escritura, no de la forma en la que la entendemos ahora, pero sí como un sistema estructurado para transmitir ideas. (Revenga, 2017)

La presencia del idioma kichwa en el antiguo imperio del Tahuantinsuyo quedó registrada en textos de cronistas indígenas, españoles y mestizos. La influencia del kichwa en el pueblo Inca, en su lengua y en su cultura fue tan fuerte que los nombres de los personajes y de los elementos registrados en estas crónicas son kichwas. La lengua kichwa fue expandiendo su territorio y debilitando otras lenguas que se hablaban en la

---

<sup>3</sup> Quipus: Del quechua *Quipu* que significa nudo.



región del Uraysuyu<sup>4</sup> lo que hoy se conoce como Quito, esto gracias a la influencia del imperio incario y su habilidad para negociar con los pueblos que oponían resistencia para al final establecer alianzas con ellos con el fin de evitar bajas de los dos lados.

Numerosos estudios se han generado para definir el proceso de expansión de la lengua kichwa y para identificar la manera en la que cada región adquirió su propio dialecto. Se supone al menos tres etapas: la preincaica, la incaica y la colonial, lo que da lugar a varias hipótesis sobre los puntos desde donde se expandió el kichwa:

- Origen serrano. – Tiene como su centro de origen al Cuzco y que su mecanismo de difusión fueron las conquistas incaicas.
- Origen costeño. – Fundamentada en las referencias de fray Martín de Murúa<sup>5</sup>, quien plantea que el inca Huayna Cápac había ordenado que se hablara el idioma de su madre, que era una variante del kichwa.
- Origen forestal. – Que el centro de expansión fue la selva alta, en la región del Amazonas, área comprendida entre Chachapoyas y Macas.
- Origen ecuatoriano. – De acuerdo con referencias del padre Juan de Velasco<sup>6</sup>, se refiere a que el kichwa existía antes de los incas y que la presencia de esta lengua causó asombro al mismo Huayna Cápac. (Kowii, 2013)

### 1.2.2 Diferencia entre quechua y kichwa (quichua)

La unidad del pueblo indígena de los Andes, que conformaba el imperio del Tawantinsuyu<sup>7</sup>, sufrió una fragmentación tras la colonización europea que impuso su lengua y su cultura, creando así una frontera lingüística que separa el quechua y el kichwa. Las principales diferencias entre estos dos dialectos son las siguientes:

TABLA 1.1  
DIFERENCIAS ENTRE QUECHUA Y KICHWA

QUECHUA	KICHWA
Tiene una marcada influencia de otras lenguas como el aymara y el guaraní, y tiene una vinculación más estricta con el Tahuantinsuyo	Tiene sus orígenes históricos en la parte norte de este mismo imperio conocido como Chinchaysuyo, en el que se encontraba lo que hoy es el territorio ecuatoriano
Existe en Perú y Bolivia	Existe en Ecuador
Cuenta con las cinco vocales del alfabeto español.	Cuenta solo con tres vocales: a, i, u.

Fuente: (Kowii, 2013)

<sup>4</sup> Uraysuyu: Palabra en kichwa que significa Norte.

<sup>5</sup> Martín de Murúa o Martín de Morúa (1525 o 1540 - Madrid, 1617 o 1618), religioso mercedario español, que pasó al Virreinato del Perú como doctrinero.

<sup>6</sup> Juan de Velasco y Pérez Petroche, sacerdote jesuita nacido en Riobamba el 6 de enero de 1727.

<sup>7</sup> Tawantinsuyu: Tahuantinsuyo en lengua kichwa.

De acuerdo con el material educativo y lingüístico de la Dirección Nacional de Educación Intercultural Bilingüe (DINEIB) y la Academia de la lengua Kichwa del Ecuador (ALKI), se puede observar que en el alfabeto del idioma kichwa unificado, no existe la grafía “q”, razón por la cual el nombre del idioma de la variante ecuatoriana de acuerdo con su alfabeto es “kichwa” y no “quichua”. (Instituto de Idiomas, Ciencias y Saberes Ancestrales, 2016)

### **1.2.3 Pérdida de la lengua kichwa**

Desde tiempos de la colonia, la lengua kichwa fue despreciada y opacada por los intentos de dominio político, religioso y económico, y a pesar de esto ha conseguido sobrevivir en muchos países de la región andina, entre ellos, Ecuador.

En el Ecuador, la lengua kichwa ha sufrido varios cambios debido a la mezcla de términos o palabras completas en español, lo que la han hecho decaer. Por esta razón las personas kichwahablantes tienden a olvidar palabras que existen en su idioma deformando así su lengua y a la vez la lengua española. Cabe resaltar que las causas de esta deformación del idioma kichwa se debe en gran parte a una política lingüística homogenizante, que pretendía que todos los ecuatorianos hablaran el idioma oficial que era el español, sin tomar en cuenta la pérdida de lenguaje y de identidad cultural. (Ayala, 2014)

Otro factor que debilita la lengua kichwa es el empobrecimiento espiritual que da lugar a la pérdida de la memoria cultural e histórica y con ello se pierde el lenguaje, produce baja autoestima y auto racismo en el pueblo indígena. Esto se evidencia cuando los mismos dirigentes, intelectuales indígenas y no indígenas que defienden el sistema bilingüe, no hablan o escriben en sus lenguas maternas. En estas circunstancias, es un reto contribuir en los procesos de liberación espiritual para defender la autodeterminación de los pueblos indígenas. (Kowii, 2015)

### **1.2.4 El kichwa unificado**

Desde los años 70s comienza un notable fortalecimiento de las organizaciones indígenas, lo que dio lugar al rescate de la cultura y de la lengua de los pueblos y las

nacionalidades indígenas del Ecuador. La Academia de la lengua kichwa, junto con la Dirección Nacional, las Direcciones Regionales y Provinciales de Educación Intercultural Bilingüe iniciaron el Programa Nacional de Alfabetización Kichwa, para ello se buscó oficializar un alfabeto kichwa unificado con el fin de establecer semejanzas en su lectura y su escritura, y así facilitar su comprensión. En esta labor participaron dirigentes indígenas de diferentes organizaciones del país, antropólogos indígenas e hispanohablantes, profesores e investigadores de las lenguas indígenas. Los principales acuerdos logrados en la estandarización del alfabeto kichwa son los siguientes:

TABLA1.2  
HISTORIA DEL ALFABETO KICHWA

Marzo y abril de 1980	Se define por primera vez el alfabeto kichwa que consta de 21 letras, 18 consonantes: c, ch, h, j, l, ll, m, n, ñ, p, q, r, s, sh, t, ts, y, z; y 3 vocales: a, i, u. Alfabeto que sería usado en la escritura en lengua kichwa y en los programas de alfabetización del Sistema de Educación Intercultural Bilingüe
Julio de 1998	Se lleva a cabo el "II Encuentro para la unificación de la lengua Kichwa", coordinado por la DENEIB con la participación de instancias comprometidas en el desarrollo de las culturas indígenas, quienes, en la búsqueda de un mejor desarrollo del idioma a fin de evitar conflictos ortográficos, formulan varios cambios en el alfabeto unificado de 1980. Como resultado de este encuentro, se establece una actualización al alfabeto kichwa unificado, que indica que la lengua kichwa utilizará 18 grafías, 15 consonantes: ch, j, k, l, ll, m, n, ñ, p, r, s, sh, t, w, y; 3 vocales: a, i, u; y la posibilidad de utilizar la (ts) para palabras preincas o protokichwas.
Marzo de 2004	La Academia de la Lengua Kichwa, luego de varios análisis lingüísticos, pedagógicos y sociales, acuerda el uso de la "h" en lugar de la "j".
16 de septiembre de 2004	Según Acuerdo Ministerial 244, se oficializa el alfabeto kichwa con 18 grafías: ch, j, k, l, ll, m, n, ñ, p, r, s, sh, t, (ts), w, y; y 3 vocales: a, i, u.

Fuente: (Instituto de Idiomas, Ciencias y Saberes Ancestrales, 2016)

### 1.2.5 El uso de las TIC en la preservación de las lenguas originarias

Los gobiernos de Latinoamérica han mostrado un gran interés en el tema de los pueblos originarios y en el rescate de sus tradiciones, costumbres y de sus lenguas originarias. El Estado a través de sus organismos e instituciones quiere contribuir para evitar que las raíces de su cultura se sigan perdiendo, un buen ejemplo es la inclusión de la Educación Intercultural Bilingüe.

Uno de los principales problemas que se debe contrarrestar, es la autodiscriminación que los pueblos indígenas tienen respecto a su lengua. El escenario para la recuperación de las lenguas originarias muestra varias dificultades, dado que el abuelo no quiere enseñar, el hijo no quiere aprender y el nieto que tiene el interés y la capacidad de

aprender, debe buscar nuevas formas de aprendizaje, y es allí donde los potenciales de las Tecnologías de la Información y de la Comunicación hacen su aporte.

La integración de las lenguas originarias en las TIC presenta varias ventajas como: mayor comunicación, mayor permanencia, mejor administración y distribución del conocimiento, mayores fuentes de conocimiento y oportunidades de investigación. Además, permite compartir experiencias y trascender las barreras del tiempo y el espacio, esto aumenta las posibilidades de los promotores de estas lenguas para llegar a más personas. Algunos ejemplos de la inclusión de las lenguas originarias en las TIC son:

- [kichwa.net](http://kichwa.net): Página web para el aprendizaje de la lengua kichwa.
- [llcanet.org/ciberaymara](http://llcanet.org/ciberaymara): Para el aprendizaje de Aymara.
- [mexica.oui.net](http://mexica.oui.net): Para el aprendizaje de náhuatl.
- [wikiguate.com.gt](http://wikiguate.com.gt): Wiki multilingüe en español, maya k'iche' y maya mam con información sobre Guatemala.
- [indigenoustweets.com](http://indigenoustweets.com): Sitio para que "twitteros" se expresen en su idioma nativo.
- [mozilla-mexico.org/firefox-en-maya/](http://mozilla-mexico.org/firefox-en-maya/): Proyecto Firefox en lenguas nativas de México.
- AbiWord: Procesador de texto de software libre traducido al quechua y al aymara.
- Windows XP en lenguaje mapuzugun: Sistema operativo que Microsoft entregará al Ministerio de Educación de Chile.
- MapuApp: Aplicación para iPhone. Es un traductor español-mapuche. (Lippenholtz, 2012)

## **1.3 Framework CakePHP**

### **1.3.1 Definición**

CakePHP es un framework libre y de código abierto, que ayuda a desarrollar aplicaciones Web en lenguaje PHP de forma estructurada y rápida. Dispone de herramientas que permiten programar la lógica específica de la aplicación, lo que da como resultado un núcleo bien probado y con mejoras constantes. La estructura básica de un proyecto de CakePHP utiliza el patrón de diseño de software Modelo-Vista-Controlador (MVC), separando la aplicación en tres partes fundamentales. (CakePHP, 2017)

### 1.3.2 Características

Entre las principales características de CakePHP se encuentran:

TABLA 1.3  
CARACTERÍSTICAS DEL FRAMEWORK CAKEPHP

Curva de aprendizaje corta	Es fácil familiarizarse con los métodos de la API principal.
Convenciones sobre la configuración	Las convenciones de CakePHP son simples y fáciles de aprender, siguiendo estas convenciones no hay que configurar nada.
Fácil instalación	Se puede instalar fácilmente ejecutando solo una línea de código.
Licencia MIT	Permite la reutilización del código dentro de software libre y propietario.
Generación automática de código	Su consola de comandos permite generar automáticamente los archivos del modelo, la vista y el controlador.
Validaciones incorporadas	Los modelos generados contienen validaciones de datos muy útiles para que la aplicación trabaje con datos limpios y correctos.
Arquitectura MVC	Visto como un paradigma o una forma de pensar, permite crear un mejor código.
URLs y rutas limpias	Permite total libertad para manipular las URLs de la aplicación.
Almacenamiento flexible	Tiene seis diferentes tipos de almacenamiento y permite al desarrollador agregar tipos de almacenamiento propios.
Localización incorporada	Permite crear archivos de traducción sin mucho esfuerzo.
Unidad de pruebas integrada	Provee clases y métodos muy útiles para realizar pruebas del sistema

Fuente: (Dása, 2016)

### 1.3.3 Ciclo de una petición CakePHP

Los pasos a alto nivel que sigue la petición de un cliente solicitando una página o recurso en CakePHP son los siguientes:

- Las reglas de reescritura del servidor web dirigen la petición a `webroot/index.php`.
- La aplicación se carga y es ligada al servidor HTTP.
- Se inicializa el Middleware<sup>8</sup> de la aplicación
- Una petición con su respuesta es despachada a través del Middleware PSR-7<sup>9</sup> que la aplicación usa. Comúnmente incluye la captura de errores y redireccionamiento.
- Si no hay respuesta del Middleware y la petición contiene información, se selecciona un controlador y una acción.
- Se llama a la acción del controlador y éste interactúa con los Modelos y Componentes requeridos.

<sup>8</sup> Middleware: Código que se ejecuta entre una solicitud y la respuesta generada.

<sup>9</sup> PSR7: Interfaces de mensajes HTTP.

- g. El Controlador delega la creación de la respuesta a la Vista usando los datos del modelo.
- h. La Vista usa Helpers y Cells para generar el cuerpo y los encabezados de la respuesta.
- i. Se envía la respuesta de vuelta a través del Middleware del Controlador.
- j. Finalmente, el servidor HTTP envía la respuesta al servidor Web. (CakePHP, 2017)

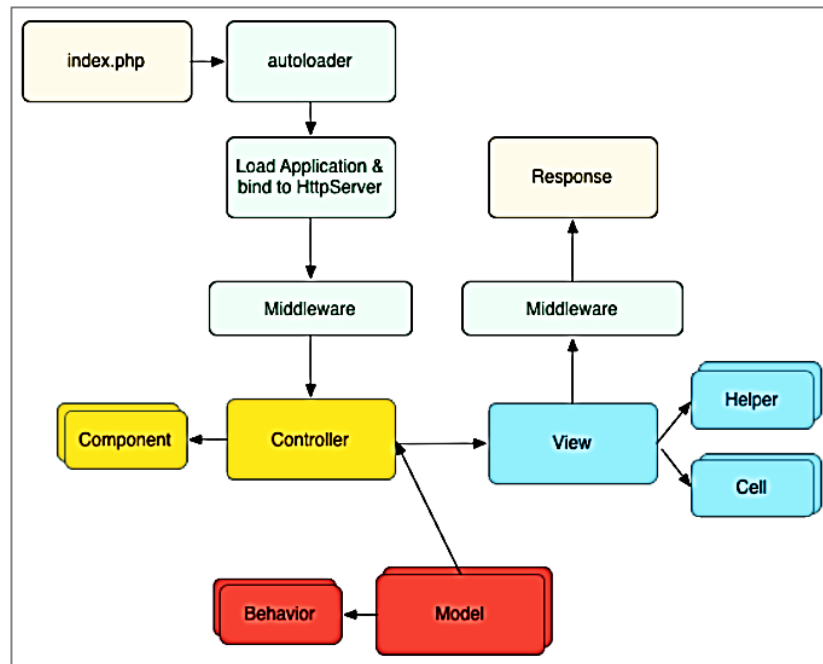


Figura 2. Ciclo de una petición CakePHP (Fuente: CakePHP 2017)

## 1.4 Framework ZURB Foundation

### 1.4.1 Definición

Zurb se ha convertido en una tecnología muy útil para crear y prototipar sitios web rápidos y adaptativos. Su framework CSS Foundation facilita el desarrollo del código de front-end escalable, para ello incorpora complementos, librerías y herramientas necesarias para trabajar con HTML, CSS y JavaScript priorizando el diseño adaptativo. Foundation se basa en el paradigma de la tecnología móvil, dado que los teléfonos inteligentes se han convertido en el principal dispositivo para navegar por la web. (Shenoy, 2015)

### 1.4.2 Características

Foundation es un framework que se enfoca en el desarrollo de aplicaciones para dispositivos móviles. Antes, los diseñadores enfocaban su trabajo en computadoras de escritorio o pantallas grandes y luego trataban de modificarlas para que sean compatibles con las pantallas de los dispositivos. Esta forma de construir aplicaciones es improductiva y muy difícil de lograr cuando un sitio web es intuitivo y lleno de funciones.

Con su enfoque primero-móvil, opta por el diseño para pantallas pequeñas y luego agrega características para pantallas de escritorio. Esto es muy útil dada la revolución de los dispositivos móviles y el acceso a Internet de hoy en día. Priorizar el diseño para dispositivos móviles también da como resultado una mejor semántica y una mejor experiencia al usuario, lo que mejora la competitividad de un sitio Web. (Shenoy, 2015) Sus principales características son:

TABLA 1.4  
CARACTERÍSTICAS DE ZURB FOUNDATION

Cuadrícula de 12 columnas	Permite escoger un diseño fijo de acuerdo con el tamaño del dispositivo.
Cuadrícula Flexbox	Se caracteriza por ser flotante, con atributos como el tamaño automático, ajustes adaptativos, la alineación y el nuevo orden de fuentes, que son muy útiles para desarrollar sitios Web complejos. Por el momento Flexbox está disponible solo con Sass y no hay una versión CSS.
Elementos tipográficos	Los elementos tipográficos de Foundation proveen estilos limpios y atractivos, esto ayuda a desarrollar aplicaciones Web con un alto nivel de coherencia para que el formato con el que se presenta la información escrita sea el más adecuado, de tal forma que cause el mayor impacto
Componentes de navegación	Son flexibles, intuitivos y fáciles de usar, lo que permite la creación de patrones de navegación comunes y con el mismo marcado de una forma fácil. Y los atributos de Medios, que permiten incrustar contenido, llamar acciones, adjuntar metadatos, mostrar animaciones y muchas otras opciones personalizables, permiten construir un diseño Web sofisticado.

Fuente: (ZURB, 2017)

Una de las principales mejoras que Foundation 6 trae en comparación con la versión 5 es la significativa reducción de su tamaño alrededor de la mitad, lo que hace que su tiempo

de carga sea más rápido. En la versión 5, los archivos CSS pesaban 160KB y los archivos JavaScript 110KB. Ahora, estos archivos pesan 60KB y 84KB respectivamente.

Junto con la reducción de tamaño, otros cambios importantes en su nueva versión son los siguientes:

- Sistema de menú completamente personalizable y modular.
- Redefinición de componentes grandes en componentes pequeños.
- Reducción de código en complementos JavaScript, reduciendo así el tamaño de cada plugin y creando más bibliotecas de utilidades compartidas.
- Reducción de estilos básicos, en su lugar los estilos CSS más simples son más fáciles de modificarlos.
- Biblioteca IU de animación basada en Sass para ayudar a agregar movimiento a los sitios Web. Simplificación de las variables Sass y mixins para crear menos opciones.
- Redefinición del JavaScript para que los componentes no accedan a su propia funcionalidad sino a utilidades universales compartidas.
- Plantilla de inicio con un generador de estilos estáticos personalizado para ayudar a aplanar archivos en documentos HTML individuales.
- Nueva herramienta de línea de comandos (CLI) que permite armar proyectos en blanco para sitios, aplicaciones y correos electrónicos con menos dependencias. Compatible con NPM<sup>10</sup>, Bower<sup>11</sup>, Meteor<sup>12</sup> y Composer. (ZURB, 2017)

## 1.5 Lenguaje de programación PHP

### 1.5.1 Historia y evolución

PHP fue desarrollado en 1994 por Rasmus Lerdorf, inicialmente ideado como un juego de herramientas para ayudar a la creación de páginas personales (Personal Home Page Tools o PHP tools), fue liberado gratis al público en 1995 como lenguaje de programación PHP. Dos años más tarde, fue lanzado un nuevo paquete del lenguaje llamado PHP/FI, que incluía un interpretador de comandos SQL. Más tarde, el programador israelí Zeev Suraski, desarrolló el analizador de PHP3 que incluía el primer recurso orientado a objetos. La versión de PHP4 fue desarrollado por Zeev y Andi Gutmans, versión que se encontró

---

<sup>10</sup> NPM: Del inglés *Node Package Manager* es un gestor de paquetes que facilita el trabajo con Node.

<sup>11</sup> Bower: Programa basado en NodeJS que se ejecuta desde la consola y que tiene un sencillo API de comandos útiles para realizar tareas de mantenimiento y administración de paquetes.

<sup>12</sup> Meteor: Plataforma para crear aplicaciones web en tiempo real construida sobre Node.js.



con el problema de creación de copias de objetos, este problema fue resuelto en la nueva versión PHP5 con la incorporación de handlers o apuntadores que permiten copiar un apuntador de un objeto original, con el objetivo de actualizar las copias de dicho objeto cuando el original es modificado. (Arias M. A., 2017)

PHP ha evolucionado en un lenguaje de scripting o de interpretación del lado del servidor robusto y ha permanecido como una de las mejores opciones para el desarrollo de aplicaciones Web, por su facilidad de uso, su escalabilidad, por ser un proyecto de código libre, por su amplio soporte para bases de datos y por su fácil incrustación en sistemas de gestión de contenido web. PHP está a libre disposición a través de internet sin pagos por licencias, sin la necesidad de programas especiales y con una comunidad de usuarios muy amplia en el caso de requerir soporte, es por esto que más de la mitad de todos los sitios web del mundo lo utilizan, y sitios como Facebook son un ejemplo de lo potente que es este lenguaje. (Beati, 2015)

### 1.5.2 Características

PHP es un lenguaje extremadamente modular y es ideal para instalarlo y usarlo en servidores web, sus principales características son:

TABLA 1.5  
CARACTERÍSTICAS DE PHP

Facilidad de uso	Su sintaxis, sus tipos de datos y sus funciones son similares a los del lenguaje C, C++ y Perl.
Compatibilidad	PHP tiene versiones disponibles para Windows, Linux, FreeBSD, Mac, Novel Netware, RISC, AIX y Solaris. Y es compatible con servidores Apache, IIS, Nginx entre otros.
Soporte de bases de datos	Da soporte a varias bases de datos, por ejemplo: MySQL, Postgres, Oracle, SQLite, Sybase, InterBase, Firebird, MSSQL, etc.
Soporte de protocolos de Internet	Da soporte a los protocolos: HTTP, XML-RPC, SOAP, IMAP, NNTP, POP3, ANMP, LDAP, y es posible interactuar con otros protocolos mediante sockets.
Software libre	Es gratuito y de código abierto.
Potente	Es veloz, robusto, estructurado y orientado a objetos.

Fuente: (Arias M. A., 2017)

## **1.6 Base de Datos MySQL**

### **1.6.1 Definición de base de datos**

Una base de datos es un conjunto de datos almacenados cuyo objetivo es brindar información a los usuarios, por medio de transacciones como inserción, actualización y eliminación de los mismos. Hay varios tipos de modelos de base de datos:

- Flat file: almacena los datos en archivos planos.
- Relacional: Almacena los datos en tablas que están relacionadas entre sí, este tipo de base de datos es el más popular.
- Orientado a objetos: Almacena los datos en clases y objetos.
- Jerárquico: Organiza los datos en forma de árbol.
- Red: Almacena los datos en tablas conectadas entre sí por referencias, lo que genera una estructura similar a una red.
- Dimensional: Los datos son organizados e integrados en varias dimensiones, lo que facilita realizar consultas sobre los datos. (Arias M. A., 2014)

### **1.6.2 Sistema de Base de Datos(SBD)**

Un sistema de gestión de bases de datos es un conjunto de programas informáticos que gestiona una base de datos. El objetivo principal de los SBD es evitar que los usuarios manipulen los datos de forma directa, para ello, proveen un marco estándar para su manipulación y organización y una interfaz estándar con la que otros programas pueden acceder a ellos. Entre los principales SBD del mercado se encuentran:

- MySQL
- PostgreSQL
- Oracle
- SQL Server (Arias M. A., 2014)

### **1.6.3 Historia y evolución de MySQL**

MySQL fue creada en 1995 por la empresa establecida en Suecia “MySQL AB” por los suecos David Axmark, Allan Larsson y el finlandés “Monty” Widenius. El objetivo principal de los creadores de MySQL fue cumplir con los estándares del Lenguaje de consultas estructurado (MySQL), y a la vez, superar a otros SGBD en velocidad, fiabilidad y usabilidad. (Torres, 2014)

Actualmente, MySQL es propiedad de Oracle y cuenta con más de 15 millones de instalaciones alrededor del mundo y entre sus usuarios se encuentran: Google, Sony, NASA, Cisco Systems, Nokia, HP, U.S. Army, entre otros. Gracias a su fácil integración con PHP, los paquetes de hospedaje de internet casi obligatoriamente incluyen MySQL. (Arias, Arias, & Durango, 2016)

### 1.6.4 Características

Entre las principales características de MySQL se encuentran las siguientes:

TABLA 1.6  
CARACTERÍSTICAS DE MySQL

Multiplataforma	Es compatible con Unix, Linux, Solaris, Windows, Alpha, SPARC, PowerPC, entre otros.
Portable	Es un sistema portable de licencia GPL, puesto que usa GNU Automake, Autoconfig y Libtool.
Interfaz de programación de aplicaciones (API)	Para el desarrollo de aplicaciones en lenguajes de programación como C, C++, Java, PHP, Perl, y otros.
Rendimiento	Su sistema de almacenamiento es transaccional y no transaccional y su sistema de reserva de memoria basado en hilos y multihilos permite que sea muy rápido. Soporta bases de datos de gran tamaño, se han probado con más de 60 mil tablas y 5 billones de registros.
Seguridad	Su sistema de privilegios y contraseñas es muy flexible y seguro. Las contraseñas son cifradas cuando se conecta con un servidor.
Soporte SQL	Tiene un soporte completo para sentencias SELECT y WHERE de la misma manera que SQL Server.
Potente	Está desarrollado en su totalidad en lenguaje C y C++. Usa una librería altamente optimizada para las funciones SQL y normalmente no utiliza reservas de memoria luego de la inicialización para consultas.

Fuente: (Torres, 2014)

## 1.7 Patrón de arquitectura de software MVC

### 1.7.1 Historia y evolución

En los años 70s, Trygve Reenskaug junto con otros inventores de Smalltalk<sup>13</sup>, introducen el concepto Modelo-Vista-Controlador, con el objetivo de encapsular ciertos

<sup>13</sup> Smalltalk: Lenguaje reflexivo de programación, orientado a objetos y con tipado dinámico.

datos junto con su procesamiento (Modelo), aislándolos de la parte que los manipula (Controlador) y la parte que los presenta (Vista).

El patrón arquitectónico MVC, del acrónimo Modelo-Vista-Controlador, fue creado como una solución modular al problema de diseñar una interfaz de usuario moderna y flexible que cumpla con requisitos como:

- Cambios en la interfaz para mejorar su eficiencia o hacerla más fácil de usar.
  - Actualización de la interfaz ante nuevas funcionalidades del sistema.
  - Evolución del sistema de ventanas de la plataforma que la aplicación utiliza.
  - Posibilidad de recibir información de diferentes medios y presentarla mediante diferentes vistas.
  - Las modificaciones de datos manipulados por la aplicación deben actualizarse inmediatamente en las vistas.
  - Posibilidad de manipular datos de la aplicación en varias interfaces al mismo tiempo.
- (Debrauwer, 2013)

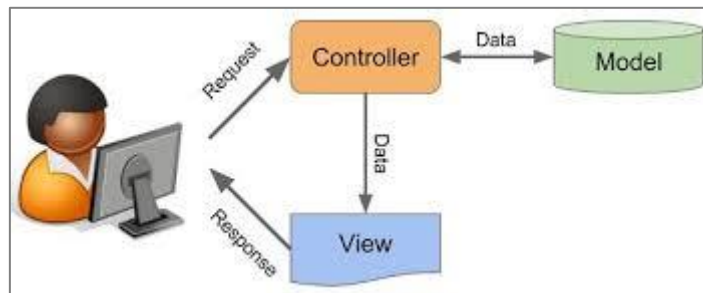


Figura 3: Funcionamiento de MVC (Fuente: <https://helloacm.com/model-view-controller-explained-in-c/>)

### 1.7.2 MVC en aplicaciones PHP

Es un patrón clásico en el diseño web y consta de tres niveles:

- El Controlador: Es el encargado de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones. (HTTP, consola de comandos, email, etc.)
- El Modelo: Es el encargado de la abstracción de la lógica relacionada con los datos permitiendo que las vistas y las acciones sean independientes del gestor de base de datos que esté usando.
- La Vista: Extrae los datos del modelo y le da el formato que requiere la presentación.

MVC separa la lógica de negocio (el modelo) y la presentación (la vista) lo que permite que el mantenimiento de las aplicaciones sea más sencillo. Por ejemplo, cuando una aplicación debe ejecutarse en un navegador estándar de un computador y en el navegador de un dispositivo móvil, solamente hay que crear vistas diferentes, manteniendo el modelo y el controlador original. (Sánchez, 2012)

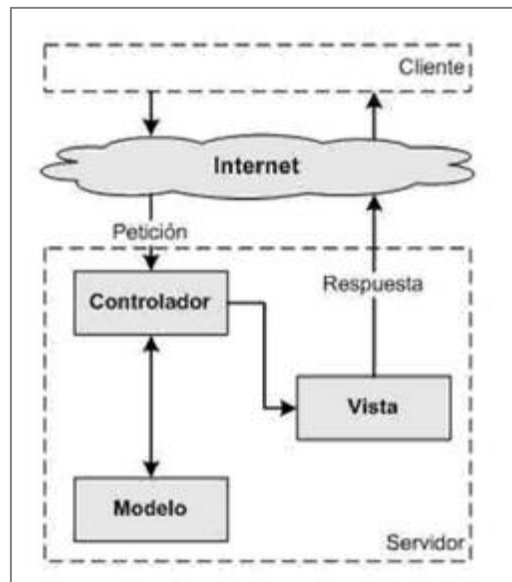


Figura 4: Patrón MVC en aplicaciones web (Fuente: Sánchez 2012)

## 1.8 Servidor Apache

### 1.8.1 Definición

El servidor Apache es un servidor web HTTP de código abierto, de libre distribución para plataformas Windows, Linux, Macintosh entre otras. Apache es una aplicación, que, gracias a su compatibilidad abierta, permite implementar un servidor local en un computador personal, sin importar el sistema en el que está ejecutándose. El servidor que compete con Apache es IIS, propiedad de Microsoft que ofrece las mismas funcionalidades, pero la gran diferencia es que es un software propietario.

