



UNIVERSIDAD TÉCNICA DEL NORTE

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

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS COMPUTACIONALES**

TEMA:

**“SISTEMA DE EVALUACIÓN PRESUPUESTARIA DEL GPI MEDIANTE
LA INTEGRACIÓN DE TECNOLOGÍA “BUSINESS PROCESS
MANAGEMENT (BPM)” y “RICH INTERNET APPLICATIONS (RIA)”**

**AUTOR: CORNELIO MOISÉS ORBE FLORES
DIRECTOR: ING. JOSÉ LUIS RODRÍGUEZ**

IBARRA – ECUADOR

2015



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital Institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad. Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1002975769		
APELLIDOS Y NOMBRES:	CORNELIO MOISÉS ORBE FLORES		
DIRECCIÓN:	SAN ANTONIO DE IBARRA, TANGUARÍN - LÍNEA FÉRREA		
EMAIL:	corne_system@hotmail.es		
TELÉFONO FIJO:	062933-070	TELÉFONO MÓVIL:	0999539955
DATOS DE LA OBRA			
TÍTULO:	SISTEMA DE EVALUACIÓN PRESUPUESTARIA DEL GPI MEDIANTE LA INTEGRACIÓN DE TECNOLOGÍA "BUSINESS PROCESS MANAGEMENT (BPM)" y "RICH INTERNET APPLICATIONS (RIA)"		
AUTOR (ES):	CORNELIO MOISÉS ORBE FLORES		
FECHA: AAAAMMDD	ENERO DEL 2015		
PROGRAMA:	<input type="checkbox"/> PREGRADO <input type="checkbox"/> POSTGRADO		
TÍTULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES		
DIRECTOR:	ING. JOSÉ LUIS RODRÍGUEZ		

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Cornelio Moisés Orbe Flores, con cédula de identidad Nro. 1002975769, en calidad de autor y titular de los derechos patrimoniales del trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación del trabajo en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 144.

Firma: 

Nombre: Cornelio Moisés Orbe Flores

Cédula: 1002975769

Ibarra, Enero del 2015



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

El señor Cornelio Moisés Orbe Flores ha trabajado en el desarrollo del proyecto de tesis **SISTEMA DE EVALUACIÓN PRESUPUESTARIA DEL GPI MEDIANTE LA INTEGRACIÓN DE TECNOLOGÍA “BUSINESS PROCESS MANAGEMENT (BPM)” y “RICH INTERNET APPLICATIONS (RIA)”** previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizándola con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.



Ing. José Luis Rodríguez
DIRECTOR DE TESIS



GAD PROVINCIAL
DE IMBABURA



PREFECTURA
DE IMBABURA

Ibarra, 11 de Diciembre del 2014

Ing. Cosme Ortega, Director de Tecnologías de la Información de la Prefectura de Imbabura

CERTIFICA:

Que, siendo auspiciantes del proyecto de grado del egresado **CORNELIO MOISES ORBE FLORES** con CI: 100297576-9 quien desarrolló su proyecto de grado con el tema **SISTEMA DE EVALUACIÓN PRESUPUESTARIA DEL GPI MEDIANTE LA INTEGRACIÓN DE TECNOLOGÍA "BUSINESS PROCESS MANAGEMENT (BPM)" Y "RICH INTERNET APPLICATIONS (RIA)"**, me es grato informar que se ha superado a satisfacción las pruebas técnicas y la revisión del cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado por parte del egresado **CORNELIO MOISES ORBE FLORES**

El egresado **CORNELIO MOISES ORBE FLORES** puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,



Ing. Cosme Ortega



DIRECTOR DE TECNOLOGÍAS DE INFORMACIÓN

IBARRA: Bolívar y Oviedo, esq. Telfs: (593 6) 2955225, 2955832, 2950939, Fax (593 6) 2955430

Email: gpi@imbabura.gob.ec / www.imbabura.gob.ec

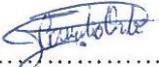


UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA
UNIVERSIDAD TÉCNICA DEL NORTE**

Yo, Cornelio Moisés Orbe Flores con cédula de identidad Nro. 1002975769, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5, 6, en calidad de autor del trabajo de grado denominado **SISTEMA DE EVALUACIÓN PRESUPUESTARIA DEL GPI MEDIANTE LA INTEGRACIÓN DE TECNOLOGÍA “BUSINESS PROCESS MANAGEMENT (BPM)” y “RICH INTERNET APPLICATIONS (RIA)”** que ha sido desarrollado para optar por el título de Ingeniero en Sistemas Computacionales, en la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

Firma: 

Nombre: Cornelio Moisés Orbe Flores

Cédula: 1002975769

Ibarra, Enero del 2015



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

A mis padres Cesar Orbe y Magdalena Flores por haberme formado en el camino del bien y por estar siempre cuando los he necesitado

A mis hermanos que me han brindado su buen ejemplo y siempre han estado apoyándome en las buenas y en las malas, en especial a mi hermano Gabriel que se ha convertido en una inspiración y una razón por la cual seguir adelante.

Cornelio Moisés Orbe Flores



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

Principalmente le agradezco a Dios por estar siempre conmigo, por apoyarme en los momentos difíciles y por acompañarme donde quiera que voy.

A mi familia por el apoyo que me han brindado, por los buenos momentos que he compartido, y por estar a mi lado cuando los he necesitado.

Al personal del departamento de Informática del Gobierno Provincial de Imbabura por el conocimiento compartido, los momentos de camaradería y por el apoyo brindado en el desarrollo del proyecto.

A los docentes de la facultad por el conocimiento adquirido y la amistad brindada, en especial al Ing. José Luis Rodríguez y a la Dra. María de la Portilla.

A mis compañeros y jefes del trabajo por el apoyo brindado, y el conocimiento compartido desinteresadamente.

A mis compañeros de carrera, por los momentos de camaradería en el transcurso de los años de estudio.

Cornelio Moisés Orbe Flores

RESUMEN

Actualmente todas las instituciones públicas y privadas del país deben llevar un control financiero de sus recursos económicos, de contabilidad y de presupuesto; para ello usan sistemas que les permitan automatizar sus procesos manteniendo la integridad de la información.

El gobierno provincial de Imbabura para la gestión del Plan de desarrollo provincial participativo recibe un presupuesto por parte del estado, y este presupuesto debe administrar y distribuir en base a los proyectos planificados para el bienestar de sus habitantes.

Por tal motivo la institución vio la necesidad de implementar un sistema para la gestión presupuestaria.

El objetivo del presente proyecto de grado es solventar las necesidades del Gobierno Provincial de Imbabura a través de la implementación de un sistema de gestión presupuestaria que permita realizar una administración, control y seguimiento del proceso presupuestario.

Para el desarrollo del sistema transaccional de evaluación presupuestaria se utilizó tecnología RIA debido a sus características y beneficios; y para la administración de procesos de negocio se utilizó tecnología BPM; una funcionalidad muy importante es la integración del sistema transaccional y el gestor de procesos de negocio BPM, esto se lo consigue mediante servicios web (web services) que proporciona la herramienta BPM, de esta manera el GPI gestiona las transacciones referentes a cada periodo presupuestario paralelamente con el proceso de negocio sin permitir de sincronizarse.

Como metodología de desarrollo se usó la metodología Scrum, ya actualmente que las metodologías ágiles son las más recomendadas cuando se desarrolla un sistema en una institución en la cual los cambios y variaciones funcionales son frecuentes.

SUMMARY

Currently, all public and private institutions in the country must take financial control of their economic resources, accounting and budget; to do this use system that allow them to automate their processes while maintaining the integrity of the information.

The provincial government of Imbabura for the management of the provincial development Plan participatory receives a budget from the state, and this budget must manage and distribute on the basis of the planned projects for the welfare of its inhabitants.

For this reason, the institution saw the need to implement a system for budget management.

The objective of this project is to address the needs of the Provincial Government of Imbabura through the implementation of a system of budget management that allows to perform a management, control and monitoring of the budget process. For the development of the transactional system of budgetary assessment using RIA technology due to its features and benefits; and for the management of business processes using BPM technology; a functionality that is very important is the integration of the transactional system and the manager of business process BPM, this is what you get by using web services (web services) that are provided with the tool BPM, this way the GPI manages the transactions relating to each budget period in parallel with the business process without allowing syncing.

As a methodology of development using the Scrum methodology, as currently the agile methodologies are the most recommended when a system is developed in an institution in which the changes and functional variations are frequent.

ÍNDICE DE CONTENIDOS

AUTORIZACIÓN DE USO Y PUBLICACIÓN	II
CERTIFICACIÓN	¡ERROR! MARCADOR NO DEFINIDO.
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE.....	¡ERROR! MARCADOR NO DEFINIDO.
DEDICATORIA	VII
AGRADECIMIENTO	VIII
RESUMEN	IX
SUMMARY	X
ÍNDICE DE CONTENIDOS.....	XI
ÍNDICE DE ILUSTRACIONES.....	XIV
ÍNDICE DE TABLAS	XVII
CAPÍTULO I	1
1 INTRODUCCIÓN	1
1.1 DESCRIPCIÓN DEL GOBIERNO PROVINCIAL DE IMBABURA.....	1
1.1.1 MISIÓN:	1
1.1.2 VISIÓN:.....	1
1.1.3 BASE LEGAL.....	1
1.1.4 FUNCIONES DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO PROVINCIAL	2
1.1.5 LOS OBJETIVOS ESTRATÉGICOS	3
1.2 PROBLEMÁTICA:.....	4
1.3 OBJETIVO GENERAL	4
1.4 OBJETIVOS ESPECÍFICOS	4
1.5 ALCANCE:	5
CAPÍTULO II	9
2 MARCO TEÓRICO	9
2.1 HERRAMIENTAS DE DESARROLLO.....	9
2.1.1 TOAD DATA MODELER:.....	9
2.1.2 BASE DE DATOS POSTGRESQL:	11

2.1.3 SERVIDOR APACHE:	14
2.1.4 LENGUAJE PHP:.....	15
2.1.5 ZEND FRAMEWORK:	16
2.1.6 APLICACIONES RIA	17
2.1.7 ADOBE FLEX:	20
2.1.8 IDE FLASH BUILDER 4.5 FOR PHP	23
2.1.9 TECNOLOGÍA BPM:.....	28
2.1.10 MODELADOR DE PROCESOS	29
2.1.11 ADMINISTRADOR DE PROCESOS DE NEGOCIO	29
2.1.12 TIBCO BUSINESS STUDIO	30
2.1.13 PROCESSMAKER.....	31
2.1.13.1 VERSIONES	32
2.1.13.2 ESPACIO DE TRABAJO	32
2.1.13.3 HOME (INICIO).....	33
2.1.13.4 DESIGNER	33
2.1.13.5 DYNAFORMS	36
2.1.13.6 TRIGGERS Y CONEXIÓN A BASE DE DATOS.....	36
2.1.13.7 DASHBOARD	37
2.1.13.8 ADMINISTRACIÓN.....	39
2.1.13.9 WEB SERVICES.....	40
2.2 METODOLOGÍA DE DESARROLLO	42
2.2.1 METODOLOGÍAS AGILES:.....	42
2.2.2. SCRUM.....	44
CAPÍTULO III.....	49
3 DESARROLLO DE LA METODOLOGÍA.....	49
3.1 PLANIFICACIÓN	49
3.1.1 ESPECIFICACIÓN DE REQUERIMIENTOS:.....	49
3.1.2 HISTORIAS DE USUARIO	49
3.1.3 DEFINICIÓN DE ROLES:.....	53
3.1.4 PRODUCT BACKLOG	55

3.2 DESARROLLO Y SEGUIMIENTO.....	56
3.2.1 DIAGRAMA DE CASOS DE USO	56
3.2.2 PECIFICACIONES DE CASOS DE USO	59
3.2.3 DESARROLLO DE LOS SPRINT	76
CAPÍTULO IV	89
4 ARQUITECTURA DEL SISTEMA:.....	89
4.1 ARQUITECTURA DE LA APLICACIÓN	89
4.1.1 ARQUITECTURA DE DESARROLLO.....	89
4.1.1.1. INTERFAZ DE USUARIO.....	90
4.1.1.2 LÓGICA DEL NEGOCIO	91
4.1.2 ARQUITECTURA DE DESPLIEGUE	93
4.1.3 ARQUITECTURA DE PROYECTOS:.....	94
4.2 DEFINICIÓN DE LOS PROCESOS DEL SISTEMA	99
4.2.1 RELACIONES Y DEPENDENCIAS.....	99
4.2.2 PERMISOS DE USUARIOS	101
4.2.3 PROCESO PRESUPUESTARIO.....	103
4.2.4 DATOS PRESUPUESTARIOS:.....	104
4.2.5 PLANIFICACIÓN	105
4.2.6 APROBACIÓN:	107
4.2.7 CONSOLIDACIÓN PRESUPUESTARIA:.....	108
CAPÍTULO V	110
5 IMPLEMENTACIÓN Y PRUEBAS.....	110
5.1 IMPLEMENTACIÓN.....	110
5.1.1 IMPLEMENTACIÓN DEL SISTEMA TRANSACCIONAL.....	110
5.1.2 IMPLEMENTACIÓN DEL GESTOR DE PROCESOS DE NEGOCIO BPM.....	111
5.2 INTEGRACIÓN DEL SISTEMA TRANSACCIONAL CON EL GESTOR DE PROCESOS BPM.....	112
5.2.1 SERVICIO PHP:	113
5.2.2 SERVICIO FLEX.....	113
5.3 CINCROIZACIÓN DEL PROCESO BPM CON EL SISTEMA TRANSACCIONAL	114
5.4 FLUJO DEL PROCESO TRANSACCIONAL Y BPM	115

CAPÍTULO VI	126
6 CONCLUSIONES Y RECOMENDACIONES	126
6.1 CONCLUSIONES	126
6.2 RECOMENDACIONES	126
BIBLIOGRAFÍA	127
6.3 ANEXOS	129
6.3.1 MANUAL DE USUARIO.....	129

ILUSTRACIÓN 1: ARQUITECTURA SUITE GPIGESTION	5
ILUSTRACIÓN 2: RELACIÓN DEL SISTEMA GPIPRESUPUESTO.....	6
ILUSTRACIÓN 3: DEPENDENCIAS DEL SISTEMA GPIPRESUPUESTO.....	7
ILUSTRACIÓN 4: PROCESO PRESUPUESTARIO	7
ILUSTRACIÓN 5: INTEGRACIÓN CON EL SISTEMA OLYMPO.....	8
ILUSTRACIÓN 6: INTEGRACIÓN DEL SISTEMA TRANSACCIONAL Y BPM.....	8
ILUSTRACIÓN 7: LOGO TOAD DATA MODELER.....	9
ILUSTRACIÓN 8: MODELO DE DATOS DISEÑADO EN TOAD DATA MODELER.....	10
ILUSTRACIÓN 9: LOGO DE POSTGRESQL	11
ILUSTRACIÓN 10: COMPONENTES POSTGRESQL.....	12
ILUSTRACIÓN 11: ESPACIO DE TRABAJO DE PGADMIN	13
ILUSTRACIÓN 12: ESPACIO DE TRABAJO NAVICAT	14
ILUSTRACIÓN 13: LOGO DE APACHE	14
ILUSTRACIÓN 14: LOGO DE PHP	15
ILUSTRACIÓN 15: LOGO SE ZEND FRAMEWORK.....	16
ILUSTRACIÓN 16: APLICACIÓN WEB TRADICIONAL	18
ILUSTRACIÓN 17: APLICACIÓN RIA.....	18
ILUSTRACIÓN 18: USOS DE APLICACIONES RIA.....	19
ILUSTRACIÓN 19: LOGO DE FLEX.....	20
ILUSTRACIÓN 20: COMPILACIÓN Y PUBLICACIÓN DE UNA APLICACIÓN FLEX.....	21
ILUSTRACIÓN 21: LOGO DE FLASH BUILDER	23
ILUSTRACIÓN 22: ESPACIO DE TRABAJO DE FLASH BUILDER 4.5	25
ILUSTRACIÓN 23: ZEND TOOL EN FLASH BUILDER 4.....	26
ILUSTRACIÓN 24: EJECUTOR DE COMANDOS DE ZEND TOOL	27
ILUSTRACIÓN 25: SERVICIOS QUE PROPORCIONA FLASH BUILDER 4.5.....	27
ILUSTRACIÓN 26: PROCESO BPM.....	28
ILUSTRACIÓN 27: ESPACIO DE TRABAJO DE TIBCO BUSINESS STUDIO	30
ILUSTRACIÓN 28: PALETA DE HERRAMIENTAS TIBCO BUSINESS STUDIO	31
ILUSTRACIÓN 29: BUZÓN DE ENTRADA DEL PROCESO BPM.....	33

ILUSTRACIÓN 30: PROCESOS IMPLEMENTADOS EN EL GPI	34
ILUSTRACIÓN 31: MENÚ PARA EL MODELAMIENTO DE PROCESOS EN PROCESSMAKER	34
ILUSTRACIÓN 32: CUADRO DE HERRAMIENTAS DE ENRUTAMIENTO	35
ILUSTRACIÓN 33: FLUJO DE PROCESOS EN PROCESSMAKER	35
ILUSTRACIÓN 34: FORMULARIO QUE PROPORCIONA PROCESSMAKER	36
ILUSTRACIÓN 35: LISTA DE BASE DE DATOS CONECTADAS EN PROCESSMAKER	36
ILUSTRACIÓN 36: TRIGGER ASIGNADO A UNA TAREA	37
ILUSTRACIÓN 37: DASHBOARD BÁSICO INTEGRADO EN PROCESSMAKER	38
ILUSTRACIÓN 38: ADVANCED DASHBOARD	38
ILUSTRACIÓN 39: CONFIGURACIONES DE PROCESSMAKER	39
ILUSTRACIÓN 40: SERVICIO WEB QUE PROPORCIONA PROCESSMAKER	40
ILUSTRACIÓN 41: OPERACIONES DEL SERVICIO WEB DE PROCESSMAKER	41
ILUSTRACIÓN 42: EJEMPLO DEL RESPONSE DE UNA OPERACIÓN DEL SERVICIO WEB..	42
ILUSTRACIÓN 43 : DIAGRAMA DE CASOS DE USO – GESTIÓN DE DATOS PRESUPUESTARIOS	56
ILUSTRACIÓN 44: DIAGRAMA DE CASOS DE USO – PLANIFICACIÓN PRESUPUESTARIA .	57
ILUSTRACIÓN 45: DIAGRAMA DE CASOS DE USO - APROBACIÓN PRESUPUESTARIA.....	58
ILUSTRACIÓN 46: DIAGRAMA DE CASOS DE USO – CONSOLIDACIÓN PRESUPUESTARIA	58
ILUSTRACIÓN 47: DISTRIBUCIÓN CAPAS DEL SISTEMA.....	89
ILUSTRACIÓN 48: LENGUAJES QUE MANEJA FLEX.....	90
ILUSTRACIÓN 49: COMUNICACIÓN ENTRE SERVICIOS PHP Y FLEX	91
ILUSTRACIÓN 50: PATRÓN USADO PARA EL DESARROLLO.....	92
ILUSTRACIÓN 51: ARQUITECTURA DE DESPLIEGUE	93
ILUSTRACIÓN 52: RELACIÓN PROYECTOS BACKEND Y FRONTEND.....	96
ILUSTRACIÓN 53: INTERRELACIÓN DE LOS PROYECTOS FRONTEND	97
ILUSTRACIÓN 54: RELACIÓN DE PROYECTOS DE LIBRERÍAS Y PROYECTOS FRONTEND	98
ILUSTRACIÓN 55: RELACIÓN GPIFRONTEND CON OTROS PROYECTOS	99
ILUSTRACIÓN 56: ORGANIZACIÓN DE LOS OBJETIVOS ESTRATÉGICOS	100
ILUSTRACIÓN 57: PROCESO PRESUPUESTARIO	103
ILUSTRACIÓN 58: PLANIFICACIÓN PRESUPUESTARIA	106

ILUSTRACIÓN 59: APROBACIÓN PRESUPUESTARIA.....	107
ILUSTRACIÓN 60: CONSOLIDACIÓN PRESUPUESTARIA	109
ILUSTRACIÓN 61: FLUJO DE LA PETICIÓN Y RESPUESTA DEL SERVICIO WEB.....	112
ILUSTRACIÓN 62: TEST DE UNA OPERACIÓN DEL SERVICIO WEB.....	114
ILUSTRACIÓN 63: FLUJO DEL PROCESO PRESUPUESTARIO.....	115
ILUSTRACIÓN 64: LOGIN CON EL USUARIO MMORALES.....	116
ILUSTRACIÓN 65: INICIO DEL EJERCICIO PRESUPUESTARIO	116
ILUSTRACIÓN 66: LOGIN EN EL PROCESO BPM, USUARIO CORTEGA.....	117
ILUSTRACIÓN 67: BUZÓN DE ENTRADA EN EL PROCESO BPM	117
ILUSTRACIÓN 68: INICIO DEL PERIODO PLANIFICACIÓN	118
ILUSTRACIÓN 69: BUZÓN DE ENTRADA EN EL PROCESO BPM	118
ILUSTRACIÓN 70: NOTIFICACIÓN DEL INICIO DEL PERIODO	119
ILUSTRACIÓN 71: ESPACIO DE TRABAJO DEL SISTEMA TRANSACCIONAL EN EL PERIODO PLANIFICACIÓN	119
ILUSTRACIÓN 72: REGISTRO DE REQUERIMIENTOS.....	120
ILUSTRACIÓN 73: BUZÓN DE ENTRADA EN EL PROCESO BPM	120
ILUSTRACIÓN 74: INICIO DEL PERIODO APROBACIÓN.....	121
ILUSTRACIÓN 75: NOTIFICACIÓN SOBRE EL PERIODO DE APROBACIÓN.....	121
ILUSTRACIÓN 76: APROBACIÓN DE REQUERIMIENTOS.....	122
ILUSTRACIÓN 77: MAIL DE APROBACIÓN DEL PROYECTO.....	122
ILUSTRACIÓN 78: INICIO DEL PERIODO CONSOLIDACIÓN.....	123
ILUSTRACIÓN 79: ASIGNACIÓN DE PARTIDAS PRESUPUESTARIAS	123
ILUSTRACIÓN 80: GENERACIÓN DE LA PROFORMA PRESUPUESTARIA	124
ILUSTRACIÓN 81: FINALIZACIÓN DEL PERIODO CONSOLIDACIÓN.....	124
ILUSTRACIÓN 82: FINALIZACIÓN DEL EJERCICIO PRESUPUESTARIO	125
ILUSTRACIÓN 83: NOTIFICACIÓN DEL PROCESO DE SINCRONIZADO.....	125

ÍNDICE DE TABLAS

TABLA 1: METODOLOGÍAS AGILES Y METODOLOGÍAS TRADICIONALES	44
---	----

TABLA 2: HISTORIA DE USUARIO 1	49
TABLA 3: HISTORIA DE USUARIO 2	50
TABLA 4: HISTORIA DE USUARIO 3	50
TABLA 5: HISTORIA DE USUARIO 4	51
TABLA 6: HISTORIA DE USUARIO 5	51
TABLA 7: HISTORIA DE USUARIO 6	52
TABLA 8: HISTORIA DE USUARIO 9	52
TABLA 9: ROLES DE LAS PERSONAS QUE INTERVIENEN EN EL SISTEMA.....	54
TABLA 10: STAKEHOLDER O USUARIOS FINALES DEL SISTEMA.....	54
TABLA 11: PRODUCT BACKLOG	55
TABLA 12: PARTIDAS PRESUPUESTARIAS	60
TABLA 13: ÍTEM PRESUPUESTARIO.....	61
TABLA 14: EJERCICIOS Y PERIODOS PRESUPUESTARIOS.....	63
TABLA 15: EJERCICIOS PRESUPUESTARIOS	64
TABLA 16: RELACIÓN PERIODOS Y EJERCICIO PRESUPUESTARIO	65
TABLA 17: AGRUPACIÓN ÍTEM Y PARTIDA PRESUPUESTARIA.....	73
TABLA 18: CONSOLIDACIÓN MEDIANTE ÍTEM PRESUPUESTARIO.....	73
TABLA 19: CONSOLIDACIÓN MEDIANTE PARTIDA PRESUPUESTARIA	74
TABLA 20: SPRINT 1 - INSTALACIÓN Y CONFIGURACIÓN DE HERRAMIENTAS.....	76
TABLA 21: SPRINT 2 – DISEÑO DEL LOGIN	77
TABLA 22: SPRINT 3 – DISEÑO DE LA INTERFAZ	78
TABLA 23: SPRINT 4 – GESTIÓN DE PARTIDAS PRESUPUESTARIAS.....	80
TABLA 24: SPRINT 5 – GESTIÓN DE EJERCICIOS PRESUPUESTARIOS	81
TABLA 25: SPRINT 6 – GESTIÓN DE PERIODOS PRESUPUESTARIOS	83
TABLA 26: SPRINT 7 – REGISTRO DE REQUERIMIENTOS	85

TABLA 27: SPRINT 8 – APROBACIÓN DE REQUERIMIENTOS	86
TABLA 28: SPRINT 9 - CONSOLIDACIÓN PRESUPUESTARIA.....	88
TABLA 29: PROYECTOS QUE INTERACTÚAN SOBRE GPIGESTION	95

CAPÍTULO I

1 INTRODUCCIÓN

1.1 DESCRIPCIÓN DEL GOBIERNO PROVINCIAL DE IMBABURA

(GPI, 2013)

El Gobierno Provincial de Imbabura (GPI) es la Institución encargada de coordinar, planificar, ejecutar y evaluar el Plan de Desarrollo Provincial Participativo; fortaleciendo la productividad, la vialidad, el manejo adecuado de sus recursos naturales y promoviendo la participación ciudadana, a fin de mejorar la calidad de vida de sus habitantes.”

1.1.1 MISIÓN:

El Gobierno Provincial de Imbabura es la institución encargada de coordinar, planificar, ejecutar y evaluar el Plan de Desarrollo Provincial Participativo; fortaleciendo la productividad, la vialidad, el manejo adecuado de sus recursos naturales y promoviendo la participación ciudadana; a fin de mejorar la calidad de vida de sus habitantes.

1.1.2 VISIÓN:

El Gobierno Provincial de Imbabura se consolida como una institución de derecho público, autónoma, descentralizada, transparente, eficiente, equitativa, incluyente y solidaria; líder del desarrollo económico, social y ambiental provincial.

1.1.3 BASE LEGAL

El Consejo Provincial de Imbabura, fue creado mediante Decreto Ejecutivo del Gobierno Nacional de 1946, pasando a denominarse Gobierno Provincial de Imbabura mediante Ordenanza Provincial publicada en Registro Oficial NQ 380 de 31 de julio del 2001.

De acuerdo al Código Orgánico de Organización Territorial, Autonomía y Descentralización, Art. 40.- Los gobiernos autónomos descentralizados provinciales, son personas jurídicas de derecho público, con autonomía política, administrativa y financiera. Estarán integrados por las funciones de participación ciudadana; legislación y fiscalización: y, ejecutiva, para el ejercicio de las funciones y competencias que le corresponden.