### 1.8.2 Características

El servidor Apache tiene gran afinidad con el lenguaje de programación PHP y cuenta con librerías que le dan soporte. Entre sus principales características se encuentran:

- Soporte completo para PHP.
- Soporte para certificados SSL y TLS.

- Contiene módulos de autenticación web, como el mod\_acces, mod\_auth y mod\_digest.
- Permite configurar y personalizar mensajes de error y negociación de contenido.
- Soporta autenticación de bases de datos basada en SGBD. (Torres, 2014)

## 1.9 Instructivo para la aplicación de la evaluación estudiantil<sup>14</sup>

### 1.9.1 Niveles y subniveles educativos

Según el Reglamento General de la LOEI en el artículo 27, consta que el Sistema Nacional de Educación (SNE) tiene tres niveles que son: inicial, básica y bachillerato.

Inicial contiene dos subniveles que son:

- Inicial 1: Subnivel no escolarizado para niños y niñas de hasta 3 años.
- Inicial 2: Para niños de entre 3 y 5 años.

Básica contiene cuatro subniveles que son:

- Preparatoria: 1° grado de Educación General Básica (EGB), estudiantes con 5 años de edad.
- Básica Elemental: 2°, 3° y 4° grados de EGB.
- Básica Media: 5°, 6° y 7° grados de EGB.
- Básica Superior: 8°, 9° y 10° grados de EGB.

El nivel de bachillerato consta de 1°, 2° y 3° curso y tiene dos categorías.

- bachillerato en Ciencias.
- Bachillerato Técnico.

El Sistema de Educación Intercultural Bilingüe (SEIB) que rige para todo los Centros Educativos Interculturales Bilingües se encuentra dentro del Sistema Nacional de Educación. Según el Acuerdo Ministerial No. 440-13 del 5 de diciembre del 2013, este sistema se implementa a través del Modelo del Sistema de Educación Intercultural.

---

<sup>14</sup> Instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación del Ecuador, actualizado a 2017.

TABLA 1.7  
COMPARACIÓN ENTRE EL SNE Y EL SEIB

Procesos de SEIB	(EIFC) Educación Infantil, Familiar y Comunitaria	(IPS) Inserción a los Procesos Semióticos	(FCAP) Fortalecimiento Cognitivo, afectivo y psicomotriz	(DDTE) Desarrollo de destrezas y técnicas de estudio	(PAI) Procesos de aprendizaje Investigativo	Bachillerato
Grados y cursos del SNE	Inicial 1 e Inicial 2	1er. Grado de EGB	2do. a 4to. grados de EGB	5to. a 7mo. grados de EGB	8avo. A 10mo. grados de EGB	De 1er. A 3er. cursos
Niveles del SNE	Educación Inicial	Educación General Básica				Bachillerato

Fuente: (MINEDUC, 2017)

### 1.9.2 Evaluación en el nivel de Educación Inicial y el subnivel de Preparatoria de EGB

En el nivel Inicial y el subnivel de Preparatoria se evalúa de manera cualitativa. En el informe de aprendizaje debe constar el reporte de desarrollo integral que describa los indicadores de evaluación de acuerdo con la siguiente escala:

TABLA 1.8: ESCALA DE EVALUACIÓN PARA EDUCACIÓN INICIAL Y PREPARATORIA

ESCALA	SIGNIFICADO	CARACTERÍSTICAS DE LOS PROCESOS
I	Inicio	El niño o niña, está empezando a desarrollar los aprendizajes previstos o evidencia dificultades para el desarrollo de éstos, para lo cual necesita mayor tiempo de acompañamiento e intervención del docente, de acuerdo con su ritmo y estilo de aprendizaje.
EP	En proceso	El niño o niña está en proceso para lograr los aprendizajes previstos, para lo cual requiere acompañamiento del docente y del representante legal durante el tiempo necesario.
A	Adquirida	El niño o niña evidencia el logro de los aprendizajes previstos en el tiempo programado.
N/E	No evaluado	Este indicador no ha sido evaluado en el quimestre.

Fuente: (MINEDUC, 2017)

### 1.9.3 Calificación de los aprendizajes

La escala de calificaciones para los estudiantes de los subniveles de básica elemental, media, superior y el nivel de bachillerato general es la siguiente:

TABLA 1.9

ESCALA DE CALIFICACIONES

ESCALA CUALITATIVA	ESCALA CUANTITATIVA
Domina los aprendizajes requeridos	9,00-10,00
Alcanza los aprendizajes requeridos	7,00-8,99
Está próximo a alcanzar los aprendizajes requeridos	4,01-6,99
No alcanza los aprendizajes requeridos	<=4

Fuente: (MINEDUC, 2017)

### 1.9.4 Exámenes

Según el Reglamento General a la LOEI, en el SNE se propone varios exámenes con diferentes propósitos, uno para mejorar el promedio obtenido y otros cuando no se ha alcanzado el promedio mínimo, estos son:

- **Examen de recuperación o de mejora del promedio:** Este examen lo puede rendir cualquier estudiante en una o más asignaturas que hubiere aprobado con un promedio anual entre 7 y 9,99 sobre 10. La nota de este examen reemplazará la nota quimestral más baja para mejorar el promedio anual.
- **Examen supletorio:** Este examen lo realiza un estudiante en una o más asignaturas que no hubiere aprobado con un puntaje promedio anual de 5 a 6,99 sobre 10. Para aprobar el examen supletorio se debe obtener una calificación mayor o igual a 7 sobre 10, aun cuando la calificación obtenida en este examen sea mayor a 7 sobre 10, solo se registrará como 7 sobre 10.
- **Examen remedial:** Este examen solo lo realizarán los estudiantes a partir de 8° año de EGB, no se exige este tipo de exámenes para estudiantes de cursos inferiores. Lo realizará un estudiante que hubiere obtenido un puntaje promedio anual menor o igual a 4,99 sobre 10 o no aprobare el examen supletorio. Aun cuando la calificación obtenida en este examen sea mayor a 7 sobre 10, solo se registrará como 7 sobre 10. Cuando un estudiante repruebe en dos o más asignaturas el examen remedial, deberá repetir el grado o curso.
- **Examen de gracia:** Este examen lo realizará un estudiante a partir de 8° año de EGB cuando reprobare el examen remedial en una sola asignatura. Si aprueba este examen será promovido de año, caso contrario deberá repetir el grado o curso. Aun cuando la calificación obtenida en este examen sea mayor a 7 sobre 10, solo se registrará como 7 sobre 10.



### 1.9.5 Procedimientos institucionales de la calificación de los aprendizajes y del refuerzo académico

El año lectivo tiene dos quimestres, en los que se deben laborar doscientos (200) días. Cada quimestre se compone por 20 semanas y está dividido en 3 parciales y un examen quimestral. Se recomienda que los estudiantes generen en cada parcial varios insumos, como mínimo dos. La nota parcial es el promedio resultante de los insumos.

En el caso de las Instituciones educativas Interculturales Bilingües, los insumos corresponden a las calificaciones de las cuatro guías de aprendizaje de la unidad de aprendizaje integrado, que corresponderá a la nota parcial.

La nota del examen quimestral corresponde a la evaluación de los aprendizajes de todo el quimestre y equivale al 20 % de la nota quimestral.

La nota quimestral se obtiene sacando un promedio de las tres notas parciales obtenidas, este valor se lo multiplica por 0.80, lo cual equivale al 80%. La nota obtenida en el examen quimestral se multiplica por 0.20, lo cual equivale al 20%, sumando las dos notas se obtiene la nota final.

En el caso de los estudiantes que asistan al proceso de Refuerzo Académico (RA), los trabajos que se realicen en el refuerzo académico deberán ser calificados y promediados con las notas obtenidas en los demás trabajos académicos, como se indica en el siguiente ejemplo:

TABLA 1.10  
CÁLCULO DE NOTA PARCIAL CON REFUERZO ACADÉMICO

PARCIAL 1 (CON REFUERZO ACADÉMICO)				
Insumo 1	Insumo 2	Insumo de RA 1	Insumo de RA 2	Promedio final parcial 1
6,00	5,65	9,00	8,65	7,32

Fuente: (MINEDUC, 2017)

En el ejemplo anterior, si no existiera refuerzo académico, el *Promedio final parcial 1* sería 5,82, calculado solo con el *Insumo 1* y el *Insumo 2*.

### 1.9.6 Evaluación del comportamiento

La evaluación del comportamiento de los estudiantes tiene como finalidad ayudarlos a

formarse integralmente, es decir, a incorporar en su desarrollo; conocimientos, destrezas y actitudes. La escala de evaluación del comportamiento se describe a continuación:

TABLA 1.11  
ESCALA DE EVALUACIÓN DEL COMPORTAMIENTO

LETRA	CALIFICACIÓN	DESCRIPCIÓN
A	MUY SATISFACTORIO	Lidera el cumplimiento de los compromisos establecidos para la sana convivencia social.
B	SATISFACTORIO	Cumple con los compromisos establecidos para la sana convivencia social.
C	POCO SATISFACTORIO	Falla ocasionalmente en el cumplimiento de los compromisos establecidos para la sana convivencia social.
D	MEJORABLE	Falla reiteradamente en el cumplimiento de los compromisos establecidos para la sana convivencia social.
E	INSATISFACTORIO	No cumple con los compromisos establecidos para la sana convivencia.

Fuente: (MINEDUC, 2017)

Cada parcial se evaluará de forma independiente. La evaluación del quimestre no será un promedio de los parciales, será una apreciación global del comportamiento del quimestre.

### 1.9.7 Evaluación de proyectos educativos

Estos proyectos se deben realizar en la Educación General Básica, debe aplicarse por cada parcial y al finalizar un quimestre. La escala con la que deben evaluar los proyectos es la siguiente:

TABLA 1.12  
ESCALA CUALITATIVA PARA EVALUAR LOS PROYECTOS EDUCATIVOS

ESCALA CUALITATIVA		DESCRIPCIÓN
EX	Excelente	Demuestra destacado desempeño en cada fase del desarrollo del proyecto escolar lo que constituye un excelente aporte a su formación integral.
MB	Muy buena	Demuestra fiabilidad en el desempeño para cada fase del desarrollo del proyecto escolar lo que constituye un aporte a su formación integral.
B	Buena	Demuestra un desempeño medianamente aceptable, en cada fase del desarrollo del proyecto escolar lo que no contribuye totalmente a su formación integral.
R	Regular	Demuestra dificultad en atender cada fase del desarrollo del proyecto escolar lo que contribuye escasamente a su formación integral.

Fuente: (MINEDUC, 2017)

Para la apreciación quimestral, se debe tomar en cuenta todas las apreciaciones parciales.  
(MINEDUC, 2017)

## **1.10 Herramientas de desarrollo**

### **1.10.1 NetBeans IDE<sup>15</sup> 8.2**

NetBeans es un entorno de desarrollo inicialmente creado por la empresa Sun Microsystems que hoy es administrada por Oracle Corporation.

NetBeans es una aplicación desarrollada en un entorno libre, inicialmente diseñada para el lenguaje Java, pero hoy es una potente herramienta para el desarrollo de aplicaciones en diferentes lenguajes, como HTML5 y PHP en una misma aplicación. NetBeans es un producto libre y sin restricciones de uso. Está compuesto por librerías especializadas para el desarrollo de aplicaciones en un entorno gráfico, lo que permite al programador preocuparse solo del código del lenguaje y no en la configuración del entorno de trabajo. (Torres, 2014)

### **1.10.2 Xampp 5.6.12**

Xampp es una aplicación de software libre que permite instalar un entorno AMP (Apache+MySQL+PHP) en sistemas Windows, Linux y MacOS. Incluye el servidor web Apache, el gestor de base de datos MySQL con un administrador gráfico, PhpMyAdmin, y los intérpretes para los lenguajes PHP y Perl. Incluye todo lo necesario para ejecutar aplicaciones web. (Ramos & Ramos, 2014)

### **1.10.3 MySQL Workbench 6.1.7**

MySQL Workbench es un software de código abierto, su entorno visual evita trabajar en un entorno D.O.S. Integra la administración de bases de datos MySQL, permitiendo diseñar, crear, y dar mantenimiento a bases de datos. (Torres, 2014)

### **1.10.4 FPDF**

FPDF (Free PDF) es una herramienta desarrollada en lenguaje PHP que permite generar documentos en formato PDF, es completamente gratuita y puede usarse en aplicaciones libres o comerciales sin ninguna limitación.

---

<sup>15</sup> En inglés *Integrated Development Environment* (IDE) o Entorno de Desarrollo Integrado en español.

Entre sus principales características se encuentran: admisión de imágenes, elección de formato de página, diversos tipos de tipos de fuentes y colores, compatible con PHP4 y PHP5 sin necesidad de ninguna extensión. (Capuñay, 2013)

## CAPITULO 2

### DESARROLLO DE LA SOLUCIÓN

#### 2.1 Arquitectura de la solución

La arquitectura del sistema se basa en el patrón arquitectónico Modelo-Vista-controlador que permite el desarrollo de una aplicación modular, fácil de mantener y que permita la integración de nuevas funcionalidades a futuro.

La capa Modelo o capa de datos es la que se encarga de gestionar los datos de la aplicación. En este caso está compuesta por el ORM del framework CakePHP y el gestor de base de datos MySQL.

La capa Vista es la encargada de presentar la interfaz con la que interactúa el usuario, contiene el código HTML, CSS y JavaScript necesario para generar las vistas que permitirán mostrar la información solicitada o capturar datos del usuario.

La capa Controlador o capa lógica es donde se encuentra toda la lógica del negocio en este caso la programación en PHP. Esta es la encargada de interactuar entre la capa Vista y la capa Modelo. La petición de usuario recibida a través de la Vista es procesada por el Controlador que envía al Modelo, y recibe de él, los datos requeridos para enviar la respuesta nuevamente al usuario a través de la Vista.

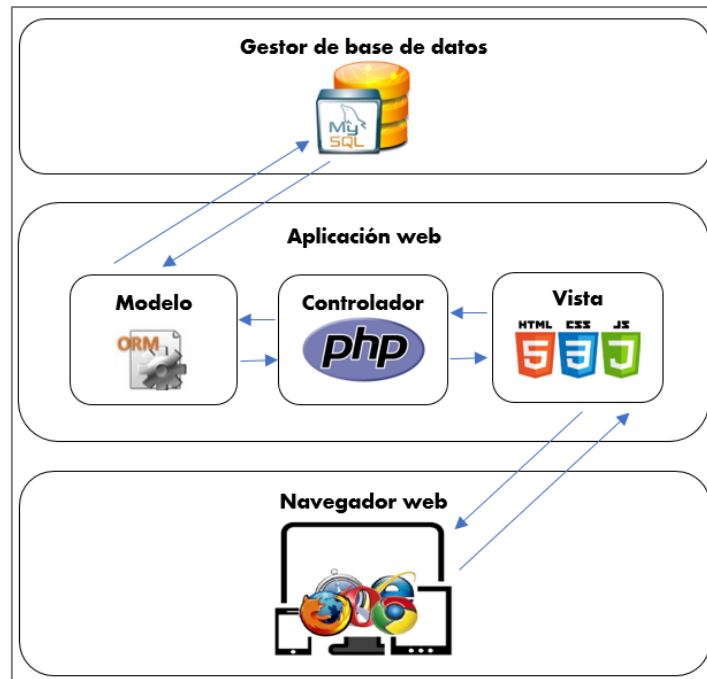


Figura 5: Arquitectura de la solución

## 2.2 Planificación

### 2.2.1 Personal involucrado

La metodología XP enfatiza la importancia de la comunicación constante entre el cliente y el programador, priorizando las entrevistas cara a cara y sin usar un lenguaje técnico, para obtener una información más precisa de las funciones que debe tener el producto final. Además, se debe tomar en cuenta la opinión de varios usuarios finales que conozcan bien los objetivos del negocio a fin de despejar las dudas que puedan presentarse en la definición de las necesidades que debe cubrir el sistema.

El personal involucrado detalla los requisitos relativos al sistema académico que determinan su funcionalidad, estableciendo las necesidades y condicionantes generales que se tomarán en cuenta durante el desarrollo del software, las mismas que están sujetas a pequeñas variaciones en el transcurso del desarrollo de acuerdo con la metodología de programación elegida.

En varias entrevistas llevadas a cabo con autoridades y docentes de la Unidad Educativa piloto, se obtuvo los requisitos que debe cumplir el Sistema Académico Web Bilingüe en forma de historias de usuario. El personal involucrado en la recopilación de información para determinar las historias de usuario y sus respectivos roles en el desarrollo del proyecto se detalla a continuación:

TABLA 2.1  
EQUIPO DE DESARROLLO DEL PROYECTO

Nombre	Chicaiza Lechón María Soraya
Rol	Cliente
Categoría profesional	Licenciada en Ciencias de la Educación Mención Gerencia Educativa
Responsabilidades	Definir las historias de usuario y validar las funcionalidades del proyecto.
Información de contacto	Otavalo, Telf.: 0967070292
Correo electrónico	sory.chicaiza@hotmail.com
Nombre	Boonter Segundo Cabascango Chiza
Rol	Programador
Categoría profesional	Egresado en Ingeniería en sistemas computacionales
Responsabilidades	Estimar el esfuerzo necesario para el desarrollo de las historias de usuario y desarrollar el código de la aplicación.
Información de contacto	Otavalo, Telf.: 0981503912
Correo electrónico	booncabas@gmail.com
Nombre	Santillán Chiza Ana Cristina
Rol	Encargada de pruebas (Tester)
Categoría profesional	Ingeniera Industrial
Responsabilidades	Validar las funcionalidades del proyecto
Información de contacto	Otavalo, Telf.: 0993895764
Correo electrónico	anylucero_14@hotmail.com
Nombre	Torres Guerra Diana María Alexandra
Rol	Encargada de pruebas (Tester)
Categoría profesional	Bachiller en secretariado
Responsabilidades	Validar las funcionalidades del proyecto
Información de contacto	Otavalo, Telf.: 0995280777
Correo electrónico	dianytg2010@yahoo.com
Nombre	Campoverde Ríos Lupe Esmeralda
Rol	Encargada de pruebas (Tester)
Categoría profesional	Licenciada en Educación Parvulario
Responsabilidades	Validar las funcionalidades del proyecto
Información de contacto	Otavalo, Telf.: 0993895764
Correo electrónico	lupitacampoverde@gmail.com

## 2.2.2 Historias de usuario

Las historias de usuario se realizaron tomando en cuenta las necesidades expresadas por el cliente. Estas historias de usuario detallan los requisitos que debe cumplir el sistema académico.

TABLA 2.2  
HISTORIA DE USUARIO – REGISTRAR USUARIOS  
HISTORIA DE USUARIO

Número: 1	Usuario: Administrador
Nombre de la historia: Registrar usuarios	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.4	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
El sistema debe permitir registrar los usuarios del sistema con 4 tipos de roles de usuario que son:	
<ul style="list-style-type: none"> <li>• Administrador</li> </ul>	

- Docente
- Estudiante
- Representante

Observaciones: Los usuarios deben crearse en base a los registros de administradores, docentes, estudiantes y representantes que estén registrados en la institución.

TABLA 2.3  
HISTORIA DE USUARIO - INGRESO AL SISTEMA  
HISTORIA DE USUARIO

Número: 2	Usuario: Administrador, Docente, Estudiante, Representante
Nombre de la historia: Ingreso al sistema	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.4	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: Los usuarios ingresarán al sistema con su nombre de usuario y su contraseña. El nombre de usuario será su número de cédula o sus nombres completos en el caso de que no tenga cédula. Su contraseña por defecto será su número de cédula o nombres completos y podrá ser cambiado posteriormente por el usuario.	
Observaciones: Las contraseñas se guardarán encriptadas y una vez cambiadas por el usuario, el administrador no sabrá cuáles son.	

TABLA 2.4  
HISTORIA DE USUARIO – PERFIL DE USUARIO  
HISTORIA DE USUARIO

Número: 3	Usuario: Administrador, Docente, Estudiante, Representante
Nombre de la historia: Perfil de usuario	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.4	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: Las funciones permitidas para los usuarios se basarán en 3 tipos de acceso que son: Lectura. Solo podrá acceder a la aplicación en modo lectura, es decir, solo podrá visualizar información o informes predefinidos ya ejecutados (Estudiantes y Representantes). Ejecución. Además de tener permisos de Lectura, podrá acceder a la aplicación para ejecutar funciones y visualizar informes más especializados (Docentes). Administración. Además de poseer los permisos del usuario de Ejecución, tendrá acceso a la parte de administración y configuración del sistema académico (Administradores).	
Observaciones: La función de cambiar contraseña será la única función que los usuarios con perfil de lectura podrán ejecutar.	

TABLA 2.5  
HISTORIA DE USUARIO – ACCESO A LA INFORMACIÓN  
HISTORIA DE USUARIO

Número: 4	Usuario: Administrador, Docente, Estudiante, Representante
Nombre de la historia: Acceso a la información	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: Los permisos de acceso a la información estarán definidos por los tipos de usuarios del sistema. Clasificación de los usuarios:	



Estudiante: Tendrá acceso a sus reportes de calificaciones y a notificaciones publicadas por sus docentes.

Representante: Tendrá acceso a los reportes de calificaciones y a notificaciones publicadas por los docentes de sus representados.

Docente: Tendrá acceso a las calificaciones de los estudiantes de los cursos a los que está asociado. Podrá ver, ingresar, editar y eliminar sus calificaciones. En caso de que el docente sea tutor de un curso, podrá ver el desempeño de los estudiantes de este curso en todas sus asignaturas. Además, el tutor será el encargado del ingreso de proyectos escolares, asistencias y comportamientos de los estudiantes de su curso de tutoría.

Administrador: Tiene el privilegio de acceder a toda la información del sistema.

Observaciones: El administrador podrá activar y desactivar el acceso al ingreso de calificaciones

TABLA 2.6  
HISTORIA DE USUARIO - REGISTRAR DOCENTES

HISTORIA DE USUARIO	
Número: 5	Usuario: Administrador
Nombre de la historia: Registrar docentes	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.2	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información del docente con sus respectivas validaciones: nombres, apellidos, cédula, título profesional, dirección, teléfono, celular, correo electrónico, fecha de ingreso y observaciones.	
Observaciones: Se desarrollará una función que permita ingresar cédulas de identificación que no cumplan la validación de cédula ecuatoriana para docentes extranjeros.	

TABLA 2.7  
HISTORIA DE USUARIO - REGISTRAR ESTUDIANTES

HISTORIA DE USUARIO	
Número: 6	Usuario: Administrador
Nombre de la historia: Registrar estudiantes	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información del estudiante con sus respectivas validaciones: nombres, apellidos, cédula, fecha de nacimiento, sexo, etnia, plantel de procedencia, problemas de salud y observaciones.	
Observaciones: El sistema debe permitir ingresar estudiantes sin cédula de identificación.	

TABLA 2.8  
HISTORIA DE USUARIO - REGISTRAR REPRESENTANTES

HISTORIA DE USUARIO	
Número: 7	Usuario: Administrador
Nombre de la historia: Registrar representantes	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información del representante con sus respectivas validaciones: nombres, apellidos, cédula, dirección, teléfono, celular, correo electrónico, ocupación y observaciones.	
Observaciones: Se desarrollará una función que permita ingresar cédulas de identificación que no cumplan la validación de cédula ecuatoriana para representantes extranjeros.	

TABLA 2.9  
HISTORIA DE USUARIO - REGISTRAR ADMINISTRADORES

HISTORIA DE USUARIO	
Número: 8	Usuario: Administrador
Nombre de la historia: Registrar administradores	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.2	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información del administrador con sus respectivas validaciones: nombres, apellidos, cédula, título profesional, dirección, teléfono, celular, correo electrónico, observaciones.	
Observaciones: La validación de cédula de identidad ecuatoriana será obligatoria.	

TABLA 2.10  
HISTORIA DE USUARIO – REGISTRAR CURSOS

HISTORIA DE USUARIO	
Número: 9	Usuario: Administrador
Nombre de la historia: Registrar cursos	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información del curso con sus respectivas validaciones: nombre, paralelo, jornada y nivel.	
Observaciones: Los niveles de inicial 1 e inicial 2 se considerarán como nivel 0.	

TABLA 2.11  
HISTORIA DE USUARIO – REGISTRAR ASIGNATURAS

HISTORIA DE USUARIO	
Número: 10	Usuario: Administrador
Nombre de la historia: Registrar asignaturas	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.2	Iteración asignada: 1
Programador responsable: Boonter Cabascango	
Descripción: El sistema tendrá un formulario que permita el ingreso de los siguientes campos de información de la asignatura con sus respectivas validaciones: nombre, área, nivel.	
Observaciones: El campo nivel solo servirá para ordenar las listas de asignaturas al momento de realizar las asignaciones a los cursos.	

TABLA 2.12  
HISTORIA DE USUARIO - MATRICULAR ESTUDIANTE

HISTORIA DE USUARIO	
Número: 11	Usuario: Administrador
Nombre de la historia: Matricular estudiante	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 1	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir matricular estudiantes validando que un estudiante no pueda matricularse en más de un curso.	
Observaciones: Se debe permitir la anulación de la matrícula o cambio de curso.	

TABLA 2.13  
HISTORIA DE USUARIO – ASIGNAR ASIGNATURAS A CURSOS

HISTORIA DE USUARIO	
Número: 12	Usuario: Administrador
Nombre de la historia: Asignar asignaturas a cursos	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir la creación de asignaturas y cursos. Estas asignaturas se asignarán a los cursos cada año.	
Observaciones: Una materia no puede asignarse dos veces a un curso.	

TABLA 2.14  
HISTORIA DE USUARIO – ASIGNAR CARGA HORARIA A DOCENTES

HISTORIA DE USUARIO	
Número: 13	Usuario: Administrador
Nombre de la historia: Asignar carga horaria a docentes	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.4	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir la asignación de las asignaturas que dictarán los docentes en los diversos cursos.	
Observaciones: No se puede asignar una asignatura en un curso dos veces a un docente.	

TABLA 2.15  
HISTORIA DE USUARIO – ASIGNAR TUTORES A CURSOS

HISTORIA DE USUARIO	
Número: 14	Usuario: Administrador
Nombre de la historia: Asignar tutores a cursos	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir la asignación de un tutor por curso. El tutor será el encargado del ingreso de calificaciones de proyectos educativos, asistencias y comportamientos.	
Observaciones: Un curso solo puede tener un tutor.	

TABLA 2.16  
HISTORIA DE USUARIO - REGISTRAR NOTAS PARCIALES

HISTORIA DE USUARIO	
Número: 15	Usuario: Administrador, Docente
Nombre de la historia: Registrar notas parciales	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 1	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir el ingreso de tres notas parciales cada quimestre. El promedio de las tres notas representará el 80% de la nota quimestral. Las notas serán sobre 10 puntos.	
Observaciones: El sistema no debe realizar aproximaciones.	

TABLA 2.17  
HISTORIA DE USUARIO - REGISTRAR NOTAS DE EXÁMENES

HISTORIA DE USUARIO	
Número: 16	Usuario: Administrador, Docente
Nombre de la historia: Registrar exámenes	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 1	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema debe permitir el ingreso de una nota de examen cada quimestre. La nota del examen representará un 20% de la nota quimestral. Los exámenes especiales para los estudiantes que no aprueben el curso en una o más asignaturas serán: Examen Supletorio, Examen Remedial y Examen de Gracia. Las notas de exámenes serán sobre 10 puntos. La validación de ingreso de notas de exámenes especiales será de acuerdo con lo que indica el instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación.	
Observaciones: El sistema no debe realizar aproximaciones.	

TABLA 2.18  
HISTORIA DE USUARIO - REGISTRAR ASISTENCIA A CLASES

HISTORIA DE USUARIO	
Número: 17	Usuario: Administrador, Docente
Nombre de la historia: Registrar asistencia a clase	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema permitirá el registro de asistencias a clase de acuerdo con los parámetros que el administrador del sistema establezca para el año lectivo en curso. Las calificaciones serán números enteros. El registro de asistencias a clase estará a cargo de los tutores de curso.	
Observaciones: La calificación será mayor o igual a cero.	

TABLA 2.19  
HISTORIA DE USUARIO - REGISTRAR COMPORTAMIENTO

HISTORIA DE USUARIO	
Número: 18	Usuario: Administrador, Docente
Nombre de la historia: Registrar comportamientos	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.3	Iteración asignada: 2
Programador responsable: Boonter Cabascango	
Descripción: El sistema permitirá el registro de comportamientos una vez cada parcial y una vez cada final de quimestre. La calificación será cualitativa usando las letras A, B, C, D, E. El registro de comportamiento estará a cargo de los tutores de curso.	
Observaciones: El sistema debe permitir la opción de calificación nulo o vacío como opción no calificado.	

TABLA 2.20  
HISTORIA DE USUARIO - REGISTRAR NOTAS DE PROYECTOS EDUCATIVOS

HISTORIA DE USUARIO	
Número: 19	Usuario: Administrador, Docente
Nombre de la historia: Registrar notas de proyectos educativos	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 0.4	Iteración asignada: 2
Programador responsable: Boonter Cabascango	

Descripción: El sistema permitirá el registro de calificación de proyectos educativos una vez cada parcial y una vez cada final de quimestre. La calificación será cualitativa usando las calificaciones: EX (Excelente), MB (Muy buena), B (Buena), R (Regular)  
 El registro de calificaciones de proyectos educativos estará a cargo de los tutores de curso.  
 Observaciones: El sistema debe permitir la opción de calificación nulo o vacío como opción no calificado.

TABLA 2.21  
 HISTORIA DE USUARIO – ADMINISTRACIÓN DE DOCUMENTOS  
 HISTORIA DE USUARIO

Número: 20	Usuario: Administrador, Docente
Nombre de la historia: Administración de documentos	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 1	Iteración asignada: 3
Programador responsable: Boonter Cabascango	
Descripción: El sistema permitirá a los docentes subir documentos como: planificaciones curriculares, planificaciones extracurriculares entre otros. Cada docente tendrá su carpeta de documentos y podrán ser revisados por los administradores.	
Observaciones: Un docente no podrá ver los documentos de otro docente.	

TABLA 2.2  
 HISTORIA DE USUARIO - PUBLICACIÓN DE NOTIFICACIONES  
 HISTORIA DE USUARIO

Número: 21	Usuario: Administrador, Docente
Nombre de la historia: Publicación de notificaciones	
Prioridad en el negocio: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input type="checkbox"/> Alto <input checked="" type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 1	Iteración asignada: 3
Programador responsable: Boonter Cabascango	
Descripción: El sistema permitirá la publicación de notificaciones para docentes, estudiantes y representantes. Estas publicaciones serán visibles de acuerdo con sus perfiles de usuario.	
Observaciones: Las notificaciones pueden dirigirse solo a cursos específicos.	

TABLA 2.23  
 HISTORIA DE USUARIO - GENERACIÓN DE REPORTE  
 HISTORIA DE USUARIO

Número: 22	Usuario: Administrador, Docente
Nombre de la historia: Generación de reportes	
Prioridad en el negocio: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Riesgo en desarrollo: <input checked="" type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo
Esfuerzo estimado: 2	Iteración asignada: 3
Programador responsable: Boonter Cabascango	
Descripción: El sistema generará los siguientes reportes:	
<ul style="list-style-type: none"> <li>• Matrícula</li> <li>• Notas parciales</li> <li>• Notas quimestrales</li> <li>• Reporte final de calificaciones</li> <li>• Promoción de curso</li> <li>• Resumen de calificaciones</li> </ul>	
Observaciones: Los reportes se generarán en PDF y Excel.	

### 2.2.3 Características de los usuarios

Las principales características de los usuarios del sistema académico se detallan a continuación:

TABLA 2.24  
CARACTERÍSTICAS DE LOS USUARIOS

Tipo de usuario	Administrador del sistema. Rector.
Formación	Ciencias de la Educación
Habilidades	Comunicación y vinculación con entornos educativos. Manejo de grupos de personas.
Actividades	Administrar, supervisar y aprobar las actividades docentes. Ingreso y modificación de datos. Asignaciones académicas.
Tipo de usuario	Administrador del sistema. Secretaria
Formación	Secretariado.
Habilidades	Manejo de sistemas informáticos
Actividades	Administrar, supervisar las actividades docentes. Ingreso y modificación de datos. Asignaciones académicas.
Tipo de usuario	Docente
Formación	Estudios de tercer nivel.
Habilidades	Conocimiento de informática, computación y acceso a Internet
Actividades	Ingreso y modificación de calificaciones, publicación de comunicados, publicación de documentos.
Tipo de usuario	Representante
Formación	Ninguna
Habilidades	Conocimiento de computación y acceso a Internet
Actividades	Revisión de calificaciones de sus representados, lectura de comunicados, lectura de información de contacto de los docentes de sus representados.
Tipo de usuario	Estudiante
Formación	Estudiante
Habilidades	Conocimiento de computación y acceso a Internet
Actividades	Revisión de sus calificaciones, lectura de comunicados, lectura de información de contacto de sus docentes.

### 2.2.4 Restricciones

El sistema académico web bilingüe, auspiciado por el GAD Municipal de Otavalo para brindar un servicio gratuito a las instituciones educativas que no cuenten con los recursos económicos para comprar un sistema hecho a la medida, requiere que el desarrollo del proyecto sea con software libre, que no implique ningún tipo de costos de licencias o pagos por uso de servicio para su funcionamiento. Por esta razón, el desarrollo del proyecto se limita al uso de lenguaje de programación PHP, base de datos MySQL/MariaDB, servidor de aplicaciones Apache, los frameworks CakePHP, Zurb Foundation y las herramientas FPDF y PHPEXcel.

## 2.2.5 Requisitos Funcionales

- **Aplicación**

Requerimientos generales de aplicación para el sistema académico.

TABLA 2.25  
REQUISITOS FUNCIONALES: APLICACIÓN

Número de requisito	RF.APL.01
Nombre de requisito	Perfil de acceso
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El perfil de acceso de los usuarios será de acuerdo con su rol de usuario.
Prioridad del requisito	Alta
Número de requisito	RF.APL.02
Nombre de requisito	Límite de acceso al usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	Existirá 3 tipos de usuarios: Lectura. Solo podrá acceder a la aplicación en modo lectura, es decir, solo podrá visualizar informes predefinidos ya ejecutados. Ejecución. Además de tener permisos de lectura, podrá acceder a la aplicación para ejecutar funciones y visualizar informes más especializados. Administración. Además de poseer los permisos del usuario de ejecución, tendrá acceso a la parte de administración y configuración del sistema académico.
Prioridad del requisito	Alta
Número de requisito	RF.APL.03
Nombre de requisito	Tipos de usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema tendrá 4 tipos de usuarios de acuerdo con su rol: <ul style="list-style-type: none"> <li>• Administradores</li> <li>• Docentes</li> <li>• Estudiantes</li> <li>• Representantes</li> </ul>
Prioridad del requisito	Alta
Número de requisito	RF.APL.04
Nombre de requisito	Permiso de acceso a la información
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	Los permisos de acceso a la información estarán definidos por los tipos de usuarios del sistema. Clasificación de los usuarios: Estudiante: Tendrá acceso a sus reportes de calificaciones y a notificaciones publicadas por sus docentes. Representante: Tendrá acceso a los reportes de calificaciones y a notificaciones publicadas por los docentes de sus representados. Docente: Tendrá acceso a las calificaciones de los estudiantes de los cursos a los que está asociado. Podrá ver, ingresar, editar y eliminar sus calificaciones. En caso de que el docente sea tutor de un curso, podrá ver el desempeño de los estudiantes de este curso en todas sus asignaturas. Además, el tutor será el encargado del ingreso y edición de asistencia, comportamiento y proyecto educativo de los estudiantes de su curso de tutoría. Administrador: Tiene el privilegio de acceder a toda la información del sistema.
Prioridad del requisito	Alta

- **Entrada de datos**

Requerimientos del sistema para el ingreso, la validación y la consolidación de datos.

TABLA 2.26  
REQUISITOS FUNCIONALES: ENTRADA DE DATOS

Número de requisito	RF.ENT.01
Nombre de requisito	Ingreso de la información
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema académico debe ser capaz de recibir, validar, cargar y consolidar los datos de docentes, estudiantes, representantes, asignaturas, cursos, calificaciones, asistencias y comportamientos registrados por los usuarios del sistema.
Prioridad del requisito	Alta
Número de requisito	RF.ENT.02
Nombre de requisito	Ingreso de calificaciones
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema académico debe ser capaz de recibir, validar, cargar y consolidar las calificaciones de acuerdo con el tipo de calificación. Notas cuantitativas: mínimo 0 y máximo 10 incluyendo dos decimales sin redondeo. Notas cualitativas: Solo las letras respectivas.
Prioridad del requisito	Alta
Número de requisito	RF.ENT.03
Nombre de requisito	Cédula de identidad no obligatoria para el registro de nuevos estudiantes.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Detalle del requisito	En el ingreso de nuevos estudiantes al sistema académico, el número de cédula de ciudadanía será opcional, al igual que en la plataforma del Ministerio de Educación. Esto se debe a que muchos alumnos son de otros países y sus cédulas de identificación incluyen letras y puntos.
Prioridad del requisito	Alta
Número de requisito	RF.ENT.04
Nombre de requisito	Campos de información de las entidades
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema permitirá ingresar los siguientes campos de información en las diferentes entidades: Estudiante: nombres, apellidos, cédula, dirección, lugar de nacimiento, fecha de nacimiento, sexo, etnia, nacionalidad, plantel de procedencia, problemas de salud, observaciones. Representante: nombres, apellidos, cédula, dirección, teléfono, celular, ocupación, correo electrónico, observaciones. Docente: nombres, apellidos, cédula, título profesional, fecha de ingreso, dirección, teléfono, celular, correo electrónico, observaciones. Administrador: nombres, apellidos, cédula, título profesional, fecha de ingreso, dirección, teléfono, celular, correo electrónico, observaciones.
Prioridad del requisito	Alta



## 2.2.6 Requisitos no funcionales

- **Arquitectura**

Requisitos necesarios para implementar el sistema académico.

TABLA 2.27  
REQUISITOS NO FUNCIONALES - ARQUITECTURA

Número de requisito	RF.ARQ.01
Nombre de requisito	Requisitos de hardware
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	Disco duro de 320 GB o superior. Procesador Pentium IV o superior. Memoria RAM de 1GB o superior. Monitor, mouse, teclado.
Prioridad del requisito	Media
Número de requisito	RF.ARQ.02
Nombre de requisito	Requisitos de software
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	Sistema operativo Windows o Linux compatible con el servidor de aplicaciones apache 2.4.x, con versión de lenguaje PHP 5.6.x y requiere el sistema de base de datos MySQL 5.6.x o MariaDB 10.1.x
Prioridad del requisito	Alta

- **Usabilidad**

Requisitos del software para que éste sea entendido, aprendido, operado y usable.

TABLA 2.28  
REQUISITOS NO FUNCIONALES: ARQUITECTURA

Número de requisito	RF.USA.01
Nombre de requisito	Acceso al sistema
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema académico será accesible a través de cualquier navegador de Internet estándar.
Prioridad del requisito	Alta
Número de requisito	RF.USA.02
Nombre de requisito	Estandarización a las normativas del ministerio de educación
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema académico debe ser útil a cualquier unidad educativa.
Prioridad del requisito	Media
Número de requisito	RF.USA.03
Nombre de requisito	Bilingüe
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema debe ser de fácil comprensión para kichwahablantes e hispanohablantes
Prioridad del requisito	Media

- **Seguridad**

Requerimientos del sistema para proteger información y datos, de manera que las personas no autorizadas no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas autorizadas.

TABLA 2.29  
REQUISITOS NO FUNCIONALES: SEGURIDAD

Número de requisito	RF.SEG.01
Nombre de requisito	Seguridad de ingreso al sistema
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	Para ingresar al sistema académico los usuarios utilizarán un usuario y una contraseña que será administrada por el usuario. Las contraseñas se almacenarán encriptadas.
Prioridad del requisito	Alta
Número de requisito	RF.SEG.02
Nombre de requisito	Seguridad de la base de datos
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	La seguridad de la base de datos estará bajo la responsabilidad del Director de Tecnología e Informática de la institución.
Prioridad del requisito	Alta
Número de requisito	RF.SEG.03
Nombre de requisito	Auditoria de calificaciones
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema debe registrar el usuario que realiza las modificaciones de las calificaciones
Prioridad del requisito	Alta
Número de requisito	RF.SEG.04
Nombre de requisito	Caducidad de sesión
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema debe cerrar automáticamente las sesiones de usuario que no muestren actividad en un tiempo determinado
Prioridad del requisito	Alta

- **Mantenibilidad**

Requisitos del software para realizar cambios de configuración o modificaciones.

TABLA 2.30  
REQUISITOS NO FUNCIONALES: MANTENIBILIDAD

Número de requisito	RF.MAN.01
Nombre de requisito	Configuración
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema debe ser configurable para adaptarse a los requerimientos de una unidad educativa bilingüe y una unidad educativa estándar. La configuración del sistema estará a cargo de los administradores del sistema bajo la dirección del Rector de la institución.
Prioridad del requisito	Media
Número de requisito	RF.MAN.02
Nombre de requisito	Modularidad
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El sistema debe ser modular ante la posibilidad de agregar nuevas funciones a futuro.
Prioridad del requisito	Media

- **Portabilidad**

Requisitos para que el sistema académico debe mantener para facilitar su traslado a otras plataformas u entornos.

TABLA 2.31  
REQUISITOS NO FUNCIONALES: PORTABILIDAD

Número de requisito	RF.MAN.01
Nombre de requisito	Cambios de entorno
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Detalle del requisito	El 100% de los componentes del sistema académico depende de las especificaciones del tipo de servidor y no de la plataforma. Al ser un sistema desarrollado en lenguaje de programación PHP con base de datos MySQL/MariaDB es un requisito fundamental seguir manteniendo el mismo tipo de servidor de aplicaciones.

### 2.3 Parámetros configurables para el sistema multi institución

Con el propósito de brindar un servicio independiente que se ajuste a las necesidades variantes de cada unidad educativa, el sistema académico bilingüe tendrá como base permitir a cada institución educativa configurar los parámetros de:

- Información de presentación de la institución
- Logos
- Tipos de calificaciones cualitativas de asignaturas y sus equivalencias
- Rangos y equivalencias de calificaciones cuantitativas de asignaturas y sus equivalencias
- Tipos de calificaciones cualitativas de proyectos educativos y sus equivalencias
- Tipos de calificaciones cualitativas de comportamiento y sus equivalencias
- Parámetros para evaluar en proyectos educativos
- Parámetros para evaluar en comportamiento
- Parámetros para evaluar en asistencia
- Nombres de ítems académicos
- Reportes de matrícula
- Reportes de promoción
- Reportes de calificaciones de estudiante
- Reportes de resúmenes de calificaciones de perfil docente

### 2.3.1 Información de presentación de la institución

Para la configuración de la información de presentación de la unidad educativa se creará la entidad “ues” que tendrá los siguientes campos de configuración de información:

- **Nombre:** Campo obligatorio que guardará el nombre de la unidad educativa. El nombre de la institución estará presente en las vistas del sistema y en los encabezados de los reportes.
- **Lema:** Campo opcional. Guardará el lema de la institución educativa y aparecerá debajo del nombre de la institución en los reportes del sistema.
- **Teléfono:** Campo opcional. Guardará el teléfono de la institución educativa y aparecerá debajo del lema de la institución en los reportes del sistema.
- **Dirección:** Campo opcional. Guardará la dirección de la institución educativa y aparecerá junto con el teléfono de la institución en los reportes del sistema.
- **Email:** Campo opcional. Guardará la dirección de correo electrónico de la institución educativa y aparecerá debajo del teléfono de la institución en los reportes del sistema.
- **Página web:** Campo opcional. Guardará la dirección de la página web de la institución educativa y aparecerá junto con el correo electrónico de la institución en los reportes del sistema.
- **Lugar:** Campo opcional. Guardará la información de ubicación de la institución educativa y aparecerá debajo del correo electrónico de la institución en los reportes del sistema.
- **Registro:** Campo opcional. Guardará información técnica como el distrito educativo al que pertenece la institución educativa. Aparecerá debajo del lugar de la institución en los reportes del sistema.

Para cada unidad educativa se generará un registro en esta entidad al momento de su creación.

### 2.3.2 Logos

Para la configuración de los logos de la unidad educativa se creará la entidad “logos” que tendrá los siguientes campos de configuración de información:

- **Nombre:** Campo obligatorio administrado por el sistema. Guardará el nombre de la imagen.

- **Logo:** Campo opcional. Guardará el nombre del archivo de la imagen que aparecerá en los reportes del sistema.
- **Logo\_dir:** Campo obligatorio administrado por el sistema. Guardará la ruta del archivo.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.

Para cada unidad educativa se generará dos registros en esta entidad al momento de su creación, un registro para el logo de la institución y otro opcional para el logo del ministerio de educación.

### 2.3.3 Tipos de calificaciones cualitativas de asignaturas y sus equivalencias

Las calificaciones cualitativas se aplican a los primeros años de educación. Para la configuración de los tipos de calificaciones cualitativas de asignaturas y sus equivalencias se creará la entidad “escalacuals” que tendrá los siguientes campos de configuración de información:

- **Abr1:** Campo opcional. Representa la opción 1 de calificación cualitativa.
- **Nombre1:** Campo opcional. Representa el significado de la abreviatura 1.
- **Descri1:** Campo opcional. Guarda la equivalencia del nombre 1.
- **Abr2:** Campo opcional. Representa la opción 2 de calificación cualitativa.
- **Nombre2:** Campo opcional. Representa el significado de la abreviatura 2.
- **Descri2:** Campo opcional. Guarda la equivalencia del nombre 2.
- **Abr3:** Campo opcional. Representa la opción 3 de calificación cualitativa.
- **Nombre3:** Campo opcional. Representa el significado de la abreviatura 3.
- **Descri3:** Campo opcional. Guarda la equivalencia del nombre 3.
- **Abr4:** Campo opcional. Representa la opción 4 de calificación cualitativa.
- **Nombre4:** Campo opcional. Representa el significado de la abreviatura 4.
- **Descri4:** Campo opcional. Guarda la equivalencia del nombre 4.
- **Abr5:** Campo opcional. Representa la opción 5 de calificación cualitativa.
- **Nombre5:** Campo opcional. Representa el significado de la abreviatura 5.
- **Descri5:** Campo opcional. Guarda la equivalencia del nombre 5.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

De acuerdo con el Instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación del Ecuador actualizado a 2017, un registro en esta entidad tendrá la siguiente información:

TABLA 2.32  
REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE ASIGNATURAS

Abr1	I
Nombre1	INICIO
Descri1	El niño o niña, está empezando a desarrollar los aprendizajes previstos o evidencia dificultades para el desarrollo de éstos, para lo cual necesita mayor tiempo de acompañamiento e intervención del docente, de acuerdo con su ritmo y estilo de aprendizaje.
Abr2	EP
Nombre2	EN PROCESO
Descri2	El niño o niña está en proceso para lograr los aprendizajes previstos, para lo cual requiere acompañamiento del docente y del representante legal durante el tiempo necesario.
Abr3	A
Nombre3	ADQUIRIDA
Descri3	El niño o niña evidencia el logro de los aprendizajes previstos en el tiempo programado.
Abr4	N/E
Nombre4	NO EVALUADO
Descri4	Este indicador no ha sido evaluado en el quimestre.
Abr5	NULL
Nombre5	NULL
Descri5	NULL
Ue_id	Administrado por el sistema
Anio_id	Administrado por el sistema

Junto con la creación de cada año lectivo se generará un nuevo registro de esta entidad para su configuración.

### 2.3.4 Rangos y equivalencias de calificaciones cuantitativas de asignaturas

Para la configuración de los rangos y equivalencias de calificaciones cuantitativas de asignaturas se creará la entidad “escalacuants” que tendrá los siguientes campos de configuración de información:

- **Notamax:** Campo obligatorio. Número entero que representa la nota máxima que admitirá el sistema para el ingreso de notas parciales y exámenes.
- **Notaaprueba:** Campo obligatorio. Número entero que representa la nota mínima que el estudiante debe obtener como promedio final para aprobar el periodo lectivo.

- **Eqparcial:** Campo obligatorio. Número decimal que representa el porcentaje al que equivale el promedio de las notas parciales en la nota quimestral.
- **Eqexamen:** Campo obligatorio. Número decimal que representa el porcentaje al que equivale la nota de examen en la nota quimestral.
- **Rango1a:** Campo opcional. Número decimal que representa la nota inicial del rango de evaluación 1.
- **Rango1b:** Campo opcional. Número decimal que representa la nota final del rango de evaluación 1.
- **Abr1:** Campo opcional. Abreviatura asignada a la calificación que se encuentra entre el rango 1a y el rango 1b.
- **Nombre1:** Campo opcional. Guarda la equivalencia de la abreviatura 1.
- **Rango2a:** Campo opcional. Número decimal que representa la nota inicial del rango de evaluación 2.
- **Rango2b:** Campo opcional. Número decimal que representa la nota final del rango de evaluación 2.
- **Abr2:** Campo opcional. Abreviatura asignada a la calificación que se encuentra entre el rango 2a y el rango 2b.
- **Nombre2:** Campo opcional. Guarda la equivalencia de la abreviatura 2.
- **Rango3a:** Campo opcional. Número decimal que representa la nota inicial del rango de evaluación 3.
- **Rango3b:** Campo opcional. Número decimal que representa la nota final del rango de evaluación 3.
- **Abr3:** Campo opcional. Abreviatura asignada a la calificación que se encuentra entre el rango 3a y el rango 3b.
- **Nombre3:** Campo opcional. Guarda la equivalencia de la abreviatura 3.
- **Rango4a:** Campo opcional. Número decimal que representa la nota inicial del rango de evaluación 4.
- **Rango4b:** Campo opcional. Número decimal que representa la nota final del rango de evaluación 4.
- **Abr4:** Campo opcional. Abreviatura asignada a la calificación que se encuentra entre el rango 4a y el rango 4b.
- **Nombre4:** Campo opcional. Guarda la equivalencia de la abreviatura 4.
- **Rango5a:** Campo opcional. Número decimal que representa la nota inicial del rango de evaluación 5.
- **Rango5b:** Campo opcional. Número decimal que representa la nota final del rango de evaluación 5.

- **Abr5:** Campo opcional. Abreviatura asignada a la calificación que se encuentra entre el rango 5a y el rango 5b.
- **Nombre5:** Campo opcional. Guarda la equivalencia de la abreviatura 5.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

De acuerdo con el Instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación del Ecuador actualizado a 2017, un registro en esta entidad tendrá la siguiente información:

TABLA 2.33  
REGISTRO CONFIGURABLE DE CALIFICACIONES CUANTITATIVAS DE ASIGNATURAS

Notamax	10
Notaaprueba	7
Eqparcial	80
Eqexamen	20
Rango1a	9.00
Rango1b	10.00
Abr1	DAR
Nombre1	Domina los aprendizajes requeridos.
Rango2a	7.00
Rango2b	8.99
Abr2	AAR
Nombre2	Alcanza los aprendizajes requeridos.
Rango3a	4.01
Rango3b	6.99
Abr3	PAAR
Nombre3	Está próximo a alcanzar los aprendizajes requeridos.
Rango4a	0.00
Rango4b	4.00
Abr4	NAAR
Nombre4	No alcanza los aprendizajes requeridos
Rango5a	NULL
Rango5b	NULL
Abr5	NULL
Nombre5	NULL
Ue_id	Administrado por el sistema
Anio_id	Administrado por el sistema

Junto con la creación de cada año lectivo se generará un nuevo registro de esta entidad para su configuración.



### 2.3.5 Tipos de calificaciones cualitativas de proyectos educativos y sus equivalencias

Para la configuración de los tipos de calificaciones cualitativas de proyectos educativos y sus equivalencias se creará la entidad “escalaproys” que tendrá los siguientes campos de configuración de información:

- **Abr1:** Campo opcional. Representa la opción 1 de calificación cualitativa de proyecto educativo.
- **Nombre1:** Campo opcional. Representa el significado de la abreviatura 1.
- **Descri1:** Campo opcional. Guarda la equivalencia del nombre 1.
- **Abr2:** Campo opcional. Representa la opción 2 de calificación cualitativa de proyecto educativo.
- **Nombre2:** Campo opcional. Representa el significado de la abreviatura 2.
- **Descri2:** Campo opcional. Guarda la equivalencia del nombre 2.
- **Abr3:** Campo opcional. Representa la opción 3 de calificación cualitativa de proyecto educativo.
- **Nombre3:** Campo opcional. Representa el significado de la abreviatura 3.
- **Descri3:** Campo opcional. Guarda la equivalencia del nombre 3.
- **Abr4:** Campo opcional. Representa la opción 4 de calificación cualitativa de proyecto educativo.
- **Nombre4:** Campo opcional. Representa el significado de la abreviatura 4.
- **Descri4:** Campo opcional. Guarda la equivalencia del nombre 4.
- **Abr5:** Campo opcional. Representa la opción 5 de calificación cualitativa de proyecto educativo.
- **Nombre5:** Campo opcional. Representa el significado de la abreviatura 5.
- **Descri5:** Campo opcional. Guarda la equivalencia del nombre 5.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

De acuerdo con el Instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación del Ecuador actualizado a 2017, un registro en esta entidad tendrá la siguiente información:

TABLA 2.34  
REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE PROYECTOS EDUCATIVOS

Abr1	EX
Nombre1	EXCELENTE
Descri1	Demuestra destacado desempeño en cada fase del desarrollo del proyecto escolar lo que constituye un excelente aporte a su formación integral.
Abr2	MB
Nombre2	MUY BUENA
Descri2	Demuestra fiabilidad en el desempeño para cada fase del desarrollo del proyecto escolar lo que constituye un aporte a su formación integral.
Abr3	B
Nombre3	BUENA
Descri3	Demuestra un desempeño medianamente aceptable, en cada fase del desarrollo del proyecto escolar lo que no contribuye totalmente a su formación integral.
Abr4	R
Nombre4	REGULAR
Descri4	Demuestra dificultad en atender cada fase del desarrollo del proyecto escolar lo que contribuye escasamente a su formación integral.
Abr5	NULL
Nombre5	NULL
Descri5	NULL
Ue_id	Administrado por el sistema
Anio_id	Administrado por el sistema

Anexo a la creación de cada año lectivo se generará un nuevo registro de esta entidad para su configuración.

### 2.3.6 Tipos de calificaciones cualitativas de comportamiento y sus equivalencias

Para la configuración de los tipos de calificaciones cualitativas de comportamiento y sus equivalencias se creará la entidad “escalacomps” que tendrá los siguientes campos de configuración de información:

- **Abr1:** Campo opcional. Representa la opción 1 de calificación cualitativa de comportamiento.
- **Nombre1:** Campo opcional. Representa el significado de la abreviatura 1.
- **Descri1:** Campo opcional. Guarda la equivalencia del nombre 1.
- **Abr2:** Campo opcional. Representa la opción 2 de calificación cualitativa de comportamiento.
- **Nombre2:** Campo opcional. Representa el significado de la abreviatura 2.
- **Descri2:** Campo opcional. Guarda la equivalencia del nombre 2.
- **Abr3:** Campo opcional. Representa la opción 3 de calificación cualitativa de comportamiento.
- **Nombre3:** Campo opcional. Representa el significado de la abreviatura 3.
- **Descri3:** Campo opcional. Guarda la equivalencia del nombre 3.

- **Abr4:** Campo opcional. Representa la opción 4 de calificación cualitativa de comportamiento.
- **Nombre4:** Campo opcional. Representa el significado de la abreviatura 4.
- **Descri4:** Campo opcional. Guarda la equivalencia del nombre 4.
- **Abr5:** Campo opcional. Representa la opción 5 de calificación cualitativa de comportamiento.
- **Nombre5:** Campo opcional. Representa el significado de la abreviatura 5.
- **Descri5:** Campo opcional. Guarda la equivalencia del nombre 5.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

De acuerdo con el Instructivo para la aplicación de la evaluación estudiantil del Ministerio de Educación del Ecuador actualizado a 2017, un registro en esta entidad tendrá la siguiente información:

TABLA 2.35  
REGISTRO CONFIGURABLE DE CALIFICACIONES CUALITATIVAS DE COMPORTAMIENTO

Abr1	A
Nombre1	MUY SATISFACTORIO
Descri1	Lidera el cumplimiento de los compromisos establecidos para la sana convivencia social.
Abr2	B
Nombre2	SATISFACTORIO
Descri2	Cumple con los compromisos establecidos para la sana convivencia social.
Abr3	C
Nombre3	POCO SATISFACTORIO
Descri3	Falla ocasionalmente en el cumplimiento de los compromisos establecidos para la sana convivencia social.
Abr4	D
Nombre4	MEJORABLE
Descri4	Falla reiteradamente en el cumplimiento de los compromisos establecidos para la sana convivencia social.
Abr5	E
Nombre5	INSATISFACTORIO
Descri5	No cumple con los compromisos establecidos para la sana convivencia social.
Ue_id	Administrado por el sistema
Anio_id	Administrado por el sistema

Anexo a la creación de cada año lectivo se generará un nuevo registro de esta entidad para su configuración.

### 2.3.7 Parámetros para evaluar en proyectos educativos

Para la configuración de los parámetros que se evaluarán en proyectos educativos se creará la entidad “valorproys” que tendrá los siguientes campos de configuración de información:

- **Código:** Campo obligatorio administrado por el sistema. Permite identificar el proyecto educativo.
- **Nombre:** Campo opcional. Guarda el nombre del proyecto educativo.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

Anexo a la creación de cada año lectivo se generará cinco nuevos registros de esta entidad para su configuración. Actualmente, el Ministerio de Educación registra solo una calificación de proyecto educativo en general, por lo que solo se utilizaría un registro y los cuatro restantes tendrían el campo nombre vacío.

### 2.3.8 Parámetros para evaluar en comportamiento

Para la configuración de los parámetros que se evaluarán en comportamiento se creará la entidad “valorcomps” que tendrá los siguientes campos de configuración de información:

- **Código:** Campo obligatorio administrado por el sistema. Permite identificar el parámetro de comportamiento.
- **Nombre:** Campo opcional. Guarda el nombre del comportamiento.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

Cerca de la creación de cada año lectivo se generará cinco nuevos registros de esta entidad para su configuración. Los nombres de estos registros tendrán los parámetros de comportamiento como: respeto, puntualidad, aseo, etc.

### 2.3.9 Parámetros para evaluar en asistencia

Para la configuración de los parámetros que se evaluarán en asistencia se creará la entidad “valorasis” que tendrá los siguientes campos de configuración de información:

- **Código:** Campo obligatorio administrado por el sistema. Permite identificar el parámetro de asistencia.
- **Nombre:** Campo opcional. Guarda el nombre de la asistencia.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

Con la creación de cada año lectivo se generarán cinco nuevos registros de esta entidad para su configuración. Los nombres de estos registros tendrán los parámetros de asistencia como: días asistidos, faltas, atrasos, etc.

### 2.3.10 Nombres de ítems académicos

Para la configuración de los ítems académicos se creará la entidad “confinoms” que tendrá los siguientes campos de configuración de información:

- **Nomarea:** Campo obligatorio. Guarda el nombre del título “ÁREA” que aparecerá en los reportes de promoción de año.
- **Nomasig:** Campo obligatorio. Guarda el nombre del título “ASIGNATURA” que aparecerá en los reportes de calificaciones y de promoción de año.
- **Nomasigpn:** Campo obligatorio. Guarda el nombre del título “ASIGNATURA” para los primeros niveles de educación básica que aparecerá en los reportes de calificaciones.
- **Nomproy:** Campo obligatorio. Guarda el nombre del título “PROYECTO EDUCATIVO” que aparecerá en los reportes de calificaciones y de promoción de año.
- **Nomcomp:** Campo obligatorio. Guarda el nombre del título “COMPORTAMIENTO” que aparecerá en los reportes de calificaciones y de promoción de año.
- **Nomasis:** Campo obligatorio. Guarda el nombre del título “ASISTENCIA” que aparecerá en los reportes de calificaciones.
- **nomnota1:** Campo opcional. Guarda el título del aporte 1 de la nota parcial que aparecerá en los reportes de calificaciones.