El desarrollo de su vida Jurídica e Institucional se rige por las siguientes normas legales:

- a) Constitución de la República del Ecuador: Capítulo tercero "Gobiernos autónomos descentralizados y regímenes especiales" Art. 252. Capítulo cuarto "Régimen de competencias" Art. 263.
- b) Código Orgánico de Planificación y Finanzas Públicas.
- b) Código Orgánico de Organización Territorial Autonomía y Descentralización.
- c) Decreto de Creación del Consejo Provincial de Imbabura
- d) Ley Orgánica de la Contraloría General del Estado, Registro Oficial N- 595 de junio 12 del 2002.
- e) Contraloría General del Estado, Normas de Control Interno para las entidades, organismos del sector público y personas jurídicas de derecho privado que dispongan de recursos públicos.
- f) Ley Orgánica del Sistema Nacional de Contratación Pública, Registro Oficial NQ 395 de 4 de agosto del 2008 y su Reglamento publicado en Registro Oficial N- 588 de 12 de mayo del 2009.
- g) Ley Orgánica del Servicio Público
- h) Acuerdo Ministerial N- 447 del 29 de diciembre del 2007, publicado en Registro Oficial N- 259 del 24 de enero del 2008. Principios y Normas del Sistema de Administración Financiera. Clasificador Presupuestario de Ingresos y Gastos y otros Acuerdos Ministeriales sobre las Normas Técnicas de Presupuesto.
- i) Ley de Régimen Tributario Interno,
- j) Ordenanzas del Consejo Provincial.
- k) Reglamentos Internos y,
- l) Otras Leyes

1.1.4 FUNCIONES DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO PROVINCIAL

(GPI, 2013)

- a) Promover el desarrollo sustentable de su circunscripción territorial provincial, garantizar la realización del buen vivir a través de la implementación de constitucionales y legales;

- b) Diseñar e implementar políticas de promoción y construcción de equidad e inclusión en su territorio, en el marco de sus competencias constitucionales y legales;
- c) Implementar un sistema de participación ciudadana para el ejercicio de los derechos y avanzar en la gestión democrática de la acción provincial;
- d) Elaborar y ejecutar el plan provincial de desarrollo, el de ordenamiento territorial y las políticas públicas en el ámbito de sus competencias y en su circunscripción territorial, de manera coordinada con la planificación nacional, regional, cantonal y parroquial, y realizar en forma permanente, el seguimiento y rendición de cuentas sobre el cumplimiento de las metas establecidas;
- e) Ejecutar las competencias exclusivas y concurrentes reconocidas por la Constitución y la ley y. en dicho marco prestar los servicios públicos, construir la obra pública provincial, fomentar las actividades provinciales productivas, así como las de vialidad, gestión ambiental, riego, desarrollo agropecuario y otras que le sean expresamente delegadas o descentralizadas, con criterios de calidad, eficacia y eficiencia, observando los principios de universalidad, accesibilidad, regularidad, continuidad, solidaridad, interculturalidad, subsidiariedad, participación y equidad;
- f) Fomentar las actividades productivas y agropecuarias provinciales, en coordinación con los demás gobiernos autónomos descentralizados;
- g) Promover los sistemas de protección integral a los grupos de atención prioritaria para garantizar los derechos consagrados en la Constitución en el marco de sus competencias;
- h) Desarrollar planes y programas de vivienda de interés social en el área rural de la provincia;
- i) Promover y patrocinar las culturas, las artes, actividades deportivas y recreativas en beneficio de la colectividad en el área rural, en coordinación con los gobiernos autónomos descentralizados de las parroquias rurales;
- j) Coordinar con la Policía Nacional, la sociedad y otros organismos lo relacionado con la seguridad ciudadana, en el ámbito de sus competencias; y,
- k) Las demás establecidas en la ley.

1.1.5 LOS OBJETIVOS ESTRATÉGICOS

- a) Fomentar el desarrollo económico provincial
- b) Consolidar el sistema de transporte y movilidad provincial
- c) Implementar el sistema de gestión ambiental provincial con enfoque intercultural y visión de cuenca hidrográfica

- d) Diseñar políticas, planes y programas, tendientes a fortalecer la inclusión social, el desarrollo cultural que permitan hacer de Imbabura una provincia equitativa, solidaria e intercultural.
- e) Generar mecanismos de articulación y lineamientos para la coordinación endógena institucional e interinstitucional.
- f) Tecnificar los procesos de administración y gestión institucional

1.2 PROBLEMÁTICA:

Actualmente el GPI cuenta con un software propietario llamado Olimpo, usado para el control de las finanzas y el presupuesto, pero dicho sistema tiene limitantes, ya que no es escalable y es inmantenible ya que no se posee el código fuente; por tal motivo el personal administrativo del GPI ha tenido que acoplarse a las limitantes que le ofrece dicho sistema.

Además existe la falta de integración entre los departamentos del GPI, y no existe un adecuado control y seguimiento del proceso presupuestario lo que ha generado incongruencia de información durante la ejecución de un ejercicio presupuestario.

1.3 OBJETIVO GENERAL

Diseñar e Implementar un Sistema Integrado de Evaluación Presupuestaria para el Gobierno Provincial de Imbabura mediante el uso de tecnologías RIA y BPM que permita integrar los departamentos de la institución y gestionar de manera ágil el proceso presupuestario.

1.4 OBJETIVOS ESPECÍFICOS

- Implementar el sistema transaccional para la gestión presupuestaria mediante el uso de tecnología RIA.
- Gestionar, controlar y realizar un seguimiento sobre los procesos presupuestarios del Gobierno Provincial de Imbabura a través de tecnología BPM
- Integrar las funcionalidades del sistema transaccional con el proceso de negocio bpm
- Integrar los departamentos del Gobierno Provincial
- Emplear la metodología SCRUM para el desarrollo del sistema de Presupuesto
- Diseñar una interfaz que permita interactuar el usuario de manera confiable asegurando su información.

1.5 ALCANCE:

El Sistema Integrado de Gestión presupuestaria es parte de una Suite Empresarial **GPI GESTION**.

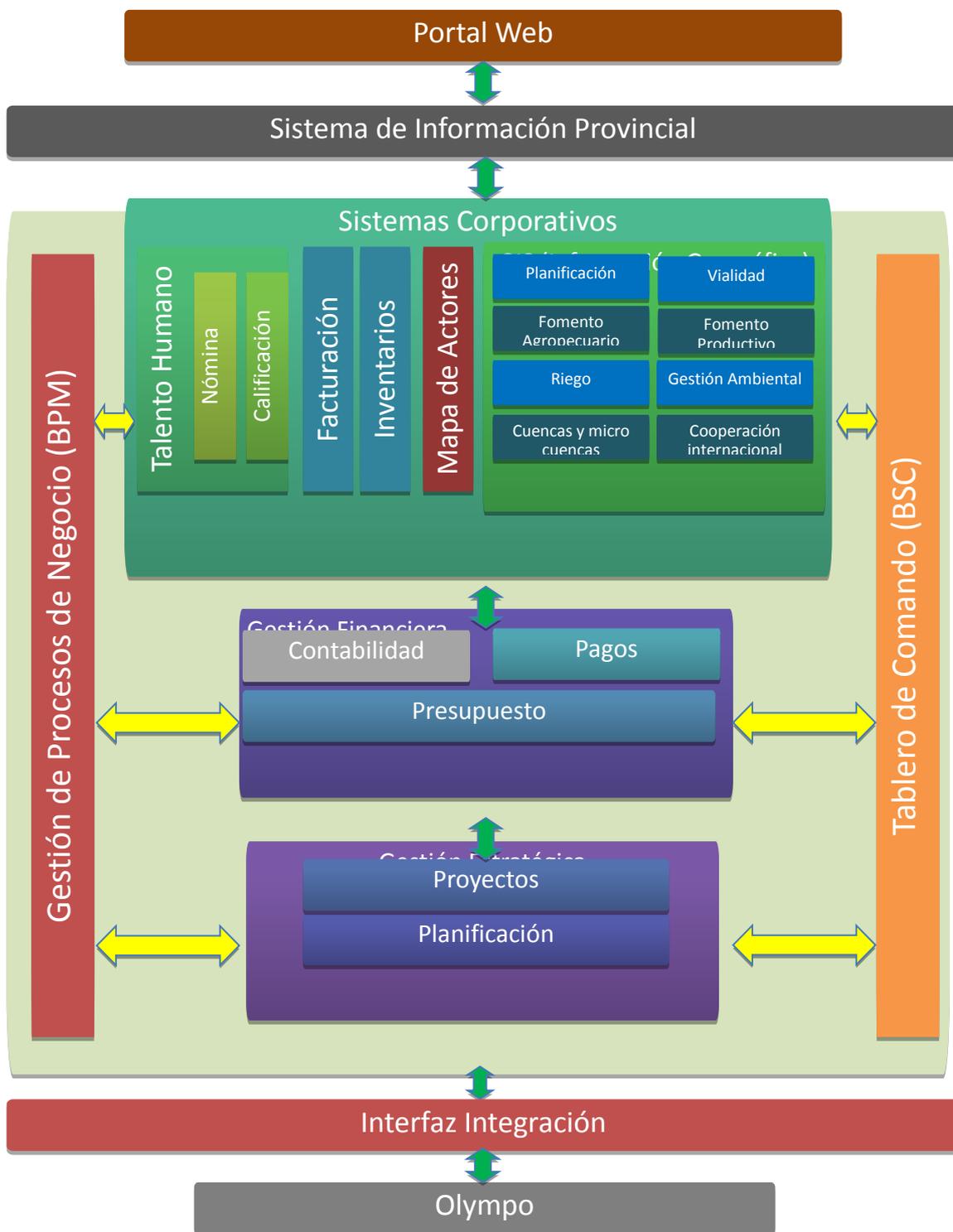


Ilustración 1: Arquitectura Suite GpiGestion

Fuente: GPI

La suite **GPI_GESTION** involucra y compromete a toda la institución con el logro de los objetivos estratégicos y establece la integración de la planificación con todo el proceso operativo y de gestión del Gobierno Provincial.

Como se observa en el grafico el Sistema de Presupuesto está contenido dentro de la Gestión Financiera; al igual que el sistema de Contabilidad y el sistema de Pagos.

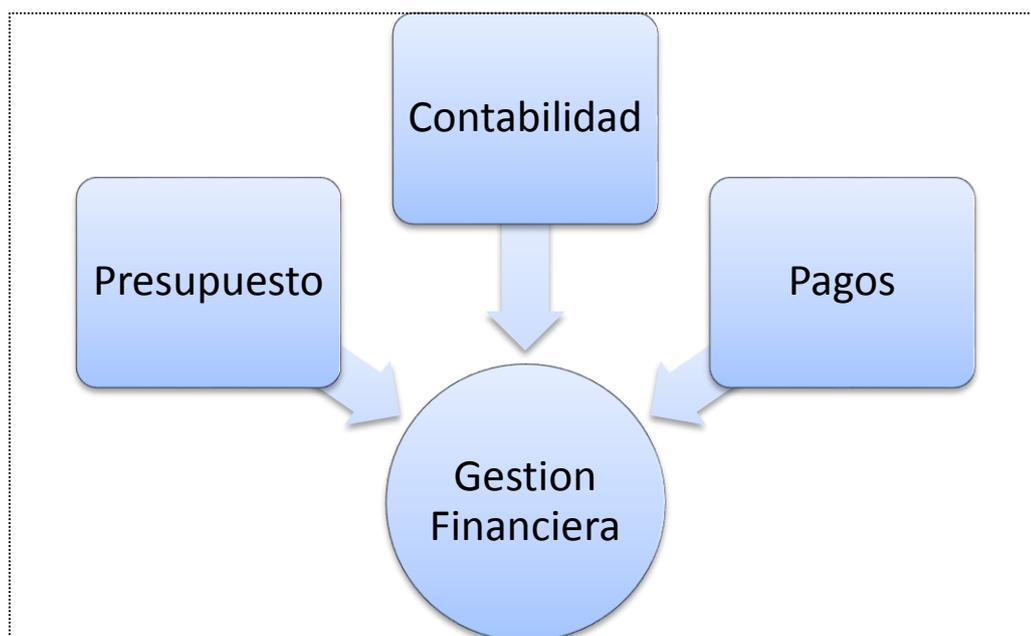


Ilustración 2: Relación del Sistema GpiPresupuesto

Fuente: Propia

Además el Sistema de Presupuesto tiene la dependencia directa con los sistemas de la Gestión Estratégica que comprenden: El sistema de planificación, y el sistema de proyectos.

Ya que un sobre los proyectos se realiza la planificación y ejecución presupuestaria.

Así también depende el sistema de Inventarios puesto que de ahí se toma la información los ítems que se va a presupuestar

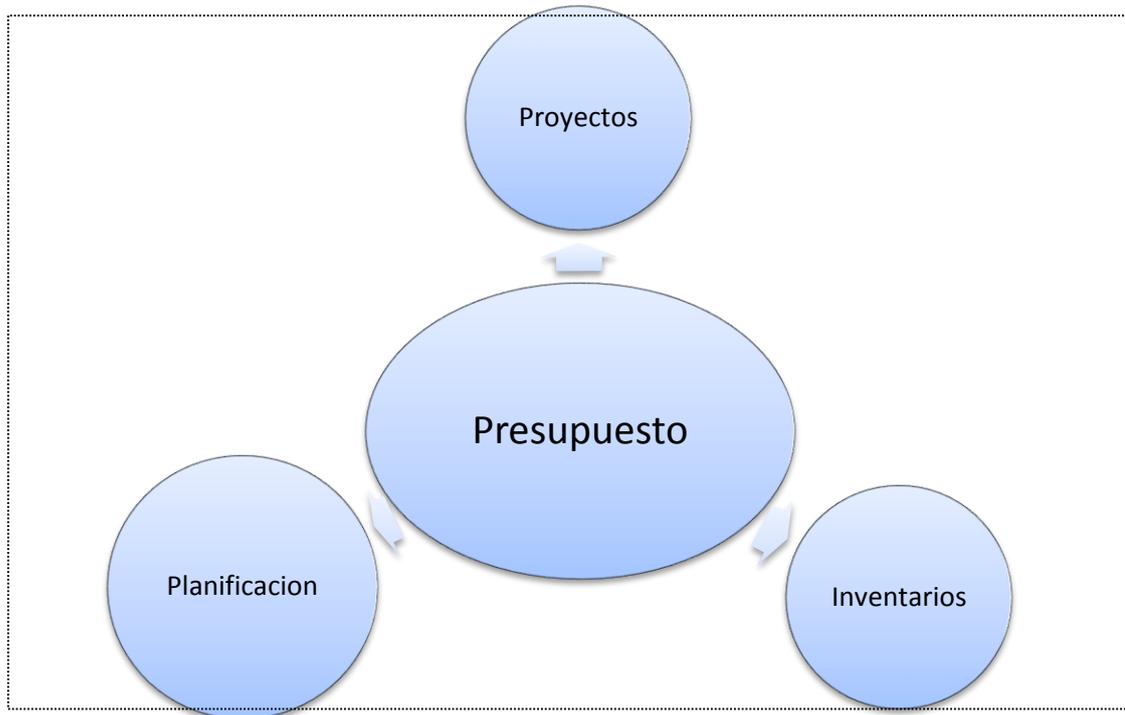


Ilustración 3: Dependencias del Sistema GpiPresupuesto

Fuente: Propia

El proceso presupuestario contiene subprocesos los cuales se describen en la siguiente ilustración:

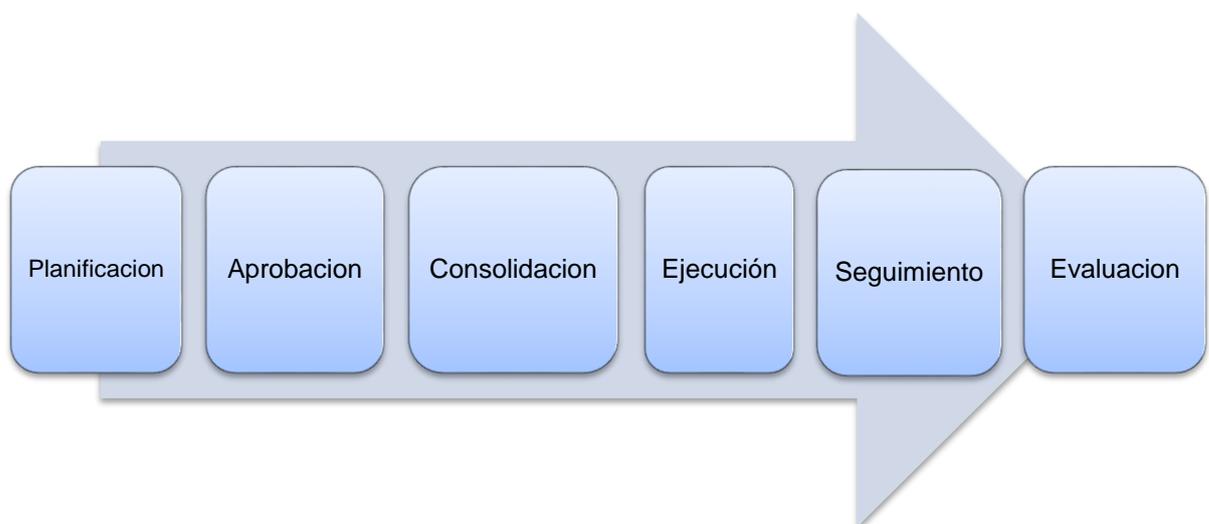


Ilustración 4: Proceso Presupuestario

Fuente: Propia

El alcance del sistema GpiPresupuesto será hasta la consolidación presupuestaria, luego se realizara una integración con el Olympo.

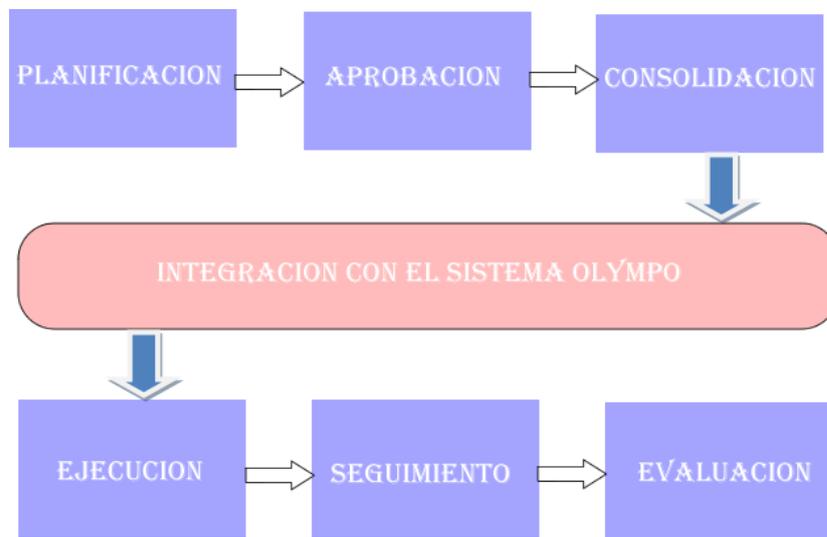


Ilustración 5: Integración con el sistema Olympo

Fuente: Propia

Uno de los objetivos muy importantes es la integración del sistema transaccional con el gestor de procesos BPM y esto se lo consigue mediante el servicio web (web services) que proporciona la herramienta BPM.

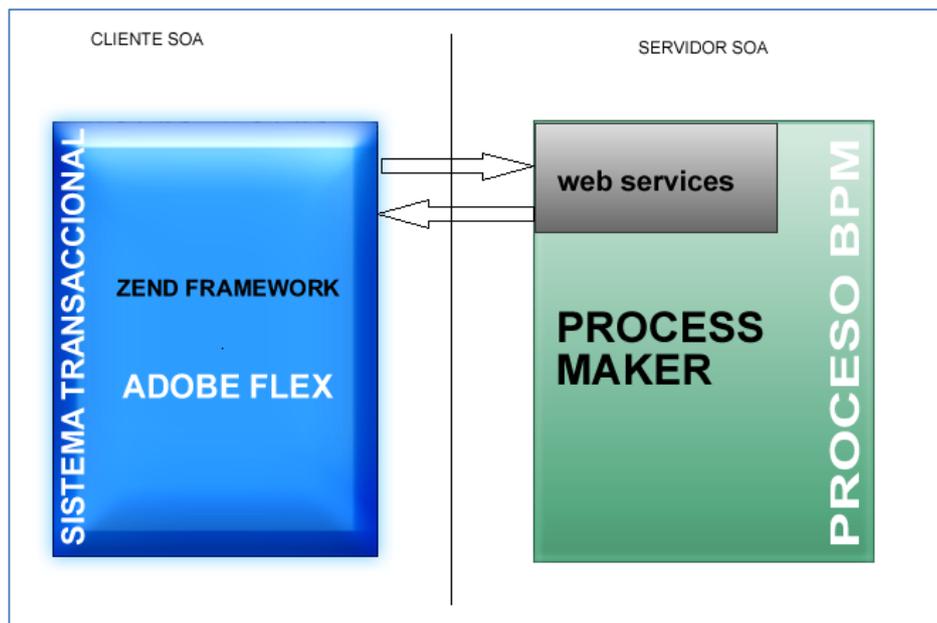


Ilustración 6: Integración del sistema transaccional y BPM

Fuente: Propia

CAPÍTULO II

2 MARCO TEÓRICO

2.1 HERRAMIENTAS DE DESARROLLO

2.1.1 TOAD DATA MODELER:

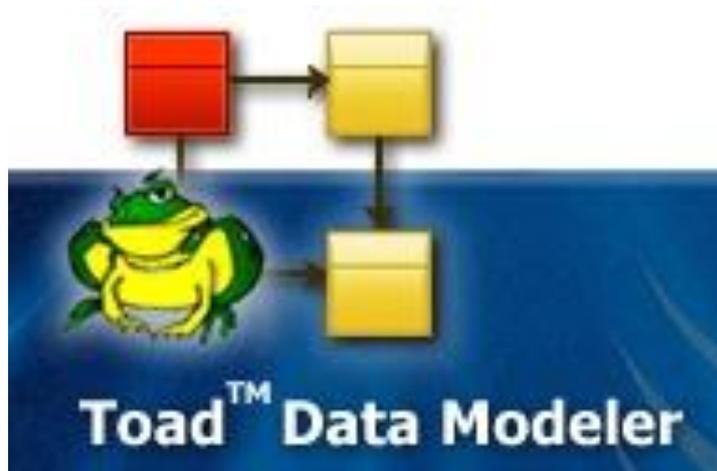


Ilustración 7: Logo Toad Data Modeler

Fuente: (Software, 2014)

Es un software free que permite diseñar, diagramar, modelar diagramas de bases de datos. Y además permite generar scripts sql a partir de estos diagramas.

Tiene las mismas características y servicios de otros programas modeladores privados como son el Power Designer.

Ofrece una interfaz muy amigable, mediante un espacio de trabajo y una paleta de herramientas con la cual se puede modelar fácilmente bases de datos.

Así también podemos relacionar las tablas gráficamente y asignar la cardinalidad y el tipo de relación.

La interfaz que ofrece Toad Data Modeler es muy amigable, cuenta con menús y herramientas que permite realizar un modelado completo en para cualquier base de datos.

Características:

- Manejo de espacios de trabajo
- Modelado simultaneo de varias bases de datos.
- Gestión de esquemas a nivel de base de datos.
- Gestión de entidades, propiedades y atributos.
- Gestión de relaciones entre las entidades.
- Manejo de vistas, funciones, secuencias, entre otros.
- Exportación del modelo físico al lenguaje SQL en la base al lenguaje establecido

Una vez que se haya modelado la base de datos se exporta el script en el lenguaje de la base de datos que se usara para gestionar la base de datos.

Se procederá a insertar datos de prueba para el funcionamiento del sistema.

En la siguiente ilustración se puede apreciar el modelo diseñado en esta herramienta.

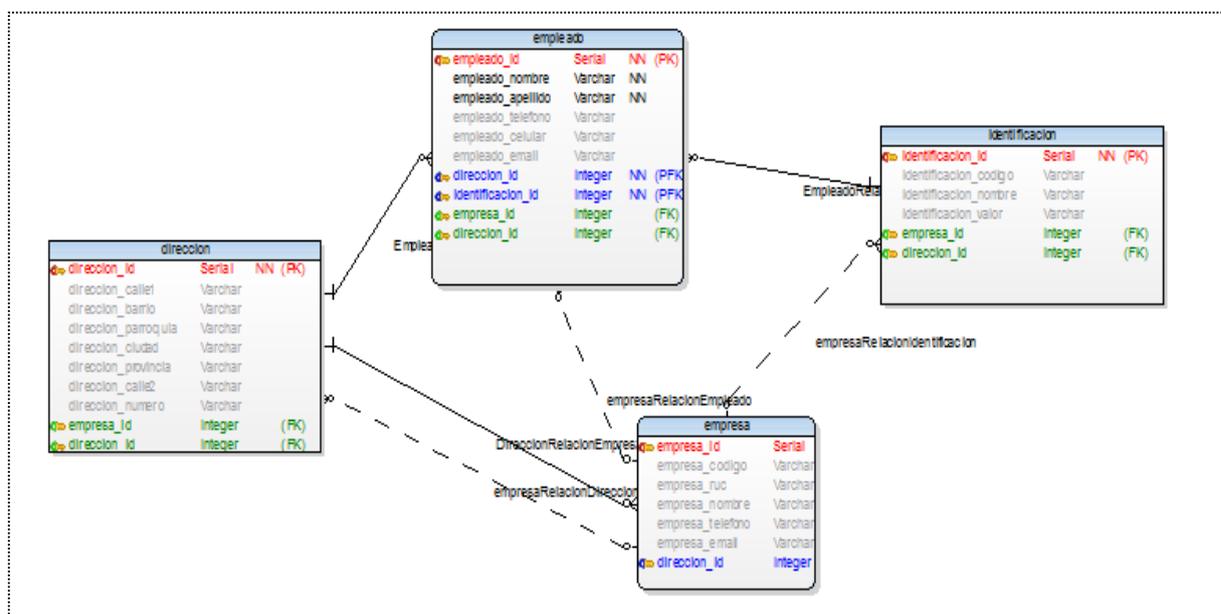


Ilustración 8: Modelo de datos diseñado en Toad Data Modeler

Fuente: Propia

2.1.2 BASE DE DATOS POSTGRESQL:



Ilustración 9: Logo de PostgreSQL

Fuente: (Martinez, 2014)

PostgreSQL es un gestor de base de datos, considerado como uno de los mejores gestores en el mercado del software, un aspecto muy importante es free, es decir que no se tiene que pagar una licencia para usarlo.

No tiene una empresa propietaria sino que es apoyada por un grupo de desarrolladores denominado PGDG (PostgreSQL Global Development Group)

Permite trabajar a nivel de usuarios y roles por usuario

Es usada en varios proyectos como moodle entre otros.

Se usara la versión 9.0 para Windows.

Posee las siguientes características:

Consultas complejas,

- triggers,
- vistas,
- integridad transaccional,
- control de concurrencia, etc.