- **nomnota2:** Campo opcional. Guarda el título del aporte 2 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota3:** Campo opcional. Guarda el título del aporte 3 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota4:** Campo opcional. Guarda el título del aporte 4 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota5:** Campo opcional. Guarda el título del aporte 5 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota6:** Campo opcional. Guarda el título del aporte 6 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota7:** Campo opcional. Guarda el título del aporte 7 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomnota8:** Campo opcional. Guarda el título del aporte 8 de la nota parcial que aparecerá en los reportes de calificaciones.
- **nomexm1:** Campo obligatorio. Guarda el título del examen de mejora 1 que aparecerá en los reportes de calificaciones.
- **nomexm2:** Campo obligatorio. Guarda el título del examen de mejora 2 que aparecerá en los reportes de calificaciones.
- **nomexsu:** Campo obligatorio. Guarda el título del examen “SUPLETORIO” que aparecerá en los reportes de calificaciones.
- **Nomexre:** Campo obligatorio. Guarda el título del examen “REMEDIAL” que aparecerá en los reportes de calificaciones.
- **Nomexgr:** Campo obligatorio. Guarda el título del examen “GRACIA” que aparecerá en los reportes de calificaciones.
- **Nivelgrado:** Campo obligatorio. Número entero que determina desde qué nivel el título “GRADO” cambia a “CURSO”.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.
- **Anio\_id:** Campo obligatorio administrado por el sistema. Guarda el id del año lectivo.

Con la creación de cada año lectivo se generará un nuevo registro de esta entidad para su configuración.

### 2.3.11 Reportes de matrícula

Para la configuración de los reportes de matrícula se creará la entidad “repmatriculas” que tendrá los siguientes campos de configuración de información:

- **parte1:** Campo obligatorio. Guarda el texto introductorio del certificado de matrícula.
- **fecha:** Campo obligatorio. Guarda el lugar y fecha de expedición del certificado de matrícula.
- **nombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte.
- **cargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte.
- **nombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte.
- **cargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte.
- **firmatutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.

Para cada unidad educativa se generará un registro en esta entidad al momento de su creación.

### 2.3.12 Reportes de promoción

Para la configuración de los reportes de promoción se creará la entidad “reppromos” que tendrá los siguientes campos de configuración de información:

- **parte1:** Campo obligatorio. Guarda el texto introductorio del certificado de promoción.
- **Parte2:** Campo obligatorio. Guarda el texto final del certificado de promoción.
- **fecha:** Campo obligatorio. Guarda el lugar y fecha de expedición del certificado de promoción.
- **nombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte.
- **cargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte.

- **nombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte.
- **cargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte.
- **firmatutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.

Para cada unidad educativa se generará un registro en esta entidad al momento de su creación.

### 2.3.13 Reportes de calificaciones de estudiantes

Para la configuración de los reportes de calificaciones de estudiantes se creará la entidad “repests” que tendrá los siguientes campos de configuración de información:

- **pnombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte parcial.
- **pcargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte parcial.
- **pnombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte parcial.
- **pcargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte parcial.
- **pfirmatutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte parcial.
- **qnombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte quimestral.
- **qcargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte quimestral.
- **qnombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte quimestral.
- **qcargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte quimestral.
- **qfirmatutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte quimestral.



- **fnombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte.
- **fcargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte.
- **fnombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte final.
- **fcargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte final.
- **ffirmatutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte final.
- **nombreq1p1:** Campo obligatorio. Guarda el nombre del reporte de la parcial 1 del quimestre 1.
- **nombreq1p2:** Campo obligatorio. Guarda el nombre del reporte de la parcial 2 del quimestre 1.
- **nombreq1p3:** Campo obligatorio. Guarda el nombre del reporte de la parcial 3 del quimestre 1.
- **nombreq2p1:** Campo obligatorio. Guarda el nombre del reporte de la parcial 1 del quimestre 2
- **nombreq2p2:** Campo obligatorio. Guarda el nombre del reporte de la parcial 2 del quimestre 2.
- **nombreq2p3:** Campo obligatorio. Guarda el nombre del reporte de la parcial 3 del quimestre 2.
- **nombreq1:** Campo obligatorio. Guarda el nombre del reporte del quimestre 1.
- **nombreq2:** Campo obligatorio. Guarda el nombre del reporte del quimestre 2.
- **nombrefinal:** Campo obligatorio. Guarda el nombre del reporte final.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.

Para cada unidad educativa se generará un registro en esta entidad al momento de su creación.

#### **2.3.14 Reportes de resúmenes de calificaciones de docentes**

Para la configuración de los reportes de calificaciones de estudiantes se creará la entidad “represums” que tendrá los siguientes campos de configuración de información:

- **nombrefirma1:** Campo opcional. Guarda el nombre de la autoridad 1 que aparecerá en el reporte de resúmenes de calificaciones.
- **cargofirma1:** Campo opcional. Guarda el cargo de la autoridad 1 que aparecerá debajo de su nombre en el reporte de resúmenes de calificaciones.
- **nombrefirma2:** Campo opcional. Guarda el nombre de la autoridad 2 que aparecerá en el reporte de resúmenes de calificaciones.
- **cargofirma2:** Campo opcional. Guarda el cargo de la autoridad 2 que aparecerá debajo de su nombre en el reporte de resúmenes de calificaciones.
- **firमतutor:** Campo obligatorio. Texto “SI” o “NO” que determina si el nombre y el cargo del tutor de curso aparecerá en el reporte de resúmenes de calificaciones.
- **nombrefmat:** Campo obligatorio. Guarda el nombre del reporte de notas parciales por materia.
- **nombreqmat:** Campo obligatorio. Guarda el nombre del reporte de notas quimestrales por materia.
- **nombrefmat:** Campo obligatorio. Guarda el nombre del reporte de notas finales por materia.
- **nombrefpcur:** Campo obligatorio. Guarda el nombre del reporte de notas parciales por curso.
- **nombreqcur:** Campo obligatorio. Guarda el nombre del reporte de notas quimestrales por curso.
- **nombrefcur:** Campo obligatorio. Guarda el nombre del reporte de notas finales por curso.
- **Ue\_id:** Campo obligatorio administrado por el sistema. Guarda el id de la institución educativa.

Para cada unidad educativa se generará un registro en esta entidad al momento de su creación.

## 2.4 Diseño

### 2.4.1 Modelo Conceptual

El Modelo Conceptual (diagrama de Modelo de Dominio) sirve para demostrar cuales son los conceptos más relevantes y sus asociaciones para el sistema.

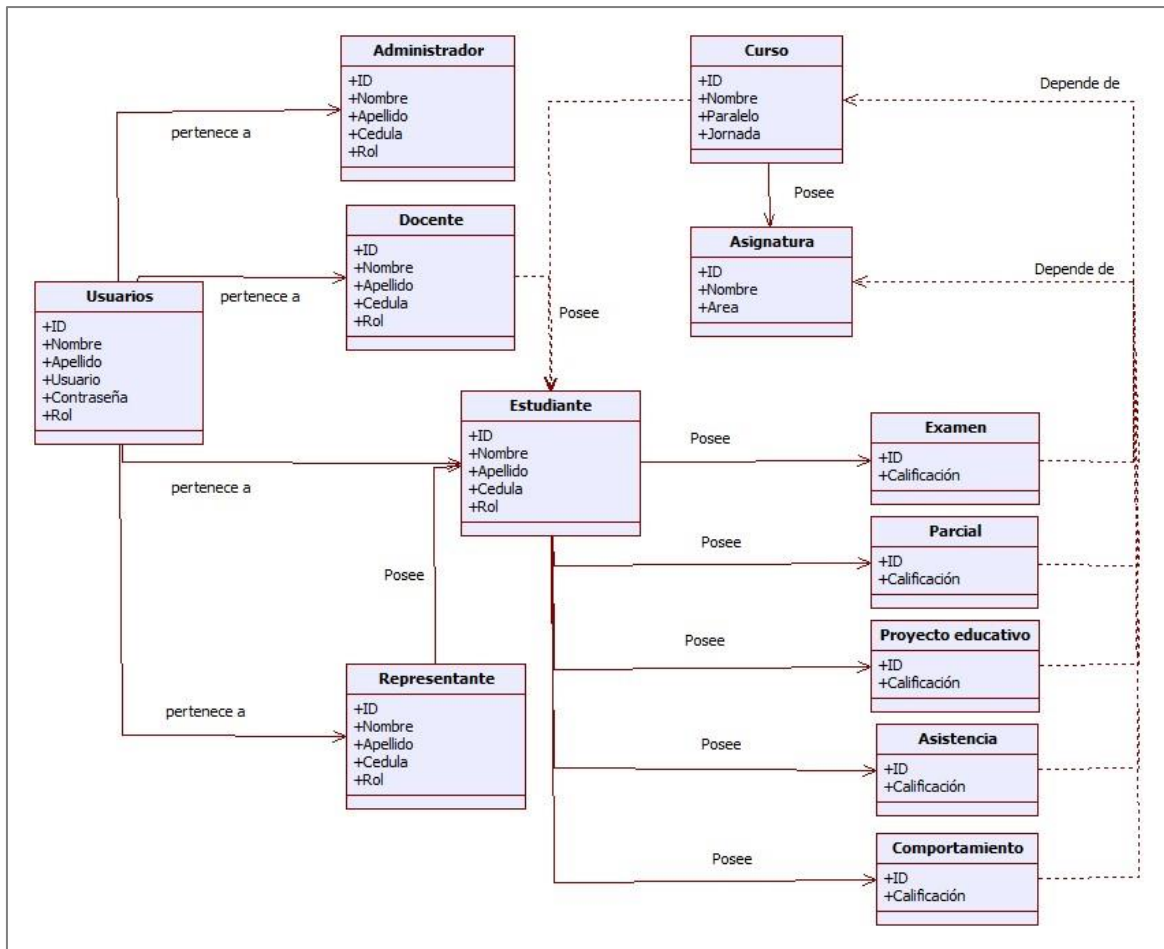


Figura 6: Modelo conceptual

### 2.4.2 Diagrama de clases

El diagrama de clases presenta en forma general la relación y la interacción que las diferentes interfaces del sistema tienen entre ellas.

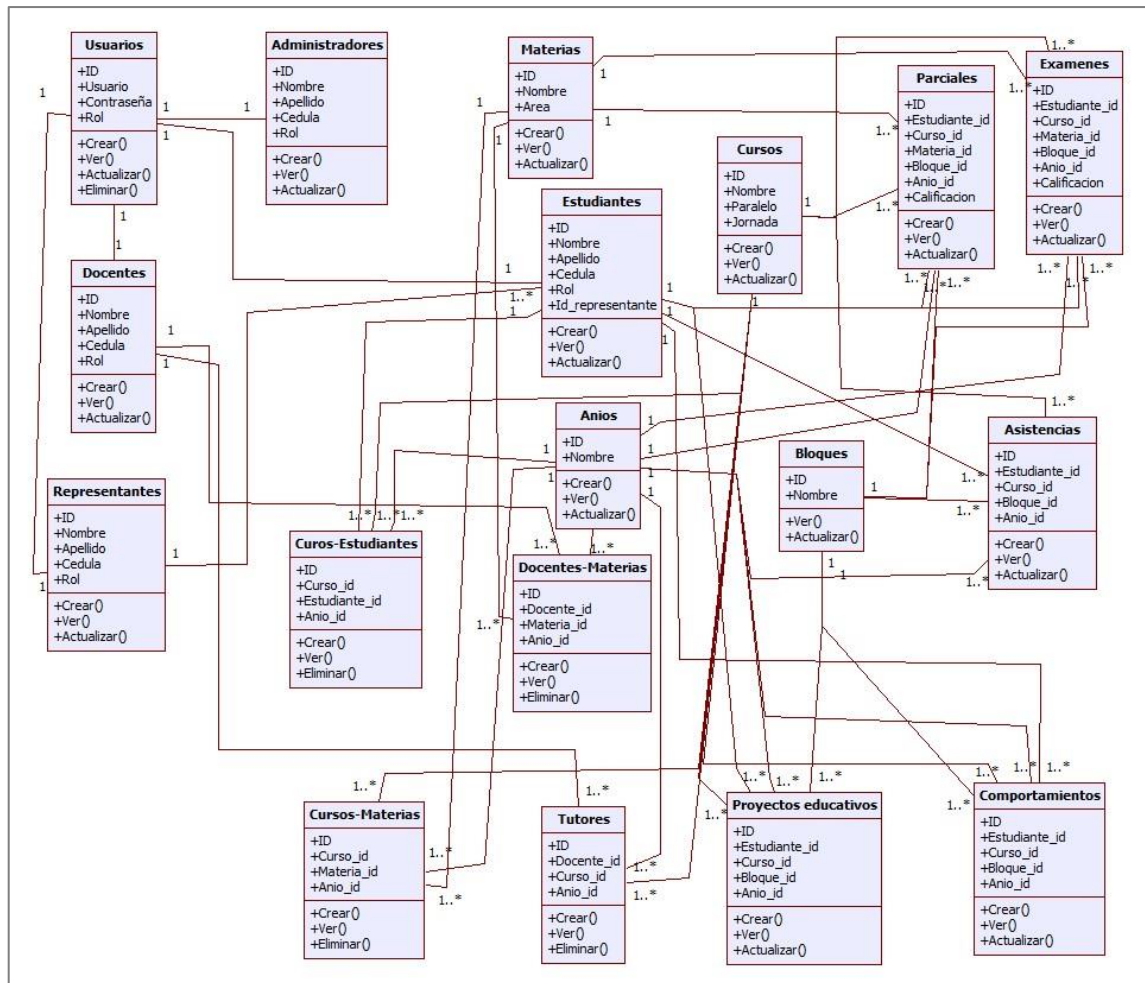


Figura 7: Diagrama de clases

### 2.4.3 Modelo Entidad Relación

El modelo entidad relación detalla la estructura de la base de datos, sus tablas, sus relaciones, los tipos de datos y las restricciones que se implementarán en el Gestor de Bases de Datos.



## 2.5 Casos de uso

La metodología XP propone el uso de las historias de usuario en lugar de los tradicionales casos de uso, sin embargo, un diagrama general por cada tipo de usuario ayuda a ver de manera resumida las tareas de ingeniería que se deben realizar para cumplir con los requerimientos especificados en las historias de usuario.

### 2.5.1 Caso de uso administrador

Las acciones generales para el tipo de usuario “Administrador” se muestran y se detallan a continuación:

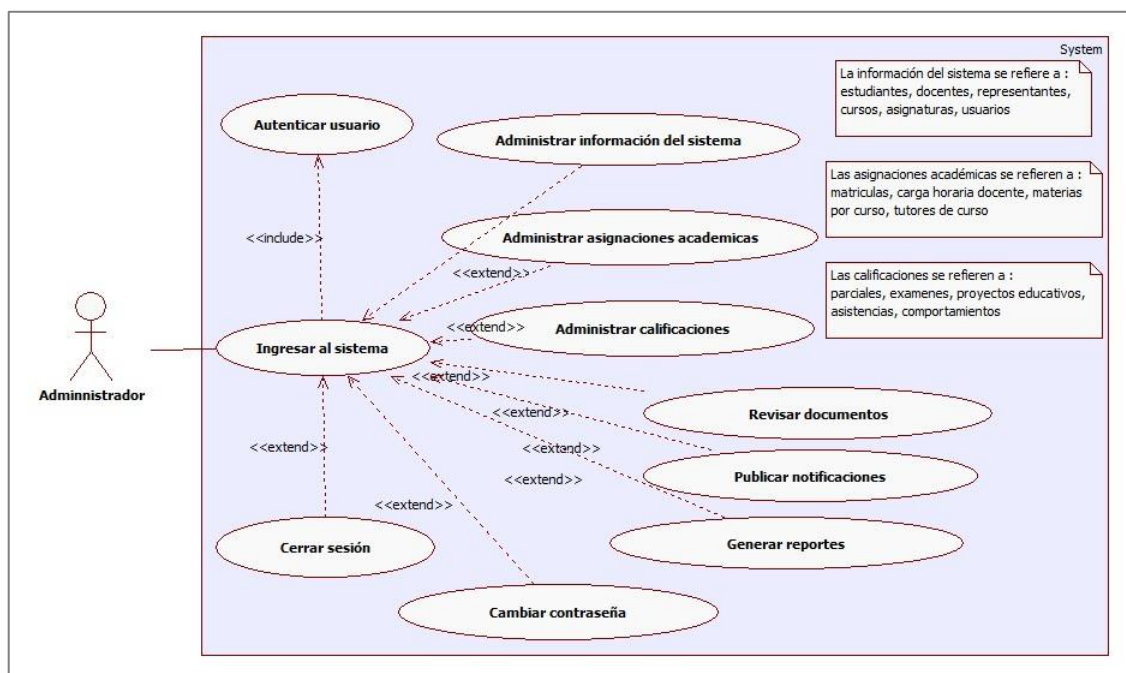


Figura 9: Caso de uso general para el tipo de usuario “Administrador”

- **Acciones principales:**

- Ingresar al sistema: Incluye la acción de autenticación como condición de ingreso.
- Administrar información el sistema: Permite al administrador ingresar, ver, editar y eliminar la información referente a estudiantes, docentes, representantes, cursos, asignaturas y usuarios del sistema.



- Administrar asignaciones académicas: Permite al administrador crear, ver, editar y eliminar matrículas, carga horaria docente, asignaturas de un curso y tutores de un curso.
- Administrar calificaciones: Permite al administrador ingresar, ver, editar y eliminar las calificaciones parciales, exámenes, proyectos educativos, asistencias y comportamientos.
- Revisar documentos: Permite al administrador ver, editar y eliminar los documentos de las carpetas de los docentes.
- Publicar notificaciones: Permite al administrador publicar notificaciones o comunicados para docentes, para todos los estudiantes o por curso, para todos los representantes o por curso.
- Generar reportes: Permite al administrador generar reportes de matrícula, promoción de año, resumen de calificaciones por estudiante y por docente.
- Cambiar contraseña: Permite al administrador cambiar su contraseña.
- Cerrar sesión: Permite al administrador salir de forma segura del sistema.

### 2.5.2 Caso de uso docente

Las acciones generales para el tipo de usuario “Docente” se muestran y se detallan a continuación:

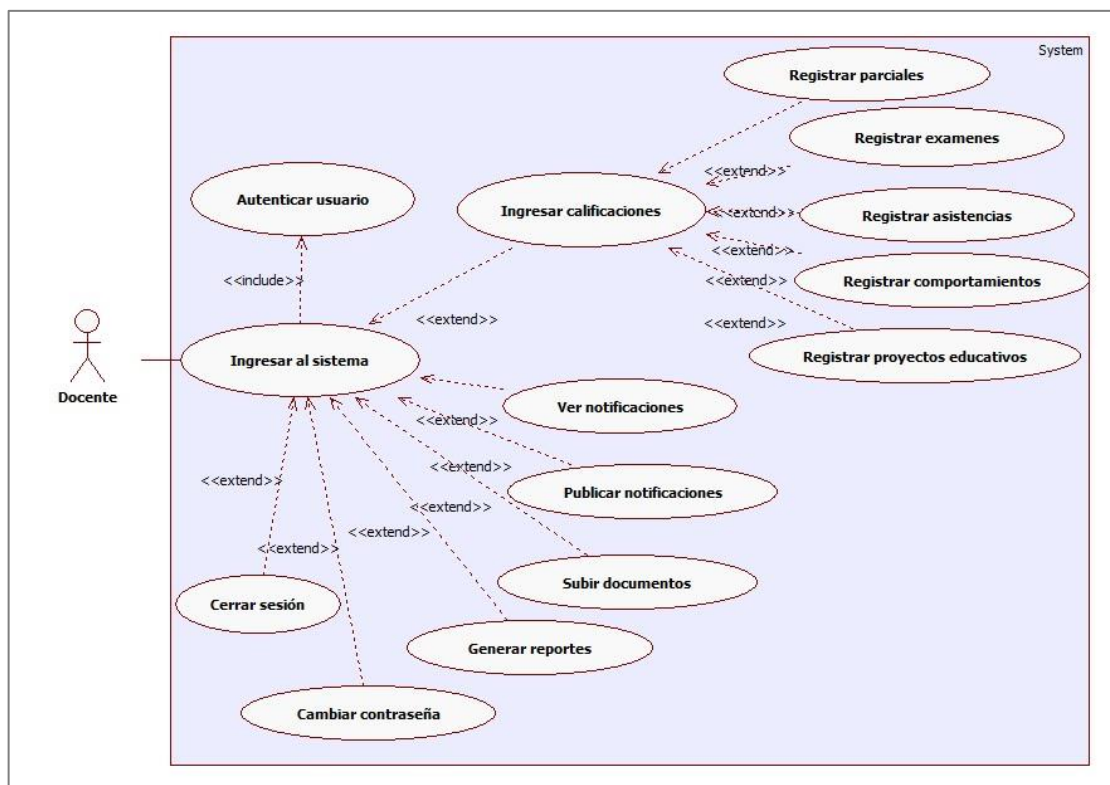


Figura 10: Caso de uso general para el tipo de usuario “Docente”

- **Acciones principales:**

- Ingresar al sistema: Incluye la acción de autenticación como condición de ingreso.
- Ingresar calificaciones: Permite al docente ingresar calificaciones de parciales, exámenes y en su curso de tutoría asignado le permite ingresar calificaciones de proyectos educativos, asistencias y comportamientos.
- Ver notificaciones: permite al docente leer notificaciones o comunicados publicados por un administrador.
- Publicar notificaciones: Permite al docente publicar notificaciones, comunicados a los estudiantes o a los representantes de los cursos en los que da clases.
- Subir documentos: Permite al docente subir, ver, editar y eliminar documentos de su carpeta asignada.
- Generar reportes: Permite al docente generar reportes de calificaciones por estudiantes y por asignaturas.
- Cambiar contraseña: Permite al docente cambiar su contraseña.
- Cerrar sesión: Permite al docente salir del sistema de forma segura.

### 2.5.3 Caso de uso estudiante

Las acciones generales para el tipo de usuario “Estudiante” se muestran y se detallan a continuación:

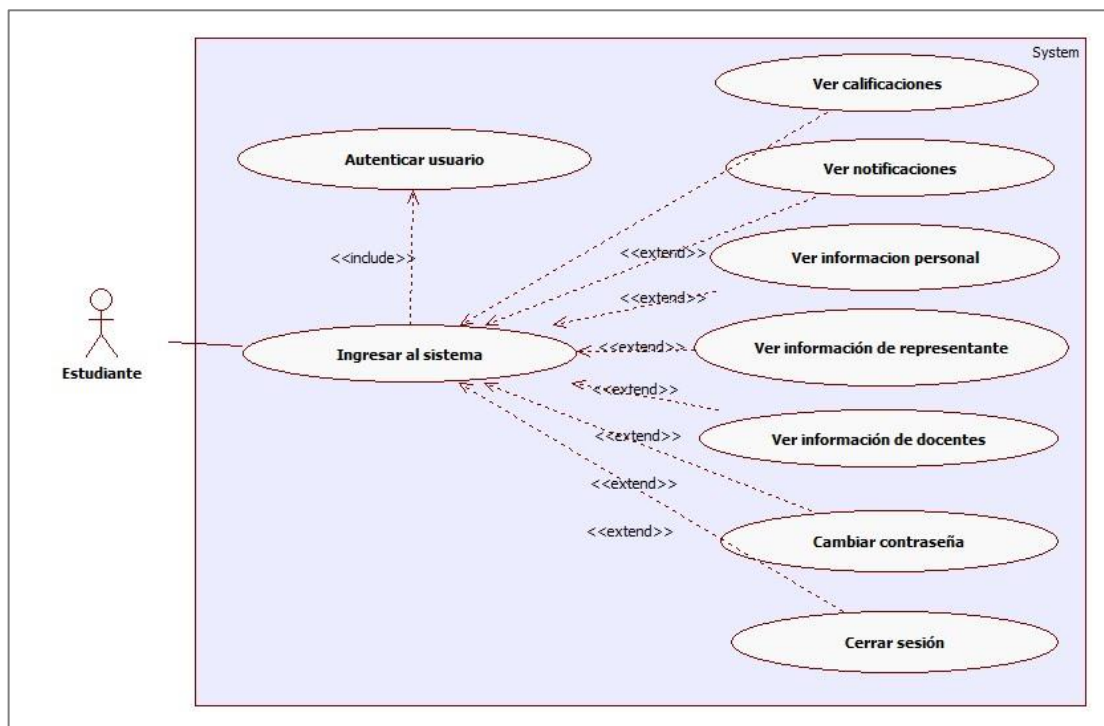


Figura 11: Caso de uso general para el tipo de usuario “Estudiante”



- **Acciones principales:**

- Ingresar al sistema: Incluye la acción de autenticación como condición de ingreso.
- Ver calificaciones: Permite al estudiante ver su rendimiento académico.
- Ver notificaciones: Permite al estudiante ver publicaciones o comunicados creados por sus docentes o por un administrador.
- Ver información personal: Permite al estudiante comprobar que su información personal sea correcta.
- Ver información de representante: Permite al estudiante ver la información de su representante.
- Ver información de docentes: Permite al estudiante ver la información de contacto de sus docentes.
- Cambiar contraseña: Permite al estudiante cambiar su contraseña.
- Cerrar sesión: Permite al estudiante salir del sistema de forma segura.

### 2.5.4 Caso de uso representante

Las acciones generales para el tipo de usuario "Representante" se muestran y se detallan a continuación:

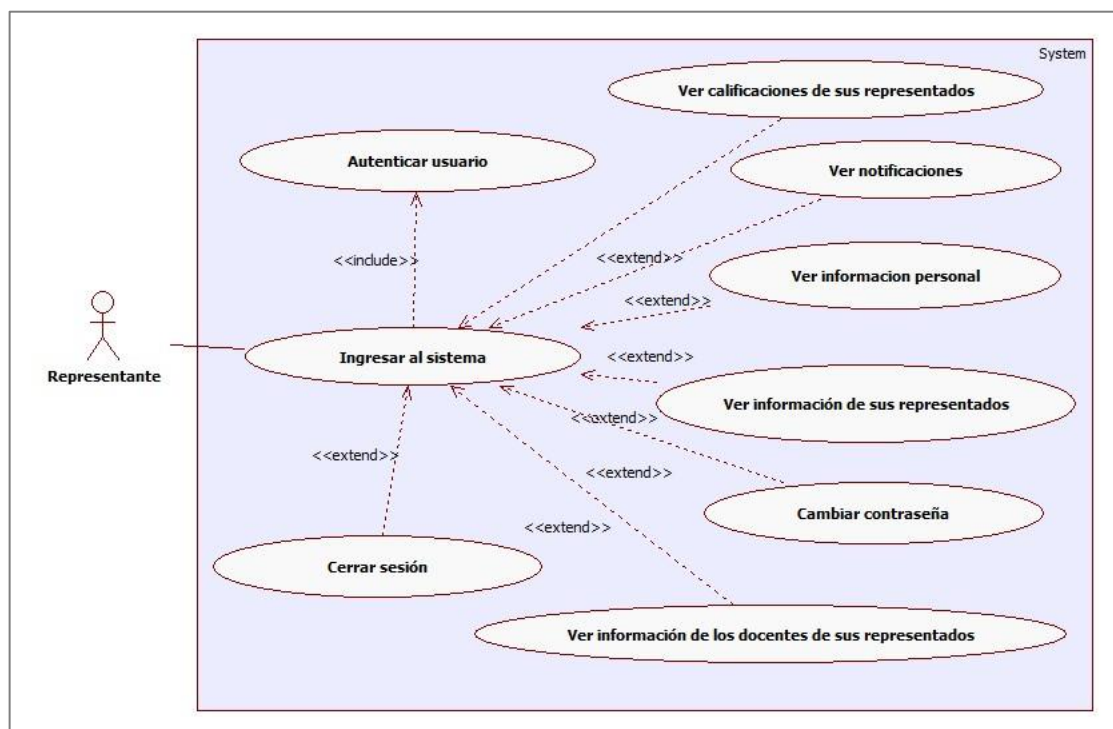


Figura 12: Caso de uso general para el tipo de usuario "Representante"

- **Acciones principales:**

- Ingresar al sistema: Incluye la acción de autenticación como condición de ingreso.
- Ver calificaciones de sus representados: Permite al representante ver el rendimiento académico de sus representados.
- Ver notificaciones: Permite al representante ver publicaciones o comunicados creados por los docentes de sus representados o por un administrador.
- Ver información personal: Permite al representante comprobar que su información personal sea correcta.
- Ver información de sus representados: Permite al representante ver la información de sus representados.
- Ver información de los docentes de sus representados: Permite al representante ver la información de contacto de los docentes de sus representados.
- Cambiar contraseña: Permite al representante cambiar su contraseña.
- Cerrar sesión: Permite al representante salir del sistema de forma segura.

## **2.6 Integración de Zurb Foundation con CakePHP**

Para el desarrollo ágil de aplicaciones web se ha elegido el framework CakePHP y para el diseño de una aplicación adaptativa se realiza la integración del framework de front-end Foundation.

### **2.6.1 Estructura del proyecto**

La instalación y creación de la estructura del proyecto se la realiza mediante el gestor de dependencias y paquetes PHP Composer<sup>16</sup> de acuerdo con la documentación oficial de instalación y creación de un nuevo proyecto con CakePHP.

El framework Foundation está disponible en varias versiones en su página oficial para que los desarrolladores puedan elegir que funcionalidades del framework necesitan integrar en su proyecto. Para el desarrollo del Sistema académico web bilingüe se utilizará la versión estándar 5.5.3 y algunos complementos de la versión 6.

---

<sup>16</sup> Composer es un gestor de paquetes a nivel de aplicación para el lenguaje de programación PHP que proporciona un formato estándar para gestionar las dependencias del software PHP y las bibliotecas requeridas.

De acuerdo con la estructura manejada por el framework CakePHP, los archivos de hojas estilo, archivos JavaScript e imágenes, se deben ubicar en sus respectivos directorios dentro del directorio principal *webroot*, como se indica a continuación:

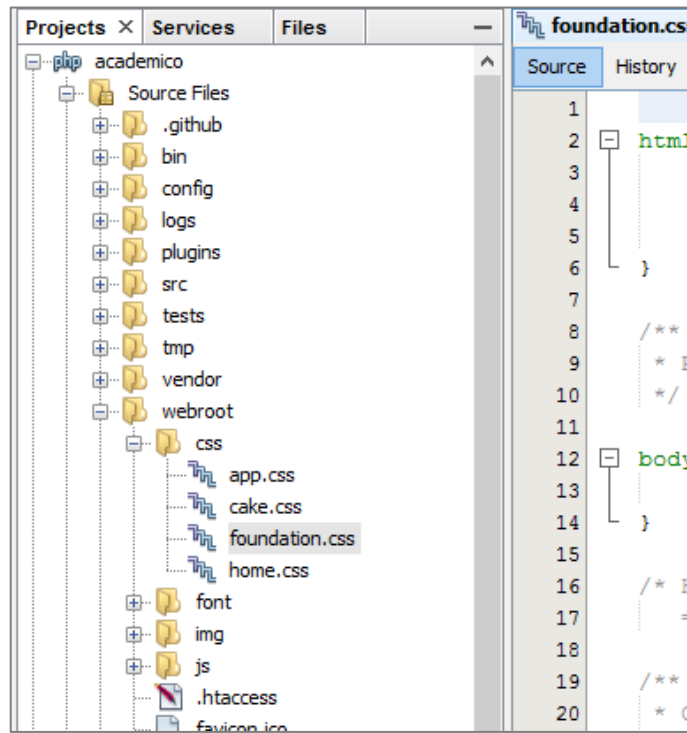


Figura 13: Estructura del proyecto con CakePHP y Foundation

La manera de referenciar los archivos CSS y JavaScript en un archivo plantilla de CakePHP con extensión “.ctp” (Cake Template Page), es mediante los métodos `htmlhelpers` del framework como se indica a continuación:

```
<?php
$cakeDescription = 'academico';
?>
<!DOCTYPE html>
<html>
<head>
    <?=$this->Html->charset() ?>
    <meta name="viewport" content="width=device-width, initial-scale=
    <title>
        <?=$cakeDescription ?>:
        <?=$this->fetch('title') ?>
    </title>
    <?=$this->Html->meta('icon') ?>
    <?=$this->Html->css('foundation') ?>
    <?=$this->Html->css('cake') ?>
    <?=$this->Html->css('app') ?>
    <?=$this->fetch('meta') ?>
    <?=$this->fetch('css') ?>
    <?=$this->fetch('script') ?>
</head>
```

Figura 14: Referencia de Foundation en un archivo plantilla de CakePHP

## 2.6.2 Propuesta de diseño adaptativo

Para el desarrollo de un sistema que sea adaptativo, se propone un diseño anti desborde horizontal, es decir, que el uso del elemento HTML conocido como *scrollbar* o barra de desplazamiento, no sea necesaria en dirección horizontal en ninguna vista del sistema.

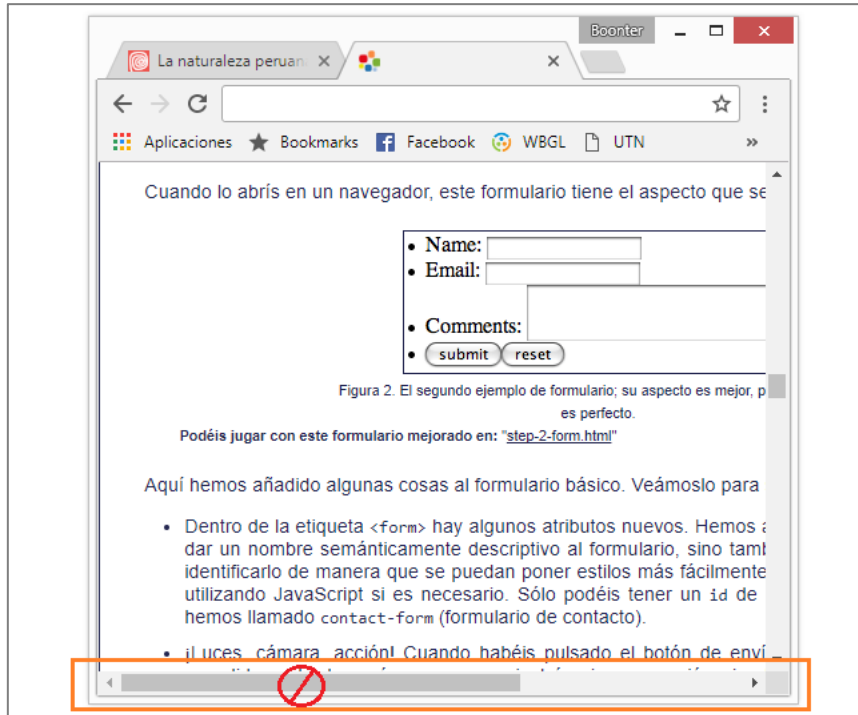


Figura 15: Propuesta de diseño anti desborde horizontal

De acuerdo con la filosofía el framework Foundation, se tomará como referencia de diseño la pantalla de un dispositivo móvil, luego se planificará las variaciones en la presentación para pantallas más grandes. Se muestra como ejemplo el diseño para el ingreso de las ocho notas de una parcial:

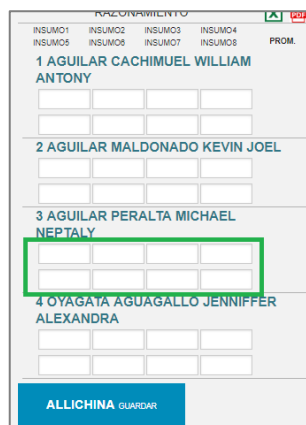
A screenshot of a mobile application interface for entering grades. The interface shows a list of students with their names and a grid of input fields for grades. The students listed are: 1 AGUILAR CACHIMUEL WILLIAM ANTONY, 2 AGUILAR MALDONADO KEVIN JOEL, 3 AGUILAR PERALTA MICHAEL NEPTALY, and 4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA. A green box highlights the input fields for the third student, '3 AGUILAR PERALTA MICHAEL NEPTALY'. At the bottom of the screen, there is a blue button labeled 'ALLICHINA GUARDAR'. The top of the screen shows a header with 'RACIONAMIENTO' and a status bar with '2:57'.

Figura 16: Ingreso de notas en dispositivo móvil pequeño

RAZONAMIENTO				
INSUMO1	INSUMO2	INSUMO3	INSUMO4	PROM.
INSUMO5	INSUMO6	INSUMO7	INSUMO8	
<b>1 AGUILAR CACHIMUEL WILLIAM ANTONY</b>				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>2 AGUILAR MALDONADO KEVIN JOEL</b>				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>3 AGUILAR PERALTA MICHAEL NEPTALY</b>				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA</b>				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

**ALLICHINA** GUARDAR

Figura 17: Ingreso de notas en dispositivo móvil mediano

RAZONAMIENTO								
INSUMO1	INSUMO2	INSUMO3	INSUMO4	INSUMO5	INSUMO6	INSUMO7	INSUMO8	PROM.
<b>1 AGUILAR CACHIMUEL WILLIAM ANTONY</b>								
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>2 AGUILAR MALDONADO KEVIN JOEL</b>								
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>3 AGUILAR PERALTA MICHAEL NEPTALY</b>								
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA</b>								
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

**ALLICHINA** GUARDAR

Figura 18: Ingreso de notas en pantalla de PC

## 2.7 Propuesta de desarrollo bilingüe

El sistema académico bilingüe tiene como uno de sus objetivos principales fomentar el uso del idioma kichwa, por esta razón se propone el uso de los dos idiomas a la vez priorizando en orden y tamaño el kichwa, debido a que la mayoría de palabras que se requieren en las vistas del sistema son palabras técnicas, la traducción y en algunos casos la elaboración de ciertas palabras estará a cargo del Departamento de Gestión Social Intercultural del Gobierno Autónomo Descentralizado Municipal del cantón Otavalo. A continuación, se muestran algunos ejemplos del uso de los dos idiomas simultáneamente para las vistas del sistema:

<b>MAWKAKPAK SHUTI</b> USUARIO
<input type="text" value="1001001001"/>

Figura 19: Formulario en kichwa-español



Figura 20: Botón en kichwa-español

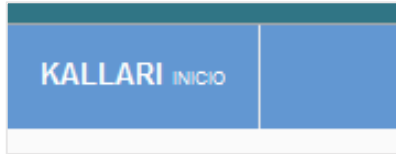


Figura 21: Barra de navegación en kichwa-español



Figura 22: Menú en kichwa-español

SHUTI <small>NOMBRES</small>	
JAQUELINE JANICE	
ERICK NATHAN	1
EDISON NOE	1

Figura 23: Encabezado de tablas en kichwa-español



Figura 24: Opciones en kichwa-español



Figura 25: Mensaje de confirmación en kichwa-español

## 2.8 Plan de desarrollo del proyecto

De acuerdo con las historias de usuario que determinan el orden y el esfuerzo necesario para desarrollar el sistema académico, se resume un plan de entrega aproximado que estará sujeto a pequeños cambios acorde con la herramienta teórica de *refactorización* contemplada en la metodología XP.

TABLA 2.36  
PLAN DE DESARROLLO DEL PROYECTO  
HISTORIAS DE USUARIO

NÚMERO	NOMBRE	ITERACIÓN	ESFUERZO (SEMANAS)
1	Registrar usuarios	1	3
2	Ingreso al sistema	1	
3	Perfil de usuario	1	
4	Acceso a la información	1	
5	Registrar docentes	1	1
6	Registrar estudiantes	1	
7	Registrar representantes	1	1
8	Registrar administradores	1	
9	Registrar cursos	1	1
10	Registrar asignaturas	1	
11	Matricular estudiante	2	1
12	Asignar asignaturas a cursos	2	2
13	Asignar carga horaria a docentes	2	
14	Asignar tutores a cursos	2	
15	Registrar notas parciales	2	1
16	Registrar notas de exámenes	2	1
17	Registrar asistencias	2	2
18	Registrar comportamientos	2	
19	Registrar notas de proyectos educativos	2	
20	Administración de documentos	3	1
21	Publicación de notificaciones	3	1
22	Generación de reportes	3	2
		TOTAL	16

## 2.9 Primera iteración

En base al plan de desarrollo del proyecto, en la primera iteración se desarrollan las funcionalidades básicas de manejo de usuarios y el registro de administradores, docentes, estudiantes, representantes, cursos y asignaturas.

TABLA 2.37  
PLAN DE DESARROLLO DE LA PRIMERA ITERACIÓN

HISTORIAS DE USUARIO EN LA PRIMERA ITERACIÓN		
NÚMERO	NOMBRE	ESFUERZO (SEMANAS)
1	Registrar usuarios	3
2	Ingreso al sistema	
3	Perfil de usuario	
4	Acceso a la información	
5	Registrar docentes	1
6	Registrar estudiantes	
7	Registrar representantes	1
8	Registrar administradores	
9	Registrar cursos	1
10	Registrar asignaturas	
TOTAL		6

### 2.9.1 Historia de usuario – Registrar usuarios

Las tareas realizadas en el desarrollo de la historia de usuario 1 se detallan a continuación:

- **Desarrollo de interfaces de administración de usuarios**

Se crea las interfaces para que el administrador del sistema pueda ingresar, ver, editar y eliminar los usuarios del sistema.

USUARIOS					
YUPAY NÚMERO	SHUTI NOMBRE	ROL	NUM. ROL	RIKUNA VER	TUKUCHINA ELIMINAR
USU-00001	PEREZ PEREZ JUAN CARLOS	ADMINISTRADOR	ADM-00001	⦿	✕
USU-00003	ANDRADE ARELLANO WASHINGTON JAVIER	DOCENTE	DCT-00005	⦿	✕
USU-00004	ANDRADE NEJER GUSTAVO ALEXANDER	DOCENTE	DCT-00006	⦿	✕
USU-00005	ANGUAYA JISAMA YANELY JISU	DOCENTE	DCT-00007	⦿	✕
USU-00006	ANRANGO MALES ARELY IRAYA	DOCENTE	DCT-00008	⦿	✕
USU-00007	ARIAS BURGA MELANIE SUYAY	DOCENTE	DCT-00009	⦿	✕
USU-00008	BAUTISTA FICHAMBA LIDIA GUADALUPE	DOCENTE	DCT-00010	⦿	✕
USU-00009	BENALCAZAR CERON JOHN SEBASTIAN	DOCENTE	DCT-00011	⦿	✕
USU-00010	BURGA ANGUAYA RAYMI APAWKI	DOCENTE	DCT-00012	⦿	✕
USU-00011	BURGA ARELLANO LIZETH PAULA	DOCENTE	DCT-00013	⦿	✕
USU-00012	BURGA ARIAS ALEN ESTEFANI	DOCENTE	DCT-00014	⦿	✕
USU-00013	CABASCANGO PERUGACHI GIUVELY MAYTE	DOCENTE	DCT-00015	⦿	✕
USU-00014	CAMUENDO ANRANGO ANA BELEN	DOCENTE	DCT-00016	⦿	✕
USU-00015	CAMUENDO ASCANTA JHON SNEIDER	DOCENTE	DCT-00017	⦿	✕
USU-00016	CASTAÑEDA GARCIA JUDITH MELANI	DOCENTE	DCT-00018	⦿	✕
USU-00017	CHALAN MALDONADO KENIN ARIEL	DOCENTE	DCT-00019	⦿	✕

Figura 26: Interfaz de administración de usuarios

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo usuario y de edición en las clases *Table* y *Entity* del modelo *Users*. El sistema permitirá la relación uno a uno de acuerdo con el diseño del modelo relacional. Los campos que determinan esta



relación son el *ID* y el *Rol* de las entidades de administradores, docentes, estudiantes y representantes.

- **Autenticación de usuarios**

Para la autenticación e identificación de usuarios se utiliza el componente de CakePHP *AuthComponent::identify()* que verifica los datos del usuario y permite el ingreso al sistema solo a aquellos que estén registrados en la base de datos.

- **Encriptación de contraseña**

De acuerdo con los requerimientos de seguridad, se codifica los métodos para que las contraseñas almacenadas en la base de datos se guarden encriptadas, el sistema generará una contraseña por defecto que será cambiada posteriormente por el usuario. El algoritmo de encriptación usado es "bcrypt"<sup>17</sup> que CakePHP lo usa por defecto desde su versión 3 por sus beneficios de seguridad.

- **Controlador**

En el controlador de usuarios se desarrollan los siguientes métodos para el manejo de sus entidades:

- *index()*: Permite listar todos los registros de usuarios.
- *view()*: Permite ver la información de un usuario específico recibiendo como parámetro su ID.
- *add()*: Permite registrar la información de un nuevo usuario. El sistema generará automáticamente el nombre de usuario y contraseña por defecto. De acuerdo con su rol, los nombres de usuario tendrán el sufijo A, D, R, E.
- *edit()*: Permite actualizar un usuario, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del usuario. Esta función permite al usuario cambiar su contraseña.
- *delete()*: Permite eliminar un usuario del sistema, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del usuario.

## 2.9.2 Historia de usuario - Ingreso al sistema

En el desarrollo de la historia de usuario 2 se realizan las siguientes tareas:

- **Desarrollo de interfaz de acceso al sistema**

---

<sup>17</sup> bcrypt es una función de cifrado de contraseñas que brinda mayor seguridad contra ataques de fuerza bruta, es usado como función de cifrado de contraseñas en algunas distribuciones de Linux.

Se diseña y se elabora la interfaz de ingreso al sistema, los usuarios accederán a la página principal del sistema y encontrarán un formulario de ingreso en el que deberán ingresar su nombre de usuario y contraseña y en el caso de ser datos correctos serán dirigidos a las tareas disponibles de acuerdo con su rol de usuario, caso contrario se desplegará un mensaje de advertencia indicando que el acceso es denegado.



GOBIERNO AUTÓNOMO  
DESCENTRALIZADO  
MUNICIPAL DE OTAVALO

SISTEMA ACADÉMICO DEL GAD MUNICIPAL DEL CANTÓN OTAVALO

**YAYKUNA** INGRESAR

**MAWKAKPAK SHUTI** USUARIO

**PAKALLA RIKUCHI** CONTRASEÑA

**KACHANA** ENMAR

Figura 27: Interfaz de acceso al sistema

### 2.9.3 Historia de usuario - Perfil de usuario

En el desarrollo de la historia de usuario número 3 se realizan las siguientes tareas:

- **Desarrollo de interfaces**

Se desarrolla interfaces diferentes para que dependiendo del tipo de usuario tengan acceso a realizar solo las tareas autorizadas.

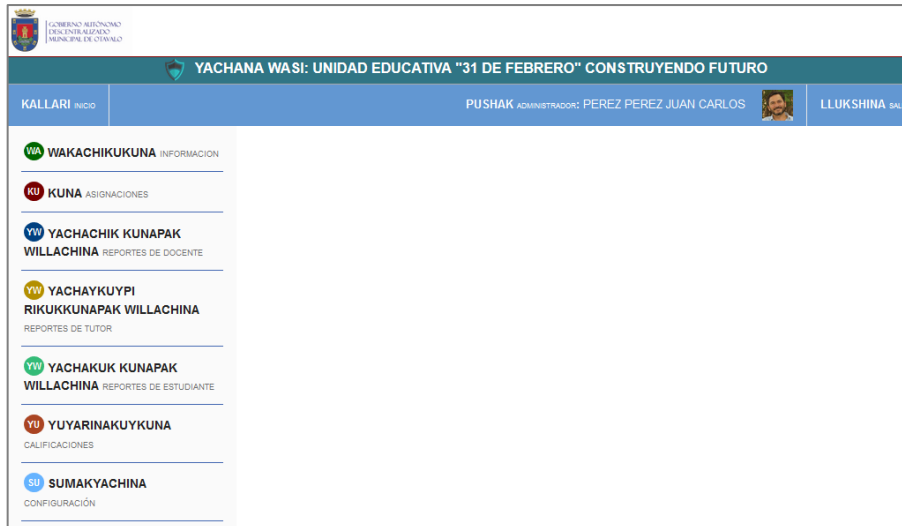


Figura 28: Interfaz de tipo de usuario Administrador



Figura 29: Interfaz de usuario Docente

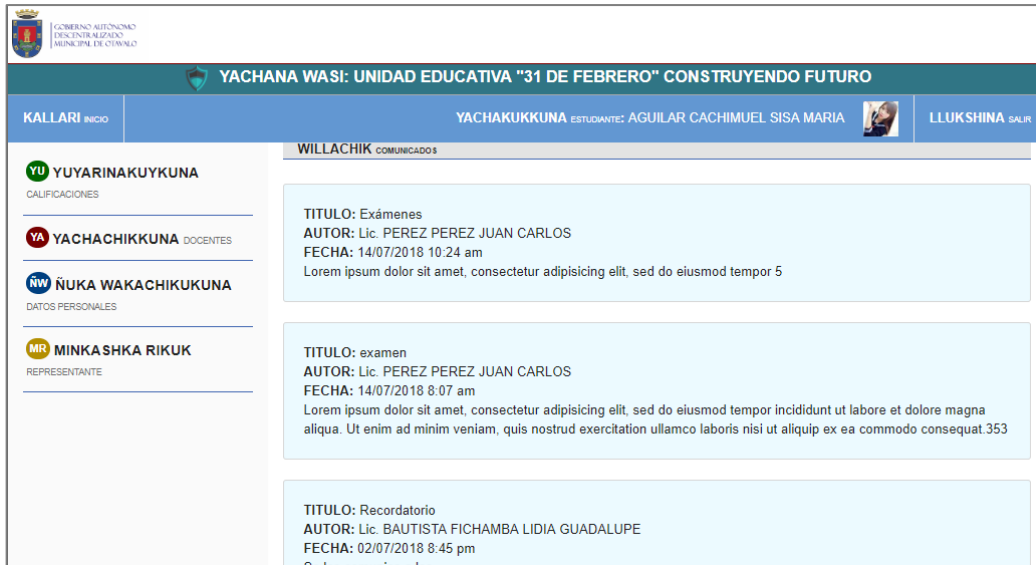


Figura 30: Interfaz de usuario Estudiante

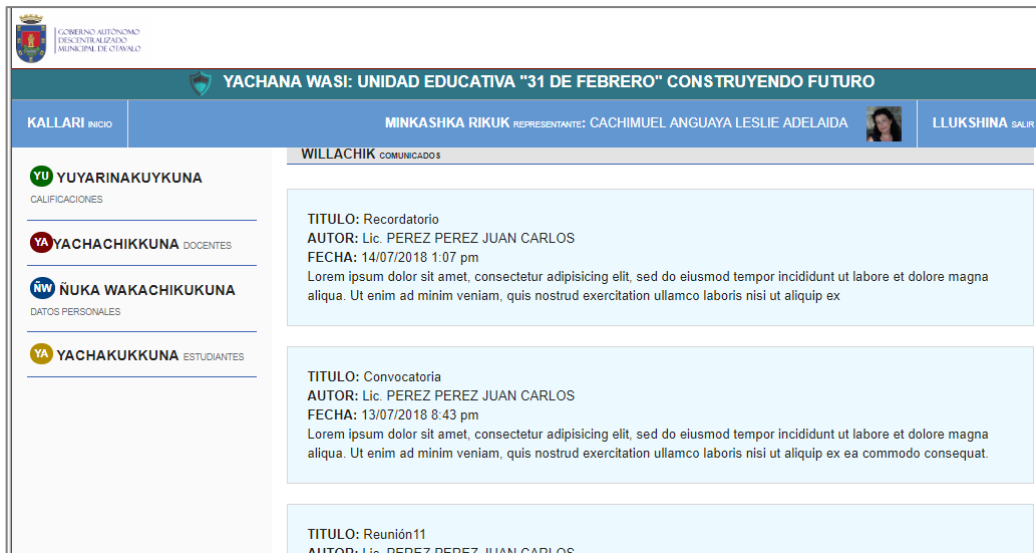


Figura 31: Interfaz de usuario Representante

## 2.9.4 Historia de usuario - Acceso a la información

En el desarrollo de la historia de usuario número 4 se llevan a cabo las siguientes tareas:

- **Autorización de acceso**

Se codifica los métodos en el controlador global de la aplicación para asignar autorización de acceso a las interfaces relacionadas con el rol de usuario. Cuando una petición sea denegada, el controlador redirigirá a la última vista de usuario que realizó la petición.

- **Limitación de acceso**

Se configura el sistema para que permitan al usuario administrador el control total sobre la información del sistema. Para los usuarios docente, estudiante y representante se limita el acceso a la información relacionada a ellos solo del año lectivo actual.

## 2.9.5 Historia de usuario – Registrar docentes

En el desarrollo de la historia de usuario número 5 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de docentes**

Se desarrollan las interfaces para ingresar, ver, editar y dar de baja a los docentes que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.

YUPAY	NÚMERO	KASTAKUNA	APELLIDOS	SHUTI	NOMBRES	KAWSAYUK	CEDULA	RIKUNA	VER	MUSHUK Y ACHINA	EDITA
DCT-00001		CABASCANGO	CHIZA	BOONTER	SEGUNDO	1003309158					
DCT-00005		ANDRADE	ARELLANO	WASHINGTON	JAVIER	1050026507					
DCT-00006		ANDRADE	NEJER	GUSTAVO	ALEXANDER	1005450455					
DCT-00007		ANGUAYA	JISAMA	YANELY	JISU	1050351152					
DCT-00008		ANRANGO	MALES	ARELY	IRAYA	1050394350					
DCT-00009		ARIAS	BURGA	MELANIE	SUYAY	1005388200					
DCT-00010		BAUTISTA	FICHAMBA	LIDIA	GUADALUPE	1005014681					
DCT-00011		BENALCAZAR	CERON	JOHN	SEBASTIAN	1050022092					
DCT-00012		BURGA	ANGUAYA	RAYMI	APAWKI	1005082027					
DCT-00013		BURGA	ARELLANO	LIZETH	PAULA	1005121460					
DCT-00014		BURGA	ARIAS	ALEN	ESTEFANI	1050534260					
DCT-00015		CABASCANGO		PERUGACHI	GIUVELY MAYTE	1050014867					
DCT-00016		CAMUENDO	ANRANGO	ANA	BELEN	1351215585					
DCT-00017		CAMUENDO	ASCANTA	JHON	SNEIDER	1050332525					
DCT-00018		CASTAÑEDA	GARCIA	JUDITH	MELANI	1005463987					
DCT-00019		CHALAN	MALDONADO	KENIN	ARIEL	1050028115					

Figura 32: Interfaz principal de administración de docentes

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo docente y de edición en las clases *Table* y *Entity* del modelo *Docentes*. El sistema transformará los campos nombres y apellidos para guardarlos en mayúsculas.

- **Controlador**

En el controlador de docentes se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de docentes.
- `view()`: Permite ver la información de un docente específico recibiendo como parámetro su ID.

- add(): Permite registrar la información de un nuevo docente.
- edit(): Permite editar un docente, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del docente.

La información de un usuario docente es necesaria para los reportes históricos de ingresos de calificaciones y de tutores de curso, por esta razón, no se codifican los métodos que permitan eliminar un docente de la base de datos cuando ya no forme parte de la institución; en su lugar, el docente será dado de baja en el sistema cambiando su estado a inactivo. Los docentes inactivos ya no son visibles para nuevas asignaciones.

### 2.9.6 Historia de usuario – Registrar estudiantes

En el desarrollo de la historia de usuario número 6 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de estudiantes**

Se desarrollan las interfaces para ingresar, ver, editar y dar de baja a los estudiantes que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.

YUPAY NÚMERO	KASTAKUNA APELLIDOS	SHUTI NOMBRES	KAWSAYUK CEDULA	RIKUNA VER	MUSHUK Y ACHINA EDITAR
EST-00001	ANRANGO VELASQUEZ	ERIKA MAITE	1005032923	👁	✎
EST-00002	ANTAMBA AMAGUAÑA	JULIETTE ANAHI	1005033061	👁	✎
EST-00003	BURGA ANGUAYA	JIMMY JAFET	1005081953	👁	✎
EST-00004	BURGA CACHIMUEL	SARA MELISSA	1004979207	👁	✎
EST-00005	BURGA ISAMA	JUDITH IRLANDA	1050480431	👁	✎
EST-00006	BURGA OYAGATA	LUIS DAVID	1004967640	👁	✎
EST-00007	CACHIMUEL ANGUAYA	MELANY LISETH	1005217037	👁	✎
EST-00008	CACHIMUEL CACHIMUEL	SAMUEL JUSTIN	1004928378	👁	✎
EST-00009	CACHIMUEL GUAMAN	JHONY STALYN	1005307267	👁	✎
EST-00010	CAMPO ISAMA	SARA DAYANA	1004918569	👁	✎
EST-00011	CAMPO MALES	KEVIN ANDERSON	1050378056	👁	✎
EST-00012	CEPEDA ANRANGO	EDWIN FABIAN	1005026883	👁	✎
EST-00013	CHALAN JETACAMA	ARIEL ISRAEL	1755724323	👁	✎
EST-00014	CHAPI ANRANGO	ERICK SNEYDER	1005089709	👁	✎
EST-00015	CHIZA CASTAÑEDA	ELSA MARINA	1050450103	👁	✎
EST-00016	CHIZA CASTAÑEDA	MARTHA CECILIA	1050449832	👁	✎

Figura 33: Interfaz principal de administración de estudiantes

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo estudiante y de edición en las clases *Table* y *Entity* del modelo *Estudiantes*. El sistema transformará los campos nombres y apellidos para guardarlos en mayúsculas. La asignación de representantes permite la relación uno a uno, un estudiante solo puede tener un representante de

acuerdo con el diseño del modelo relacional. El campo cédula para estudiantes no es obligatorio.

- **Controlador**

En el controlador de Estudiantes se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de estudiantes.
- `view()`: Permite ver la información de un estudiante específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo estudiante.
- `edit()`: Permite editar un estudiante, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del estudiante.

La información de los estudiantes que se retiran de la institución o que ya se graduaron es necesaria para los reportes históricos, por esta razón, no se codifican los métodos que permitan eliminar un estudiante de la base de datos, en su lugar, el estudiante será dado de baja del sistema cambiando su estado a inactivo. Los estudiantes inactivos ya no son visibles para nuevas asignaciones.

### 2.9.7 Historia de usuario - Registrar representantes

En el desarrollo de la historia de usuario número 7 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de representantes**

Se desarrollan las interfaces para ingresar, ver, editar y dar de baja a los representantes que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.

		MINKASHKA RIKUKKUNA REPRESENTANTES			MUSHUK NUEVO	
		MASKANA BUSCAR				
	YUPAY	KASTAKUNA	SHUTI	KAWSAYUK	RIKUNA	MUSHUK Y
	NOMERO	APELLIDOS	NOMBRES	CECULA	VER	ACHINA
						EDITA
PANTES	REP-00001	ANTAMBA AMAGUAÑA	MELANIE SULAY	1050445830	👁	✍
	REP-00002	BAUTISTA TOCAGON	KAROLA	1004942916	👁	✍
	REP-00003	BENALCAZAR ENRIQUEZ	ELVIS SEBASTIAN	1005011190	👁	✍
	REP-00004	BURGA ARELLANO	DIANA ESTEFANIA	1004761266	👁	✍
	REP-00005	BURGA BURGA	LUIS GUSTAVO	1005117518	👁	✍
	REP-00006	BURGA BURGA	ÑUSTA DAYANA	1721603072	👁	✍
	REP-00007	BURGA CRIOLLO	PEDRO ANTONIO	1950167849	👁	✍
TURAS	REP-00008	BURGA ISAMA	ELIAS STALIN	1005226376	👁	✍
	REP-00009	BURGA ISAMA	JORDAN DAVID	1005338908	👁	✍
	REP-00010	BURGA LEON	ELISEO SALOMON	1050186780	👁	✍
	REP-00011	CACHIMUEL ANGUAYA	LESLIE ADELAIDA	1005216955	👁	✍
	REP-00012	CAMPO CAMPO	NESLY YARINA	1050037587	👁	✍
	REP-00013	CAMUENDO ASCANTA	ERIKA JANETH	1005197155	👁	✍
	REP-00014	ENRIQUEZ GRUJALVA	ADRIANA MARIBEL	1004977078	👁	✍
	REP-00015	GUALACATA PERUGACHI	XAVIER ALEXANDER	1004887236	👁	✍

Figura 34: Interfaz principal de administración de representantes

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo representante y de edición en las clases *Table* y *Entity* del modelo *Representantes*. El sistema transformará los campos nombres y apellidos para guardarlos en mayúsculas.