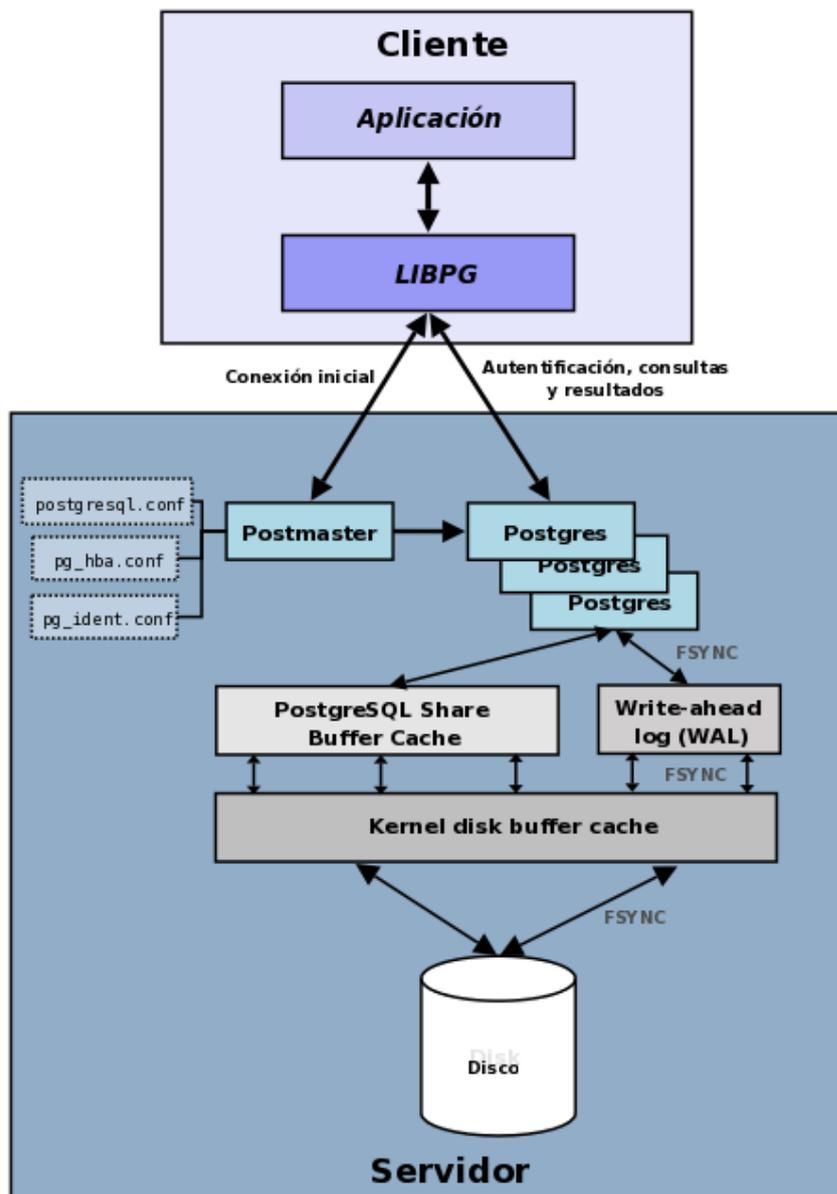


Ilustración 10: Componentes PostgreSQL

Fuente: (PF, 2012)

Es un IDE para PostgreSQL en Windows, posee interfaz gráfica muy amigable, que permite gestionar bases de datos gráficamente o mediante scripts, para la creación de la base de datos del sistema GpiPresupuesto se ejecutara el script generado a través de Toad Data Modeler, y para hacer un seguimiento y modificaciones de la base de datos se lo hará visualmente.

Permite realizar consultas personalizadas y una administración completa sobre la base de datos.

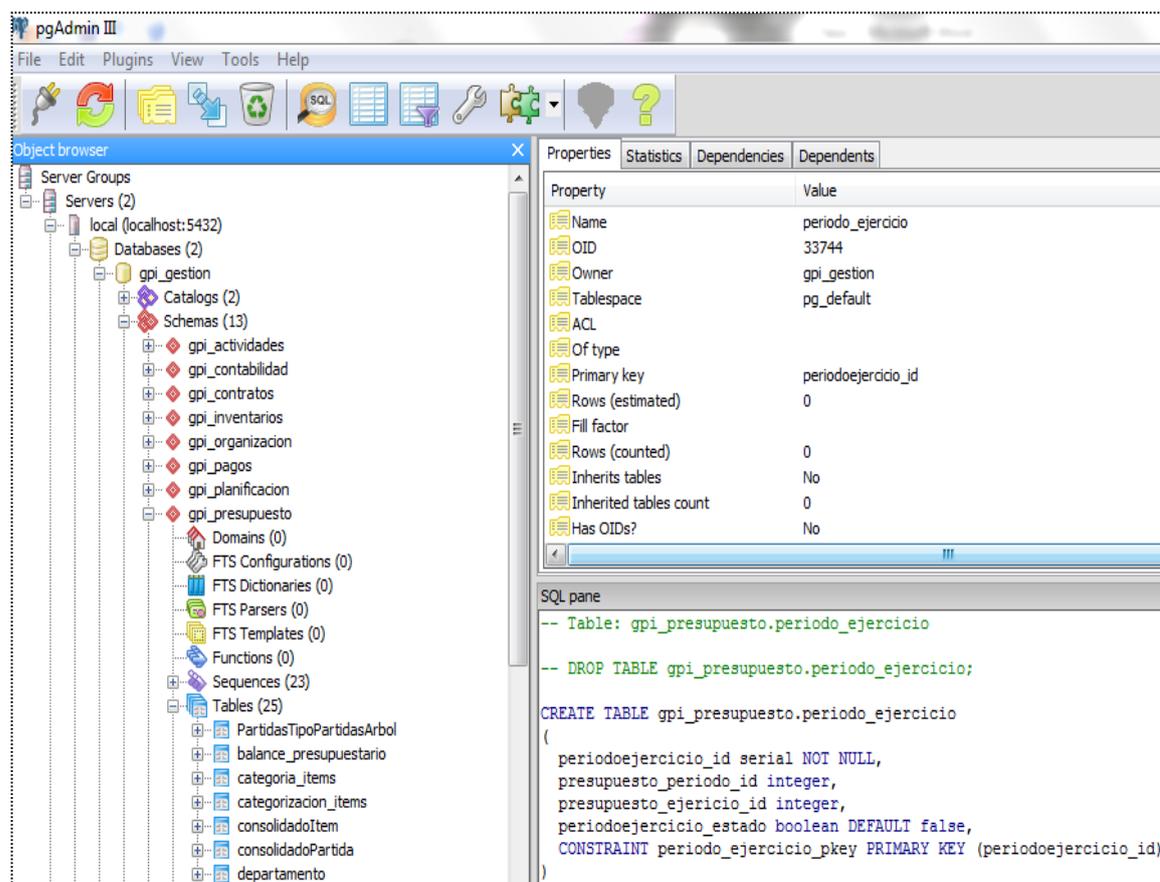


Ilustración 11: Espacio de Trabajo de PgAdmin

Fuente: Propia

Navicat Premium

Es un software que permite interactuar gráficamente la base de datos PostgreSQL, ofrece una interfaz sencilla pero muy rica en herramientas y funcionalidades, al igual que PgAdmin, mediante Navicat Premium también se puede gestionar entidades, relaciones, vistas, consultas, funciones pero además ofrece también un gestor de modelado de base de datos.

En la siguiente ilustración se aprecia el espacio de trabajo y las herramientas que proporciona esta herramienta

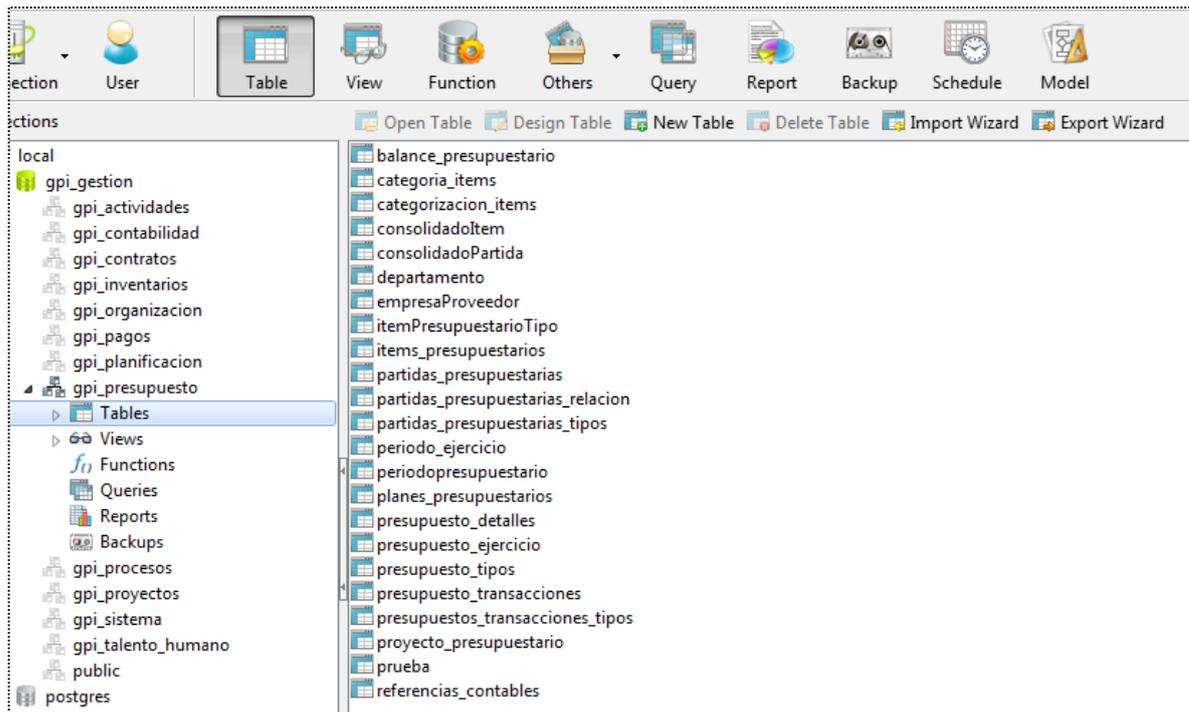


Ilustración 12: Espacio de Trabajo Navicat

Fuente: Propia

2.1.3 SERVIDOR APACHE:



Ilustración 13: Logo de Apache

Fuente: (Foundation, 2014)

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. (AJBDSOFT, 2013)

El nombre "Apache" fue elegido entre el respeto por los nativos americanos de la tribu indígena Apache, conocido por sus habilidades superiores en la estrategia de la guerra y su resistencia inagotable. También hace un retruécano lindo en "un servidor web en parches" -un servidor a partir de una serie de parches- pero esto no fue su origen. El grupo de desarrolladores que lanzó este nuevo software pronto comenzó a llamarse "Grupo Apache".

(Rouse, 2013)

Apache server es de código libre y es apoyada por una comunidad de desarrolladores sin fin de lucro en todo el mundo.

Apache server ha sido uno de los ejes para el desarrollo de múltiples aplicaciones web, es usado en algunos proyectos como Joomla, mambo, etc.

Al igual que otros servidores web, los recursos se consumen mediante un navegador web mediante el puerto 80 o el 8080 a través del protocolo http o https

Aunque viene instalado por defecto en Linux y en algunas otras plataformas, para Windows es necesario descargar el instalador de la página oficial, instalarlo y configurarlo

Una característica muy importante es que soporta PHP, es por eso que se usará como servidor para el sistema GpiPresupuesto ya que Zend Framework está basado en PHP.

Actualmente existen algunos paquetes que ya vienen integrados el servidor apache junto con algunos lenguajes de programación y aplicaciones ya configurados, como son el AppServer, Xampp y Zend Server.

2.1.4 LENGUAJE PHP:



Ilustración 14: Logo de PHP

Fuente: (Group, 2014)

PHP (acrónimo recursivo de **PHP: Hypertext Preprocessor**) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. (PHP, 2013)

Una de las características más importantes es un lenguaje de servidor, es decir que el código PHP se ejecutara en el servidor y no en el cliente.

Se eligió este lenguaje primeramente porque es compatible con servidores como Apache Server que es servidor con el cual se desarrollara esta aplicación, segundamente porque Zend Framework está desarrollado sobre PHP y tanto PHP como Zend Framework son apoyados por la misma organización.

2.1.5 ZEND FRAMEWORK:



Ilustración 15:Logo se Zend Framework

Fuente:(Technologies, 2014)

(Gonzales B. , 2009)

Zend Framework (ZF) es un framework de código abierto para desarrollar aplicaciones web y servicios web con PHP5.ZF es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de ZF es algo único; cada componente está construido con una baja dependencia de otros componentes. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado.

Aunque se pueden utilizar de forma individual, los componentes de la biblioteca estándar de Zend Framework conforman un potente y extensible framework de aplicaciones web al combinarse. ZF ofrece un gran rendimiento y una robusta implementación MVC, una abstracción de base de datos fácil de usar, y un componente de formularios que implementa la prestación de formularios HTML y validación para que los desarrolladores puedan consolidar todas las operaciones usando de una manera sencilla la interfaz orientada a objetos.

Para el desarrollo del sistema GpiPresupuesto se usará Zend Framework como herramienta de persistencia de datos. Es decir para convertir las tablas relacionales de base de datos en objetos PHP.

Una gran ventaja es que Zend Framework es modular, es decir que se puede usar por separado los módulos dependiendo del uso requerido.

Existen otros framework MVC que también han sido desarrollados en PHP, como son Symfony, pero la particularidad de Zend Framework es que proporciona un protocolo para la comunicación con Actionscript.

Los módulos utilizados en el desarrollo de la aplicación son Zend_Db y Zend_Amf **Zend_Db**, que implementa las funcionalidades la conexión con la base de datos y la interacción con los elementos de la misma.

El módulo **Zend_Db_Adapter** proporciona drivers para las siguientes bases de datos relacionales:

- IBM DB2
- MySQL
- Microsoft SQL Server,
- Oracle
- PostgreSQL
- SQLite

Zend_Amf: provee apoyo para el Formato de Mensajes de Actionscript Action Message (AMF) de Adobe, que permite la comunicación entre Adobe Flash y PHP.

Específicamente, proporciona una aplicación para un servidor gateway que tramita las solicitudes enviadas desde Flash Player al servidor, mapeando estos requerimientos al objeto y a sus métodos de clase, como así también a llamadas arbitrarias de comunicación.(Gonzales B. , 2009)

2.1.6 APLICACIONES RIA

RIA es un acrónimo de Rich Internet Application, son Aplicaciones Web Ricas, es decir que la aplicación se ejecuta desde el cliente.

Una aplicación RIA se aproxima a una aplicación de escritorio ya que no hace una actualización completa de la pantalla si no que solamente de una sección de la aplicación cuando lo requiera, es decir cuando se interactuó con información de la base de datos.

Una aplicación RIA es un gran referente de las aplicaciones web 2.0, ya que contiene aplicaciones y contenido multimedia, y proporciona servicios para el usuario, entonces el usuario se convierte en protagonista del contenido web y no solo en espectador.

En la web 2.0 encontramos páginas con contenido interactivo.

Entre ellos encontramos a páginas como:

- Redes sociales
- Blogs
- Wikis
- Videosharing

Arquitectura:

En las siguientes ilustraciones se puede apreciar las diferencias entre una aplicación web tradicional con una aplicación RIA

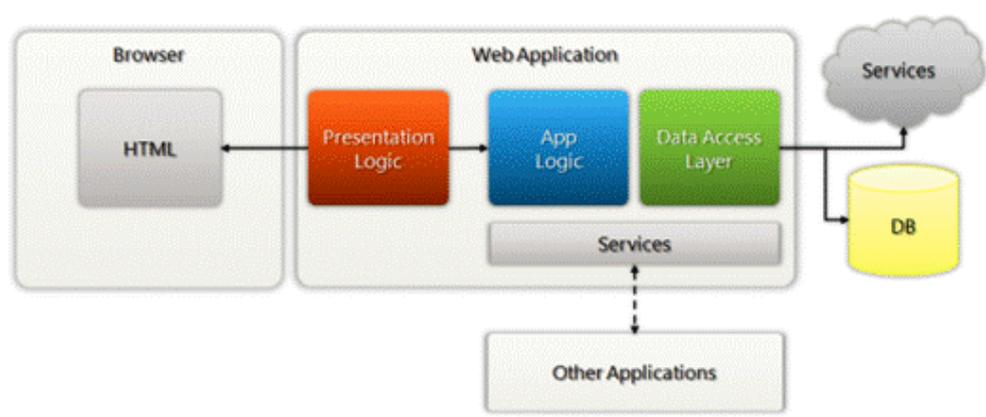


Ilustración 16: Aplicación Web tradicional

Fuente: (Gabriellebet, 2014)

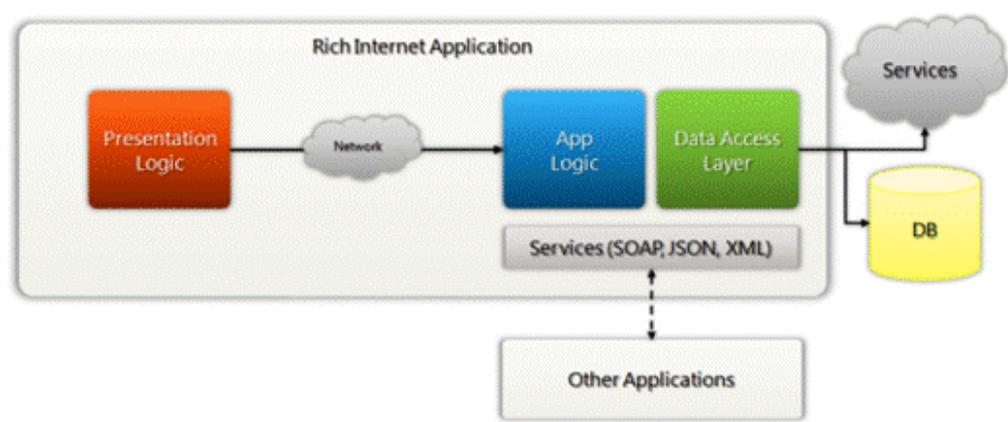


Ilustración 17: Aplicación RIA

Fuente:(Gabriellebet, 2014)

Una Aplicación RIA puede ser utilizada en tres ambientes:

- Aplicaciones de Escritorio
- Aplicaciones Web
- Aplicaciones Móviles



Ilustración 18: Usos de Aplicaciones RIA

Fuente: (Pyme, 2013)

Beneficios:

- Solamente hace peticiones al servidor cuando sea necesario, es decir cuando se quiera obtener información de la base de datos.
- Contiene aplicaciones multimedia: audio, video, etc.
- Se puede acceder desde cualquier navegador, (el navegador debe contener el plugin necesario, flash player en el caso de una aplicación Flex)
- Distribución de la aplicación entre el cliente y el servidor

Herramientas de Desarrollo

Actualmente existen varias herramientas que permiten desarrollar Aplicaciones RIA, A continuación las herramientas más usadas para el desarrollo de aplicaciones RIA.

- Flex – Adobe, (actualmente Apache)
- SilverLight - Microsoft
- OpenLaszlo - Libre
- Rich Faces - JBoss
- PrimerFaces

Aplicaciones

Existen varias aplicaciones en la web que han sido desarrolladas sobre tecnología RIA

- Netflix
- Flickr
- Meebo

2.1.7 ADOBE FLEX:



Ilustración 19: Logo de Flex

Fuente: (Ireland, Adobe Flash Builder 4.7 Premium, 2014)

Flex es un framework que permite crear aplicaciones web 2.0 y también aplicaciones de escritorio, permite que la aplicación se ejecute en flash player, es ideal para desarrollar aplicaciones RIA, que contengan reproductores multimedia, aplicaciones de mapas, entre otros.

El framework Flex contiene tres tipos de lenguaje:

MXML: Significa Macromedia XML, este lenguaje se usa para el desarrollo de la interfaz de la aplicación, la inserción de componentes propios de flex y personalizados.

Actionscript 3.0: es un lenguaje orientado a objetos, se usa para el desarrollo de la lógica de la aplicación para el cliente y de la comunicación con los servicios que se desea consumir (puede ser mediante Zend Framework o Blase)

FXG: Flash XML Graphigs, permite crear gráficos.

Las aplicaciones flex se implementan como archivos swf, lo cual es una gran ventaja ya que un archivo swf es muy pequeño y eso hace que la aplicación pueda descargarse mucho más rápido en el cliente.

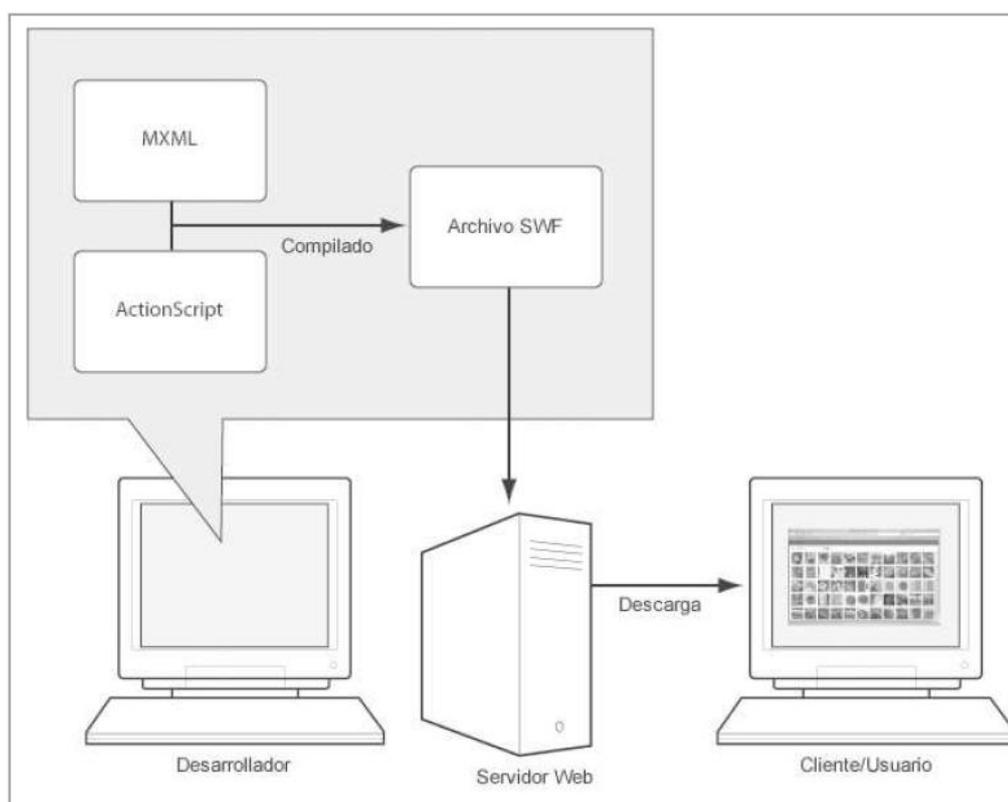


Ilustración 20: Compilación y publicación de una aplicación Flex

Fuente: (Carrizo, 2011)

Sdk de Flex:

Software Development Kit es una biblioteca de componentes, compilador y herramientas de documentación.

El SDK hace más fácil el desarrollo de una aplicación flex, por ejemplo para el diseño de un componente solamente se inserta la etiqueta del componente mediante tags xml y listo, también para componentes personalizados.

Se puede desarrollar aplicaciones Flex usando cualquier editor de textos y el compilador que contiene el SDK.

Aplicaciones AIR:

Adobe Integrated Runtime, se usa para el desarrollo de aplicaciones en escritorio, una ventaja muy importante de las aplicaciones AIR, es que es independiente del sistema operativo, es decir que funcionara en Windows, Linux, Mac, etc.

Razones para Usar Flex:

Desarrollo de Aplicaciones: la gran variedad de componentes que trae consigo Flex además de los componentes personalizados que se pueden desarrollar hace que el desarrollo de una aplicación Flex sea mucho más rápida, una aplicación Flex se ejecuta en un navegador web o se puede implementar como aplicación AIR.

Interactividad: esta es una característica muy destacable de flex, ya que es posible desarrollar aplicaciones con gran variedad de efectos y transiciones; además del fácil acceso con los datos de un servicio.

Arquitectura MVC: Aunque el desarrollador es libre de implementar la arquitectura MVC en su aplicación, existen algunas bibliotecas que permiten la implementación de esta arquitectura, lo cual hace que el código sea más organizado.

Es Gratis: Aunque el IDE Flash Builder, que es el más apropiado para el desarrollo de una aplicación Flex, es pagado, es decir, toca pagar por una licencia de una copia de este IDE.

Pero el Framework Flex es totalmente gratis, se puede desarrollar aplicaciones mediante un editor de textos y el compilador que trae consigo la SDK de flex, sin tener que pagar absolutamente nada.

Es abierto: el código de Flex es totalmente abierto, esto hace posible que se desarrolló en componentes personalizados lo cual es una gran ayuda al momento de integrar en una aplicación.

Existen empresas que se dedican al desarrollo de componentes Flex como son: FlexLib, klovis; estos componentes se los puede descargar libremente del internet

Así también existen empresas que desarrollan componentes personalizados propietarios, es decir suben al internet un demo, pero si quiere implementar en su aplicación tiene que pagar una licencia.

(Robison Elijah, 2011)

Comunicación con Servicios: debido a que Flex es un framework de interfaz de usuario, para comunicarse con una base de datos tiene que consumir servicios.

Estos pueden ser:

PHP by Zend

Http

Web Service

XML

2.1.8 IDE FLASH BUILDER 4.5 FOR PHP



Ilustración 21: Logo de Flash Builder

Fuente: (Ireland, Flex, 2014)

Es una herramienta (IDE) de desarrollo basada en Eclipse creada en asociación con Zend Technologies para crear aplicaciones móviles, web y de escritorio usando PHP, Actionscript y el Framework Flex. Además, utiliza las herramientas de prueba de gran alcance para crear aplicaciones para empresas.

Existen otras opciones para desarrollar un sistema en Flex, como es usando un editor de código junto con el SQK para compilar el programa, pero la opción más recomendable es usar Flash Builder.

A continuación las características más relevantes:

- editor de código personalizable
- completado de código
- informativo de documentación del framework
- compilado automático
- herramientas para depuración
- Utilidades para el desarrollo

Existen dos modos de instalar Flash Builder, la una es con una instalación independiente, la otra es a través de eclipse mediante un plugin.

Perspectivas: es una la estructura del espacio de trabajo que ofrece Flex Builder 4, son usadas dependiendo del trabajo o la arquitectura en la cual se desarrolla, por ejemplo si se mapea objetos mediante Zend Framework se selecciona la perspectiva que lleva el mismo nombre, si se desarrolla con Flex se trabajara en la perspectiva Flash, y es la que viene seleccionada por defecto.

Debido a que Flash Builder fue desarrollado basado en Eclipse se puede personalizar una perspectiva y guardarla.