- **Controlador**

En el controlador de Representantes se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de representantes.
- `view()`: Permite ver la información de un representante específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo representante.
- `edit()`: Permite editar un representante, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del representante.

La información de los representantes no se puede eliminar por estar relacionada con los estudiantes, por esta razón, no se codifican los métodos que permitan eliminar un representante de la base de datos, en su lugar, el representante será dado de baja del sistema cambiando su estado a inactivo. Los representantes inactivos ya no son visibles para nuevas asignaciones.

### 2.9.8 Historia de usuario - Registrar administradores

En el desarrollo de la historia de usuario número 8 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de administradores**

Se desarrollan las interfaces para ver, editar y dar de baja a los administradores que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.

YUPAY NÚMERO	KASTAKUNA APELLIDOS	SHUTI NOMBRES	KAWSAYUK CEDULA	RIKUNA VER	MUSHUK Y ACHINA EDITAR
ADM-00001	PEREZ PEREZ	JUAN CARLOS	1719328955	👁	✎
ADM-00002	GUERRA	MARIA	1003309158	👁	✎

< >

Página 1 de 1, mostrando 2 administradores de 2 totales

Figura 35: Interfaz principal de administración de administradores



- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo administrador y de edición en las clases *Table* y *Entity* del modelo *Admins*. El sistema transformará los campos nombres y apellidos para guardarlos con mayúsculas.

- **Controlador**

En el controlador de administradores se desarrollan los siguientes métodos para el manejo de sus entidades:

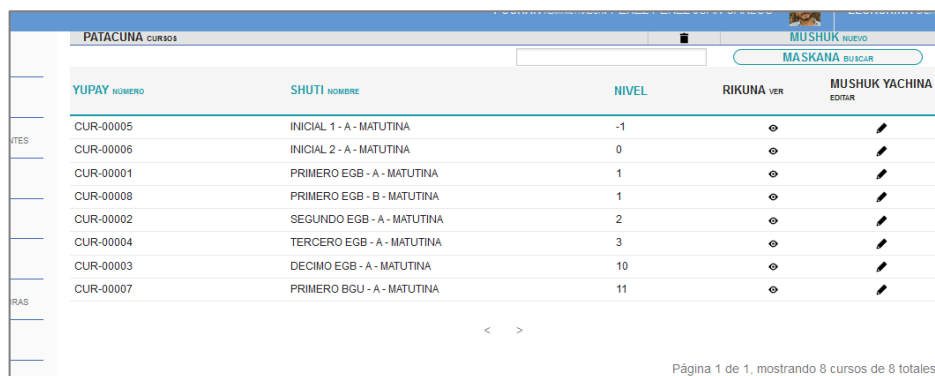
- `index()`: Permite listar todos los registros de administradores.
- `view()`: Permite ver la información de un administrador específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo administrador.
- `edit()`: Permite editar un administrador, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del administrador.
- `delete()`: Permite eliminar un administrador del sistema, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del administrador.

### 2.9.9 Historia de usuario - Registrar cursos

En el desarrollo de la historia de usuario número 9 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de cursos**

Se desarrollan las interfaces para ver, editar y dar de baja a los cursos que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.



PATAACUNA CURSOS		MUSHUK NUEVO		
		MASKANA BUSCAR		
YUPAY NUMERO	SHUTI NOMBRE	NIVEL	RIKUNA VER	MUSHUK YACHINA EDITAR
CUR-00005	INICIAL 1 - A - MATUTINA	-1	👁	✎
CUR-00006	INICIAL 2 - A - MATUTINA	0	👁	✎
CUR-00001	PRIMERO EGB - A - MATUTINA	1	👁	✎
CUR-00008	PRIMERO EGB - B - MATUTINA	1	👁	✎
CUR-00002	SEGUNDO EGB - A - MATUTINA	2	👁	✎
CUR-00004	TERCERO EGB - A - MATUTINA	3	👁	✎
CUR-00003	DECIMO EGB - A - MATUTINA	10	👁	✎
CUR-00007	PRIMERO BGU - A - MATUTINA	11	👁	✎

Página 1 de 1, mostrando 8 cursos de 8 totales

Figura 36: Interfaz principal de administración de cursos

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nuevo curso y edición en las clases *Table* y *Entity* del modelo *Cursos*. El sistema transformará el campo nombre para guardarlos con mayúsculas.

- **Controlador**

En el controlador de cursos se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de cursos.
- `view()`: Permite ver la información de un curso específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo curso.
- `edit()`: Permite editar un curso, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del curso.

La información de un curso es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar un curso de la base de datos cuando ya no esté vigente, en su lugar, el curso será dado de baja en el sistema cambiando su estado a inactivo. Los cursos inactivos ya no son visibles para nuevas asignaciones.

### 2.9.10 Historia de usuario - Registrar asignaturas

En el desarrollo de la historia de usuario número 10 se llevan a cabo las siguientes tareas:

- **Interfaces de administración de asignaturas**

Se desarrollan las interfaces para ver, editar y dar de baja a las asignaturas que están registrados en la base de datos. La autorización de acceso a estas interfaces la tiene el usuario con rol de administrador.

YUPAY	NUMERO	SHUTI	NOMBRE	AREA	NIVEL	RIKUNA	VER	MUSHUK YACHI	NA
	ASG-00001	MATEMATICAS		MATEMATICAS	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00002	DIBUJO		EXPRESIÓN ARTISTICA	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00003	CIENCIAS NATURALES		CIENCIAS NATURALES	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00004	ALGEBRA		ALGEBRA	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00005	INVESTIGACIÓN		INVESTIGACIÓN	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00006	DIBUJO TECNICO		DIBUJO	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00007	COMPRESION		DESARROLLO	BÁSICA - BACHILLERATO		👁	✎	
	ASG-00008	RAZONAMIENTO		A. RAZONAMIENTO	INICIAL - PREPARATORIA		👁	✎	
	ASG-00009	DESARROLLO MOTRIZ		A. D. MOTRIZ	INICIAL - PREPARATORIA		👁	✎	

Página 1 de 1, mostrando 9 asignaturas de 9 totales

Figura 37: Interfaz principal de administración de asignaturas

- **Validación de datos**

Se codifican los métodos para la validación y el tratamiento de la información que se obtiene de los formularios de ingreso de nueva asignatura y de edición en las clases *Table* y *Entity* del modelo *Materias*. El sistema transformará el campo nombre y área para guardarlos con mayúsculas.

- **Controlador**

En el controlador de asignaturas se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de asignaturas.
- `view()`: Permite ver la información de una asignatura específica recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de una nueva asignatura.
- `edit()`: Permite editar una asignatura, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID de la asignatura.

La información de una asignatura es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar una asignatura de la base de datos cuando ya no esté vigente, en su lugar, la asignatura será dada de baja en el sistema cambiando su estado a inactivo. Las asignaturas inactivas ya no son visibles para nuevas asignaciones.

### 2.9.11 Pruebas de aceptación

De acuerdo con la metodología XP, al final de una iteración se deben realizar pruebas de aceptación en base a las historias de usuario definidas en el plan de desarrollo de cada iteración.

Los casos de prueba se realizan con datos de prueba en el equipo del desarrollador utilizando la aplicación Xampp como servidor local. Los casos de prueba realizados en la primera iteración se detallan a continuación:

TABLA 2.38  
CASO DE PRUEBA – REGISTRAR USUARIOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 1

Nombre	Registrar usuarios
Descripción	El sistema debe permitir ingresar datos de los usuarios y almacenarlos en la base de datos
Condiciones de ejecución	El administrador, docente, estudiante o representante debe estar registrado en la base de datos
Entrada	<ul style="list-style-type: none"> <li>• El administrador ingresa en la interfaz de administración de usuarios</li> <li>• Selecciona la opción “Ingresar nuevo”</li> <li>• Selecciona el tipo de nuevo usuario: “administrador”, “docente”, “estudiante” o “representante”.</li> <li>• Selecciona el nuevo usuario de una lista de opciones.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos</li> <li>• Nuevo registro de usuario guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.39  
CASO DE PRUEBA – INGRESO AL SISTEMA  
CASO DE PRUEBA – HISTORIA DE USUARIO 2

Nombre	Ingreso al sistema
Descripción	El sistema debe presentar una interfaz de ingreso al sistema
Condiciones de ejecución	El usuario debe estar registrado en la base de datos del sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario ingresa la URL del sistema en un navegador de Internet.</li> <li>• El sistema presenta una página de ingreso.</li> <li>• El usuario ingresa su nombre de usuario y su contraseña, luego hace clic en el botón “Entrar”.</li> <li>• Si los datos son correctos, el sistema permite el ingreso.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Ingreso al sistema si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.40  
CASO DE PRUEBA – PERFIL DE USUARIO  
CASO DE PRUEBA – HISTORIA DE USUARIO 3

Nombre	Perfil de usuario
Descripción	El sistema debe presentar interfaces diferentes de acuerdo con el tipo de usuario
Condiciones de ejecución	El usuario debe estar registrado en la base de datos y debe ingresar al sistema con su nombre de usuario y contraseña
Entrada	<ul style="list-style-type: none"> <li>• El usuario ingresa al sistema con su nombre de usuario y contraseña.</li> <li>• El sistema identifica el rol del usuario.</li> </ul>

	<ul style="list-style-type: none"> <li>El sistema presenta una interfaz de usuario asociada a su rol.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>Interfaces diferentes dependiendo del rol de usuario</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.41  
CASO DE PRUEBA – ACCESO A LA INFORMACIÓN  
CASO DE PRUEBA – HISTORIA DE USUARIO 4

Nombre	Acceso a la información
Descripción	Se comprueban los privilegios asociados con los roles de usuario
Condiciones de ejecución	El usuario debe estar registrado en la base de datos del sistema
Entrada	<ul style="list-style-type: none"> <li>El usuario ingresa al sistema con su nombre de usuario y contraseña.</li> <li>El sistema identifica el rol de usuario.</li> <li>El sistema brinda los privilegios asociados al rol de usuario.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>El usuario administrador debe tener todos los privilegios.</li> <li>El usuario docente debe tener acceso a ver y modificar solo lo que sea parte de su labor docente.</li> <li>Los estudiantes y representantes solo deben poder ver la información relacionada a ellos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.42  
CASO DE PRUEBA – REGISTRAR DOCENTES  
CASO DE PRUEBA – HISTORIA DE USUARIO 5

Nombre	Registrar docentes
Descripción	El sistema debe permitir ingresar datos de los docentes y almacenarlos en la base de datos
Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>El usuario administrador ingresa al sistema</li> <li>El sistema presenta una interfaz de administración de docentes</li> <li>El administrador selecciona la opción “Nuevo”</li> <li>El administrador ingresa la información del docente y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>Validación de datos.</li> <li>Nuevo registro de docente guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.43  
CASO DE PRUEBA – REGISTRAR ESTUDIANTES  
CASO DE PRUEBA – HISTORIA DE USUARIO 6

Nombre	Registrar estudiantes
Descripción	El sistema debe permitir ingresar datos de los estudiantes y almacenarlos en la base de datos

Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario administrador ingresa al sistema</li> <li>• El sistema presenta una interfaz de administración de estudiantes</li> <li>• El administrador selecciona la opción “Nuevo”</li> <li>• El administrador ingresa la información del estudiante y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Nuevo registro de estudiante guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.44  
CASO DE PRUEBA – REGISTRAR REPRESENTANTES  
CASO DE PRUEBA – HISTORIA DE USUARIO 7

Nombre	Registrar representantes
Descripción	El sistema debe permitir ingresar datos de los representantes y almacenarlos en la base de datos
Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario administrador ingresa al sistema</li> <li>• El sistema presenta una interfaz de administración de representantes</li> <li>• El administrador selecciona la opción “Nuevo”</li> <li>• El administrador ingresa la información del representante y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Nuevo registro de representante guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.45  
CASO DE PRUEBA – REGISTRAR ADMINISTRADORES  
CASO DE PRUEBA – HISTORIA DE USUARIO 8

Nombre	Registrar administradores
Descripción	El sistema debe permitir ingresar datos de los administradores y almacenarlos en la base de datos
Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario administrador ingresa al sistema</li> <li>• El sistema presenta una interfaz de administración de administradores</li> <li>• El administrador selecciona la opción “Nuevo”</li> <li>• El administrador ingresa la información del administrador y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Nuevo registro de administrador guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.46  
CASO DE PRUEBA – REGISTRAR CURSOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 9

Nombre	Registrar cursos
Descripción	El sistema debe permitir ingresar datos de los cursos y almacenarlos en la base de datos
Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario administrador ingresa al sistema</li> <li>• El sistema presenta una interfaz de administración de cursos</li> <li>• El administrador selecciona la opción “Nuevo”</li> <li>• El administrador ingresa la información del curso y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Nuevo registro de curso guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.47  
CASO DE PRUEBA – REGISTRAR ASIGNATURAS  
CASO DE PRUEBA – HISTORIA DE USUARIO 10

Nombre	Registrar asignaturas
Descripción	El sistema debe permitir ingresar datos de las asignaturas y almacenarlos en la base de datos
Condiciones de ejecución	El usuario administrador debe ingresar al sistema
Entrada	<ul style="list-style-type: none"> <li>• El usuario administrador ingresa al sistema</li> <li>• El sistema presenta una interfaz de administración de asignaturas</li> <li>• El administrador selecciona la opción “Nuevo”</li> <li>• El administrador ingresa la información de la asignatura y presiona el botón “Guardar”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Validación de datos.</li> <li>• Nuevo registro de asignatura guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

## 2.10 Segunda iteración

Para la segunda iteración se realizan los módulos que permitan realizar la matrícula de estudiantes lo que generará la estructura de calificaciones para que los docentes puedan ingresar notas parciales, exámenes, proyectos educativos, asistencia y comportamiento.

Además, se desarrollan los módulos que permiten realizar las asignaciones de asignaturas a cursos, la carga horaria de trabajo docente, la asignación de docentes tutores a los cursos

y el ingreso de calificaciones de parciales, exámenes, asistencias, comportamientos y proyectos educativos. Las historias de usuario que se desarrollan en la segunda iteración se detallan a continuación:

TABLA 2.48  
PLAN DE DESARROLLO DE LA SEGUNDA ITERACIÓN  
HISTORIAS DE USUARIO EN LA SEGUNDA ITERACIÓN

NÚMERO	NOMBRE	ESFUERZO (SEMANAS)
11	Matricular estudiante	1
12	Asignar asignaturas a cursos	2
13	Asignar carga horaria a docentes	
14	Asignar tutores a cursos	
15	Registrar notas parciales	1
16	Registrar notas de exámenes	1
17	Registrar asistencias	2
18	Registrar comportamientos	
19	Registrar notas de proyectos educativos	
	TOTAL	7

### 2.10.1 Historia de usuario – Matricular estudiante

Las tareas realizadas en el desarrollo de la historia de usuario 11 se detallan a continuación:

- **Desarrollo de interfaces para administración de matrículas**

Se crea las interfaces para que el administrador del sistema pueda crear, ver, y eliminar las matrículas de estudiantes.

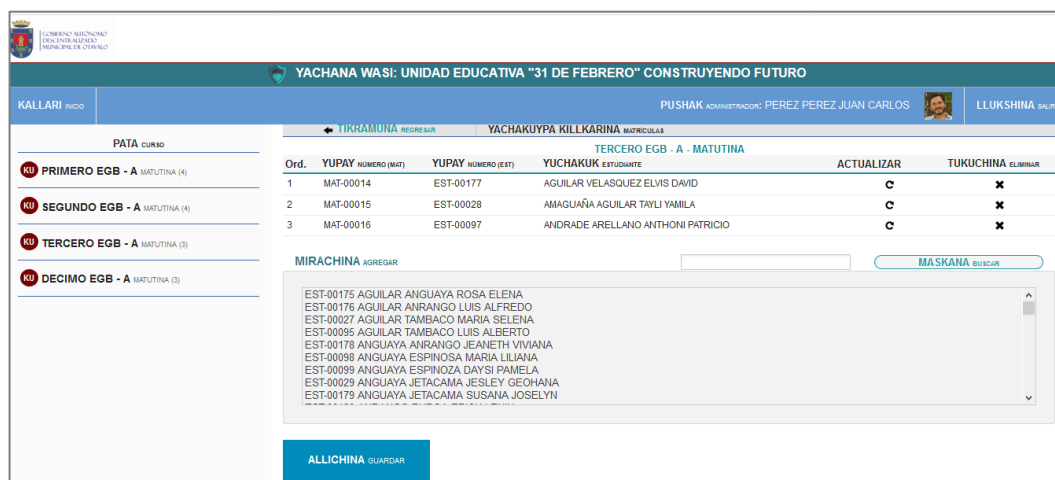


Figura 38: Interfaz de registrar matrículas



- **Validación de datos**

Se codifica los métodos en las clases *Table* y *Entity* del modelo *CursosEstudiantes* que permitan ingresar nuevos registros de acuerdo con el modelo relacional uno a muchos, el *ID* del modelo *Estudiantes* y el *ID* del modelo *Cursos* determinan esta relación. Por cada matrícula se generan más de 200 nuevos registros en la base de datos por la estructura de calificaciones del estudiante, por esta razón, se restringe el límite de matriculación múltiple a un máximo de 5 estudiantes a la vez, a fin de no sobrepasar el límite de tiempo de espera de respuesta de una petición, de acuerdo con el tiempo máximo de ejecución en la configuración estándar de un servidor Apache. Además, se valida que los estudiantes desplegados en la lista de selección solo sean los que aún no se encuentran matriculados.

- **Creación de la estructura de calificaciones**

Se codifica los métodos que permitan crear la totalidad de registros de calificaciones de un estudiante en el momento en que se realice su matriculación, la creación de los nuevos registros de calificaciones tienen prioridad, una vez terminada la creación de registros de calificaciones se registra la nueva matrícula y se devuelve un mensaje de confirmación.

### **2.10.2 Historia de usuario – Determinar asignaturas de un curso**

Las tareas realizadas en el desarrollo de la historia de usuario 12 se detallan a continuación:

- **Desarrollo de interfaces para administración de asignaturas por cursos**

Se crea las interfaces para que el administrador del sistema pueda crear, ver, y eliminar las asignaciones de asignaturas a cursos.

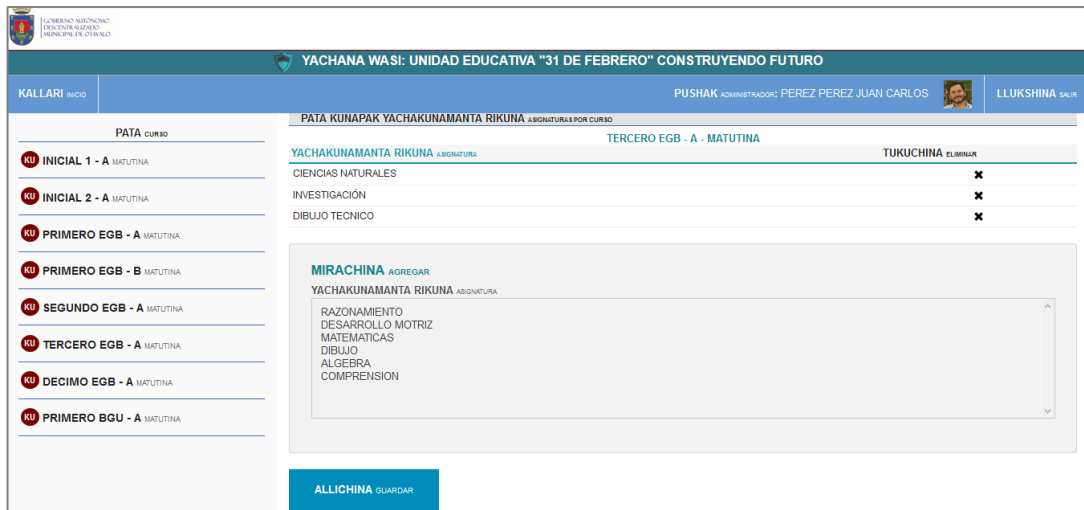


Figura 39: Interfaz de asignación de asignaturas a curso

- **Validación de datos**

Se codifican los métodos en las clases *Table* y *Entity* del modelo *CursosMaterias* que permitan ingresar nuevos registros de acuerdo con el modelo relacional uno a muchos, el *ID* del modelo *Cursos* y el *ID* del modelo *Materias* determinan esta relación. Para evitar la asignación de una misma asignatura dos veces a un curso se define en la tabla *CursosMaterias* una clave *Unique* compuesta.

- **Controlador**

En el controlador de *CursosMaterias* se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de *CursosMaterias* por curso.
- `add()`: Permite registrar la información de una nueva asignación de una asignatura a un curso.
- `delete()`: Permite eliminar un registro de *CursosMaterias*, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del registro.

### 2.10.3 Historia de usuario – Asignar carga horaria a docentes

Las tareas realizadas en el desarrollo de la historia de usuario 13 se detallan a continuación:

- **Desarrollo de interfaces para administración de carga horaria docente**

Se crea las interfaces para que el administrador del sistema pueda crear, ver, y eliminar las asignaciones de carga horaria a los docentes.

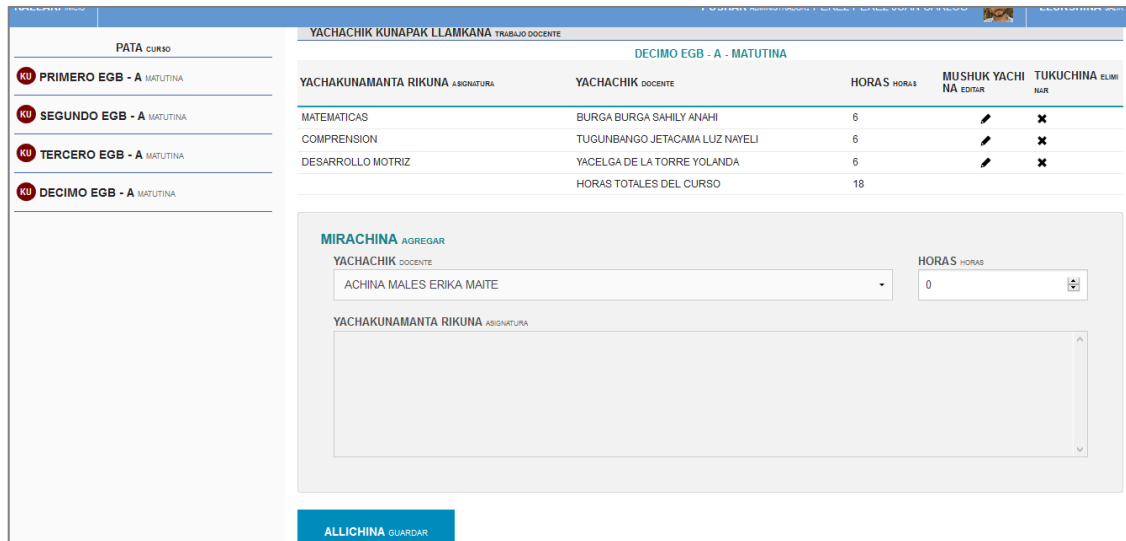


Figura 40: Interfaz de asignación de carga horaria a docentes

- **Validación de datos**

Se codifica los métodos en las clases *Table* y *Entity* del modelo *DocentesMaterias* que permitan ingresar nuevos registros de acuerdo con el modelo relacional uno a muchos, el *ID* del modelo *Docentes* y el *ID* del modelo *Materias* determinan esta relación. Para evitar la asignación de una misma asignatura dos veces a un docente en un mismo curso se define en la tabla *DocentesMaterias* una clave *Unique* compuesta.

- **Controlador**

En el controlador de *CursosMaterias* se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de *DocentesMaterias* por docente.
- `add()`: Permite registrar la información de una nueva asignación de una asignatura de un curso a un docente.
- `delete()`: Permite eliminar un registro de *DocentesMaterias*, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del registro.

## 2.10.4 Historia de usuario – Asignar tutores a cursos

Las tareas realizadas en el desarrollo de la historia de usuario 14 se detallan a continuación:

- **Desarrollo de interfaces para administración de tutores**

Se crea las interfaces para que el administrador del sistema pueda crear, ver, y eliminar las asignaciones de tutores a cursos.

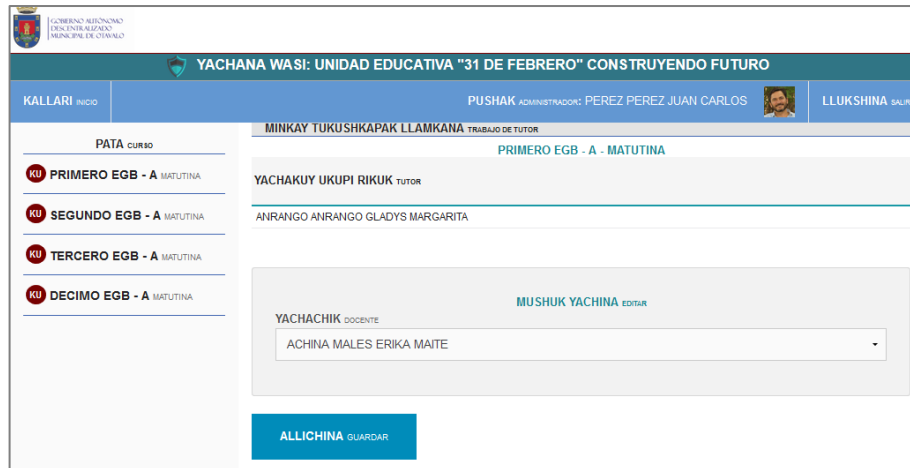


Figura 41: Interfaz de asignación de tutores a cursos

- **Validación de datos**

Se codifica los métodos en las clases *Table* y *Entity* del modelo *Tutores* que permitan ingresar nuevos registros de acuerdo con el modelo relacional uno a muchos, el *ID* del modelo *Docentes* y el *ID* del modelo *Cursos* determinan esta relación. Para evitar la asignación de un mismo tutor dos veces a un mismo curso se define en la tabla *Tutores* una clave *Unique* compuesta.

- **Controlador**

En el controlador de tutores se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar todos los registros de tutores por curso.
- `add()`: Permite registrar la información de una nueva asignación de un tutor a un curso.
- `delete()`: Permite eliminar un registro de tutores, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del registro.

### 2.10.5 Historia de usuario – Registrar notas parciales

Las tareas realizadas en el desarrollo de la historia de usuario 15 se detallan a continuación:

- **Desarrollo de interfaz de ingreso de notas parciales**

Se crea las interfaces para que el administrador y el docente puedan ingresar, ver, y modificar las notas parciales de los estudiantes.

	INSUMO1	INSUMO2	INSUMO3	INSUMO4	INSUMO5	INSUMO6	INSUMO7	INSUMO8	PROM.
1 AGUILAR CACHIMUEL WILLIAM ANTONY									
2 AGUILAR MALDONADO KEVIN JOEL									
3 AGUILAR PERALTA MICHAEL NEPTALY									
4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA									

Figura 42: Interfaz el ingreso de notas parciales

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de la interfaz de ingreso de notas parciales en las clases *Table* y *Entity* del modelo *Parciales*. El sistema verificará que las notas ingresadas sean válidas ( $\geq 0$ ,  $\leq$  a la nota máxima configurada) y realizará el cálculo del campo promedio antes de modificar el registro en la base de datos.

- **Controlador**

En el controlador de parciales se desarrollan los siguientes métodos para el manejo de sus entidades:

- `add()`: Permite crear un nuevo registro de notas parciales. Esta función es utilizada en el momento de crear la matrícula de estudiante.
- `editd()`: Permite que los docentes puedan ingresar las notas parciales por curso.
- `edita()`: Permite que los administradores puedan ingresar las notas parciales por alumno.

La información de las notas parciales es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar los registros de notas parciales.

### 2.10.6 Historia de usuario – Registrar notas de exámenes

Las tareas realizadas en el desarrollo de la historia de usuario 16 se detallan a continuación:

- **Desarrollo de interfaz de ingreso de notas de exámenes**

Se crea las interfaces para que el administrador y el docente puedan ingresar, ver, y modificar las notas de exámenes de los estudiantes.

RAZONAMIENTO	
1 AGUILAR CACHIMUEL WILLIAM ANTONY	10
2 AGUILAR MALDONADO KEVIN JOEL	
3 AGUILAR PERALTA MICHAEL NEPTALY	
4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA	

Figura 43: Interfaz de ingreso de notas de exámenes

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de la interfaz de ingreso de notas de exámenes en las clases *Table* y *Entity* del modelo *Exámenes*. El sistema verificará que las notas ingresadas sean válidas ( $\geq 0$ ,  $\leq$  que la nota máxima configurada).

- **Controlador**

En el controlador de exámenes se desarrollan los siguientes métodos para el manejo de sus entidades:

- `add()`: Permite crear un nuevo registro de notas de exámenes. Esta función es utilizada en el momento de crear la matrícula de estudiante.

- editd(): Permite que los docentes puedan ingresar las notas de exámenes por curso.
- edita(): Permite que los administradores puedan ingresar las notas de exámenes por alumno.

La información de las notas de exámenes es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar los registros de notas de exámenes.

### 2.10.7 Historia de usuario – Registrar asistencias

Las tareas realizadas en el desarrollo de la historia de usuario 17 se detallan a continuación:

- **Desarrollo de interfaz de ingreso de asistencias**

Se crea las interfaces para que el administrador y el docente puedan ingresar, ver, y modificar los registros de asistencias de los estudiantes.

FALTAS JUSTIFICADAS	
1 AGUILAR CACHIMUEL WILLIAM ANTONY	<input type="text"/>
2 AGUILAR MALDONADO KEVIN JOEL	<input type="text"/>
3 AGUILAR PERALTA MICHAEL NEPTALY	<input type="text"/>
4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA	<input type="text"/>

ALLICHINA GUARDAR

Figura 44: Interfaz de ingreso de asistencias

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de la interfaz de ingreso de asistencias en las clases *Table* y *Entity* del modelo *Asistencias*. El sistema verificará que las asistencias sean números enteros antes de guardar los registros en la base de datos.

- **Controlador**

En el controlador de asistencias se desarrollan los siguientes métodos para el manejo de sus entidades:

- add(): Permite crear un nuevo registro de asistencias. Esta función es utilizada en el momento de crear la matrícula de estudiante.
- editd(): Permite que los docentes puedan ingresar las asistencias por curso.
- edita(): Permite que los administradores puedan ingresar las asistencias por alumno.