A continuación las perspectivas que nos ofrece Flash Builder:

- Flash (es la que viene por defecto, para desarrollo)
- Flash Debug
- SVN Repository Explorer
- Code Tracing
- CVS Repository Explorer
- Database Debug
- Database Development
- Flash Profile
- Java
- Java Browsing
- Java Type Hierarchy
- JavaScript
- PHP

- PHP Profile
- Planning
- Remote System Explorer
- Resource
- SVN Repository Explorer
- Team Synchronizing
- Web
- Web Browsers Tools
- XML
- Zend Framework

Espacio de Trabajo:

El espacio de trabajo es muy amigable, y para aquellos desarrolladores que están acostumbrados a trabajar en Eclipse se hará mucho más fácil familiarizarnos con el IDE, proporciona una paleta de componentes propios de Flex, mediante los cuales nosotros se puede arrastrar y soltar el componente seleccionado en el área de trabajo, esta acción se realizara muchas veces en un diseño de una interfaz gráfica, o para aquellos que están acostumbrados a digitar en código, también existe esta opción, a través de tags, llamando a los componentes se puede construir una interfaz muy llamativa y completa solamente programando.

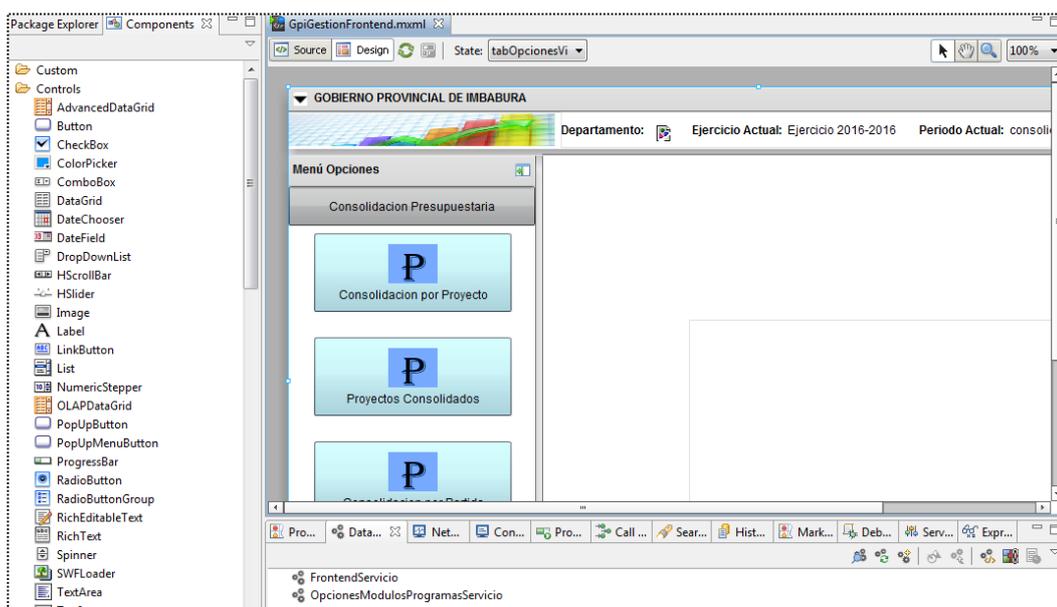


Ilustración 22: Espacio de Trabajo de Flash Builder 4.5

Fuente: Propia

Integración con Zend Framework:

Flash Builder 4 for PHP fue desarrollado con el objetivo de mantener una sincronización con Zend Framework y PHP, ya que la empresa que las dos herramientas fueron desarrollados por la misma empresa.

Primeramente se configura Zend Framework en el servidor de Aplicaciones, cuando se haya instalado Flash Builder, este ya cuenta con una herramienta para ejecutar comandos Zend, como son el mapeo de tablas, relaciones, etc.

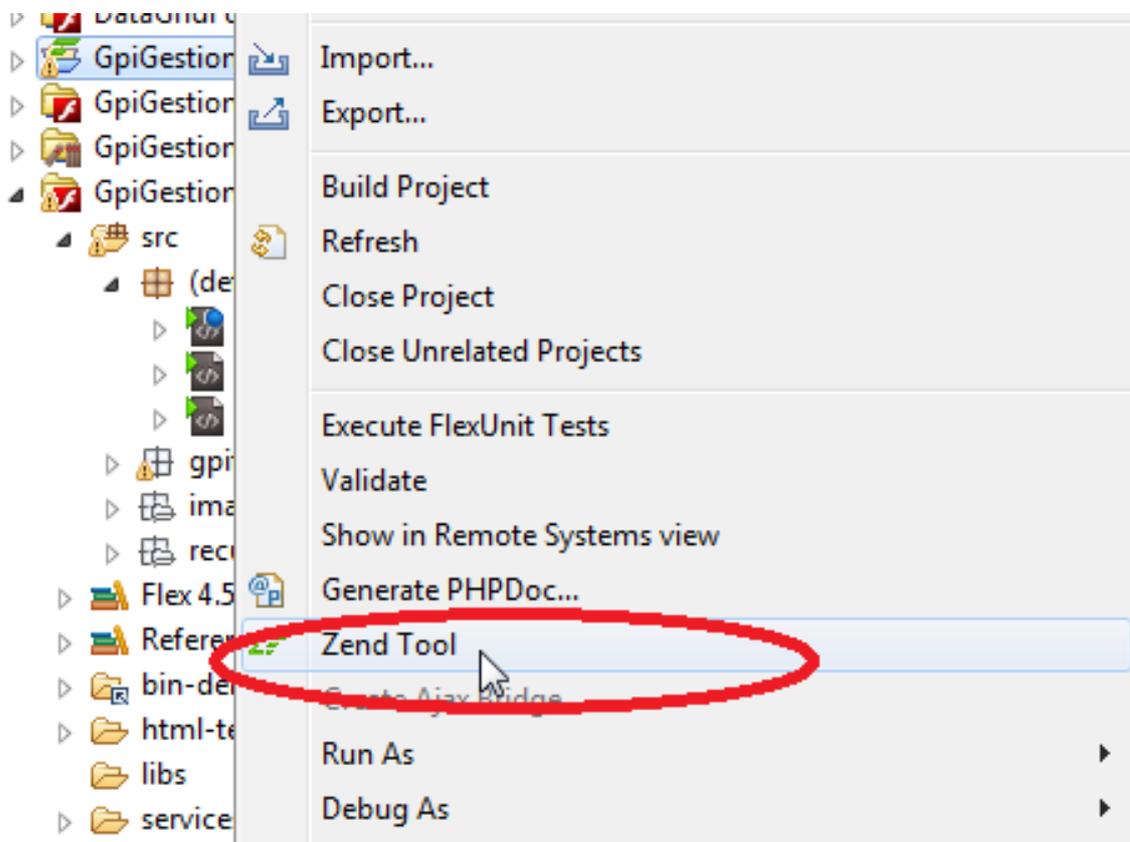


Ilustración 23: Zend Tool en Flash Builder 4

Fuente: Propia

Zend Tool es un ejecutor de comandos zend usado en Flash Builder, mediante esta funcionalidad se ejecuta sentencias de persistencia, como por ejemplo para mapear tablas, relaciones, etc.

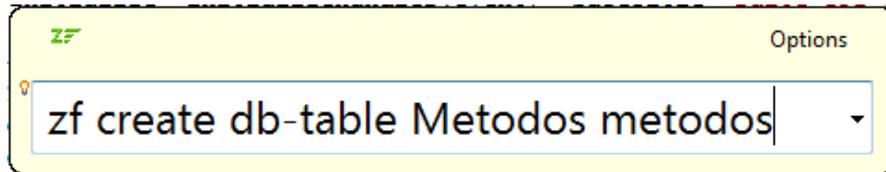


Ilustración 24: Ejecutor de Comandos de Zend Tool

Fuente: Propia

Servicios:

Flash Builder ofrece wizards que permiten crear y sincronizar los servicios Flex con los servicios de otras tecnologías.

PHP by Zend: Conecta o crea funciones en clases PHP usando Zend Framework.

Http: Conecta a uno o a muchos URLs http. Soporta XML y formatos de datos JSON

WSDL: Conecta a un servicio web SOAP usando definición WSDL

XML: conecta a un archivo XML local o a una URL retornando una respuesta XML.

Estos servicios se gestionan mediante wizards o asistentes que facilitan el manejo y la conexión con los servicios remotos de otras tecnologías

En la siguiente ilustración se puede apreciar los servicios que proporciona Flash Builder.

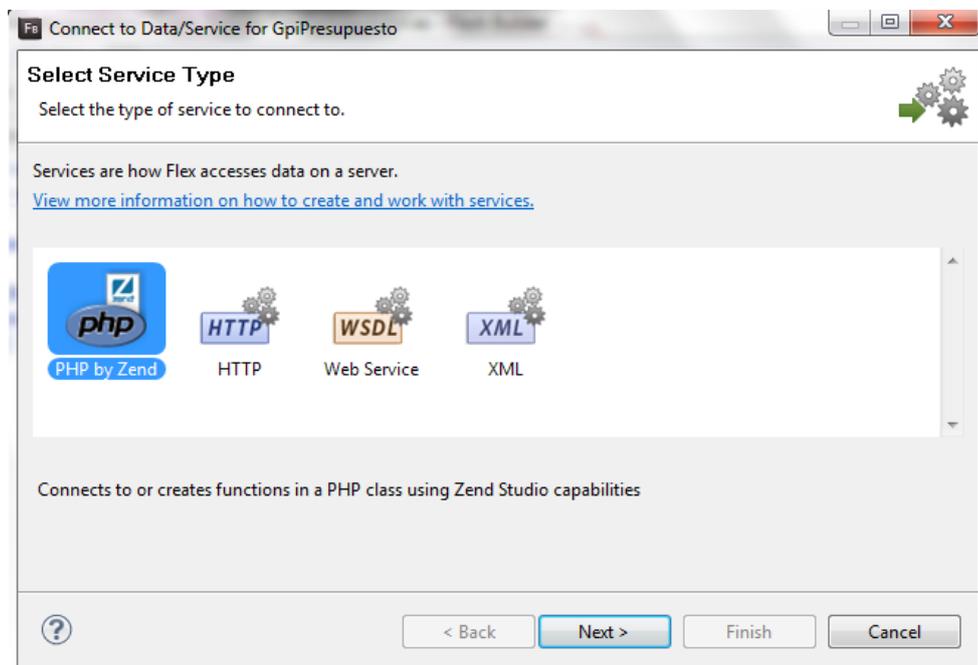


Ilustración 25: Servicios que proporciona Flash Builder 4.5

Fuente: Propia

2.1.9 TECNOLOGÍA BPM:

(KIRAN GARIMELLA)

Business Process Management (BPM) es un conjunto de métodos, herramientas y tecnologías utilizados para diseñar, representar, analizar y controlar procesos de negocio operacionales. BPM es un enfoque centrado en los procesos para mejorar el rendimiento que combina las tecnologías de la información con metodologías de proceso y gobierno. BPM es una colaboración entre personas de negocio y tecnólogos para fomentar procesos de negocio efectivos, ágiles y transparentes.

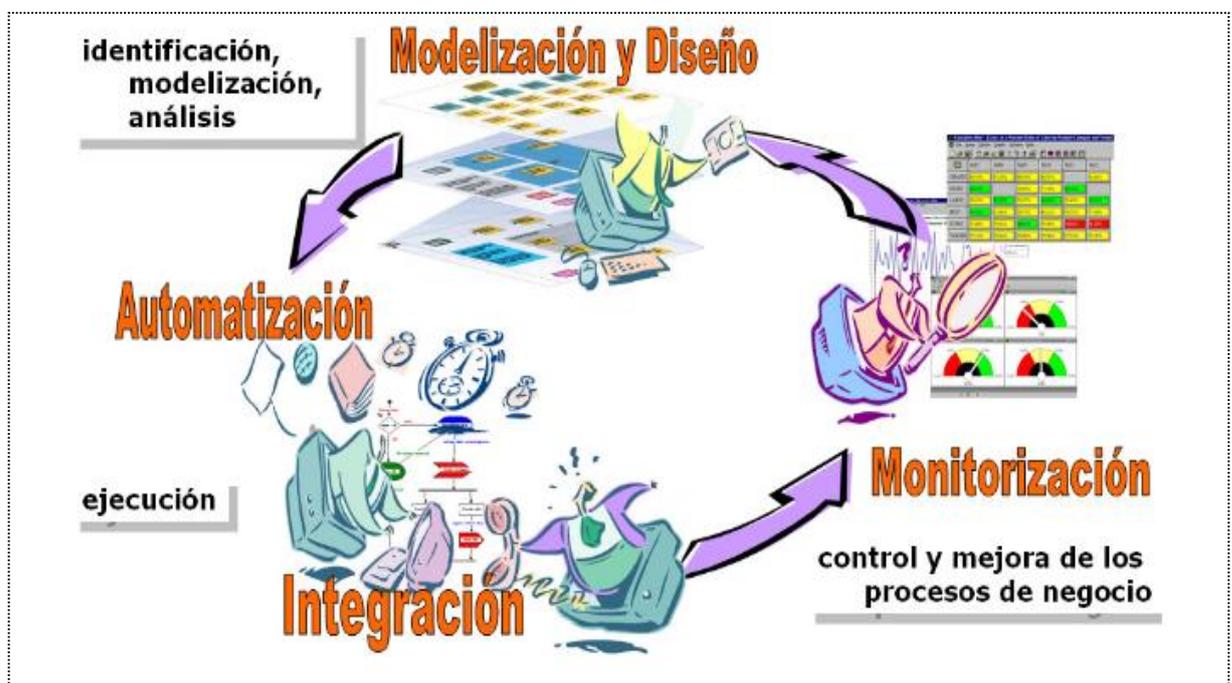


Ilustración 26: Proceso BPM

Fuente: (Club-BPM, 2010)

Existen diversas herramientas tanto libres como pagadas que nos permiten modelar procesos de negocio.

Es necesario diferenciar las herramientas desarrolladas solamente para diagramar procesos de negocio y las herramientas con las que se implementara el proceso BPM.

2.1.10 MODELADOR DE PROCESOS

Nos ayudan solamente a determinar cuáles son los procesos del negocio, y solamente eso, estas herramientas proveen de un conjunto de instrumentos los cuales nos ayudan a diagramar un proceso de negocio, algunas son básicas como el caso de Microsoft Visio; mientras que otras son más complejas, es decir poseen muchos más instrumentos para organizar los procesos, inclusive poseen un compilador para determinar si el diagrama está modelado correctamente, de estar mal saldrán mensajes de errores, y no se quitarán hasta que se corrijan y se compile nuevamente.

A continuación algunos ejemplos de modeladores BPM.

- Microsoft Visio
- Adobe LiveCycle Workflow
- Bisil – Enj
- Savvion Business Manager™ 6.5
- TIBCO BPM

2.1.11 ADMINISTRADOR DE PROCESOS DE NEGOCIO

Estas herramientas son las que nos permiten administrar, visualizar, realizar un seguimiento de los procesos de principio a fin, y establecer los actores en los mismos.

En estas herramientas obviamente también se debe modelar procesos, pero con la diferencia de las herramientas anteriores, estas son las que van a generar un motor de procesos y el proceso en la organización deberá fluir conforme a las reglas de enrutamiento de los procesos.

El motor de procesos debe alojarse en un servidor de aplicaciones, y los usuarios deberán acceder a este mediante un cliente http, es decir con un navegador web se accede tanto a la administración como al seguimiento de los casos y procesos.

Se recomienda diagramar el proceso de negocio primero en una herramienta de modelado y luego diagramar en la herramienta de administración de procesos para su implementación.

A continuación de algunas herramientas de administración de procesos:

- Bonita BPM
- JBPM
- ProcessMaker

Para la implementación de procesos presupuestarios en el GPI, y la integración con el sistema transaccional GpiPresupuesto se determinó que la herramienta más apropiada será ProcessMaker ya que es desarrollada en php, mientras que las otras en java, además por otras características que serán presentadas más adelante.

2.1.12 TIBCO BUSINESS STUDIO

Es una herramienta BPM que permite modelar procesos de negocio mediante el estándar BPMP, es libre y permite simular procesos reales de las empresas.

Es basado en eclipse de manera que los desarrolladores que están familiarizados a trabajar en eclipse el aprendizaje se realizara mucho más rápido, proporciona un espacio de trabajo amigable en él se diseña gráficamente los procesos de negocio, sus flujos, y sus relaciones entre sí.

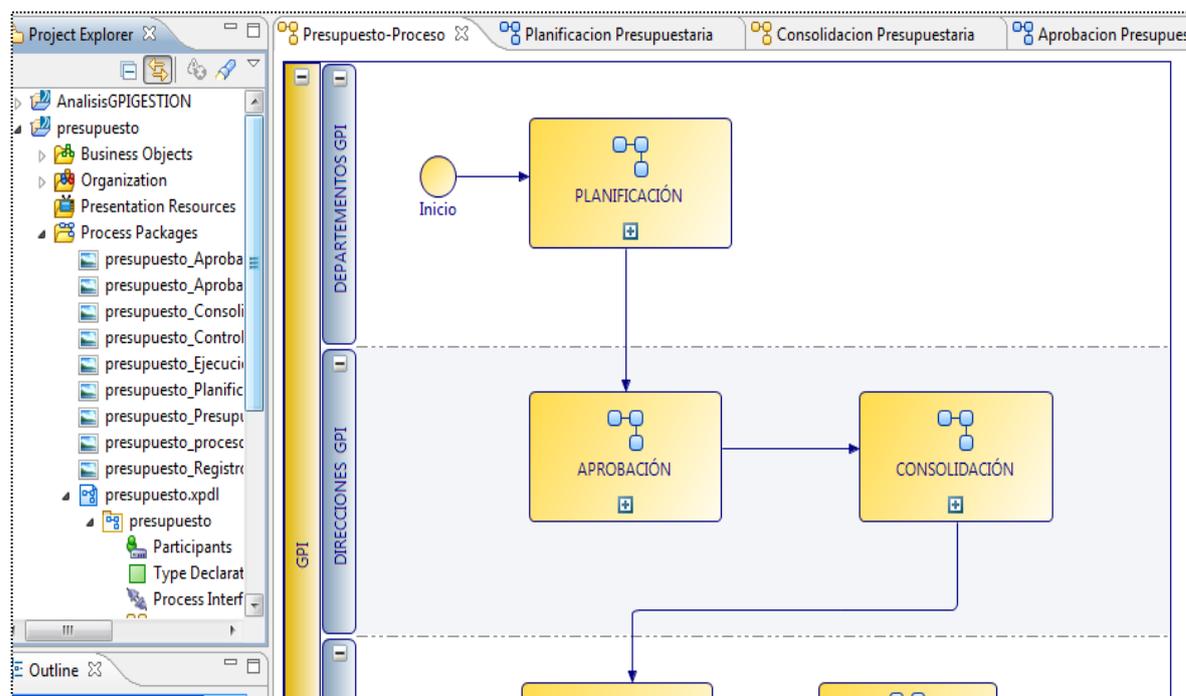


Ilustración 27: Espacio de Trabajo de TIBCO Business Studio

Fuente: Propia

Además posee una paleta de controles BPMP que facilitan el modelado de los procesos BPM.

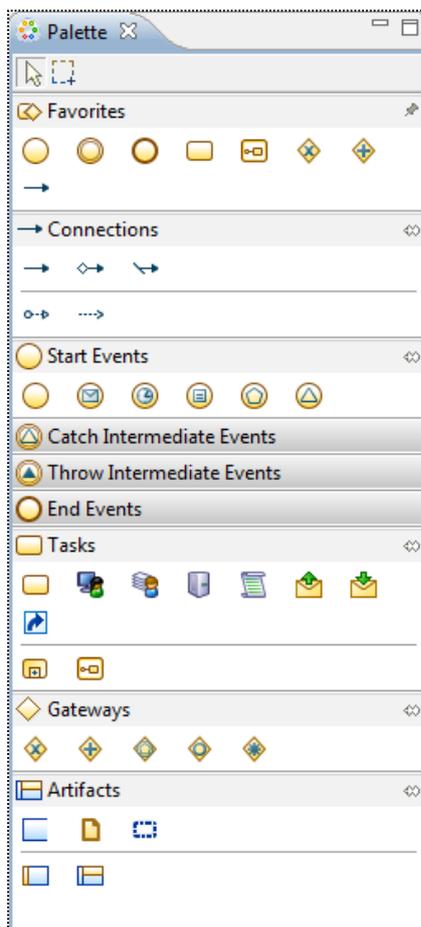


Ilustración 28: Paleta de Herramientas TIBCO Business Studio

Fuente: Propia

2.1.13 PROCESSMAKER

ProcessMaker es un gestor de procesos de negocio (BPM) de código abierto líder de procesos de negocio (BPM) y software de flujo de trabajo diseñado para pequeñas y medianas empresas y organizaciones proporcionando automatización del flujo de trabajo y gestión de procesos de negocio.

Su aplicación de usuario amigable ayuda a los usuarios de negocio para gestionar el flujo de trabajo eficiente y eficaz. Bajo experiencia programador puede diseñar y ejecutar workflows que ayuda en la reducción de papeleo.

ProcessMaker es una herramienta BPM basada en la web que facilita el uso del software de gestión de procesos desde cualquier lugar. Es flexible y personalizable según sus necesidades de negocio, lo que es más ProcessMaker conseguir fácilmente se conecta con sus aplicaciones empresariales existentes incluyendo ERP, CRM, BI y mucho más que asegurar que no haya duplicación de tareas.

(Solutions Bista)

Una de las características de ProcessMaker es libre, aunque también existe la versión empresarial, esta última contiene los plugins que son extensiones para funcionalidades extras muy importantes también para la administración de procesos.

Otra característica también es que los plugins se pueden importar en la versión free mediante la compra o accediendo a un plan pack plugin a través de la empresa Colosa, que es la que tiene los derechos sobre ProcessMaker.

Entre estos plugins tenemos:

- Simple reporting: usado para reportes sobre el proceso, los datos, las variables y los usuarios que interactúan en el proceso BPM
- Advanced Dashboards: Son cuadros de mando avanzados y personalizables sobre el proceso, sus tareas y variables

2.1.13.1 VERSIONES

ProcessMaker inició con la versión 2.030 y a medida del paso del tiempo y de la detección de bugs ha ido evolucionando generando una nueva versión cada vez más completa y estable.

La versión más actual y estable es la 2.5.2 y es la que usaremos para la implementación BPM del GPI.

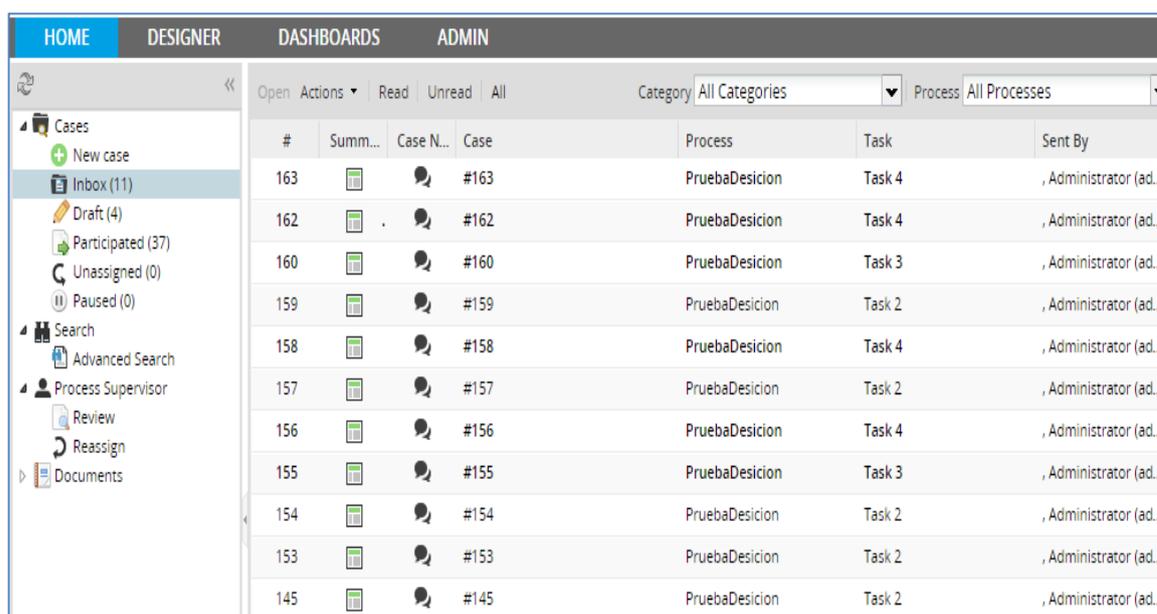
2.1.13.2 ESPACIO DE TRABAJO

ProcessMaker no solo contiene el espacio de trabajo para el modelamiento de flujo de procesos, también posee una bandeja de entrada en donde llegan las notificaciones de los procesos y casos cuando el proceso ya se encuentre activo, también posee Dashboards y la pantalla de administración de datos de configuración, asignación de usuarios, entre otros.

Estas ventanas de processmaker se visualizaran de acuerdo al permiso que tenga el rol del usuario logeado

2.1.13.3 HOME (INICIO)

Contiene una bandeja de entrada en donde se visualizara las notificaciones de las tareas que debe realizar el usuario, las participaciones en los casos, y también la opción para iniciar un nuevo caso; esta última acción se podrá efectuar siempre y cuando el usuario tenga participación en una tarea inicial de un proceso.



The screenshot shows the 'HOME' tab of the ProcessMaker interface. On the left, there is a sidebar with navigation options: 'Cases' (with sub-items: 'New case', 'Inbox (11)', 'Draft (4)', 'Participated (37)', 'Unassigned (0)', 'Paused (0)'), 'Search' (with sub-items: 'Advanced Search'), 'Process Supervisor' (with sub-items: 'Review', 'Reassign'), and 'Documents'. The main area displays a table of cases. The table has columns for '#', 'Summ...', 'Case N...', 'Case', 'Process', 'Task', and 'Sent By'. The data rows show cases numbered 163 down to 145, all belonging to the 'PruebaDesicion' process and assigned to 'Task 4' or 'Task 3'. The 'Sent By' column for all entries is ', Administrator (ad...'. Above the table, there are filters for 'Open', 'Actions', 'Read', 'Unread', 'All', 'Category' (set to 'All Categories'), and 'Process' (set to 'All Processes').

#	Summ...	Case N...	Case	Process	Task	Sent By
163			#163	PruebaDesicion	Task 4	, Administrator (ad...
162			#162	PruebaDesicion	Task 4	, Administrator (ad...
160			#160	PruebaDesicion	Task 3	, Administrator (ad...
159			#159	PruebaDesicion	Task 2	, Administrator (ad...
158			#158	PruebaDesicion	Task 4	, Administrator (ad...
157			#157	PruebaDesicion	Task 2	, Administrator (ad...
156			#156	PruebaDesicion	Task 4	, Administrator (ad...
155			#155	PruebaDesicion	Task 3	, Administrator (ad...
154			#154	PruebaDesicion	Task 2	, Administrator (ad...
153			#153	PruebaDesicion	Task 2	, Administrator (ad...
145			#145	PruebaDesicion	Task 2	, Administrator (ad...

Ilustración 29: Buzón de entrada del proceso bpm

Fuente: Propia

2.1.13.4 DESIGNER

ProcessMaker provee de un modelador de procesos siguiendo las convenciones bpmn, (Modelo y Notación de Procesos de Negocio), consta de una paleta de herramientas que nos permiten crear formas y modelar un flujo de procesos totalmente funcional.