La información de asistencias es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar los registros de asistencias.

### 2.10.8 Historia de usuario – Registrar comportamientos

En el desarrollo de la historia de usuario número 18 se llevan a cabo las siguientes tareas:

- **Desarrollo de interfaz de ingreso de comportamientos**

Se crea las interfaces para que el administrador y el docente puedan ingresar, ver, y modificar los registros de comportamientos de los estudiantes.

The screenshot shows a web application interface. At the top, there is a navigation bar with a back arrow, the text 'TIKRAMUNA', and a 'REGRESAR' link. To the right, it says 'YACHAKUY UKUPI RIKUK TUTOR / PARCIAL 1 - QUIMESTRE 1'. Below this, the main content area is titled 'SEGUNDO EGB - A - MATUTINA'. Underneath, the word 'RESPETO' is centered. There is a table with four rows of student names and a dropdown menu for each row. The first three rows have empty dropdowns, and the fourth row has 'B' selected. Below the table, there is a blue button labeled 'ALLICHINA GUARDAR'. To the right of the button, a dropdown menu is open, showing options A, B, C, D, and E, with 'B' highlighted in blue.

RESPETO	
1 AGUILAR CACHIMUEL WILLIAM ANTONY	
2 AGUILAR MALDONADO KEVIN JOEL	
3 AGUILAR PERALTA MICHAEL NEPTALY	
4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA	B

ALLICHINA GUARDAR

- A
- B
- C
- D
- E

Figura 45: Interfaz de ingreso de comportamientos

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de la interfaz de ingreso de comportamientos en las clases *Table* y *Entity* del modelo *Comportamientos*. El sistema verificará que las calificaciones de los comportamientos sean cualitativas (A, B, C, D, E dependiendo de la configuración) antes de guardar los registros en la base de datos.



- **Controlador**

En el controlador de comportamientos se desarrollan los siguientes métodos para el manejo de sus entidades:

- `add()`: Permite crear un nuevo registro de comportamientos. Esta función es utilizada en el momento de crear la matrícula de estudiante.
- `editd()`: Permite que los docentes puedan ingresar las calificaciones de comportamientos por curso.
- `edita()`: Permite que los administradores puedan ingresar las calificaciones de comportamientos por alumno.

La información de comportamientos es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar los registros de comportamientos.

### 2.10.9 Historia de usuario – Registrar proyectos educativos

En el desarrollo de la historia de usuario número 19 se llevan a cabo las siguientes tareas:

- **Desarrollo de interfaz de ingreso de proyectos educativos**

Se crea las interfaces para que el administrador y el docente puedan ingresar, ver, y modificar los registros de proyectos educativos de los estudiantes.

The screenshot shows a web interface for entering educational projects. At the top, there is a navigation bar with a back arrow, the text 'TIKRAMUNA REGRESAR', and 'YACHAKUY UKUPI RIKUK TUTOR / PARCIAL 2 - QUIMESTRE 1'. Below this is a header 'SEGUNDO EGB - A - MATUTINA'. The main content area is titled 'PROYECTO EDUCATIVO' and contains a table with four rows of student names. Each row has a dropdown menu to its right. The first three rows have empty dropdowns, while the fourth row has a dropdown menu open, showing the options 'EX', 'MB', 'B', and 'R'. Below the table is a blue button labeled 'ALLICHINA GUARDAR'.

PROYECTO EDUCATIVO	
1 AGUILAR CACHIMUEL WILLIAM ANTONY	
2 AGUILAR MALDONADO KEVIN JOEL	
3 AGUILAR PERALTA MICHAEL NEPTALY	
4 OYAGATA AGUAGALLO JENNIFFER ALEXANDRA	EX MB B R

ALLICHINA GUARDAR

Figura 46: Interfaz de ingreso de proyectos educativos

- **Validación de datos**

Se codifica los métodos para la validación y el tratamiento de la información que se obtiene de la interfaz de ingreso de proyectos educativos en las clases *Table* y *Entity* del modelo *Proyectos*. El sistema verificaraá que las calificaciones de proyectos

educativos sean cualitativas (EX, MB, B, R dependiendo de la configuración) antes de guardar los registros en la base de datos.

- **Controlador**

En el controlador de proyectos educativos se desarrollan los siguientes métodos para el manejo de sus entidades:

- add(): Permite crear un nuevo registro de proyectos educativos. Esta función es utilizada en el momento de crear la matrícula de estudiante.
- editd(): Permite que los docentes o administradores puedan ingresar las calificaciones de proyectos educativos por curso.
- edita(): Permite que los docentes o administradores puedan ingresar las calificaciones de proyectos educativos por alumno

La información de proyectos educativos es necesaria para los reportes históricos de calificaciones, por esta razón, no se codifican los métodos que permitan eliminar los registros de proyectos educativos.

### 2.10.10 Pruebas de aceptación

Para las pruebas de aceptación de las historias de usuario definidas en el plan de desarrollo de la segunda iteración se opta por hacer uso del servicio de alojamiento gratuito de 000Webhost<sup>18</sup> para subir el avance del proyecto. Los casos de prueba se realizan con datos reales de la institución. Estos casos de prueba realizados se detallan a continuación:

TABLA 2.49  
CASO DE PRUEBA – MATRICULAR ESTUDIANTE  
CASO DE PRUEBA – HISTORIA DE USUARIO 11

Nombre	Matricular estudiante
Descripción	El sistema debe permitir registrar la matrícula de un estudiante a un curso
Condiciones de ejecución	El administrador debe ingresar al sistema, debe haber cursos y estudiantes registrados en el sistema.
Entrada	<ul style="list-style-type: none"> <li>• El administrador ingresa en la interfaz de administración de matrículas</li> <li>• Selecciona la opción “Nuevo”.</li> <li>• Selecciona el curso.</li> <li>• Selecciona el o los estudiantes de una lista de opciones.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro de matrícula guardado en la base de datos, estructura de calificaciones creada y mensaje de confirmación si los datos son correctos.</li> </ul>

<sup>18</sup> 000Webhost es un servicio de alojamiento para aplicaciones PHP, MySQL. Servicio brindado por la empresa HOSTINGER para que desarrolladores evalúen sus aplicaciones de forma gratuita, sin necesidad de tarjeta de crédito y sin publicidad.

	<ul style="list-style-type: none"> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.50  
CASO DE PRUEBA – ASIGNAR ASIGNATURAS A CURSOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 12

Nombre	Asignar asignaturas a cursos
Descripción	El sistema debe permitir asignar las asignaturas que se enseñarán en un curso
Condiciones de ejecución	El administrador debe ingresar al sistema, debe haber cursos y asignaturas registrados en el sistema.
Entrada	<ul style="list-style-type: none"> <li>• El administrador ingresa en la interfaz de administración de Asignaturas por curso</li> <li>• Selecciona la opción “Nuevo”.</li> <li>• Selecciona el curso.</li> <li>• Selecciona la o las asignaturas de una lista de opciones.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.51  
CASO DE PRUEBA – ASIGNAR CARGA HORARIA A DOCENTES  
CASO DE PRUEBA – HISTORIA DE USUARIO 13

Nombre	Asignar carga horaria a docentes
Descripción	El sistema debe permitir asignar las asignaturas que enseñará un docente en uno o varios cursos.
Condiciones de ejecución	El administrador debe ingresar al sistema, debe haber cursos, asignaturas y docentes registrados en el sistema.
Entrada	<ul style="list-style-type: none"> <li>• El administrador ingresa en la interfaz de administración de Asignaturas por docente</li> <li>• Selecciona la opción “Nuevo”.</li> <li>• Selecciona el curso.</li> <li>• Selecciona la asignatura y el docente de las listas de opciones.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.52  
CASO DE PRUEBA – ASIGNAR TUTORES A CURSOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 14

Nombre	Asignar tutores a cursos
Descripción	El sistema debe permitir asignar docentes tutores a los cursos.
Condiciones de ejecución	El administrador debe ingresar al sistema, debe haber cursos y docentes registrados en el sistema.
Entrada	<ul style="list-style-type: none"> <li>• El administrador ingresa en la interfaz de administración de tutores.</li> </ul>

	<ul style="list-style-type: none"> <li>• Selecciona la opción “Nuevo”.</li> <li>• Selecciona el curso.</li> <li>• Selecciona el docente de una lista de opciones.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro de tutores guardado en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.53  
CASO DE PRUEBA – REGISTRAR NOTAS PARCIALES  
CASO DE PRUEBA – HISTORIA DE USUARIO 15

Nombre	Registrar notas parciales
Descripción	El sistema debe permitir registrar notas parciales
Condiciones de ejecución	El administrador o docente debe ingresar al sistema, debe haber estudiantes matriculados y con su estructura de calificaciones creada.
Entrada	<ul style="list-style-type: none"> <li>• El administrador o docente ingresa en la interfaz de ingreso de notas parciales.</li> <li>• Ingresa notas cuantitativas mayores o iguales a 0 y menores o iguales a 10</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Notas parciales guardadas en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.54  
CASO DE PRUEBA – REGISTRAR NOTAS DE EXAMENES  
CASO DE PRUEBA – HISTORIA DE USUARIO 16

Nombre	Registrar notas de exámenes
Descripción	El sistema debe permitir registrar notas de exámenes
Condiciones de ejecución	El administrador o docente debe ingresar al sistema, debe haber estudiantes matriculados y con su estructura de calificaciones creada.
Entrada	<ul style="list-style-type: none"> <li>• El administrador o docente ingresa en la interfaz de ingreso de notas de exámenes.</li> <li>• Ingresa notas cuantitativas mayores o iguales a 0 y menores o iguales a 10</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Notas de exámenes guardadas en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.55  
CASO DE PRUEBA – REGISTRAR ASISTENCIAS  
CASO DE PRUEBA – HISTORIA DE USUARIO 17

Nombre	Registrar asistencias
Descripción	El sistema debe permitir registrar asistencias
Condiciones de ejecución	El administrador o docente debe ingresar al sistema, debe haber estudiantes matriculados y con su estructura de calificaciones creada.
Entrada	<ul style="list-style-type: none"> <li>• El administrador o docente ingresa en la interfaz de ingreso de asistencias.</li> <li>• Ingresa los números de días asistidos.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Asistencias guardadas en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.56  
CASO DE PRUEBA – REGISTRAR COMPORTAMIENTOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 18

Nombre	Registrar comportamientos
Descripción	El sistema debe permitir registrar comportamientos
Condiciones de ejecución	El administrador o docente debe ingresar al sistema, debe haber estudiantes matriculados y con su estructura de calificaciones creada.
Entrada	<ul style="list-style-type: none"> <li>• El administrador o docente ingresa en la interfaz de ingreso de comportamientos.</li> <li>• Selecciona las calificaciones cualitativas de las listas desplegables.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Comportamientos guardados en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.57  
CASO DE PRUEBA – REGISTRAR PROYECTOS EDUCATIVOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 19

Nombre	Registrar proyectos educativos
Descripción	El sistema debe permitir registrar proyectos educativos
Condiciones de ejecución	El administrador o docente debe ingresar al sistema, debe haber estudiantes matriculados y con su estructura de calificaciones creada.
Entrada	<ul style="list-style-type: none"> <li>• El administrador o docente ingresa en la interfaz de ingreso de proyectos educativos.</li> <li>• Selecciona las calificaciones cualitativas de las listas desplegables.</li> <li>• Hace clic en el botón “GUARDAR”.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Proyectos educativos guardados en la base de datos y mensaje de confirmación si los datos son correctos.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

## 2.11 Tercera iteración

En base al plan de desarrollo, para la tercera iteración se realizan los módulos de administración de documentos, comunicados y reportes del sistema. Las historias de usuario que se desarrollan en la tercera iteración se detallan a continuación:

TABLA 2.58  
PLAN DE DESARROLLO DE LA TERCERA ITERACIÓN

NÚMERO	NOMBRE	ESFUERZO (SEMANAS)
20	Administración de documentos	3
21	Publicación de notificaciones	
22	Generación de reportes	3
	TOTAL	6

### 2.11.1 Historia de usuario – Administración de documentos

Las tareas realizadas en el desarrollo de la historia de usuario 20 se detallan a continuación:

- **Desarrollo de interfaces para administración de documentos**

Se crea las interfaces para que los administradores y docentes puedan subir, ver, editar y eliminar documentos.



The screenshot shows a web interface for document management. At the top, there is a breadcrumb trail: "MUTUSHKIA PANKA documentos / PEREZ PEREZ JUAN CARLOS". Below this is a table with the following columns: "YUPAY NÚMERO", "SHUTI NOMBRE", "MUTUSHKIA PANKA DOCUMENTO", "PACHA FECHA", and "RIKUNA ve R". The table contains five rows of document entries:

YUPAY NÚMERO	SHUTI NOMBRE	MUTUSHKIA PANKA DOCUMENTO	PACHA FECHA	RIKUNA ve R
DOC-00012	Silabo	Silabo matemáticas.docx	02/07/2018 5:09 pm	🔍
DOC-00013	Planificación	planificación parcial1.docx	02/07/2018 5:11 pm	🔍
DOC-00014	Nómina 2DO EGB C	Nómina segundo EGB C.xlsx	02/07/2018 5:12 pm	🔍
DOC-00019	MYDOC1	datosboon.pdf	12/07/2018 9:33 am	🔍
DOC-00020	3b	Openshift caaapp.docx	12/07/2018 11:47 pm	🔍

Figura 47: Interfaz principal de administración de documentos

MUTUSHKIA PANKA DOCUMENTOS

MUSHUK NUEVO

SHUTI NOMBRE \*

ALLIPACHA WILLACHIK DETALLES

RIKURISHPA KATINA OBSERVACIONES

MUTUSHKIA PANKA DOCUMENTO \*

Seleccionar archivo | Ningún archivo seleccionado

ALLICHINA GUARDAR

Figura 48: Interfaz para subir documentos

- **Validación de datos**

Para la validación y el tratamiento de la información que se obtiene del formulario para subir documentos se utiliza el plugin para CakePHP llamado "Upload" en la clase *Table* del modelo *Documentos*. El sistema guardará el archivo en el directorio "webroot" y transformará el nombre del documento a un número único de identificación manteniendo su extensión. Esto permite subir varias versiones de un archivo con el mismo nombre al sistema.

- **Controlador**

En el controlador de documentos se desarrollan los siguientes métodos para el manejo de entidades y archivos:

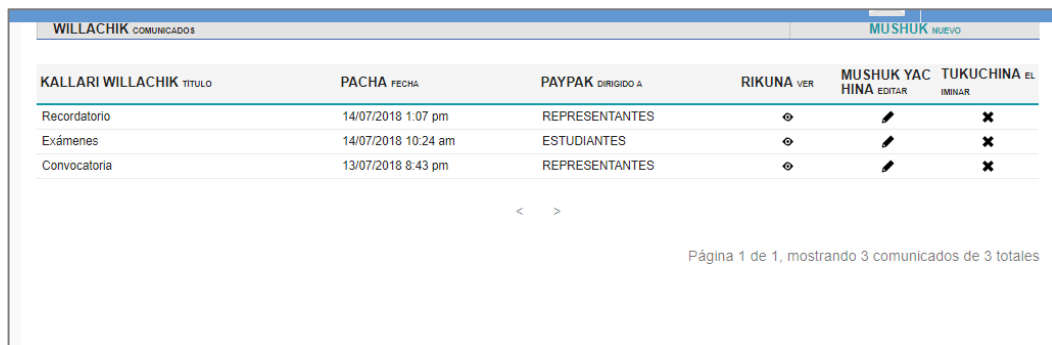
- `index()`: Permite listar todos los registros de documentos.
- `view()`: Permite ver la información de un documento específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo documento y subir un nuevo archivo al directorio webroot del sistema.
- `edit()`: Permite editar los campos de un registro de documentos, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del documento.
- `delete()`: Permite eliminar un registro de documentos junto con su archivo, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del documento.

### 2.11.2 Historia de usuario – Publicación de notificaciones

Las tareas realizadas en el desarrollo de la historia de usuario 21 se detallan a continuación:

- **Desarrollo de interfaces para administración de notificaciones**

Se crea las interfaces para que los administradores y docentes puedan crear, ver, editar y eliminar notificaciones.



KALLARI WILLACHIK TITULO	PACHA FECHA	PAYPAK DIRIGIDO A	RIKUNA VER	MUSHUK YAC HINA EDITAR	TUKUCHINA ELIMINAR
Recordatorio	14/07/2018 1:07 pm	REPRESENTANTES	👁	✎	✖
Exámenes	14/07/2018 10:24 am	ESTUDIANTES	👁	✎	✖
Convocatoria	13/07/2018 8:43 pm	REPRESENTANTES	👁	✎	✖

< >

Página 1 de 1, mostrando 3 comunicados de 3 totales

Figura 49: Interfaz principal para administración de comunicados

- **Validación de datos**

Se codifica los métodos en las clases *Table* y *Entity* del modelo *Mensajes* que permitan publicar un mensaje de manera general para un tipo de rol de usuario o para un curso o varios cursos específicos. Cuando un mensaje sea dirigido de forma general para un tipo de rol de usuario el campo *cursos\_ids* se guardará vacío y cuando el mensaje sea dirigido para cursos específicos los IDs de los cursos destinatarios se guardarán en forma de cadena de texto utilizando la función *implode()* de PHP.

- **Controlador**

En el controlador de mensajes se desarrollan los siguientes métodos para el manejo de sus entidades:

- `index()`: Permite listar los mensajes.
- `view()`: Permite ver la información de un mensaje específico recibiendo como parámetro su ID.
- `add()`: Permite registrar la información de un nuevo mensaje.
- `edit()`: Permite editar los campos de un registro, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del mensaje.



- delete(): Permite eliminar un registro de mensajes, tiene como condición que la petición sea de tipo *Post* y recibe como parámetro el ID del mensaje.

### 2.11.3 Historia de usuario – Generación de reportes

Para la elaboración de reportes se utilizará las librerías de FPDF y PHPEXcel.

- **Integración de FPDF y CakePHP**

Para integrar la librería FPDF en CakePHP se debe instalar el paquete llamado “setasign” con el gestor de dependencias “composer”.

```
C:\xampp\htdocs\academico>composer require setasign/fpdf
Failed to decode response: zlib_decode(): data error
Retrying with degraded mode, check https://getcomposer.org/doc/articles/troubleshooting.md#degraded-mode for more info
Using version ^1.8 for setasign/fpdf
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
 - Installing setasign/fpdf (1.8.1): Loading from cache
Writing lock file
Generating autoload files
> Cake\Composer\Installer\PluginInstaller::postAutoloadDump
C:\xampp\htdocs\academico>
```

Figura 50: Instalación de setasign/fpdf

La instalación se realiza en el directorio “vendor”, su estructura se muestra a continuación:

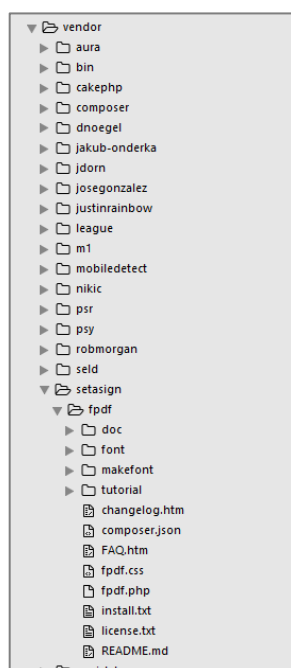


Figura 51: Estructura de setasign/fpdf en el proyecto

Para utilizar la librería desde el controlador de reportes se la instala de la siguiente forma:

```
1 <?php
2 namespace App\Controller;
3
4 use App\Controller\AppController;
5 use Setasign\Fpdf;
6
...
56 public function parcial()
57 {
58     $pdf = new \FPDF('P','mm','A4');
59 }
```

Figura 52: Llamado y utilización de fpdf

- **Integración de PHPExcel y CakePHP**

Para integrar la librería PHPExcel en CakePHP se debe instalar el plugin llamado “dakota/cake-excel” con el gestor de dependencias “composer”.

```
C:\xampp\htdocs\academico>composer require dakota/cake-excel
Using version ^2.0 for dakota/cake-excel
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 2 installs, 0 updates, 0 removals
 - Installing phpooffice/phpexcel (1.8.1): Loading from cache
 - Installing dakota/cake-excel (v2.0.1): Loading from cache
Package phpooffice/phpexcel is abandoned, you should avoid using it. Use phpooffice/phpspreadsheet instead.
Writing lock file
Generating autoload files
> Cake\Composer\Installer\PluginInstaller::postAutoloadDump
C:\xampp\htdocs\academico>
```

La instalación se realiza en el directorio “vendor”, su estructura se muestra a continuación:

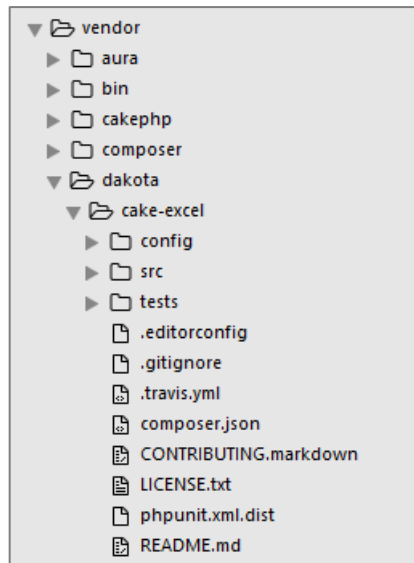


Figura 53: Estructura de dakota/cake-excel en el proyecto

- **Codificación**

Para la generación de reportes se crean las clases controladoras: "ReportespdfController", "ReportesresumenpdfController" y "ReportesxlsController".

- ReportespdfController: Contiene las funciones y métodos necesarios para generar los reportes en formato PDF de: matrícula por curso, matrícula por estudiante, parcial por curso, parcial por estudiante, quimestral por curso, quimestral por estudiante, final por curso, final por estudiante, promoción por curso y promoción por estudiante.
- ReportesresumenpdfController: Contiene las funciones y métodos necesarios para generar los reportes en formato PDF de: parcial por curso, parcial por asignatura, quimestral por curso, quimestral por asignatura, final por curso y final por asignatura.
- ReportesxlsController: Contiene las funciones y métodos necesarios para generar los reportes en formato XLS de: parcial por curso, parcial por asignatura, quimestral por curso, quimestral por asignatura, final por curso y final por asignatura.

#### 2.11.4 Pruebas de aceptación

Para las pruebas de aceptación de las historias de usuario definidas en el plan de desarrollo de la tercera iteración se continúa utilizando el servicio de alojamiento gratuito de 000Webhost.com. Los casos de prueba se realizan con datos reales de la institución. Los casos de prueba realizados se detallan a continuación:

TABLA 2.59  
CASO DE PRUEBA – ADMINISTRACIÓN DE DOCUMENTOS  
CASO DE PRUEBA – HISTORIA DE USUARIO 20

Nombre	Administración de documentos
Descripción	El sistema debe permitir subir documentos a los docentes para revisión de los administradores.
Condiciones de ejecución	El docente o administrador debe ingresar al sistema.
Entrada	<ul style="list-style-type: none"> <li>• El docente o administrador ingresa en la interfaz de administración de documentos</li> <li>• El docente selecciona la opción “NUEVO”</li> <li>• El docente llena los campos de información y selecciona el archivo.</li> <li>• El docente hace clic en el botón “GUARDAR”.</li> <li>• Luego de subido el documento, el docente o administrador selecciona la opción “EDITAR” para modificar los campos de información o la opción “ELIMINAR” para eliminar el documento.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro de documento guardado en la base de datos, nuevo documento subido al directorio de documentos del sistema al momento de su creación.</li> <li>• Mensajes de confirmación si el documento se ha subido, modificado o eliminado con éxito.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.60  
CASO DE PRUEBA – PUBLICACIÓN DE NOTIFICACIONES  
CASO DE PRUEBA – HISTORIA DE USUARIO 21

Nombre	Publicación de notificaciones
Descripción	El sistema debe permitir publicar notificaciones a docentes y administradores.
Condiciones de ejecución	El docente o administrador debe ingresar al sistema.
Entrada	<ul style="list-style-type: none"> <li>• El docente o administrador ingresa en la interfaz de administración de notificaciones.</li> <li>• El docente o administrador selecciona la opción “NUEVO”</li> <li>• El docente o administrador llena los campos de información del comunicado y hace clic en el botón “GUARDAR”.</li> <li>• Luego de creado el comunicado, el docente o administrador selecciona la opción “EDITAR” para modificar los campos de información o la opción “ELIMINAR” para eliminar el comunicado.</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Nuevo registro de comunicado guardado en la base de datos al momento de su creación.</li> <li>• mensajes de confirmación si el comunicado se ha creado, modificado o eliminado con éxito.</li> <li>• Mensaje de alerta si los datos son incorrectos.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

TABLA 2.61  
CASO DE PRUEBA – GENERACIÓN DE REPORTES  
CASO DE PRUEBA – HISTORIA DE USUARIO 21

Nombre	Publicación de notificaciones
Descripción	El sistema debe permitir publicar notificaciones a docentes y administradores.
Condiciones de ejecución	El administrador, docente, estudiante o representante deben ingresar al sistema.
Entrada	<ul style="list-style-type: none"> <li>• El administrador, docente, estudiante o representante ingresa en la interfaz de reportes disponible para su tipo de usuario.</li> <li>• El administrador, docente, estudiante o representante hace clic en la opción disponible de reporte (PDF o Excel).</li> </ul>
Resultado esperado	<ul style="list-style-type: none"> <li>• Descarga del reporte seleccionado.</li> </ul>
Encargada de pruebas (Tester)	Ing. Santillán Chiza Ana Cristina
Evaluación de la prueba	Satisfactoria

# CAPÍTULO 3

## RESULTADOS

### 3.1 Instalación

El sistema está desarrollado para que una persona con conocimientos básicos en administración de sitios web pueda realizar su instalación de forma fácil y con pocos requerimientos de hardware y software.

#### 3.1.1 Requerimientos

La instalación no dependerá de la plataforma (Windows, Linux, etc.), dependerá de los componentes de software y de hardware. Los requisitos básicos para la instalación se detallan a continuación:

TABLA 3.1  
REQUERIMIENTOS DE HARDWARE Y SOFTWARE

Servidor	Apache 2.4.x
Lenguaje de programación	PHP versión 5.6.x
SGBD	MySQL 5.6.x o MariaDB 10.1.x
Espacio disponible en disco duro	4BG o superior
Memoria RAM	1GB o superior
Periféricos (E/S)	Monitor, mouse, teclado
Conexión a internet	3Mbps o superior
Navegador de Internet	Google Chrome, Mozilla Firefox, Windows Internet Explorer, entre otros.

#### 3.1.2 Alojamiento

La estructura del proyecto creado con CakePHP integra en un solo directorio todos los archivos y componentes necesarios. Para el despliegue de la aplicación se copia el directorio del proyecto en la carpeta de aplicaciones del servidor.

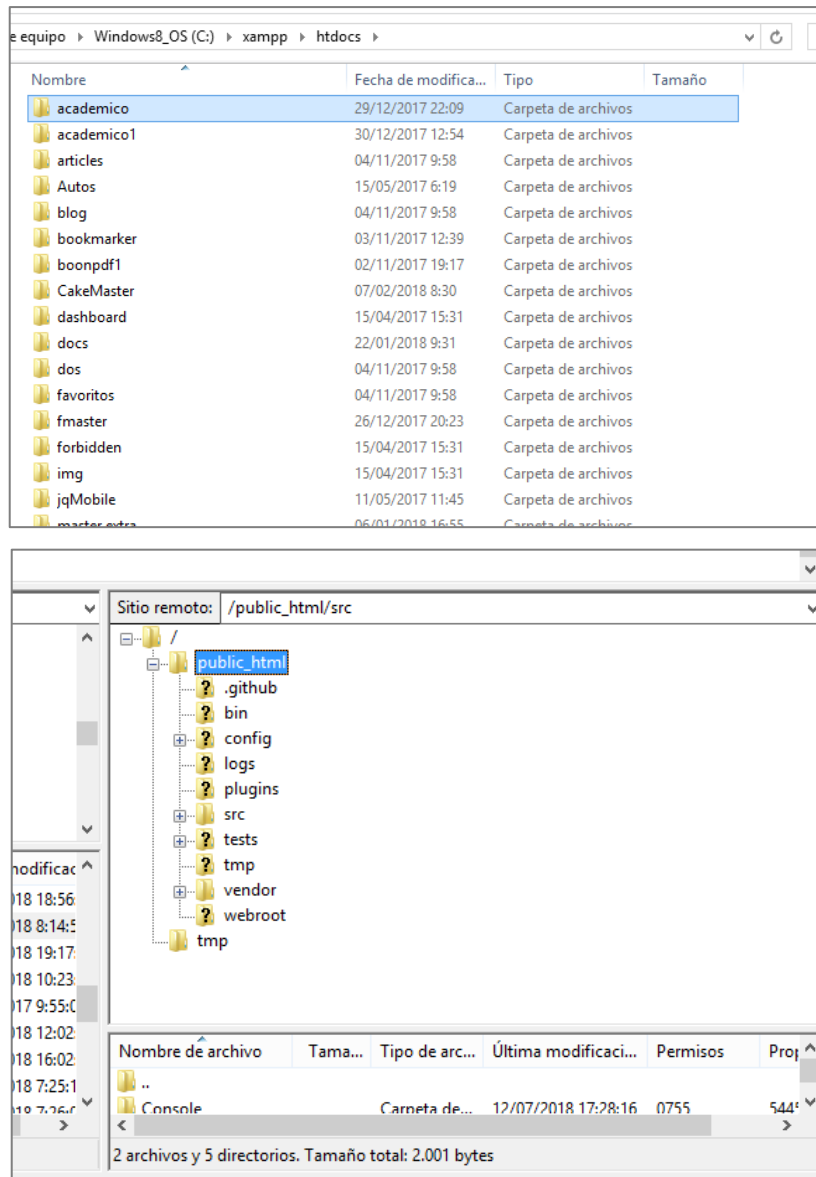


Figura 54: Alojamiento del Proyecto (Localhost y host remoto)

### 3.1.3 Conexión con la base de datos

Para realizar la conexión con la base de datos no se necesita ningún complemento o configuración especial, CakePHP tiene los drivers necesarios para la conexión con MySQL, solo es necesario llenar los parámetros de conexión “database”, “username” y “password” del archivo de configuración.

```

*/
'Datasources' => [
  'default' => [
    'className' => 'Cake\Database\Connection',
    'driver' => 'Cake\Database\Driver\Mysql',
    'persistent' => false,
    'host' => 'localhost',
    /**
     * CakePHP will use the default DB port based on
     * the configuration selected
     * MySQL on MAMP uses port 8889, MAMP users
     * the following line and set the port accordingly
     */
    //'port' => 'non_standard_port_number',
    'username' => 'root',
    'password' => 'root',
    'database' => 'academico',
    'encoding' => 'utf8',
    'timezone' => 'UTC',
    'flags' => [],
    'cacheMetadata' => true,
    'log' => false,
  ],
],

```

Figura 55: Conexión con la base de datos

### 3.2 Despliegue

El sistema permite al responsable de la administración de la aplicación crear las instituciones educativas que usarán el servicio gratuito. Para crear una nueva institución educativa se requiere los nombres, apellidos y cédula del rector de la institución, lo que genera un usuario con rol “ADMINISTRADOR” que ya puede ingresar al sistema y hacer uso de todas sus funcionalidades.

The screenshot shows a web application interface for managing educational units. On the left, there is a sidebar titled "INSTITUCIONES" with a sub-section "ADMIN". Below it, a list of existing units is shown, including "UNIDAD EDUCATIVA \*31 DE FEBRERO\*", "UNIDAD EDUCATIVA 24 DE MAYO", "UNIDAD EDUCATIVA 11 DE AGOSTO", "UNIDAD EDUCATIVA 22 DE JUNIO", "UNIDAD EDUCATIVA 9 DE OCTUBRE", and "27 de febrero". The main content area is titled "YACHANA WASI UNIDAD EDUCATIVA" and "MUSHUK NUEVO". It contains a form for creating a new unit with the following fields: "SHUTI NOMBRE" (text input), "ALMACENAMIENTO (GB)" (text input with a dropdown arrow), "SHUTI NOMBRES" (text input), "KA STAKUNA APELLIDOS" (text input), and "KAW SAYUK CEDULA" (text input). At the bottom of the form is a blue button labeled "ALLICHINA GUARDAR".

Figura 56: Administración de unidades educativas



### 3.2.1 Funciones del sistema para el tipo de usuario administrador

El sistema permite al administrador realizar las siguientes funciones:

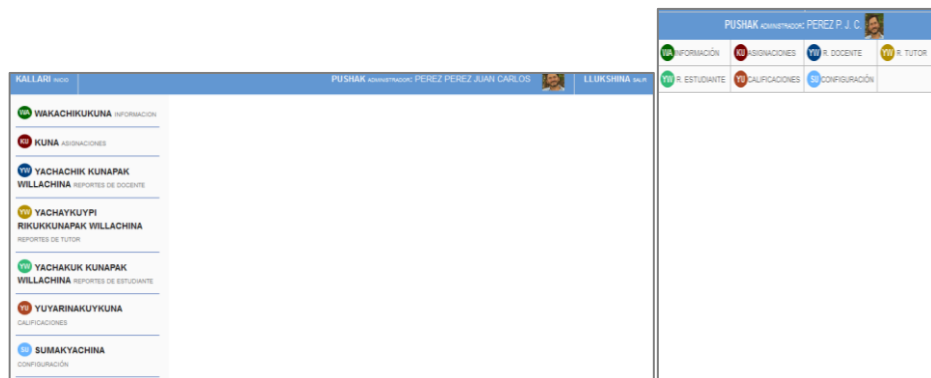


Figura 57: Opciones del usuario Administrador (Vista computador y dispositivo móvil)

- **Información:** Permite ingresar, ver, editar y eliminar la información de estudiantes, representantes, docentes, curso, materias, usuarios, documentos y mensajes.



Figura 58: Opciones de la función Información (vista computador y dispositivo móvil)

- **Asignaciones:** Permite asignar asignaturas a los cursos, realizar la matrícula de estudiantes, asignar el trabajo docente y el trabajo de tutores de curso.

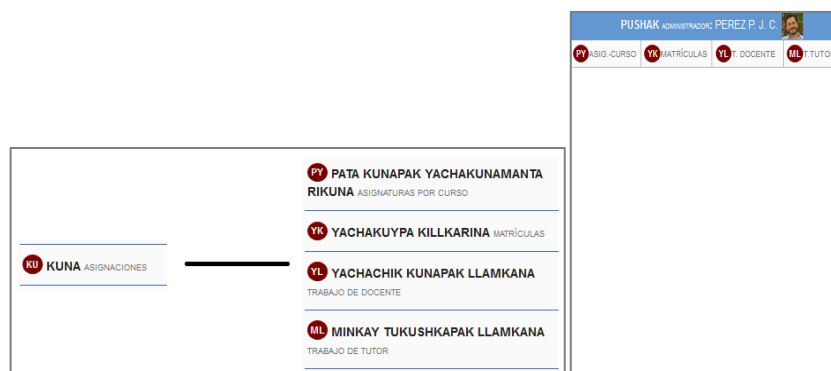


Figura 59: Opciones de la función Asignaciones (Vista computador y dispositivo móvil)

- **Reportes de docente:** Permite ver los reportes de calificaciones de acuerdo con el distributivo de cada docente.

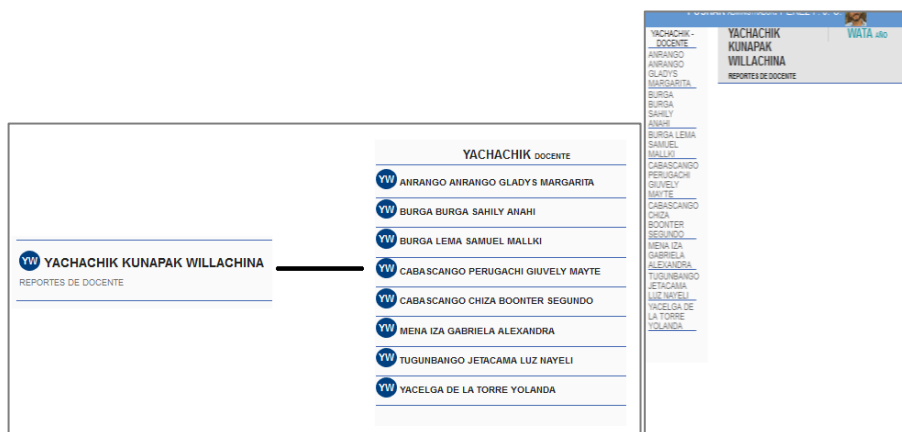


Figura 60: Opciones de la función Reportes de docente (Vista computador y dispositivo móvil)

- **Reportes de tutor:** Permite ver los reportes de resúmenes de calificaciones de perfil tutor, matrícula y promoción de año.



Figura 61: Opciones de la función Reportes de tutor (Vista computador y dispositivo móvil)

- **Reportes de estudiante:** Permite ver los reportes de calificaciones que se entregan a los estudiantes o a sus representantes al final de un bloque de estudio.



Figura 62: Opciones de la función Reportes de estudiante (Vista computador y dispositivo móvil)

- **Calificaciones:** Permite editar las calificaciones de asignaturas, proyectos educativos, comportamiento y asistencia.



Figura 63: Opciones de la función Calificaciones (Vista computador y dispositivo móvil)

- **Configuración:** Permite modificar los parámetros de configuración detallados en el capítulo 2 numeral 3.

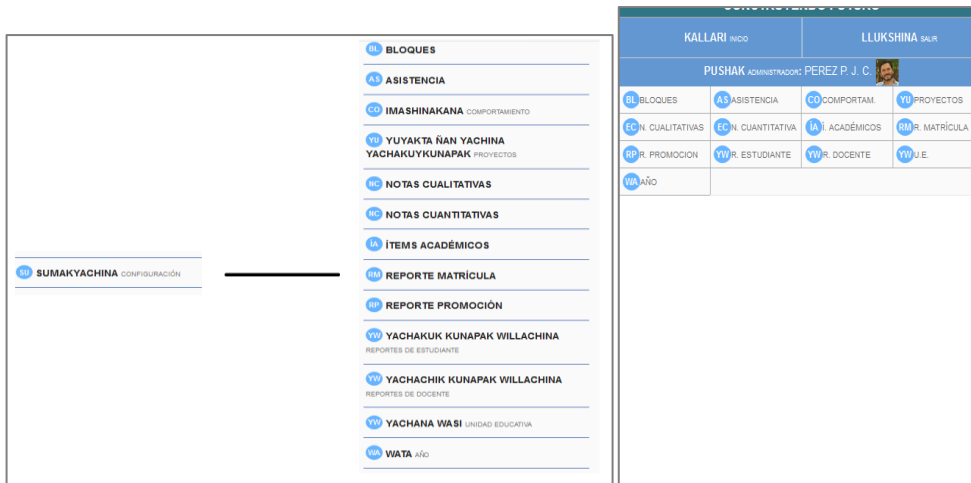


Figura 64: Opciones de la función Configuración (Vista computador y dispositivo móvil)

### 3.2.2 Funciones del sistema para el tipo de usuario docente

El sistema permite al docente realizar las siguientes funciones:

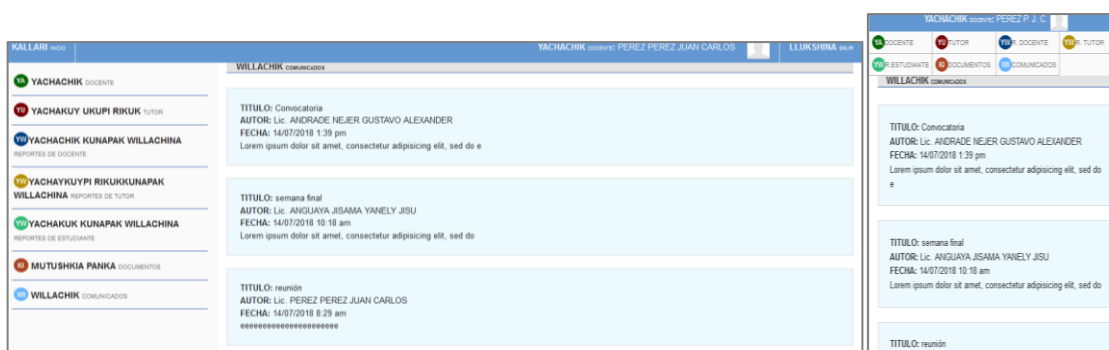


Figura 65: Opciones del usuario Docente (Vista computador y dispositivo móvil)

- **Calificaciones:** Permite ingresar a su distributivo docente para ingresar las calificaciones en sus cursos asignados. La condición es que el administrador del sistema active al menos un bloque de ingreso de calificaciones.

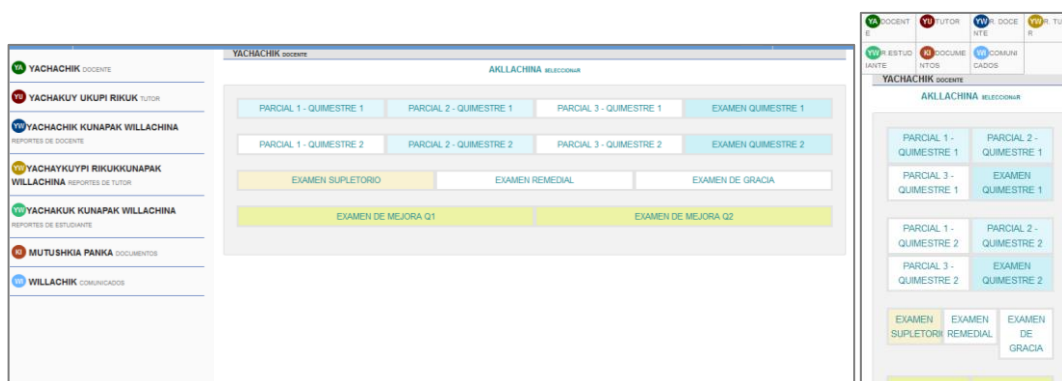


Figura 66: Opciones de la función Calificaciones (Vista computador y dispositivo móvil)

- **Tutor:** Permite ingresar al docente de aula o tutor de curso ingresar calificaciones de proyectos escolares, comportamiento y asistencia en su curso asignado, la condición es que el administrador del sistema active al menos un bloque de ingreso de calificaciones.

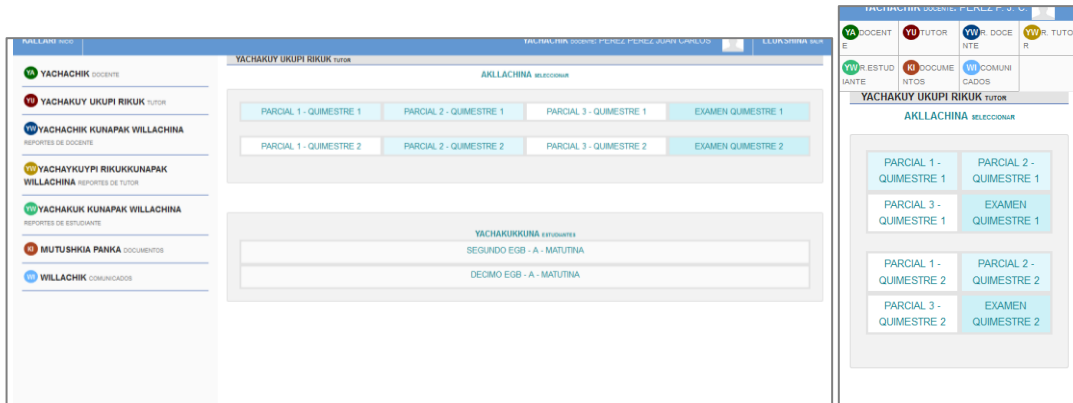


Figura 67: Opciones de la función Tutor (Vista computador y dispositivo móvil)

- **Reportes de docente:** Permite al docente generar los reportes de calificaciones de sus ingresos de calificaciones curso y asignaturas.

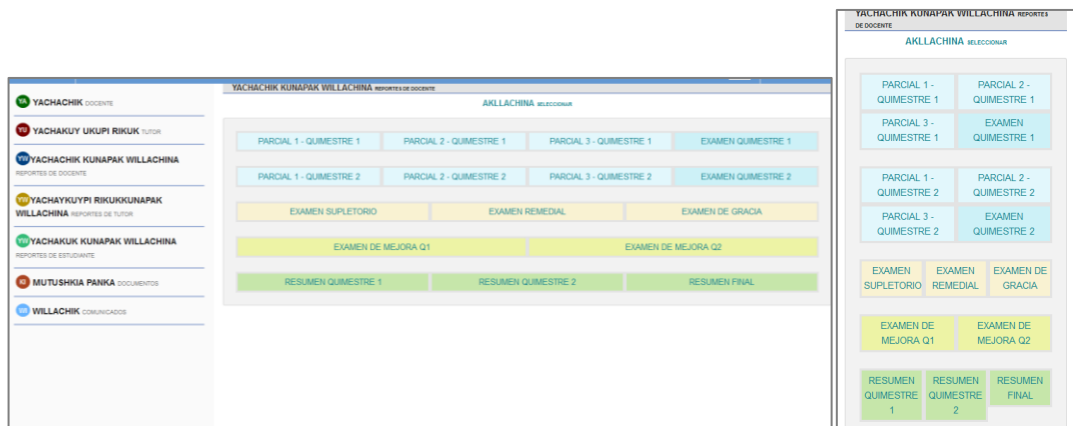


Figura 68: Opciones de la función Reportes de docente (Vista computador y dispositivo móvil)

- **Reportes de tutor:** Permite al tutor generar los reportes de resúmenes de calificaciones del curso, reportes de matrículas, reportes de promoción y sus ingresos de calificaciones.



Figura 69: Opciones de la función Reportes de tutor (Vista computador y dispositivo móvil)

- **Reportes de estudiante:** Permite al docente de aula o tutor de curso generar los reportes de calificaciones que se entregarán a los representantes de los estudiantes.



Figura 70: Opciones de la función Reportes de estudiante (Vista computador)

- **Documentos:** Permite al docente subir documentos al sistema. Estos documentos los puede revisar el administrador y agregar observaciones.

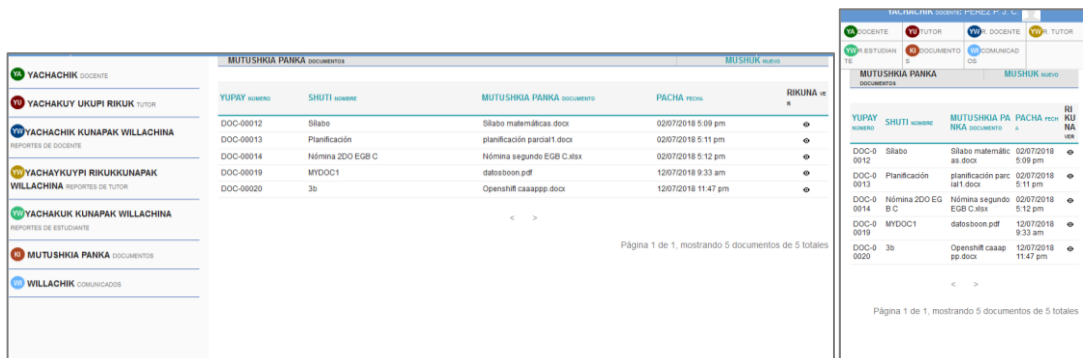


Figura 71: Opciones de la función Documentos (Vista computador y dispositivo móvil)

- **Comunicados:** Permite al docente publicar comunicados a los estudiantes o a sus representantes de los cursos que forman parte de su distributivo docente.

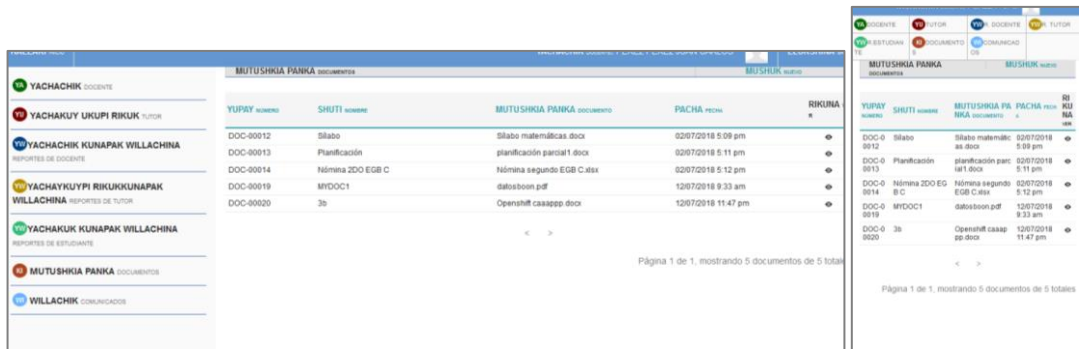


Figura 72: Opciones de la función Comunicados (Vista computador y dispositivo móvil)

### 3.2.3 Funciones del sistema para el tipo de usuario estudiante

El sistema permite al estudiante realizar las siguientes funciones:

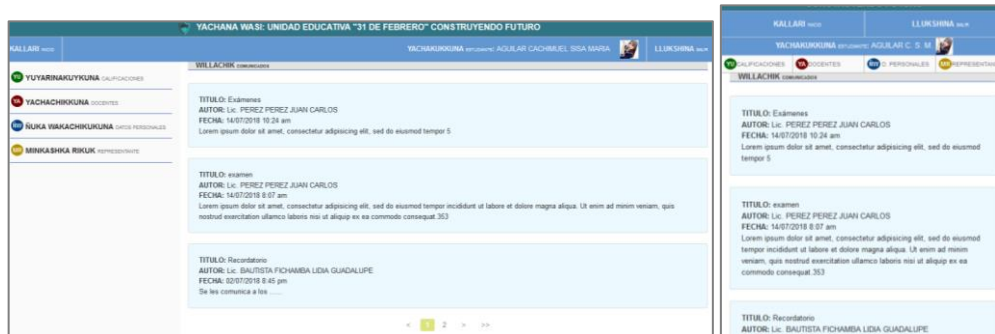


Figura 73: Opciones del usuario Estudiante (Vista computador y dispositivo móvil)

- **Calificaciones:** Permite ver sus reportes de calificaciones.



Figura 74: Opciones de la función Calificaciones (Vista computador y dispositivo móvil)

- **Docentes:** Permite ver la información de contacto de sus docentes.

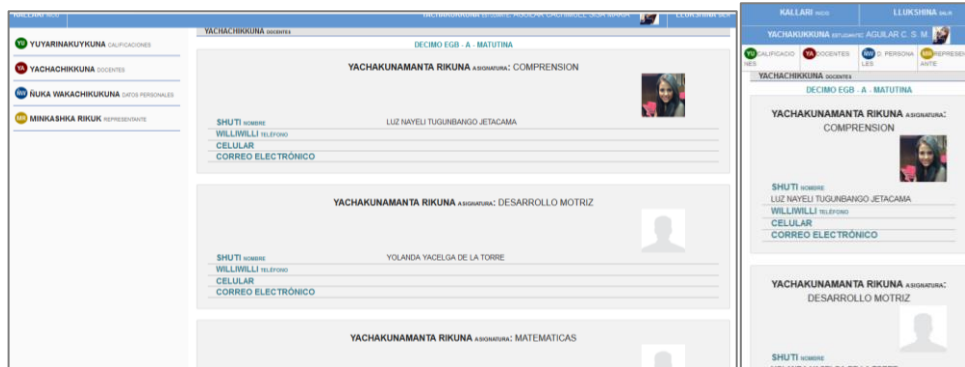


Figura 75: Opciones de la función Docentes (Vista computador y dispositivo móvil)

- **Información personal:** Permite ver sus datos personales.



Figura 76: Opciones de la función Datos personales (Vista computador y dispositivo móvil)

- **Representante:** Permite ver los datos personales de su representante.



Figura 77: Opciones de la función Representante (Vista computador y dispositivo móvil)



### 3.2.4 Funciones del sistema para el tipo de usuario representante

El sistema permite al representante realizar las siguientes funciones:



Figura 78: Opciones del usuario Representante (Vista computador y dispositivo móvil)

- **Calificaciones:** Permite ver los reportes de calificaciones de sus representados.



Figura 79: Opciones de la función Calificaciones (Vista computador y dispositivo móvil)

- **Docentes:** Permite ver la información de contacto de los docentes de sus representados.



Figura 80: Opciones de la función Docentes (Vista computador y dispositivo móvil)

- **Información personal:** Permite ver sus datos personales.



Figura 81: Opciones de la función Datos personales (Vista computador y dispositivo móvil)

- **Estudiantes:** Permite ver los datos personales de los estudiantes que son sus representados.



Figura 82: Opciones de la función Estudiantes (Vista computador y dispositivo móvil)



## CONCLUSIONES

1. El uso de la metodología XP en el desarrollo del sistema académico bilingüe ha permitido que el equipo de trabajo pueda obtener los resultados esperados. El trabajo colectivo, el respeto mutuo, la simplicidad en el diseño y las entregas cortas para identificar si los avances del desarrollo van por el rumbo correcto, se destacan en la obtención de un trabajo más productivo.
2. Para que la funcionalidad del sistema académico trascienda en el tiempo fue de vital importancia el análisis de los posibles cambios en las normativas de evaluación del Ministerio de Educación. Este análisis permitió establecer los parámetros configurables para el desarrollo de un sistema académico multiinstitucional que se adapte a las necesidades cambiantes del usuario final.
3. El uso del framework CakePHP en el desarrollo de este proyecto permitió que el esfuerzo dedicado esté centrado en la programación de funciones de configuración del sistema y no en el desarrollo de los CRUDS de las diferentes entidades, con lo que se obtuvo el resultado esperado de agilizar el trabajo de desarrollo.
4. Con los resultados obtenidos en el desarrollo de las interfaces del sistema se pudo comprobar que el framework de front-end ZURB Foundation es una herramienta eficaz para el desarrollo de aplicaciones web adaptativas.
5. La inclusión de los idiomas kichwa y español de forma simultánea en las interfaces del sistema, brinda el aporte social esperado, más allá de facilitar las labores académicas a los usuarios, contribuyendo con la preservación de las raíces culturales del pueblo indígena del cantón Otavalo.

## RECOMENDACIONES

1. Es de vital importancia considerar la curva de aprendizaje que implica usar el framework CakePHP al momento de realizar la estimación del esfuerzo de desarrollo. La mayoría de la información oficial del framework y la ayuda para resolver conflictos se encuentran disponibles en inglés, esto puede retrasar los plazos establecidos para realizar las entregas planificadas.
2. En el desarrollo de sistemas que impliquen el idioma kichwa se debe considerar que esta lengua ancestral no es totalmente traducible. Muchas palabras técnicas necesarias en el desarrollo de software tendrán que mantenerse en español.
3. Al trabajar con la metodología XP en el desarrollo de un sistema, prácticas no aplicables como la programación en parejas, o roles inexistentes como el “couch” o entrenador pueden incrementar el esfuerzo necesario para el desarrollo, esto implica retrasos en los tiempos de entrega.
4. Para el desarrollo de aplicaciones web adaptativas, se debe considerar la compatibilidad de los elementos CSS con los posibles navegadores de internet que el usuario puede usar. Es necesario usar elementos CSS secundarios para navegadores no compatibles o no actualizados.

## Referencias

- Arias, A., Arias, M., & Durango, A. (2016). *Curso de Desarrollo Web: 2ª edición*. IT Campus Academy.
- Arias, M. A. (2014). *Bases De Datos Con Mysql*. Createspace.
- Arias, M. A. (2017). *Aprende Programación Web con PHP y MySQL*. IT Campus Academy.
- Ayala, E. (2014). *Interculturalidad en el Ecuador*. Obtenido de <http://www.uasb.edu.ec/UserFiles/380/File/Interculturalidad%20en%20el%20Ecuador.pdf>
- Beati, H. (2015). *PHP-Creación de páginas Web dinámicas*. Buenos Aires: Alfaomega.
- CakePHP. (2017). *CakePHP.org*. Obtenido de <https://book.cakephp.org/2.0/es/cakephp-overview/what-is-cakephp-why-use-it.html>
- CakePHP. (2017). *CakePHPCookbook*. Obtenido de [https://book.cakephp.org/3.0/\\_downloads/es/CakePHPCookbook.pdf](https://book.cakephp.org/3.0/_downloads/es/CakePHPCookbook.pdf)
- Capuñay, O. (2013). *Desarrollo Web con PHP*. Amazon Digital Services LLC .
- Constitución. (2008). *Constitución de la Republica del Ecuador 2008*. Obtenido de [http://www.asambleanacional.gob.ec/sites/default/files/documents/old/constitucion\\_de\\_bolsillo.pdf](http://www.asambleanacional.gob.ec/sites/default/files/documents/old/constitucion_de_bolsillo.pdf)
- Dāsa, R. (2016). *Learn CakePHP: With Unit Testing*. Nueva York: Apress.
- Debrauwer, L. (2013). *Patrones de diseño en java*. Barcelona: Eni.
- Díaz, M., & Collazo, A. (2013). *La programación extrema*. Obtenido de ResearchGate: [https://www.researchgate.net/publication/318211906\\_La\\_programacion\\_extrema](https://www.researchgate.net/publication/318211906_La_programacion_extrema)
- Instituto de Idiomas, Ciencias y Saberes Ancestrales. (2016). *KICHWA Yachakukunapa Shimiyuk Kamu*. Obtenido de <http://www.saberesancestrales.gob.ec/wp-content/uploads/downloads/2017/02/KICHWA-BAJA.pdf>
- Kowii, A. (01 de 04 de 2013). *VISIBILIZACIÓN DEL KICHWA*. Obtenido de POLÍTICAS LINGÜÍSTICAS EN EL ECUADOR: <http://repositorio.uasb.edu.ec/bitstream/10644/3827/1/TD037-DECLA-Kowii-invisibilizacion.pdf>
- Kowii, A. (2015). *SENDEROS PARA RECONSTRUIR EL SUMAK KAWSAY*. Obtenido de <http://repositorionew.uasb.edu.ec/bitstream/10644/4489/1/CON-PAP-Kowi,%20A-Senderos-s.pdf>
- Láinez, J. (2015). *Desarrollo de Software Ágil: Extreme Programming y Scrum*. IT Campus Academy.

- Lippenholtz, B. (2012). *El uso de las Nuevas Tecnologías de la Información y de la Comunicación (Tic) en la enseñanza de las lenguas originarias*. Salamanca: Congreso Iberoamericano de las Lenguas en la Educación y en la Cultura.
- MINEDUC. (2017). *INSTRUCTIVO PARA LA APLICACIÓN DE LA EVALUACIÓN ESTUDIANTIL*. Obtenido de [http://consultaseducacion.com/wp-content/uploads/2017/10/instructivo\\_para\\_la\\_aplicacion\\_de\\_la\\_evaluacion\\_estudiantil\\_febrero-2017-1.pdf](http://consultaseducacion.com/wp-content/uploads/2017/10/instructivo_para_la_aplicacion_de_la_evaluacion_estudiantil_febrero-2017-1.pdf)
- Pantaleo, G., & Rinaudo, L. (2015). *Ingeniería de Software*. Mexico: Alfaomega.
- Ramos, A., & Ramos, M. (2014). *APLICACIONES WEB*. Madrid: Ediciones Parainfo, SA.
- Ramos, D., Noriega, R., Laínez, R., & Durango, A. (2017). *Curso de Ingeniería de Software: 2ª Edición*. IT Campus Academy.
- Revenga, D. (2017). *La organización política y la administración del imperio inca*. Obtenido de <http://uvadoc.uva.es/handle/10324/25497>
- Sánchez, M. (2012). *Manual de Desarrollo Web basado en ejercicios y supuestos prácticos*. Málaga: CreateSpace.
- Shenoy, A. (2015). *Introducing Zurb Foundation 6*. Nueva York: Apress.
- Torres, M. (2014). *Desarrollo de aplicaciones web con PHP*. Lima: Macro.
- Torres, M. (2014). *Desarrollo de aplicaciones web con PHP y MySQL*. Lima: Macro.
- Wells, D. (2013). *Extreme Programming: A gentle introduction*. Obtenido de <http://www.extremeprogramming.org/>
- ZURB. (2017). *Foundation.ZURB.com*. Obtenido de <https://foundation.zurb.com/>