Para acceder al diseño de un proceso, primeramente tenemos que crearlo, esta acción es muy fácil, solamente se requiere el nombre y la descripción del mismo.

Esta herramienta soporta varios procesos, en la ilustración podemos observar varios procesos creados, y el menú para crear un nuevo proceso o editarlo.

HOME		DESIGNER	DASHBOARDS	ADMIN
+ New ✎ Edit ⊘ Deactivate ⊖ Delete 📄 Export 📄 Import				
	Process Title	Category	Status	User Owner
<input type="checkbox"/>	Aprobacion	No Category	Active	Administrator
<input type="checkbox"/>	Consolidacion	No Category	Active	Administrator
<input checked="" type="checkbox"/>	Gestion Presupuesto	No Category	Active	Administrator
<input type="checkbox"/>	Planificacion	No Category	Active	Administrator
<input type="checkbox"/>	PruebaDecision	No Category	Active	Administrator

Ilustración 30: Procesos Implementados en el GPI

Fuente: Propia

El menú de modelamiento se desplegará realizando click derecho en el espacio de diseño.

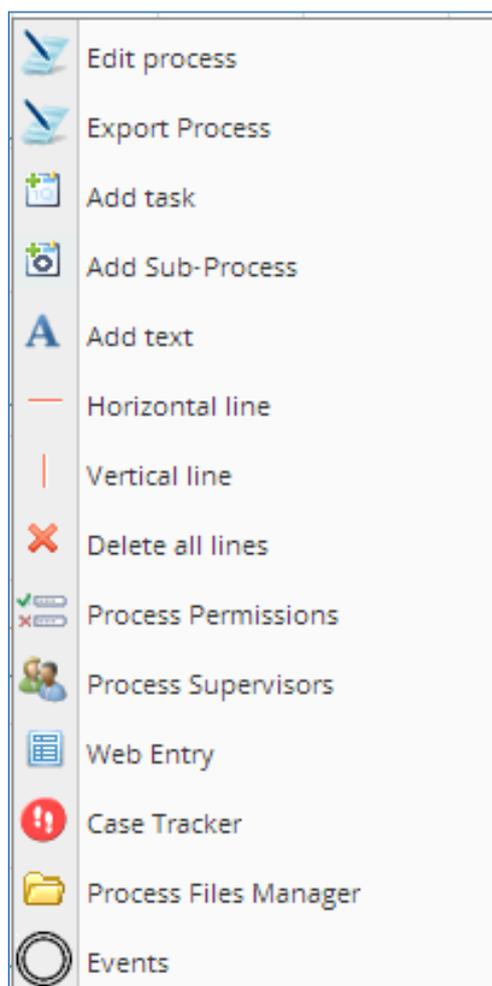


Ilustración 31: Menú para el modelamiento de procesos en processmaker

Fuente: Propia

Este menú nos proporcionara las funcionalidades básicas para modelar un flujo de trabajo; en el podemos encontrar funcionalidades como editar el proceso, exportar el proceso, añadir tarea, esta última es muy importante y es la base para el flujo del proceso, así como también podemos añadir líneas horizontales y verticales; y textos que nos permiten organizar y documentar de mejor manera el flujo del proceso.

Para las conexiones entre tareas es necesario usar la paleta de herramientas que se encuentra en la parte superior del espacio de diseño, existen reglas simples como conexiones directas así como complejas, por ejemplo reglas de decisión, enrutamiento paralelo.



Ilustración 32: Cuadro de herramientas de enrutamiento

Fuente: Propia

Es necesario en todo flujo de proceso asignar una tarea inicial y otra final; esto se lo consigue mediante la paleta de herramientas de enrutamiento, solamente con la acción de arrastrar y soltar en la tarea que se quiere asignar.

A continuación un ejemplo del flujo básico de un proceso.

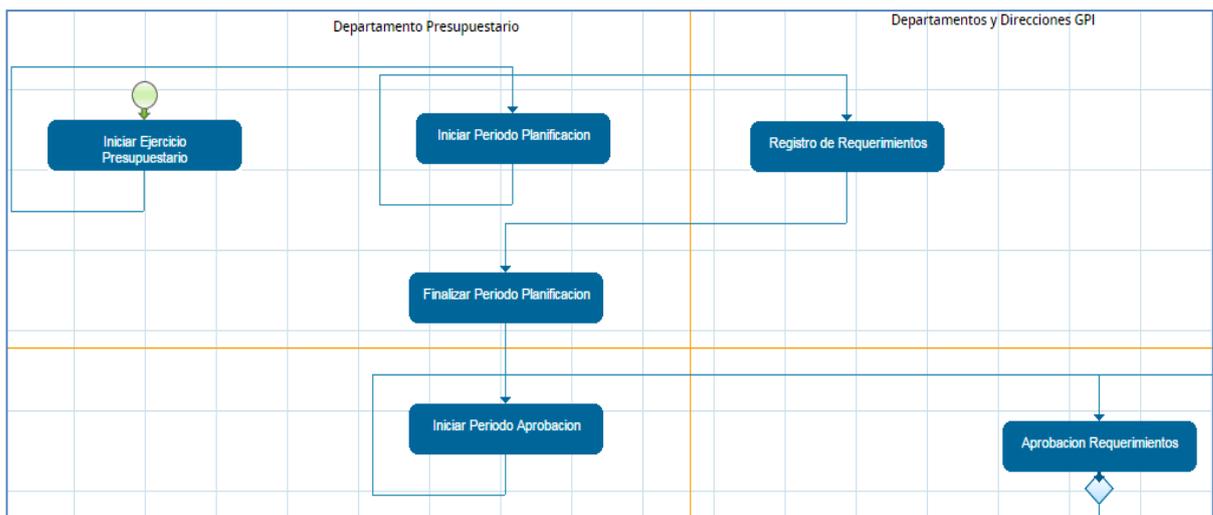


Ilustración 33: Flujo de procesos en processmaker

Fuente: Propia

2.1.13.5 DYNAFORMS

Una característica muy importante de processmaker es que nos permite crear formularios dinámicos personalizables que servirán de gran ayuda para el ingreso, procesamiento y resultado de información.

Entre los elementos web se pueden crear dentro de un formulario tenemos, cajas de texto, botones, radiobuttons, checkBox, links, tablas, entre otros.

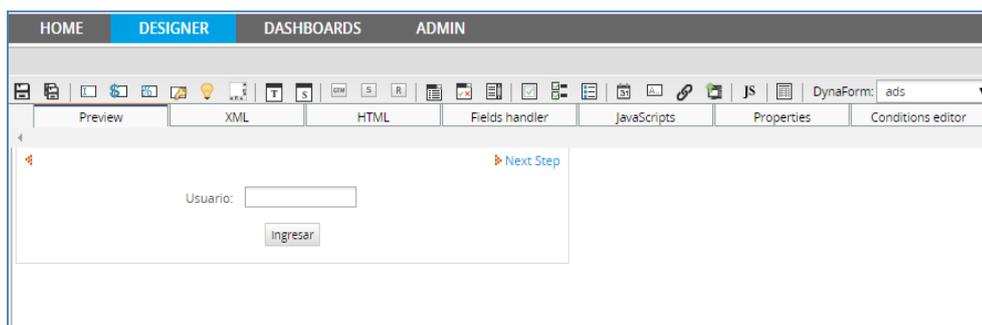


Ilustración 34: Formulario que proporciona processmaker

Fuente: Propia

Estos DynaForms posteriormente se asignaran a las tareas según el requerimiento y funcionalidad del proceso.

2.1.13.6 TRIGGERS Y CONEXIÓN A BASE DE DATOS

Podemos conectarnos a una base de datos desde el motor processmaker, ya sea para obtener información o ejecutar alguna actualización.

Para realizar una conexión a la base de datos debemos asignar la dirección ip (host) del servidor bd, el nombre de la base de datos, el usuario y el password de la misma.

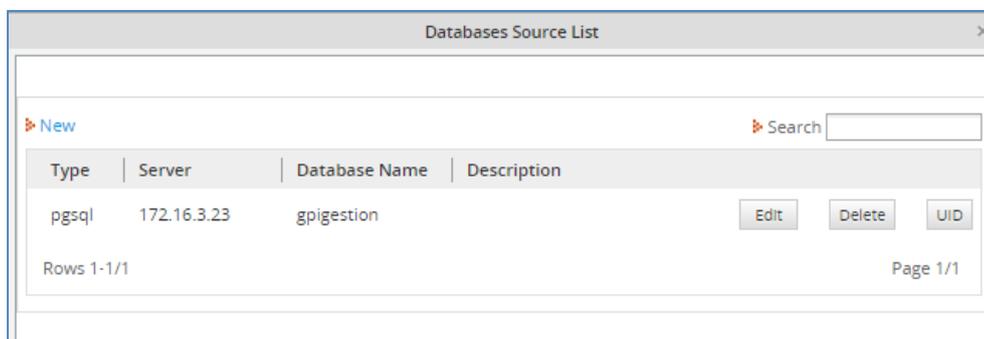


Ilustración 35: Lista de base de datos conectados en processmaker

Fuente: Propia

Una vez realizada la conexión se pueden ejecutar sentencias sql directamente a la base de datos, estas sentencias en processmaker son conocidas como Triggers o disparadores, ya que dependen de la acción de alguna tarea para ejecutarse.

Cada tarea tiene la funcionalidad para asignar Triggers antes o después del enrutamiento.

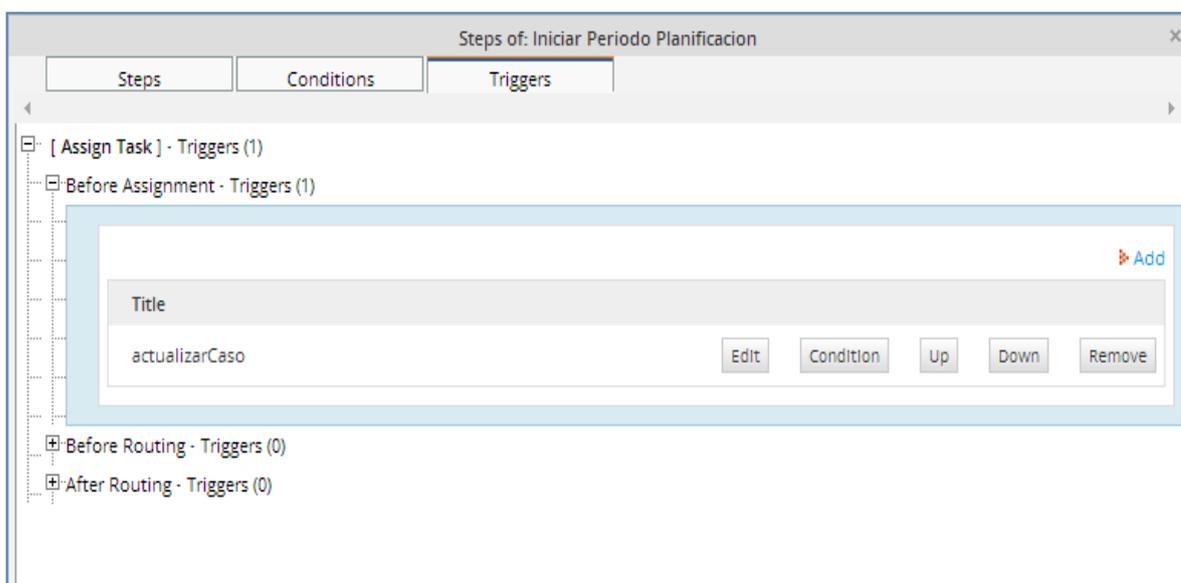


Ilustración 36: Trigger asignado a una tarea

Fuente: Propia

2.1.13.7 DASHBOARD

Uno de los objetivos principales de la buena gestión de procesos de negocio es mejorar continuamente los procesos, esto es posible mediante la medición y seguimiento de los mismos.

ProcessMaker proporciona tablero de comandos en forma de widgets (ventanas flotantes), con diferentes funcionalidades que permiten visualizar, supervisar y monitorear el flujo de los procesos y obtener reportes reales tanto gráficos como a través de tablas.

La versión free de processmaker proporciona un solo dashboard funcional; este mide los casos abiertos y completados asignados a un proceso.

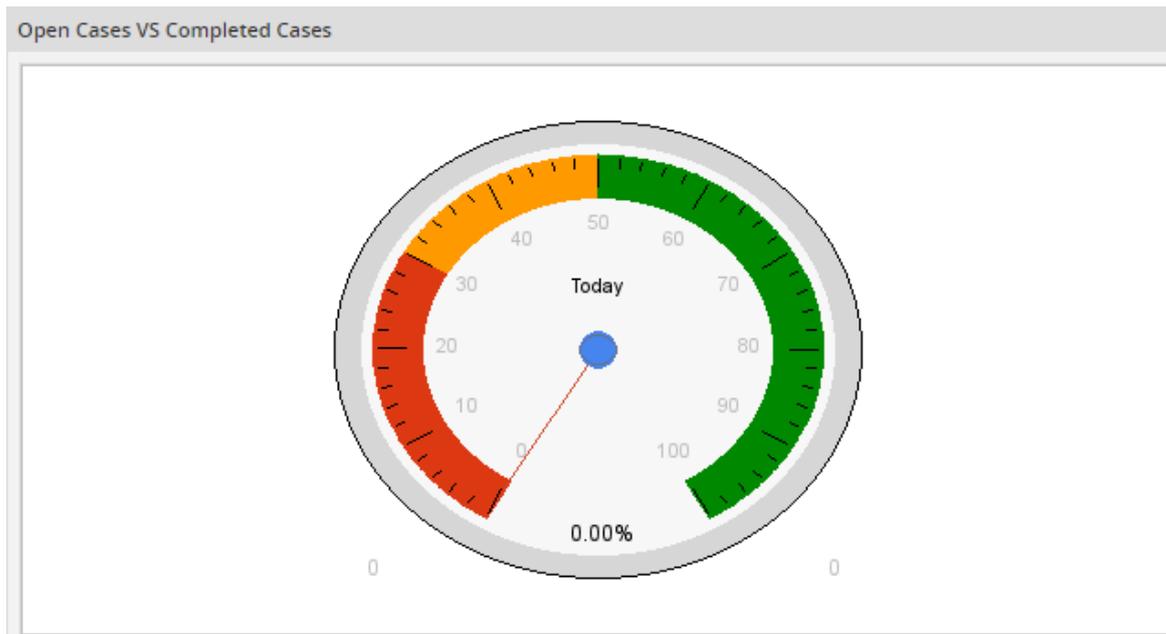


Ilustración 37: Dashboard básico integrado en processmaker

Fuente: Propia

La versión empresarial de processmaker proporciona dashboard mucho más funcionales y avanzados, mediante estos se obtiene reportes más específicos y personalizables.

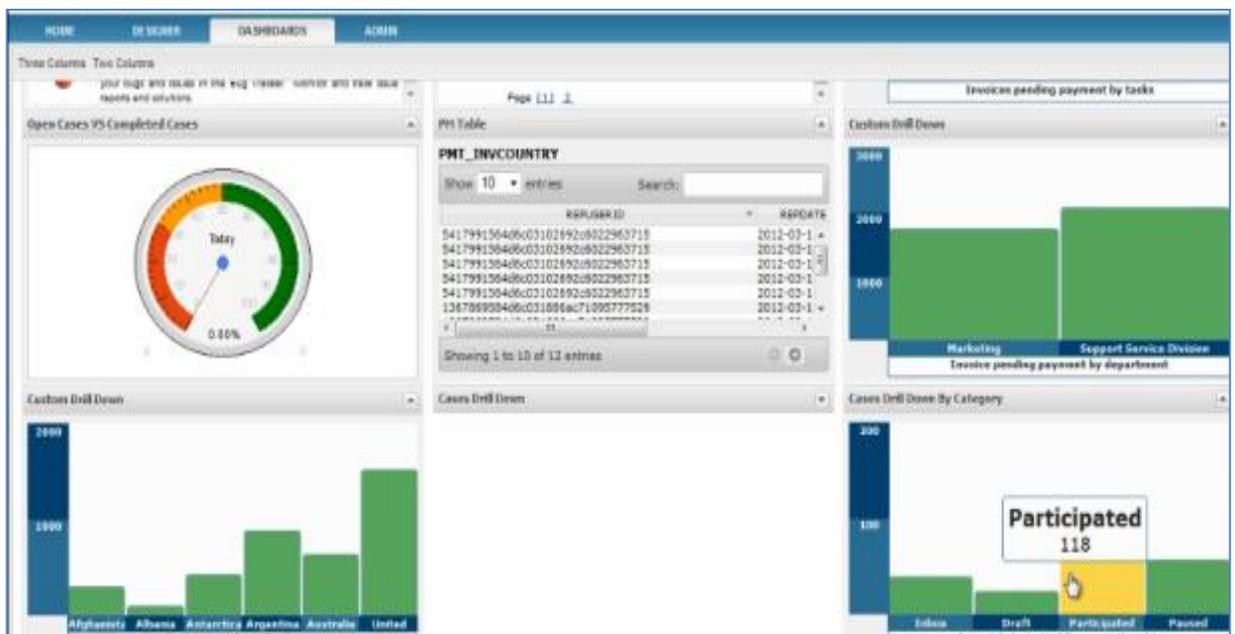


Ilustración 38: Advanced Dashboard

Fuente:(Colosa)

2.1.13.8 ADMINISTRACIÓN

En esta sección encontramos configuraciones para el funcionamiento de processmaker, administración de usuarios, entre otros.

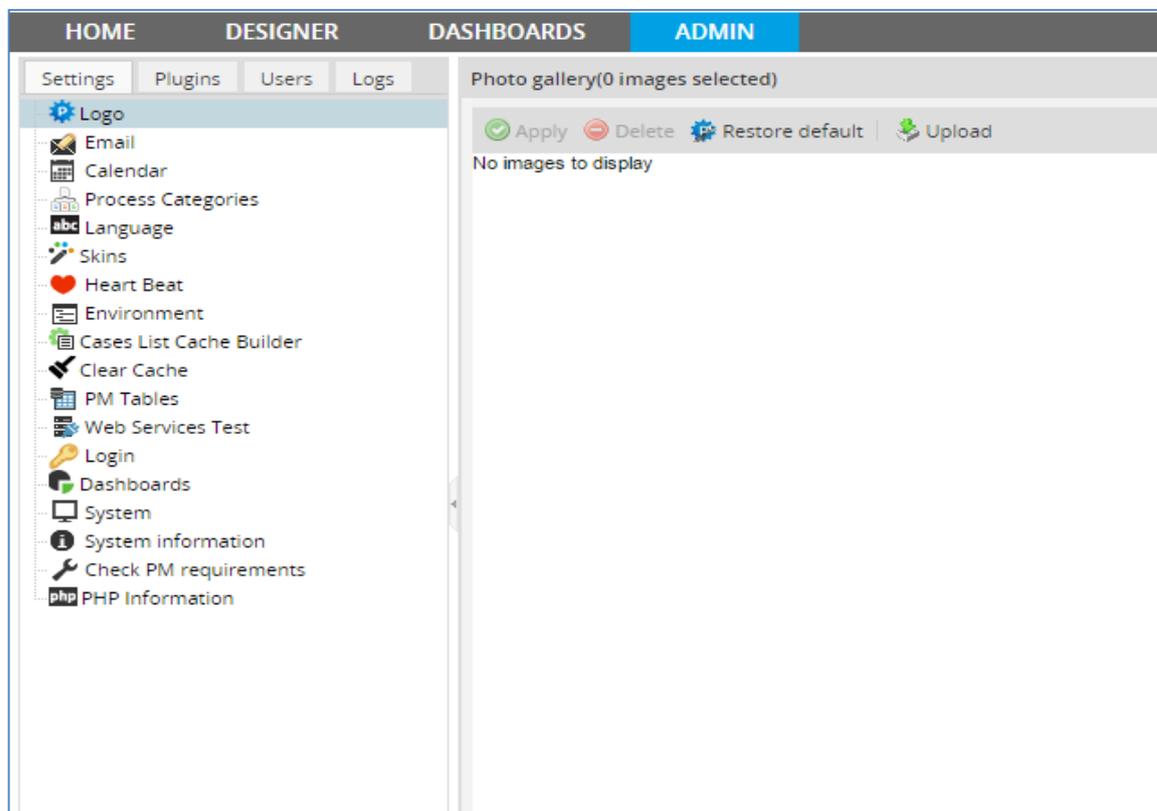


Ilustración 39: configuraciones de processmaker

Fuente: Propia

Entre las opciones más importantes podemos encontrar:

Mail: Configura las notificaciones mediante el servidor smtp

Lenguaje: Permite instalar y habilitar varios lenguajes, dependiendo en que región se use el programa

Web Services Test: Proporciona pruebas directas con los servicios web que posee.

Dashboard: Administra los tableros de comando y los personaliza

Plugins: Permite importar y habilitar funcionalidades extras mediante plugins, esta opción tienen acceso la versión empresarial.

Usuarios: Administra los usuarios que interactúan en el proceso, estos usuarios posteriormente se asignaran a una tarea.

Logs: Contiene la descripción sobre los eventos y notificaciones enviadas y por enviar, para que los eventos se completen es necesario ejecutar el archivo cron.php.

2.1.13.9 WEB SERVICES

La integración con otros sistemas es una de los objetivos de processmaker, esto se lo puede realizar a través de webservices (servicios web).

Los servicios web que proporciona processmaker son publicados mediante wsdl y soap 1.2; y la dirección web en la cual son publicados es la siguiente:

<http://localhost:port/sysworkflow/en/classic/services/wsdl2>

En donde localhost, es el dominio del servidor (puede ser también la ip de la pc) en donde se encuentra instalado processmaker y el puerto es del servidor en donde se encuentra instalado.

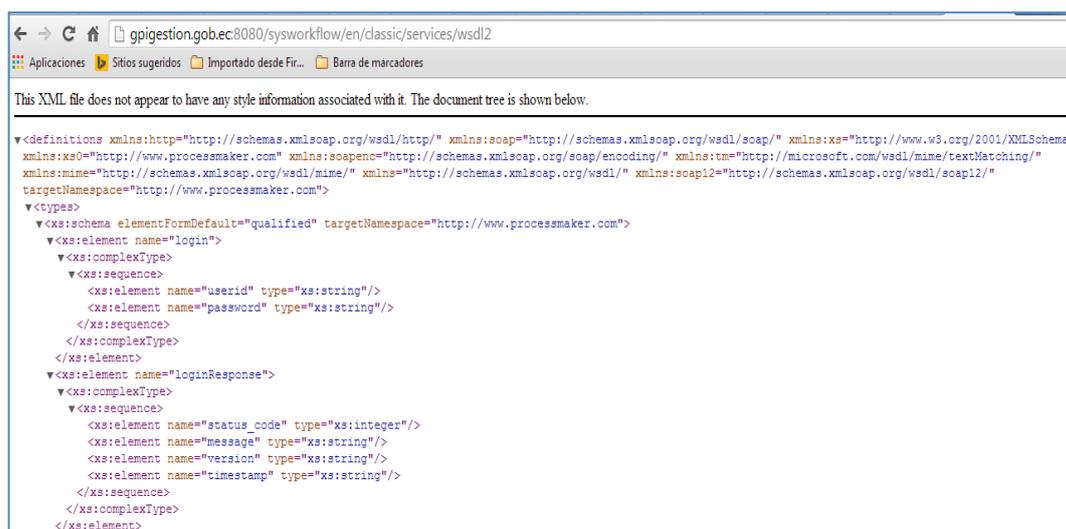


Ilustración 40: servicio web que proporciona processmaker

Fuente: Propia

Este servicio web que nos proporciona processmaker, contiene diversas operaciones que nos permiten interactuar con el proceso bpm desde un cliente soa, o desde un sistema externo.

Entre las operaciones más utilizadas son las siguientes:

- **Login:** permite iniciar session bpm, si el usuario y contraseña con correctos.

- **NewCase:** se crea un nuevo caso en el proceso bpm, recibe como parámetros el id de la session bpm, la tarea inicial y el proceso actual.
- **CaseList:** Retorna la lista de los casos asignados al usuario logeado, recibe como parámetro el id de la session bpm.
- **TaskList:** Retorna la lista de las tareas existentes en el flujo del proceso, recibe como parámetro el id de la session bpm, el id del caso.
- **CaseRoute:** Permite mover el estado del proceso, es decir si la tarea actual en el proceso es la número 1, esta operación asigna la siguiente tarea como actual, de acuerdo a las reglas de enrutamiento establecidas. Recibe como parámetro el id de la session bpm, el id del caso, y el valor delIndex que representa el estado del proceso en el flujo.

Para realizar un test de estas operaciones se debe realizar desde un software cliente soap.

Para un ejemplo usaremos el software SoapUi -5.0.0

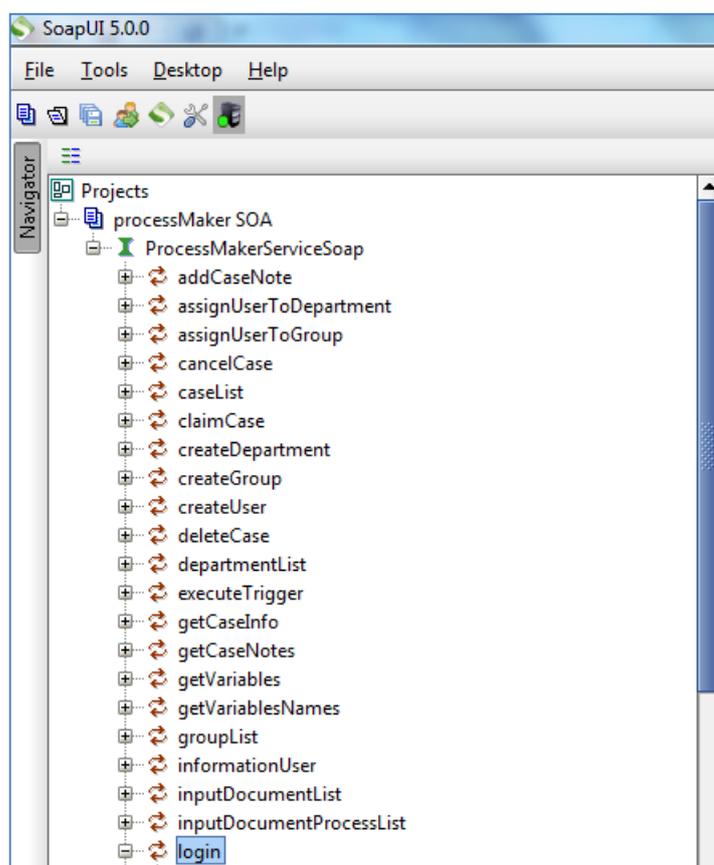


Ilustración 41: Operaciones del servicio web de processmaker

Fuente: Propia

Para obtener el request de una operación se ingresan los parámetros y se deployara e inmediatamente se desplegará una venta con el objeto Response, si los datos de los parámetros son incorrectos en el objeto response saldrá un mensaje informando el error del mismo.

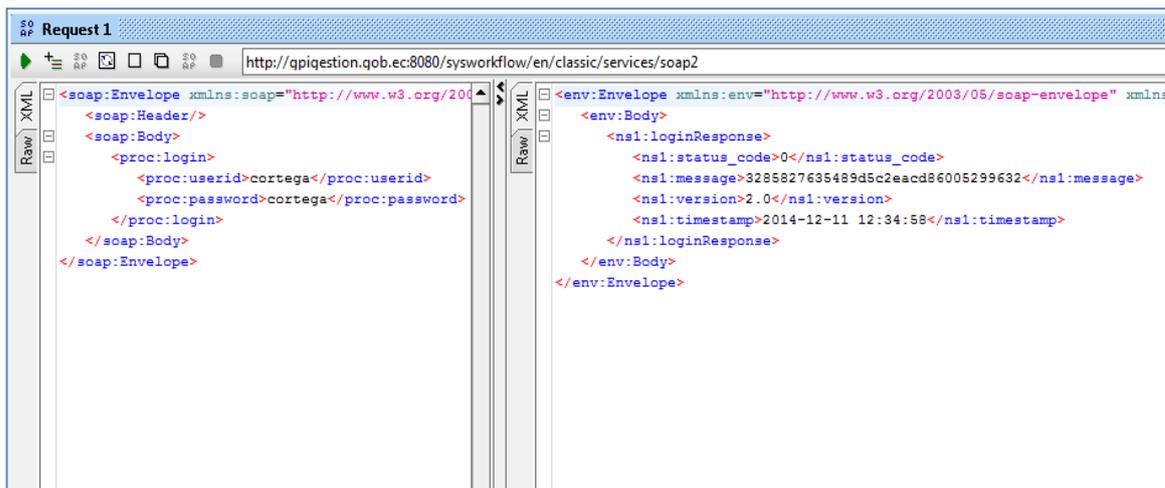


Ilustración 42: Ejemplo del response de una operación del servicio web

Fuente: Propia

2.2 METODOLOGÍA DE DESARROLLO

2.2.1 METODOLOGÍAS AGILES:

Principios del Manifiesto Ágil

(Calderon A, 2010)

- La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.

- Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- El software que funciona es la medida principal de progreso.
- Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- La simplicidad es esencial.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

Metodologías Ágiles vs Metodologías Tradicionales

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Especialmente preparadas para cambios durante el proyecto.	Cierta resistencia a los cambios.
Impuestas internamente (por el equipo).	Impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/normas.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.

Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos
Pocos roles	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

Tabla 1: Metodologías Ágiles y Metodologías Tradicionales

Fuente: (Calderon A, 2010)

(Calderon A, 2010)

Beneficios de las metodologías ágiles en el desarrollo de sistemas de software:

1. Simplificación de la sobrecarga de procesos
2. Calidad mejorada
3. Mejorar la previsibilidad a través de una mejor gestión del riesgo
4. Dar prioridad a los riesgos.
5. Evaluación de riesgos en paralelo.
6. Mejor perfil de productividad
7. Capacidad para aprovechar las inversiones realizadas

2.2.2. SCRUM

Definición:

(Management, 2010)

Scrum es, actualmente, uno de los métodos ágiles para desarrollo de software de mayor difusión en la industria, junto con Extreme Programming (XP). Su nombre proviene del rugby, deporte en el que un Scrum es una jugada que permite reiniciar el juego luego de una falta accidental. La elección del nombre busca rescatar el principio de trabajo en equipo que se observa en un Scrum de rugby: varios jugadores se toman de los hombros y se esfuerzan para lograr por sí solos y rápidamente un objetivo común, que consiste en adueñarse de la pelota y llevarla hacia adelante.

Características de un proceso Scrum

La metodología Scrum asume que el proceso de desarrollo de software es impredecible, y lo trata como a una “caja negra” controlada, en vez de manejarlo como un proceso completamente definido. Ésta es una de las principales diferencias entre Scrum y otras metodologías, como los modelos de espiral o de cascada, en los cuales el proceso de desarrollo se define por completo desde el inicio. Por tratar de planificar el proceso en forma completa desde el principio, las metodologías tradicionales fallan al toparse con algunos problemas habituales del desarrollo de software, como la falta de comprensión de los requerimientos al empezar el proceso, el cambio en los requerimientos durante el proceso, o la dificultad para prever los resultados del uso de nuevas herramientas y tecnologías.

Otra diferencia de Scrum con las metodologías tradicionales es que no trata el proceso de desarrollo de software como un proceso lineal, en el que se sigue la secuencia de análisis, diseño, codificación y testing.

En Scrum, el proyecto puede iniciarse con cualquier actividad, y cambiar de una a otra en cualquier momento.

Un proyecto administrado mediante Scrum se organiza en iteraciones, llamadas Sprints, que normalmente tienen entre dos y cuatro semanas de duración. Al principio de cada sprint se establece una lista de requerimientos llamada Backlog, que debe completarse cuando éste finalice. A diario se realizan breves reuniones del equipo de desarrollo, en las que se exponen los avances y los problemas encontrados, y se señalan posibles caminos para resolverlos (la resolución detallada de estos problemas no debe determinarse durante la reunión, para mantener su brevedad).

VENTAJAS

- Alineamiento entre cliente y equipo
- Gestión regular de las expectativas del cliente
- Resultados a corto plazo
- Retorno de inversión (ROI). (Se hace lo importante)
- Equipo motivado
- Flexibilidad y adaptación a los cambios.
- Calidad del producto final

Fases de Scrum:

(Hidalgo L, 2010)

- Planificación del sprint,
- Seguimiento del Sprint y
- Revisión del sprint

Planificación del sprint:

En esta fase se define el Product Backlog. Si todavía no ha sido definido, consiste en una lista priorizada de requisitos del sistema y es un documento vivo, que puede ser continuamente actualizado. En cada iteración el Product Backlog es revisado por el equipo. También se lleva a cabo la planificación del primer Sprint. La planificación de cualquier sprint es la jornada de trabajo previa al inicio de cualquier sprint y en la cual se determinan cuáles son los objetivos y el trabajo que se deben cubrir en esa iteración. En esta reunión se obtiene una lista de tareas que se denomina Sprint Backlog, y el lema u objetivo principal del sprint.

Seguimiento del Sprint:

A lo largo de esta fase se llevan a cabo breves reuniones diarias, para ver el avance de las tareas y el trabajo que está previsto para la jornada. En estas reuniones solo están presentes el Scrum Master y el equipo, las preguntas que se realizan suelen ser tres:

1. Qué trabajo se ha realizado desde la reunión anterior.
2. Qué trabajo que se va a hacer hasta la próxima reunión.
3. Qué impedimentos que deben solventarse para proseguir con el trabajo.

Revisión del sprint

Una vez finalizado el Sprint, se realiza un análisis y revisión del incremento generado. En esta reunión se presentan los resultados finales y se recomienda siempre tener preparada una demo. Existen múltiples razones para recomendar tener una demo al final de cada sprint, entre ellas la mejora del feedback con los interesados, reconocimiento del trabajo, un esfuerzo por finalizar las cosas o un correctivo en caso de tener una demo mal desarrollada.

Roles y Responsabilidades

(Guanotasig J, 2010)

Equipo de Desarrollo (Scrum Team):

Es el equipo del proyecto y tiene la autoridad para decidir en las acciones necesarias y para auto-organizarse con la finalidad de alcanzar los objetivos del sprint.

Propietario del Producto (Product Owner):

Es la única persona del proyecto conocedora del entorno de negocio del cliente y de la visión del producto y es el responsable de obtener el resultado de mayor valor posible para el cliente. También es el responsable de la financiación necesaria para el proyecto, de tomar las decisiones que afecten a cómo va a ser el resultado final, fechas de lanzamiento y el retorno de inversión. Por regla general y si no se trata de proyectos internos, el propietario del producto suele ser el responsable del proceso de adquisición del cliente.

El equipo está involucrado en la estimación del esfuerzo de las tareas del ProductBacklog, en la creación del sprint Backlog, etc.

Scrum Master (Líder de Proyecto):

Es el encargado de garantizar el funcionamiento de los procesos y de la metodología. Es importante darse cuenta que Scrum Master es más que un rol, es la responsabilidad de funcionamiento de modelo, por tanto muchas veces es aconsejable utilizar a personas y puestos más adecuados según la organización. Un Scrum master debe interactuar tanto con el equipo como con el cliente y con los gestores.

Documentos

(PF, 2012)

Product Backlog

El product backlog es un documento de alto nivel para todo el proyecto. Contiene descripciones genéricas de todos los requerimientos, funcionalidades deseables, etc. priorizadas según su retorno sobre la inversión (ROI). Es el qué va a ser construido. Es abierto y cualquiera puede modificarlo. Contiene estimaciones grosomodo, tanto del valor para el negocio, como del esfuerzo de desarrollo requerido.

Por ejemplo, si dos características tienen el mismo valor de negocio la que requiera menos tiempo de desarrollo tendrá probablemente más prioridad, debido a que su ROI será más alto.

Sprint Backlog

El sprint backlog es un documento detallado donde se describe el cómo el equipo va a implementar los requisitos durante el siguiente sprint. Las tareas se dividen en horas con ninguna tarea de dura superior a 16 horas. Si una tarea es mayor de 16 horas, deberá ser rota en mayor detalle. Las tareas en el sprint backlog nunca son asignadas, son tomadas por los miembros del equipo del modo que les parezca oportuno.

CAPÍTULO III

3 DESARROLLO DE LA METODOLOGÍA

3.1 PLANIFICACIÓN

3.1.1 ESPECIFICACIÓN DE REQUERIMIENTOS:

Para determinar los requerimientos se realizó un estudio y posteriormente un análisis del funcionamiento del proceso presupuestario en el gobierno provincial. Además se realizó un estudio comparativo de la manera que gestionan la información y los procesos en otras instituciones.

Esto sirvió para estructurar las historias de usuario del sistema GpiPresupuesto, detallando de manera global las necesidades a solucionar y las funcionalidades más importantes de la estructura del sistema.

3.1.2 HISTORIAS DE USUARIO

HISTORIA DE USUARIO
Numero: 1
Nombre: Gestión de Usuarios y Permisos
Usuario: Administrador
Iteración Asignada: 1
Prioridad del Negocio: Baja
Descripción: Debido a que la funcionalidad del sistema GpiPresupuesto se basa en el permiso y acceso que tenga el usuario que se encuentre logeado. Es por eso que la administración de usuarios y asignación de permisos es importantísima. Los usuarios que usaran el sistema serán los siguientes: <ul style="list-style-type: none">• Administrador• Usuario de Presupuesto• Usuario de Departamento• Director de Departamento
Observaciones: De los permisos que tenga cada usuario dependerá los menús y opciones que tenga en el sistema GpiPresupuesto

Tabla 2: Historia de usuario 1

Fuente: Propia

HISTORIA DE USUARIO
Numero: 2
Nombre: Gestión de Partidas Presupuestarias
Usuario: Usuario de Presupuesto
Iteración Asignada: 2
Prioridad del Negocio: Media
<p>Descripción:</p> <p>Las partidas presupuestarias representan el conjunto de campos, compuestos por códigos y descripciones que se utiliza para ordenar sistemáticamente la información presupuestaria de ingresos y gastos, normalmente es la unión o interrelación de los catálogos y clasificadores presupuestarios.</p> <p>Las partidas presupuestarias estarán clasificadas de manera jerárquica en base al CLASIFICADOR PRESUPUESTARIO DE INGRESOS Y GASTOS DEL SECTOR PÚBLICO generado por el Ministerio de Finanzas de Ecuador.</p> <p>Esta funcionalidad comprende la creación, edición, visualización, jerarquización y eliminación de una partida presupuestaria, así como la selección única una partida presupuestaria</p>
Observaciones:

Tabla 3: Historia de Usuario 2
Fuente: Propia

HISTORIA DE USUARIO
Numero: 3
Nombre: Gestión de Ejercicios Presupuestarios
Usuario: Usuario de Presupuesto
Iteración Asignada: 3
Prioridad del Negocio: Media
<p>Descripción:</p> <p>Un ejercicio presupuestario es un lapso de tiempo en el cual se realiza un proceso presupuestario.</p> <p>El ejercicio presupuestario deberá tener una fecha de inicio y una fecha de fin y estas no se deben repetir.</p> <p>Esta funcionalidad comprende la creación, edición, visualización y eliminación de un ejercicio presupuestario.</p> <p>Solamente deberá existir un ejercicio activo a la vez.</p>
<p>Observaciones:</p> <p>Si existen más de un ejercicio presupuestario, se encontrara uno activo solamente, los demás ejercicios estarán inactivos.</p> <p>Para validar que el ejercicio que se quiera activar es el correcto se realizara una comparación con la fecha actual del sistema</p>

Tabla 4: Historia de Usuario 3
Fuente: Propia

HISTORIA DE USUARIO
Numero: 4
Nombre: Gestión de Periodos Presupuestarios
Usuario: Usuario de Presupuesto
Iteración Asignada: 4
Prioridad del Negocio: Media
<p>Descripción:</p> <p>Un periodo presupuestario es un lapso de tiempo en el cual se realiza una fase del proceso presupuestario.</p> <p>Un periodo presupuestario se encuentra obligatoriamente dentro de un ejercicio presupuestario.</p> <p>Los periodos presupuestarios pueden ser:</p> <ul style="list-style-type: none"> • Planificación • Aprobación • Ejecución
<p>Observaciones:</p> <p>Solamente puede estar un periodo presupuestario activo a la vez.</p> <p>Solamente el usuario de presupuesto puede iniciar y terminar un periodo presupuestario.</p>

Tabla 5: Historia de Usuario 4

Fuente: Propia

HISTORIA DE USUARIO
Numero: 5
Nombre: Registro de Requerimientos
Usuario: Usuario de Departamento, Director de Departamento
Iteración Asignada: 5
Prioridad del Negocio: Alta
<p>Descripción:</p> <p>El registro de requerimientos se realizara en el periodo de planificación, durante este periodo los usuarios deben planificar y formular los ítems presupuestarios necesarios para la ejecución de un proyecto o actividad programática.</p> <p>Un requerimiento ingresado constara de:</p> <ul style="list-style-type: none"> • Fecha Transacción • Descripción • Ítem Presupuestario • Cantidad • Monto Unitario • Monto Total
Observaciones

Tabla 6: Historia de Usuario 5

Fuente: Propia

HISTORIA DE USUARIO
Numero: 6
Nombre: Aprobación de Requerimientos
Usuario: Director de Departamento
Iteración Asignada: 6
Prioridad del Negocio: Media
Descripción: Esta operación la realizara los directores de departamento durante el periodo de aprobación.
Observaciones: El director de departamento revisa los requerimientos ingresados por los usuarios y aprueba los que encuentran la información congruente pero si encuentra requerimientos incensarios o que sobrepasen el presupuesto asignado puede modificar o eliminar.

Tabla 7: Historia de Usuario 6

Fuente: Propia

HISTORIA DE USUARIO
Numero: 7
Nombre: Consolidación Presupuestaria
Usuario: Usuario de Presupuesto, Director de Departamento
Iteración Asignada: 7
Prioridad del Negocio: Alta
Descripción: La consolidación presupuestaria es el proceso de agrupar todos los requerimientos ingresados y aprobados correspondientes a un proyecto. Esta agrupación se la puede realizar mediante partida presupuestaria o ítem presupuestario.
Observaciones: Cabe señalar que cuando hayan sido consolidados todos los proyectos se procede a generar el Plan Anual de Compras y la Proforma Presupuestaria

Tabla 8: Historia de Usuario 9

Fuente: Propia

3.1.3 DEFINICIÓN DE ROLES:

Scrum usa diferentes tipos de roles, estos son definidos según su interacción con el sistema.

Product Owner: es la persona encargada de llevar a cabo la recolección de información para los requerimientos y la definición de las historias de usuario y de que el sistema se esté desarrollando conforme a las historias de usuario establecidas, es el representante del cliente.

Además interactúa con el equipo de desarrollo brindándoles apoyo y sugiriendo ideas constantemente.

Scrum Master: es la persona encargada de que el desarrollo del sistema avance conforme a las historias de usuario, es como el líder del equipo, pero en Scrum no existen líderes, solamente cumple las funciones de mayor responsabilidad que sus compañeros de equipo y les representa.

Equipo de desarrollo (Product Team): son los encargados de desarrollar el proyecto conforme a las historias de usuario, y requerimientos; estas personas debe trabajar en equipo y comunicándose constantemente; tienen un representante que es el Scrum Master.

Stakeholders: son las terceras personas que interactúan con el proyecto, es decir son los proveedores, clientes, etc.

En la arquitectura del sistema GpiPresupuesto son los usuarios finales que son los empleados de la institución.

NOMBRE	DESCRIPCION	RESPONSABILIDAD	ROL
GPI – Departamento de Presupuesto	Administra los datos del sistema y proporciona información presupuestaria	Gestiona los datos presupuestarios, importantes para el funcionamiento del proceso	Product Owner
Ing. Roberto López	Representante del equipo de desarrollo	Está pendiente del avance del proyecto y de las reuniones diarias, y del proceso Scrum	ScrumMaster
Ing. Marcelo Jingo	Desarrollador	Desarrolla el módulo de	Scrum Team

		proyectos	
Ing. Fernando Miño	Infraestructura	Estructura los servidores y los ambientes para la implementación del sistema	Scrum Team
Cornelio Orbe	Desarrollador	Desarrolla el módulo de Presupuesto	Scrum Team
Aníbal Benalcázar	Desarrollador	Desarrolla del módulo de Contratos	Scrum Team
Cristian IpiALES	Desarrollador	Desarrolla el módulo de Inventarios	Scrum Team
Ing. José Luis Rodríguez	Director de Tesis	Realiza un seguimiento del avance del aplicativo y de la documentación	Scrum Team

Tabla 9: Roles de las personas que intervienen en el sistema

Fuente: Propia

Usuarios:

Son los que interactúan con el producto terminado, es decir son los empleados de la institución que serán asignados a usar el sistema GpiPresupuesto.

NOMBRE	RESPONSABILIDAD	ROL
Usuario de Presupuesto	Administra los datos presupuestarios: Partidas presupuestarias Ejercicios periodos	Stakeholder
Usuario de Departamento	Realiza la planificación presupuestaria para su departamento, así como la consolidación del mismo	Stakeholder
Director de Departamento	Realiza el ingreso de requerimientos, y además aprueba los requerimientos realizados por los usuarios de su departamento, también puede consolidar los proyectos y generar reportes	Stakeholder

Tabla 10: Stakeholder o usuarios finales del sistema

3.1.4 PRODUCT BACKLOG

A continuación el artefacto más importante de la metodología, pila del producto o product backlog, se encuentra detallado en forma global o en forma macro las principales funcionalidades del sistema GpiPresupuesto

ID	Nombre (Requerimiento)	Importancia	Sprint
1	Instalación y Configuración de Herramientas	3	1
2	Diseño del Login	3	2
3	Diseño de la Interfaz	3	3
7	Gestión de Partidas Presupuestarias	5	7
9	Gestión de Ejercicios Presupuestarios	5	9
10	Gestión de Periodos Presupuestarios	5	10
11	Registro de Requerimientos	10	11
12	Aprobación de Requerimientos	10	12
13	Consolidación Presupuestaria	10	13

Tabla 11: Product Backlog

Fuente: Propia

3.2 DESARROLLO Y SEGUIMIENTO

A continuación se define la relación de los actores o usuario del sistema con el funcionamiento del sistema, estas relaciones se especifican en los casos de uso y se visualizaran mediante diagramas de casos de uso.

3.2.1 DIAGRAMA DE CASOS DE USO

Diagrama de Gestión Datos Presupuestarios

El actor principal es el usuario de Presupuesto, puede ser el Jefe de Presupuesto o el Secretario de Presupuesto; se encarga de gestionar los datos presupuestarios, que son los datos maestros sobre el cual funcionara el sistema.

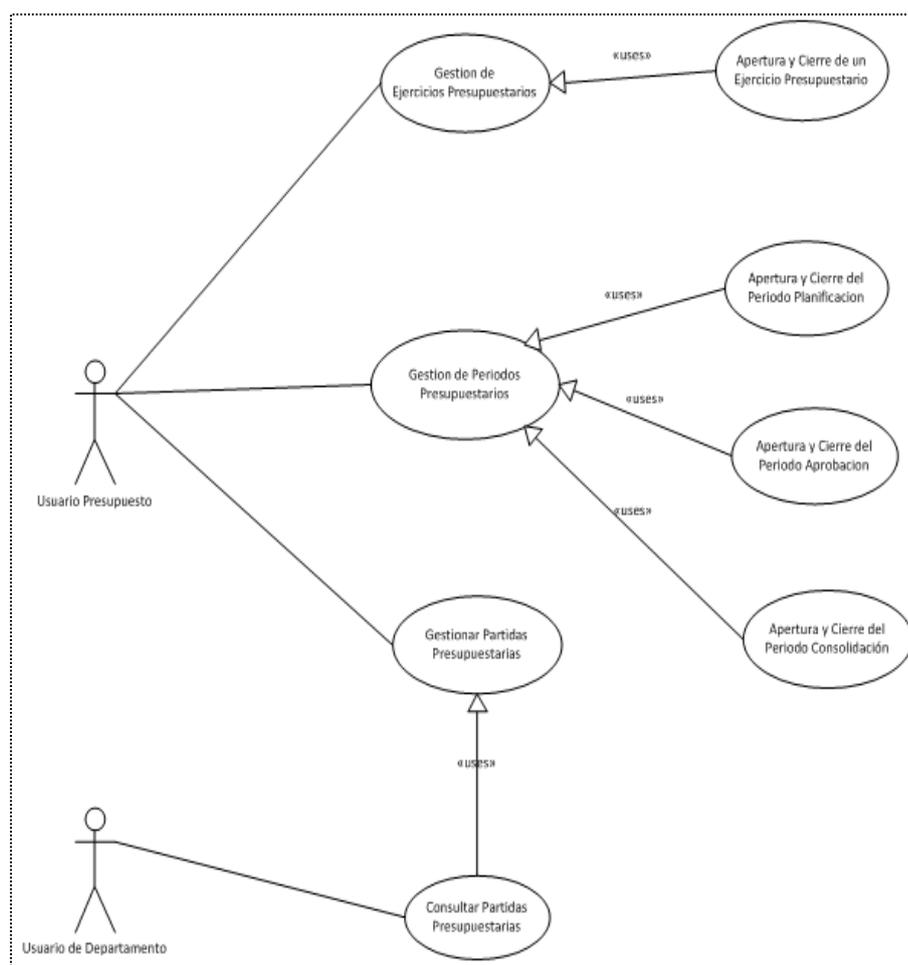


Ilustración 43: Diagrama de Casos de Uso – Gestión de Datos Presupuestarios

Fuente: Propia

Proceso Presupuestario

Diagrama de Planificación Presupuestaria

El actor principal es el usuario de Departamento, y es la persona que debe realizar el registro de requerimientos sobre los proyectos planificados, el director de departamento también tendrá acceso al registro de requerimientos, pero se ha puesto solamente el Usuario de departamento como referencia.

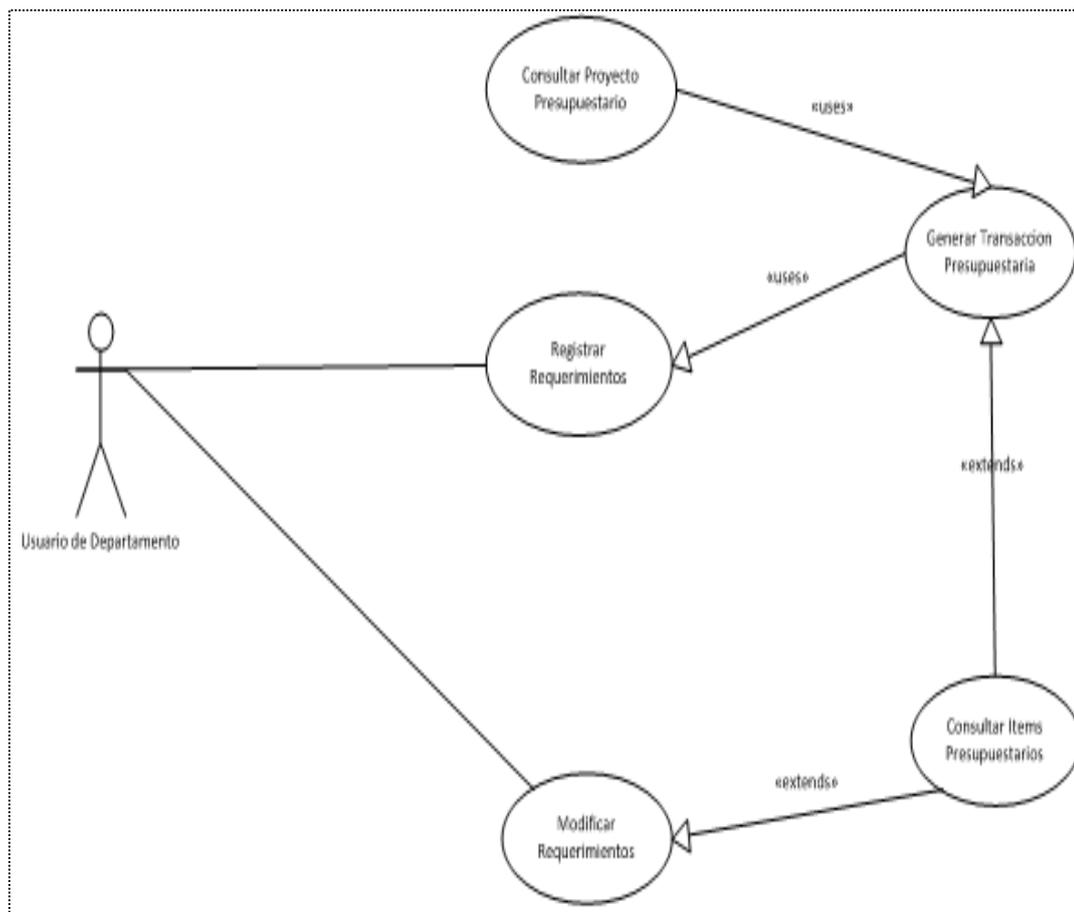


Ilustración 44: Diagrama de Casos de Uso – Planificación Presupuestaria

Fuente: Propia

Diagrama Aprobación Presupuestaria

El actor es el director de departamento, y es la persona encargada de realizar la revisión de los requerimientos registrados por los usuarios de su departamento.

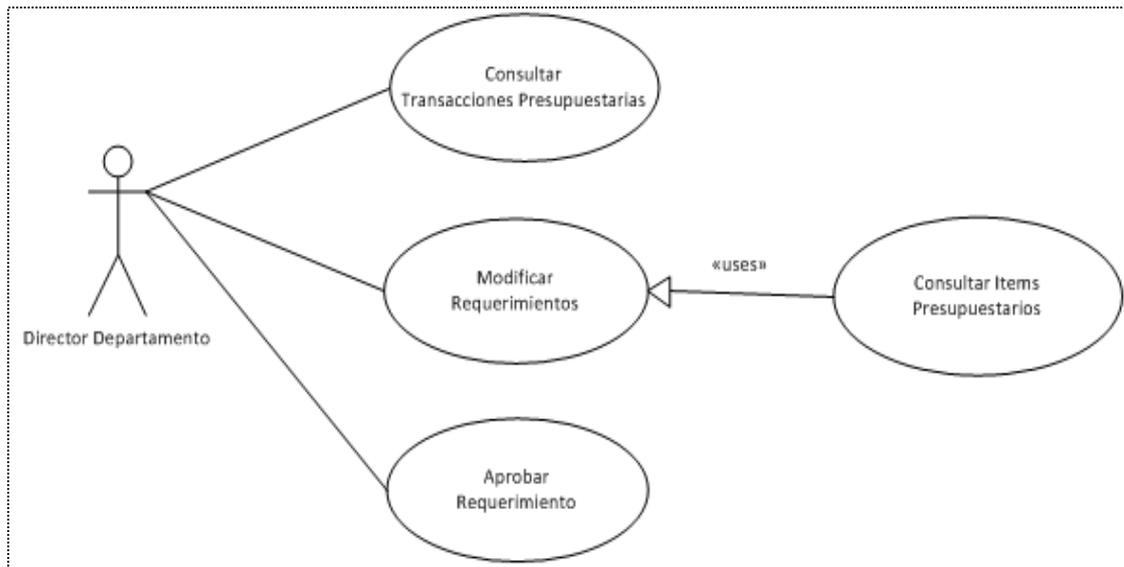


Ilustración 45: Diagrama de Casos de Uso - Aprobación Presupuestaria

Fuente: Propia

Diagrama de Consolidación Presupuestaria

Los actores son el director de departamento, el usuario de departamento, y el usuario de presupuesto; los tres usuarios interactúan con el sistema en el periodo de consolidación presupuestaria

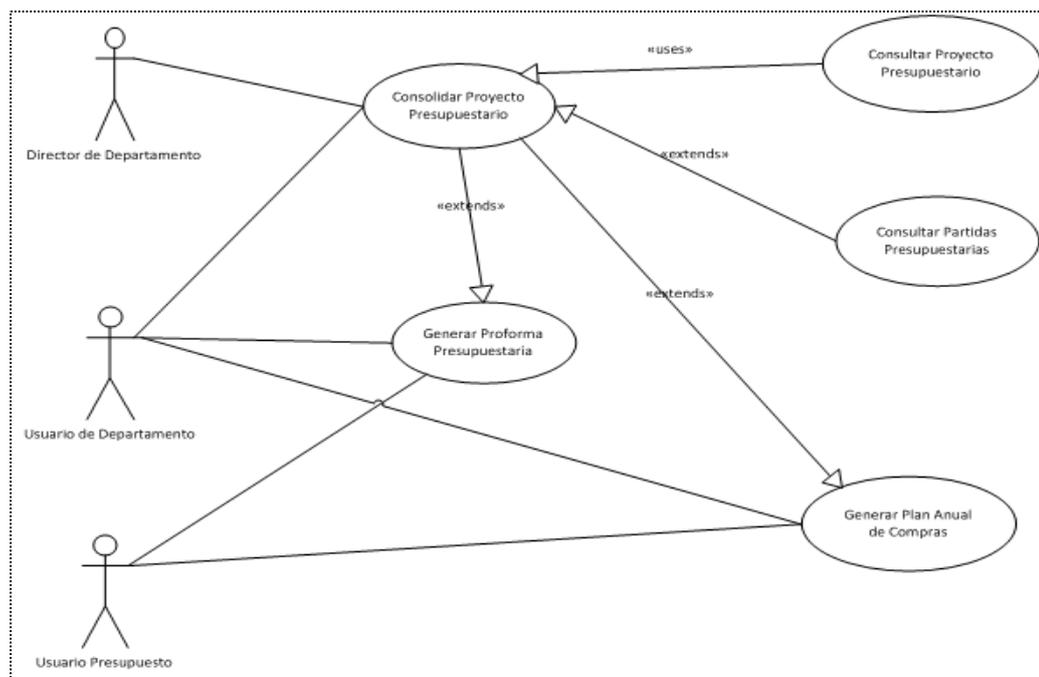


Ilustración 46: Diagrama de Casos de Uso – Consolidación Presupuestaria

Fuente: Propia

3.2.2 PECIFICACIONES DE CASOS DE USO

❖ GPIPE-CU-001

Nombre: Gestionar Partidas Presupuestarias

Actor: Usuario Presupuesto

Descripción:

Las partidas presupuestarias representan el conjunto de campos, compuestos por códigos y descripciones que se utiliza para ordenar sistemáticamente la información presupuestaria de ingresos y gastos, normalmente es la unión o interrelación de los catálogos y clasificadores presupuestarios.

Las partidas presupuestarias estarán clasificadas de manera jerárquica en base al CLASIFICADOR PRESUPUESTARIO DE INGRESOS Y GASTOS DEL SECTOR PÚBLICO generado por el Ministerio de Finanzas de Ecuador.

Se especificará el uso de un plan presupuestario que contendrá un grupo de partidas presupuestarias. En otras palabras un plan presupuestario es un catálogo de partidas presupuestarias.

Cada ítem presupuestario obligatoriamente deberá tener una partida presupuestaria asignada.

La gestión de partidas presupuestarias comprende:

- La creación de una partida presupuestaria
- La edición de una partida presupuestaria
- La visualización de una partida presupuestaria
- El registro de una partida presupuestaria

Para el ingreso de una nueva partida presupuestaria se especificará el código, el nombre, la descripción, y el padre de la partida presupuestaria en el caso que se cree una partida presupuestaria hija.

Si la partida presupuestaria raíz no deberá tener ninguna partida presupuestaria padre.

Ejemplos:

CODIGO	PARTIDA PRESUPUESTARIA
1.0	INGRESOS CORRIENTES
1.0	IMPUESTOS
1.1.1	Sobre la Renta, Utilidades y Ganancias de Capital
5.0	GASTOS CORRIENTES

Tabla 12: Partidas Presupuestarias

Fuente: Propia

Pre Condiciones:

No aplica

Post Condiciones:

- Consulta de partidas presupuestarias
- Asignación de partidas presupuestarias

❖ **GPIPRE-CU-002**

Nombre: Consultar Partidas Presupuestarias

Actor: Usuario Presupuesto

Descripción:

La consulta de partidas presupuestarias comprende solamente la visualización del catálogo de partidas presupuestarias.

Esta funcionalidad es llamada desde la gestión de ítems presupuestarios.

Pre Condiciones:

Deben existir partidas presupuestarias registradas.

Post Condiciones:

Asignación de partidas presupuestarias.

❖ **GPIPRE-CU-003**

Nombre: Asignación de Partidas Presupuestarias

Actor: Usuario Presupuesto

Descripción:

La asignación de partidas presupuestarias consiste en relacionar una partida presupuestaria a un ítem presupuestario.

Esta funcionalidad es llamada en la consolidación presupuestaria

Ejemplo:

CODIGO ITEM	NOMBRE ITEM	CODIGO PARTIDA
ITEM0040	Licencia de Software	6.3.07.01
ITEM0050	Impresora HP F300	6.3.07.01

Tabla 13: Ítem Presupuestario

Fuente: Propia

Nota: la partida presupuestaria 6.3.07.01 corresponder a *Desarrollo de Sistemas Informáticos* en el catálogo de partidas presupuestarias.

Pre Condiciones:

Deben existir partidas presupuestarias registradas

Post Condiciones:

No aplica

❖ **GPIPRE-CU-004**

Nombre: Consultar Ítems Presupuestarios

Actor: Usuario de Departamento

Descripción:

Un ítem presupuestario es todo aquello que sea susceptible de presupuestar y se someta a un comprometimiento y devengado.

En los ítems presupuestarios está comprendida la lista de requerimientos que el usuario del sistema solicite.

Esta funcionalidad es llamada desde el registro y la modificación de requerimientos.

Para la inserción de un requerimiento es necesario seleccionar uno o varios ítems presupuestarios mientras que para la modificación de un requerimiento se necesitaran la selección solamente de un ítem presupuestario.

Entre los Ítems presupuestarios se encuentran:

- Productos (CPC nivel 8)
- Rubros
- Cargos
- Servicios(CPC nivel 8)

El sistema GPI Presupuesto solamente se realizara la búsqueda de productos y servicios.

Pre Condiciones:

Deberán existir ítems presupuestarios registrados

Post Condiciones:

- Registro de requerimientos
- Modificación de requerimientos

❖ GIPRE-CU-005

Nombre: Gestión Ejercicios Presupuestarios

Actor: Usuario Presupuesto

Descripción:

Un ejercicio presupuestario es el espacio de tiempo que incluye la duración del proceso presupuestario, este comprende entre septiembre del año actual y diciembre del año siguiente.

El proceso presupuestario se divide en periodos que son descritos en el caso de uso GPIPE-CU-014; normalmente un ejercicio presupuestario toma el nombre de la fecha que inicia concatenado con la fecha final y solamente puede existir un ejercicio presupuestario a la vez.

Ejemplo:

1. Ejercicio 2012-2013
2. Ejercicio 2013-2014

PERIODO				PERIODO		PERIODO		PERIODO							
sep-12	oct-12	nov-12	dic-12	ene-13	feb-13	mar-13	abr-13	may-13	jun-13	jul-13	ago-13	sep-13	oct-13	nov-13	dic-13
EJERCICIO															

Tabla 14: Ejercicios y periodos presupuestarios

Fuente: Propia

La gestión de ejercicios presupuestarios comprende:

- Crear un ejercicio presupuestario
- Modificar un ejercicio presupuestario
- Eliminar un ejercicio presupuestario
- Guardar un ejercicio presupuestario
- Iniciar un ejercicio presupuestario
- Terminar un ejercicio presupuestario

Pre Condiciones:

No aplica

Post Condiciones:

- Iniciar un ejercicio presupuestario
- Terminar un ejercicio presupuestario
- Gestionar periodos presupuestarios

❖ **GIPRE-CU-006**

Nombre: Iniciar Ejercicio Presupuestario

Actor: Usuario Presupuesto

Descripción:

EL inicio o la apertura de un ejercicio presupuestario es el primer paso que es necesario en el proceso presupuestario, después de esta acción que lleva a cabo el usuario de presupuesto se realizan todas las demás subprocesos.

El estado del ejercicio presupuestario se determinara a través de un indicador que activo o inactivo.

Ejemplo:

EJERCICIO	ESTADO
Ejercicio 2012 - 2013	Inactivo
Ejercicio 2013 - 2014	Activo
Ejercicio 2014 - 2015	Inactivo

Tabla 15: Ejercicios Presupuestarios

Fuente: Propia

Pre Condiciones:

No aplica

Post Condiciones:

- Gestionar Periodos Presupuestarios
- Iniciar Periodo Proforma

❖ **GIPRE-CU-007**

Nombre: Gestionar Periodos Presupuestarios

Actor: Usuario Presupuesto

Descripción:

Un periodo presupuestario es un lapso de tiempo dentro de un ejercicio presupuestario.

Este periodo define las tareas a ejecutarse dentro del tiempo establecido.

Los periodos presupuestarios definidos en el sistema Gpi Presupuesto son:

- Proforma
- Aprobación
- Consolidación

Al igual que en los ejercicios presupuestarios, solamente puede haber un periodo activo a la vez.

PERIODO	ESTADO
Proforma	Inactivo
Aprobación	Activo
Ejecución	Inactivo

Un ejercicio presupuestario contiene varios periodos presupuestarios. La relación existente se puede apreciar en la siguiente ilustración:

PROFORMA				APROBACION		CONSOLIDACION		EJECUCION							
SEP	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEP	OCT	NOV	DIC
EJERCICIO															

Tabla 16: Relación periodos y ejercicio presupuestario

Fuente: Propia

La gestión de periodos presupuestarios comprende:

- Crear un periodo presupuestario
- Modificar un periodo presupuestario

- Eliminar un periodo presupuestario
- Guardar un periodo presupuestario
- Iniciar un periodo presupuestario
- Terminar un periodo presupuestario

Pre Condiciones:

Deben existir ejercicios presupuestarios registrados y un ejercicio presupuestario activo.

Post Condiciones:

Actividades afines al subproceso establecido en dicho periodo.

❖ **GPIPRE-CU-008**

Nombre: Iniciar Periodo Proforma

Actor: Usuario Presupuesto

Descripción:

Es la apertura del primer periodo dentro del ejercicio presupuestario, esta acción la lleva a cabo el usuario de presupuesto.

Después de haber sido iniciado este periodo los usuarios del sistema debe ingresar los requerimientos en los proyectos de sus departamentos.

Pre Condiciones:

Debe existir un ejercicio presupuestario activo

Deben existir periodos presupuestarios registrados

Post Condiciones:

Registro de Requerimientos

❖ **GIPRE-CU-009**

Nombre: Registrar Requerimientos

Actor: Usuario del Sistema

Descripción:

El registro de requerimientos es la acción que los usuarios del sistema incluirán en los proyectos planificados los ítems presupuestarios que crean convenientes para la ejecución del proyecto.

Esta acción se realizara durante el periodo de proforma, una vez que el usuario de presupuesto cierre o inactive el periodo de proforma los usuarios ya no podrán registrar los requerimientos.

Para el registro de uno o varios requerimientos se seleccionara primeramente un proyecto presupuestario, y luego se generara una transacción presupuestaria.

Pre Condiciones:

- Iniciar el periodo proforma
- Gestionar tipos de transacciones
- Gestionar ítems presupuestarios

Post Condiciones:

Modificar Requerimientos

❖ **GIPRE-CU-010**

Nombre: Generar Transacción Presupuestaria

Actor: Usuario de Departamento

Descripción:

Una transacción presupuestaria es un conjunto de acciones llevaran consigo le realización de una actividad de acuerdo al periodo activo.

Las transacciones presupuestarias serán usadas en los periodos de planificación.

La transacción presupuestaria consta de dos componentes que se interactúan en forma de maestro detalle.

La cabecera de la transacción presupuestaria consta de:

- Una fecha de transacción presupuestaria
- Una descripción que contendrá la razón de la transacción presupuestaria.
- Referencias al Proyecto o Actividad Programática.
- El estado de aprobación, en el caso que la transacción sea visualizada en el periodo de aprobación.
- El monto total presupuestado que es la suma de los montos de todos los ítems presupuestados en el detalle.

El detalle de la transacción presupuestaria consta de:

- Selección de Ítems Presupuestarios,
- El o los ítems presupuestarios seleccionados
- La cantidad de ítems presupuestarios
- El monto unitario a presupuestar en el ítem presupuestario seleccionado.
- El monto total que es la multiplicación de la cantidad con el monto unitario.

Pre Condiciones:

Deben existir proyectos y actividades programáticas registradas.

Deben existir productos registrados

Post Condiciones:

- Cierre del periodo planificación
- Modificación de Requerimientos

❖ **GIPRE-CU-011**

Nombre: Consultar Proyectos Presupuestarios

Actor: Usuario de Departamento

Descripción:

La información de los proyectos presupuestarios y actividades programáticas correspondientes a cada departamento proviene del Módulo Gpi Planificación y el Modulo Gpi Proyectos. Los proyectos son consultados son de acuerdo al departamento activo o asignado al usuario.

Pre Condiciones:

Deben existir proyectos presupuestarios y actividades programáticas registradas

Post Condiciones:

Generación de transacciones presupuestarias

❖ **GIPRE-CU-012**

Nombre: Modificar Requerimientos

Actor: Usuario de Departamento

Descripción:

La modificación de un requerimiento es la corrección y/o el cambio de la información registrada en un requerimiento o transacción presupuestaria.

Esta acción puede realizarla el usuario de Departamento en el periodo de planificación y el director del departamento en el periodo de aprobación.

La funcionalidad modificación de requerimientos comprende:

- Cambiar la fecha de transacción
- Cambiar el ítem presupuestado

- Eliminar el ítem presupuestado
- Cambiar la cantidad de ítems presupuestados
- Cambiar el monto del ítem presupuestado
- Ingresar un nuevo ítem presupuestario

Pre Condiciones:

El requerimiento debe estar registrado

Post Condiciones:

Aprobar el requerimiento (en el caso del periodo aprobación)

❖ **GPIPRE-CU-013**

Nombre: Apertura Periodo Aprobación

Actor: Usuario Presupuesto

Descripción:

El periodo de aprobación es el siguiente del periodo de proforma en el proceso presupuestario

Una vez que el usuario de presupuesto de por terminado el periodo de proforma se inicia el periodo de aprobación.

En este periodo los directores de departamento deberán realizar la aprobación de los requerimientos proformados.

Pre Condiciones:

El ejercicio presupuestario debe estar activo.

El periodo de aprobación debe ser registrado previamente.

Post Condiciones:

Aprobación de requerimientos presupuestarios.

❖ **GIPRE-CU-014**

Nombre: Consultar Transacciones Presupuestarias

Actor: Director de Departamento, Usuario de Departamento

Descripción:

La consulta de las transacciones presupuestarias consiste en visualizar las transacciones que han sido generadas por los usuarios de sistema en un periodo establecido.

En el caso de la aprobación presupuestaria el director consultara las transacciones generadas en el periodo de proforma con el fin de aprobar o modificar las que se encuentren mal estructuradas.

Pre Condiciones:

Deben existir transacciones presupuestarias registradas

Post Condiciones:

Aprobación de requerimientos en el caso que el periodo de aprobación se encuentre activo.

❖ **GIPRE-CU-015**

Nombre: Aprobar Requerimiento

Actor: Director de Departamento

Descripción:

Es la acción que realizan los directores de los departamentos durante el periodo de aprobación.

Para llevar a cabo este proceso se realiza la revisión de los requerimientos registrados en el periodo de planificación, si la información es congruente se procede a aprobar el requerimiento, caso contrario se modifica para luego aprobarlo o simplemente no se aprueba dicho requerimiento.

La aprobación de una transacción se maneja mediante un estado, así el estado de aprobación puede ser 0 o 1; o dicho de otra manera: activo o inactivo.

También podrá aprobar masivamente todas las transacciones, así como también desaprobar todas las transacciones.

Pre Condiciones:

El periodo de aprobación debe estar activo

Las transacciones presupuestarias deben estar correctamente registradas

Post Condiciones:

Cierre del periodo de aprobación

❖ **GPIPRE-CU-016**

Nombre: Apertura Periodo Consolidación

Actor: Usuario Presupuesto

Descripción:

La consolidación presupuestaria dentro del proceso presupuestario es el siguiente periodo después de la aprobación presupuestaria.

La consolidación presupuestaria se realizara a nivel de proyecto presupuestario y se trabajara solamente con las transacciones presupuestarias que han sido aprobadas.

Pre Condiciones:

El ejercicio presupuestario debe estar activo

El periodo de consolidación debe estar correctamente registrado.

Post Condiciones:

Consolidación de un proyecto presupuestario

❖ **GPIPRE-CU-017**

Nombre: Consolidar Proyecto Presupuestario

Actor: Director de Departamento, Usuario Presupuesto

Descripción:

La consolidación presupuestaria es el proceso de agrupar todas las transacciones presupuestarias generadas durante el periodo de proforma.

Esta agrupación puede ser mediante los ítems presupuestarios y mediante la partida presupuestaria.

En la siguiente tabla se encuentra las transacciones generadas en el proyecto X

Proyecto X

Transacción	Ítem	Partida
Transacción 1	Ítem 1	2.0.1
	Ítem 2	2.0.5
Transacción 2	Ítem 5	6.0.10
Transacción 3	Ítem 1	2.0.1
Transacción 4	Ítem 5	6.0.10
Transacción 5	Ítem 6	6.0.10

Tabla 17: Agrupación Ítem y Partida Presupuestaria

Fuente: Propia

Consolidación Mediante Ítem Presupuestario:

Proyecto X

Ítem	Partida
Ítem 1	2.0.1
Ítem2	2.0.5
Ítem 5	6.0.10
Ítem 6	6.0.10

Tabla 18: Consolidación mediante Ítem Presupuestario

Fuente: Propia

Consolidación Mediante Partida Presupuestaria:

Proyecto X

Partida Presupuestaria
2.0.1
2.0.5
6.0.10

Tabla 19: Consolidación mediante Partida Presupuestaria

Fuente: Propia

Pre Condiciones:

El periodo de consolidación debe estar activo

Post Condiciones:

Generación de la proforma presupuestaria

Cierre del periodo de consolidación

❖ **GPIPRE-CU-018**

Nombre: Generar Proforma Presupuestaria

Actor: Director de Departamento, Usuario Presupuesto

Descripción:

La proforma presupuestaria es la compilación de todos los proyectos presupuestarios y actividades programáticas correspondientes a un departamento, en los cuales se detallan los requerimientos consolidados a nivel partida presupuestarias.

La proforma presupuestaria se muestra mediante un reporte y puede ser exportado en Excel o pdf.

Pre Condiciones:

Todos los proyectos deben estar consolidados

El periodo de consolidación debe estar activo

Post Condiciones:

Cierre del periodo de consolidación

❖ **GPIPRE-CU-019**

Nombre: Generar Plan Anual de Compras

Actor: Director de Departamento, Usuario Presupuesto

Descripción:

La proforma presupuestaria es la compilación de todos los proyectos presupuestarios y actividades programáticas correspondientes a un departamento, en los cuales se detallan los requerimientos consolidados a nivel ítem presupuestario.

La proforma presupuestaria se muestra mediante un reporte y puede ser exportado en Excel o pdf.

Pre Condiciones:

Todos los proyectos deben estar consolidados

El periodo de consolidación debe estar activo

Post Condiciones:

Cierre del periodo de consolidación.

3.2.3 DESARROLLO DE LOS SPRINT

Sprint 1: Instalación y Configuración de Herramientas

Se estructura el ambiente de desarrollo para el sistema, las instalaciones y configuraciones de las herramientas.

También se estructurara los Proyectos de desarrollo de acuerdo con la estructura propuesta.

Sprint	Duración(días)			
1	10			
ID	Tarea	Estimación(días)	Estado	Responsable
1	Instalación y configuración del Servidor Apache	0.5	Terminado	Cornelio Orbe
2	Instalación y Configuración de Zend Framework	0.5	Terminado	Cornelio Orbe
3	Instalación y Configuración de PHP	0.5	Terminado	Cornelio Orbe
4	Instalación de Flash Builder for PHP	1	Terminado	Cornelio Orbe
5	Integración de Flash Builder con Zend Framework	1	Terminado	Cornelio Orbe
6	Integración de Flash Builder con Apache Server	1	Terminado	Cornelio Orbe
7	Creación y Configuración del Proyecto GpiGestionBackend	1	Terminado	Cornelio Orbe
9	Creación del Proyecto GpiGestionFrontend	1	Terminado	Cornelio Orbe
10	Creación del Proyecto GpiComponentes	1	Terminado	Cornelio Orbe
11	Creación del Proyecto GpiSistema	1	Terminado	Cornelio Orbe
12	Creación del Proyecto GpiPresupuesto	1	Terminado	Cornelio Orbe

Tabla 20: Sprint 1 - Instalación y Configuración de Herramientas

Fuente: Propia

Sprint 2: Diseño del Login

Se establecerá la asignación de permisos a los usuarios, los usuarios tendrán diferentes privilegios del sistema, tanto transaccionales como de menús y reportes.

El objetivo es que en el ingreso al sistema el usuario digite el nombre de usuario y su contraseña y dinámicamente visualice menús y opciones que ofrece el sistema según su perfil asignado.

Sprint	Duración(días)			
2	20			
ID	Tarea	Estimación(días)	Estado	Responsable
1	Creación entidades de Usuarios	2	Terminado	Cornelio Orbe
2	Creación entidades de Permisos	2	Terminado	Cornelio Orbe
3	Relacionar Usuarios y Permisos	3	Terminado	Cornelio Orbe
4	Mapeo de las entidades mediante Zend Framework	1	Terminado	Cornelio Orbe
5	Creación de las entidades y servicios PHP	0.5	Terminado	Cornelio Orbe
6	Creación de los servicios y valueObjects en flex	0.5	Terminado	Cornelio Orbe
7	Diseño de la parte visual del Login	4	Terminado	Cornelio Orbe
8	Validación del usuario y Contraseña ingresados	5	Terminado	Cornelio Orbe
9	Relación del perfil de usuario con las opciones del menú	2	Terminado	Cornelio Orbe

Tabla 21: Sprint 2 – Diseño del Login

Fuente: Propia

Sprint 3: Diseño de la Interfaz

La interfaz es el espacio de trabajo sobre el cual se desarrollara la parte gráfica del sistema, se realizara mediante un Layout.

Que contendrá un menú de opciones, una barra para la funcionalidad Crud, y búsqueda; un banner, y el espacio de trabajo sobre la cual trabajaran los módulos.

Sprint	Inicio	Fin	Duración(días)			
3			46			
ID	Tarea	Estimación(días)	Estado	Responsable		
1	Creación entidades opciones menús	1	Terminado	Cornelio Orbe		
2	Creación entidades módulos	1	Terminado	Cornelio Orbe		
3	Relacionar opciones menús con módulos	1	Terminado	Cornelio Orbe		
4	Mapeo de las entidades mediante Zend Framework	1	Terminado	Cornelio Orbe		
5	Creación de las entidades y servicios PHP	2	Terminado	Cornelio Orbe		
6	Creación de los servicios y value objects en flex	2	Terminado	Cornelio Orbe		
7	Diseño del Layout del sistema	5	Terminado	Cornelio Orbe		
8	Diseño e implementación del espacio de trabajo	3	Terminado	Cornelio Orbe		
9	Diseño e implementación de las opciones de menús	5	Terminado	Cornelio Orbe		
10	Diseño e implementación de la barra Crud	5	Terminado	Cornelio Orbe		
11	Diseño e implementación de la opción Buscar Genérica	10	Terminado	Cornelio Orbe		
12	Diseño e implementación de la Arquitectura Modular del Sistema	10	Terminado	Cornelio Orbe		

Tabla 22: Sprint 3 – Diseño de la Interfaz

Fuente: Propia

Sprint 4: Gestión de Partidas Presupuestarias

La gestión de partidas presupuestarias se realizara únicamente por el usuario de presupuesto, se ingresara el catálogo de partidas presupuestarias en forma jerárquica en base al clasificador presupuestario actual.

Sprint	Inicio	Fin	Duración(días)			
4			34			
ID	Tarea	Estimación(días)	Estado	Responsable		
1	Creación de entidades de partida presupuestaria	2	Terminado	Cornelio Orbe		
2	Creación de entidades de partida presupuestaria relación	1	Terminado	Cornelio Orbe		
3	Relacionar partida presupuestaria con partida presupuestaria relación	1	Terminado	Cornelio Orbe		
4	Mapeo de las entidades mediante Zend Framework	1	Terminado	Cornelio Orbe		
5	Creación de las entidades y servicios PHP	1	Terminado	Cornelio Orbe		
6	Análisis, diseño e Implementación de la estructura jerárquica del catálogo de partidas presupuestarias en PHP	3	Terminado	Cornelio Orbe		
7	Creación de los servicios y value objects en flex	2	Terminado	Cornelio Orbe		
8	Análisis, diseño e Implementación de la estructura jerárquica del catálogo de partidas presupuestarias en Flex	5	Terminado	Cornelio Orbe		
9	Creación de un componente árbol para la visualización de la estructura jerárquica de las partidas presupuestarias	5	Terminado	Cornelio Orbe		
10	Implementar funcionalidad de inserción de una partida presupuestaria raíz	3	Terminado	Cornelio Orbe		

11	Implementar funcionalidad de inserción de una partida presupuestaria hija	3	Terminado	Cornelio Orbe
12	Implementar funcionalidad de edición de una partida presupuestaria	1	Terminado	Cornelio Orbe
13	Implementar funcionalidad de eliminación de una partida presupuestaria	2	Terminado	Cornelio Orbe
14	Implementar funcionalidad de registro de una partida presupuestaria	3	Terminado	Cornelio Orbe
15	Creación de Componente Árbol de Selección de Partidas Presupuestarias	1	Terminado	Cornelio Orbe
16	Creación del Módulo de Partidas Presupuestarias	0.5	Terminado	Cornelio Orbe
17	Relacionar el Modulo de Partidas Presupuestarias con el menú Partidas Presupuestarias	0.5	Terminado	Cornelio Orbe

Tabla 23: Sprint 4 – Gestión de Partidas Presupuestarias

Fuente: Propia

Sprint 5: Gestión de Ejercicios Presupuestarios

La gestión de los ejercicios presupuestarios se lo realizara en base a la planificación y estructuración del ejercicio presupuestario, solamente el usuario de presupuesto podrá realizar esta administración.

Sprint	Inicio	Fin	Duración			
5			20			
ID	Tarea			Estimación	Estado	Responsable
1	Creación de entidades del ejercicio presupuestario			2	Terminado	Cornelio Orbe

2	Mapeo de las entidades mediante Zend Framework	1	Terminado	Cornelio Orbe
3	Creación de las entidades y servicios PHP	1	Terminado	Cornelio Orbe
4	Creación de los servicios y value objects en flex	1	Terminado	Cornelio Orbe
5	Creación del componente grid de visualización de ejercicios presupuestarios	1	Terminado	Cornelio Orbe
6	Implementar funcionalidad de inserción de un ejercicio presupuestario	1	Terminado	Cornelio Orbe
7	Implementar funcionalidad de edición de un ejercicio presupuestario	1	Terminado	Cornelio Orbe
8	Implementar funcionalidad de eliminación de una ejercicio presupuestario	1	Terminado	Cornelio Orbe
9	Implementar funcionalidad de registrar de una ítem presupuestario	1	Terminado	Cornelio Orbe
10	Implementar funcionalidad de iniciar un ejercicio presupuestario	3	Terminado	Cornelio Orbe
11	Implementar funcionalidad de terminar un ejercicio presupuestario	3	Terminado	Cornelio Orbe
12	Creación del módulo de gestión de ejercicios presupuestarios	1	Terminado	Cornelio Orbe
13	Relacionar el módulo de gestión de ejercicios presupuestarios con el menú Ejercicios Presupuestarios	1	Terminado	Cornelio Orbe
14	Creación del módulo de activación de ejercicios presupuestarios	1	Terminado	Cornelio Orbe
15	Relacionar el módulo de activación de ejercicios presupuestarios con el menú Activar Ejercicio Presupuestario	1	Terminado	Cornelio Orbe

Tabla 24: Sprint 5 – Gestión de Ejercicios Presupuestarios

Fuente: Propia

Sprint 6: Gestión de Periodos Presupuestarios

Se asignaran periodos presupuestarios al ejercicio presupuestario activo, y se gestionara el inicio y fin de los periodos del ejercicio

Sprint	Inicio	Fin	Duración(días)			
6			20			
ID	Tarea			Estimación(días)	Estado	Responsable
1	Creación de entidades del periodo presupuestario			1	Terminado	Cornelio Orbe
2	Relacionar el periodo presupuestario con el ejercicio presupuestario			1	Terminado	Cornelio Orbe
3	Creación de las entidades y servicios PHP			1	Terminado	Cornelio Orbe
4	Creación de los servicios y value objects en flex			1	Terminado	Cornelio Orbe
5	Creación del componente grid de visualización de periodos presupuestarios			1	Terminado	Cornelio Orbe
6	Implementar funcionalidad de inserción de un periodo presupuestario			1	Terminado	Cornelio Orbe
7	Implementar funcionalidad de edición de un periodo presupuestario			1	Terminado	Cornelio Orbe
8	Implementar funcionalidad de eliminación de un periodo presupuestario			1	Terminado	Cornelio Orbe
9	Implementar funcionalidad de registro de un periodo presupuestario			1	Terminado	Cornelio Orbe
10	Creación del componente grid de activación de periodos presupuestarios			1	Terminado	Cornelio Orbe
11	Implementar funcionalidad de activación de periodo presupuestario			3	Terminado	Cornelio Orbe
12	Implementar funcionalidad de inactivación de un periodo presupuestario			3	Terminado	Cornelio Orbe
13	Creación del módulo de gestión de periodos presupuestarios			1	Terminado	Cornelio Orbe

14	Relacionar el módulo de gestión de periodos presupuestarios con el menú Periodos Presupuestarios	1	Terminado	Cornelio Orbe
15	Creación del módulo de activación de periodos presupuestarios	1	Terminado	Cornelio Orbe
16	Relacionar el módulo de activación de periodos presupuestarios con el menú Activación de Periodos Presupuestarios	1	Terminado	Cornelio Orbe

Tabla 25: Sprint 6 – Gestión de Periodos Presupuestarios

Fuente: Propia

Sprint 7: Registro de Requerimientos

El registro de requerimientos se lo realizara a través de una transacción presupuestaria, constara de dos componentes, en maestro detalle, y realizara una referencia a la búsqueda de productos que gestiona el sistema Gpi Inventarios.

Esta actividad solamente se podrá realizar en el periodo de planificación presupuestaria, es cuando los usuarios de los departamentos tendrán que presupuestar los requerimientos necesarios para llevar a cabo la ejecución de los proyectos planificados en su departamento.

Sprint	Inicio	Fin	Duración(días)			
7			30			
ID	Tarea	Estimación(días)	Estado	Responsable		
1	Creación de entidades de la transacción presupuestaria	2	Terminado	Cornelio Orbe		
3	Creación de entidades del detalle de transacción presupuestaria	2	Terminado	Cornelio Orbe		
5	Relacionar la transacción presupuestaria con el detalle de transacción presupuestaria	2	Terminado	Cornelio Orbe		

6	Mapeo de las entidades mediante Zend Framework	1	Terminado	Cornelio Orbe
7	Creación de las entidades y servicios PHP	1	Terminado	Cornelio Orbe
8	Creación de los servicios y value objects en flex	1	Terminado	Cornelio Orbe
9	Creación del componente formulario de transacción presupuestaria	5	Terminado	Cornelio Orbe
10	Implementar funcionalidad de relación con el proyecto presupuestario	2	Terminado	Cornelio Orbe
11	Implementar funcionalidad de asignación de fecha de registro	1	Terminado	Cornelio Orbe
13	Creación del componente detalle de transacción presupuestaria	15	Terminado	Cornelio Orbe
14	Implementar funcionalidad de búsqueda de ítems presupuestarios	10	Terminado	Cornelio Orbe
15	Implementar funcionalidad de asignación de ítems presupuestarios	3	Terminado	Cornelio Orbe
16	Implementar funcionalidad de modificar un ítem presupuestario	1	Terminado	Cornelio Orbe
17	Implementar funcionalidad de eliminación de un ítem presupuestario	1	Terminado	Cornelio Orbe
18	Implementar funcionalidad de asignación de cantidad de ítems presupuestarios	1	Terminado	Cornelio Orbe
19	Implementar funcionalidad de asignación del monto a los ítems presupuestarios	1	Terminado	Cornelio Orbe
20	Creación del componente maestro detalle de transacción presupuestaria	1	Terminado	Cornelio Orbe
21	Implementar funcionalidad del registro de una transacción presupuestaria	1	Terminado	Cornelio Orbe
22	Implementar funcionalidad de la edición de una transacción presupuestaria	1	Terminado	Cornelio Orbe

23	Implementar funcionalidad de la eliminación de una transacción presupuestaria	1	Terminado	Cornelio Orbe
24	Implementar funcionalidad del registro de una transacción presupuestaria	1	Terminado	Cornelio Orbe
25	Implementar funcionalidad del cálculo del monto de una transacción presupuestaria	3	Terminado	Cornelio Orbe
26	Implementar funcionalidad del ingreso de un detalle de transacción presupuestaria	5	Terminado	Cornelio Orbe
27	Implementar funcionalidad de la edición de un detalle de transacción presupuestaria	3	Terminado	Cornelio Orbe
28	Implementar funcionalidad de la eliminación de un detalle de transacción presupuestaria	3	Terminado	Cornelio Orbe
30	Creación del módulo del componente maestro detalle de la transacción presupuestaria	1	Terminado	Cornelio Orbe
31	Relación del módulo de transacción presupuestaria con el menú Transacción presupuestaria	1	Terminado	Cornelio Orbe
32	Relación del proyecto presupuestario con el componente maestro detalle de transacción presupuestaria	1	Terminado	Cornelio Orbe
33	Creación del componente grid de visualización de transacciones presupuestarias	1	Terminado	Cornelio Orbe
34	Relación del proyecto presupuestario con el componente grid de transacciones presupuestarias	1	Terminado	Cornelio Orbe

Tabla 26: Sprint 7 – Registro de Requerimientos

Fuente: Propia

Sprint8: Aprobación de Requerimientos

Para la aprobación presupuestaria se determinara cuales requerimientos registrados están correctamente estructurados, si están mal, el director de departamento los modificara, o simplemente no los aprueba, si están bien, les aprueba para proceder a la consolidación, para esta funcionalidad se reutilizara el componente de la transacción presupuestaria maestro detalle.

Sprint	Inicio	Fin	Duración(días)			
8			21			
ID	Tarea	Estimación(días)	Estado	Responsable		
1	Reutilización del componente transacción presupuestaria grid para la aprobación presupuestaria	5	Terminado	Cornelio Orbe		
2	Implementar funcionalidad de aprobación y desaprobación individual de transacciones presupuestarias	3	Terminado	Cornelio Orbe		
3	Implementar funcionalidad de aprobación y desaprobación masiva de transacciones presupuestarias	3	Terminado	Cornelio Orbe		
4	Reutilización del componente maestro detalle de transacción presupuestaria para la aprobación presupuestaria	3	Terminado	Cornelio Orbe		
5	Implementar funcionalidad de aprobación y desaprobación en la transacción presupuestaria	2	Terminado	Cornelio Orbe		
6	Creación de componente de reporte de proyecto presupuestario con las transacciones presupuestarias aprobadas y no aprobadas	3	Terminado	Cornelio Orbe		
7	Relación del componente de proyectos presupuestarios con el componente de transacciones presupuestarias grid	1	Terminado	Cornelio Orbe		
8	Relación del componente de reporte de proyectos presupuestarios aprobados con el menú Aprobación Presupuestaria	1	Terminado	Cornelio Orbe		

Tabla 27: Sprint 8 – Aprobación de Requerimientos

Fuente: Propia

Sprint 9: Consolidación Presupuestaria

La consolidación presupuestaria se realizara a nivel de proyecto presupuestario, cada departamento es el responsable de realizar esta actividad en el periodo que le corresponde, cuando todos los proyectos de un departamento hayan sido consolidados se podrá generar el Plan Anual de Compras y la Proforma Presupuestaria.

Sprint	Inicio	Fin	Duración			
8			40			
ID	Tarea	Estimación	Estado	Responsable		
1	Creación de entidades requerimiento agrupación partida	2	Terminado	Cornelio Orbe		
2	Creación de entidades requerimiento agrupación ítem	2	Terminado	Cornelio Orbe		
3	Relacionar el requerimiento agrupación partida con el requerimiento agrupación ítem	1	Terminado	Cornelio Orbe		
4	Creación de las entidades y servicios PHP	1	Terminado	Cornelio Orbe		
5	Creación de los servicios y value objects en flex	1	Terminado	Cornelio Orbe		
6	Creación del componente grid de visualización de proyectos presupuestarios consolidados	1	Terminado	Cornelio Orbe		
7	Implementar funcionalidad de consolidación de proyecto presupuestarios	1	Terminado	Cornelio Orbe		
8	Implementar funcionalidad de agrupación de ítems presupuestarios de un proyecto	3	Terminado	Cornelio Orbe		
9	Implementar funcionalidad de asignar partida presupuestaria por ítem presupuestario	2	Terminado	Cornelio Orbe		
10	Implementar funcionalidad de asignar partida presupuestaria en forma masiva	3				
11	Implementar funcionalidad de selección múltiple e individual	2	Terminado	Cornelio Orbe		

12	Registro de los requerimientos consolidados por ítem presupuestario	2	Terminado	Cornelio Orbe
13	Reporte de requerimientos consolidados por partida presupuestaria	2		
14	Registro de los requerimientos consolidados por partida presupuestaria	2	Terminado	Cornelio Orbe
15	Creación del módulo de proyectos presupuestarios consolidados	1	Terminado	Cornelio Orbe
16	Relacionar el módulo de proyectos presupuestarios consolidados con el menú Consolidación Presupuestaria	1	Terminado	Cornelio Orbe
17	Implementación funcionalidad plan anual de compras por departamento	2	Terminado	Cornelio Orbe
18	Implementación funcionalidad plan anual de compras general	2	Terminado	Cornelio Orbe
19	Creación de los módulos y menús del plan anual de compras	1	Terminado	Cornelio Orbe
20	Implementación de la funcionalidad reportes del plan anual de compras	1	Terminado	Cornelio Orbe
21	Implementación de la funcionalidad de proforma presupuestaria resumido	2	Terminado	Cornelio Orbe
22	Implementación de la funcionalidad de proforma presupuestaria jerárquica	2	Terminado	Cornelio Orbe
23	Implementación de la funcionalidad de proforma presupuestaria por departamento y general	1	Terminado	Cornelio Orbe
24	Creación de los módulos y menús de la proforma presupuestaria	1	Terminado	Cornelio Orbe
25	Implementación de la funcionalidad de reportes de la proforma presupuestaria	1	Terminado	Cornelio Orbe

Tabla 28: Sprint 9 - Consolidación Presupuestaria

Fuente: Propia

CAPÍTULO IV

4 ARQUITECTURA DEL SISTEMA:

4.1 ARQUITECTURA DE LA APLICACIÓN

4.1.1 ARQUITECTURA DE DESARROLLO

El sistema GpiPresupuesto está compuesto de cuatro paquetes principales:

1. Interfaz de usuario (GUI),
2. Servicios del negocio (Business Logic),
3. Objetos del negocio (Business Entity)
4. Objetos de persistencia (Data Access Objects).

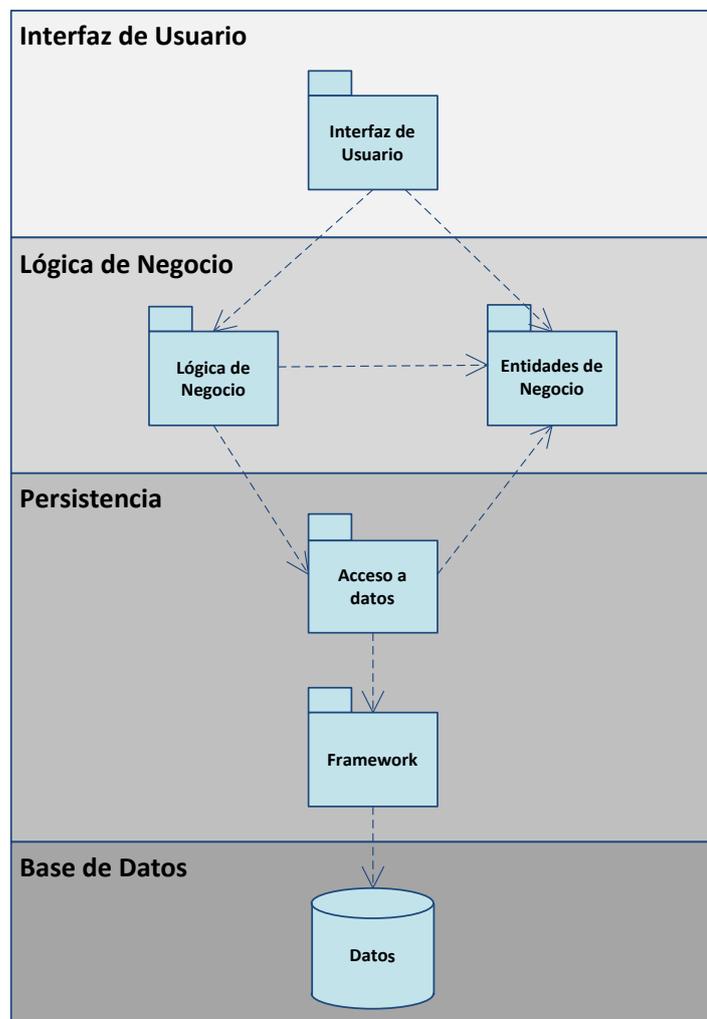


Ilustración 47: Distribución Capas del Sistema

Fuente: GPI

4.1.1.1. INTERFAZ DE USUARIO

Es la parte visible mediante la cual el usuario interactúa con el sistema, es decir, la parte gráfica como son las ventanas, los menús y opciones y demás funcionalidades que el usuario puede visualizar.

Gracias al IDE Flash Builder 4 es posible crear interfaces de manera rápida, ya que posee una paleta de herramientas para diseñar gráficamente las ventanas así como también permite asignar estilos para que estas ventanas sean más vistosas.

También se puede graficar una interfaz a través de código mxml.



Ilustración 48: Lenguajes que maneja Flex

Fuente: Propia

Los componentes serán diseñados en componentes mx y spark.

Flex se definen dos tipos de componentes: Spark y MX. Los componentes Spark son nuevos para Flex 4 y se definen en la spark. Los componentes de dominio enviado en versiones anteriores de Flex y se definen en el mx

Las principales diferencias entre spark y los componentes de MX son cómo utilizar los estilos CSS con los componentes y la forma de la máscara (skin). Para los componentes de contenedores, hay diferencias adicionales acerca de cómo realizar el diseño de los contenedores.

Spark y MX define algunos de los mismos componentes. Por ejemplo, el Spark define un control de botón en el paquete de spark.components, y MX define un control de botón en el paquete mx.controls. Cuando un componente está disponible en la Spark y MX, Adobe recomienda que utilice el componente de la Spark.

Spark y MX también definir los componentes que son únicos. Por ejemplo, el Spark define los componentes para realizar efectos tridimensionales. MX define los componentes de visualización de datos, tales como la cuadrícula de datos y los controles AdvancedDataGrid, no incluidos en el Spark. Las aplicaciones a menudo contiene una mezcla de componentes Spark y MX.

4.1.1.2 LÓGICA DEL NEGOCIO

Esta capa contiene los paquetes de Servicios de Negocio y Entidades de Negocio. Contiene la lógica para el manejo de las operaciones del negocio.

La lógica del negocio está dada por los servicios y Value Objects VO, tanto Flex como php

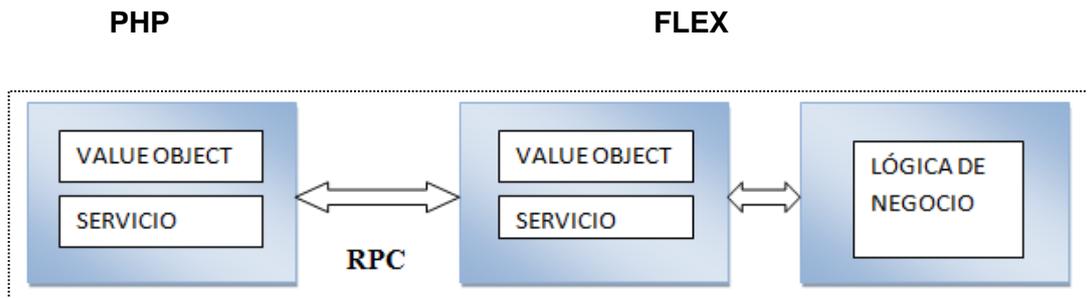


Ilustración 49: Comunicación entre servicios php y flex

Fuente: Propia

Servicios PHP

Los servicios php son clases php que contienen la referencia a la persistencia, es decir que interactúan con los objetos mapeados mediante ZF y además contienen los métodos públicos que dan la funcionalidad CRUD y otras funcionalidades y los métodos privados que son complemento de los métodos públicos.

Los value object en php son clases que por lo general tienen el mismo nombre que la entidad mapeada de la base de datos. Esta clase es incluida en el servicio php.

Pero su creación es opcional, es decir no es obligatorio, es decisión del desarrollador.

Si es creado el value object en php el value object de flex tomara el mismo nombre, pero si no ha sido creado, flex asignara automáticamente un nombre para el value object.

Servicios FLEX

Los Data Services DS o también conocidos como servicios de Flex son funcionalidades de Flex que contienen la referencia a los servicios php.

- PHP by Zend
- Http
- WSDL
- XML

El servicio que se usara es el primero, este tipo de servicio referencia a los servicios php.

Para crear o sincronizar un servicio flex, se usara el asistente que proporciona el IDE Flash Builder.

Los servicios flex son generados automáticamente así como también los value objects. Como se había comentado anteriormente, si los value objects php fueron creados e incluidos en los servicios, entonces flex generara los value objects flex con el mismo nombre de los value objects php, caso contrario el asistente de flash Builder genera automáticamente los value object y le asigna un nombre.

Lógica de Negocio

Es el conjunto de clases, interfaces, métodos, y funcionalidades que hacen posible el funcionamiento del sistema.

Contiene la programación con la que funciona un proceso, validaciones e interacciones con los servicios flex.

Está estructurada según el patrón MVCE: Modelo Vista Controlador Evento

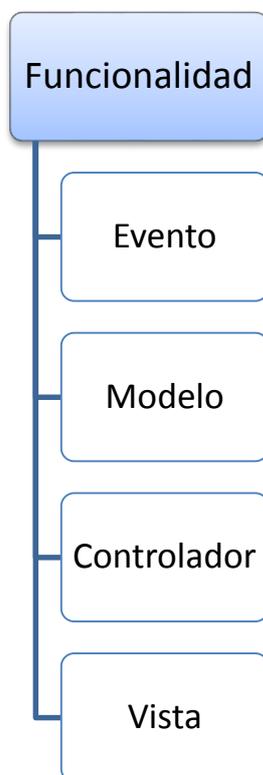


Ilustración 50: Patrón usado para el desarrollo

Fuente: Propia

Persistencia

Esta capa contiene el paquete de Objetos de Acceso de Datos, que brinda una interfaz transparente para la interacción con el Framework el cual enviará al Driver el conjunto de sentencias para interactuar con la Base de Datos.

Para ello se usará Zend Framework que contienen una línea de comandos para la interacción con la base de datos, como la conexión, configuración, referencia a tablas, esquemas y relaciones con la base de datos.

Zend Framework almacena en un archivo de extensión .zf las diferentes transacciones de persistencia que se realice con la línea de comandos de Zend

4.1.2 ARQUITECTURA DE DESPLIEGUE

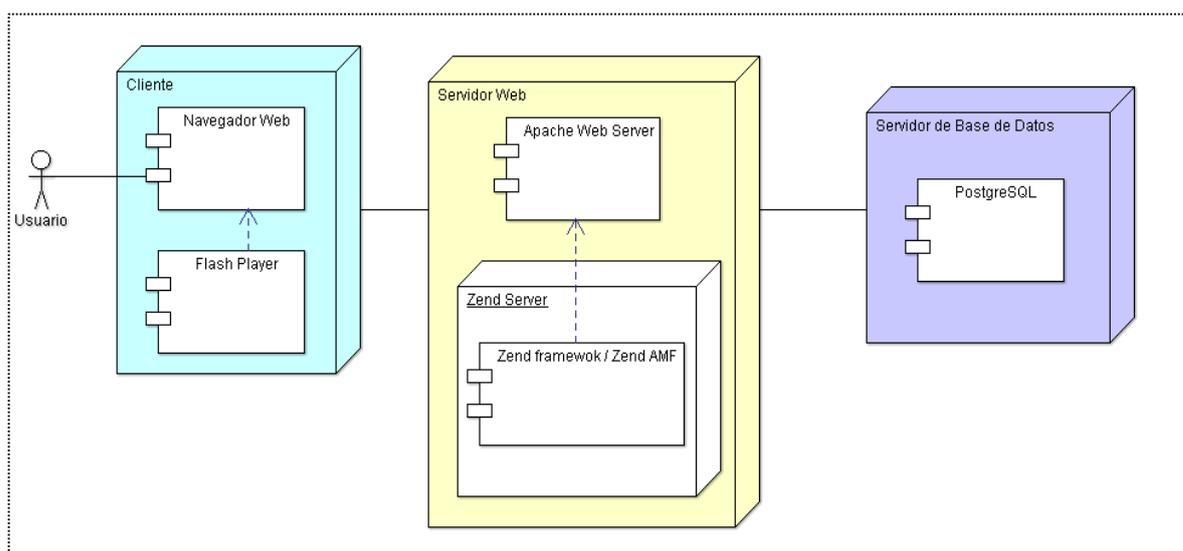


Ilustración 51: Arquitectura de Despliegue

Fuente: GPI

Nodo Cliente

Las estaciones de trabajo deben tener instalado un navegador de internet (web browser).

Puede ser cualquier navegador, para la demostración del funcionamiento del sistema GpiPresupuesto se escogió el opera.

Es necesario tener instalado Flash Player 11 o posterior.

Ya que el sistema se carga en el navegador al igual que una aplicación flash.

Nodo Servidor Web

El servidor web deberá constar de apache Server, debidamente configurado con las librerías zend y amf.

Servidor de Base de Datos

Este servidor deberá contar con la base de datos Postgres debidamente configurado y con los datos corporativos de la institución.

4.1.3 ARQUITECTURA DE PROYECTOS:

La suite de Aplicaciones GpiGestion consta de varios proyectos, descritos según su funcionalidad, aunque cada proyecto desarrollado es un sistema, dentro de la Suite GpiGestion se define como módulo o subsistema.

En la arquitectura constan varios tipos de proyectos según su funcionalidad dentro de la misma.

A continuación los proyectos usados en la arquitectura:

- **Proyecto Backend:** es el que gestiona la información con el servidor y la base de datos. Contiene los objetos de persistencia y los servicios que son llamados desde Flex.
- **Proyecto Frontend:** es aquel que contiene la lógica de negocio, y la interfaz con la cual el usuario podrá interactuar con el sistema.
- **Proyecto Librería:** es aquel proyecto que es empaquetado y puede ser llamado desde otro un proyecto Frontend, es usado para la comunicación de Proyectos.

Los proyectos son creados independientemente en un workspace o espacio de trabajo, según el desarrollador y sus requerimientos, pero todos los proyectos Flex apuntan a un mismo proyecto Backend y se comunican entre sí a través de proyectos Librería.

En la siguiente tabla se puede visualizar los proyectos que interactúan dentro de la arquitectura GpiGestion y su influencia sobre el mismo.

Proyecto	Tipo	Funcionalidad
GpiBackend	Backend	Gestiona los objetos de persistencia y la comunicación con el servidor y la base de datos
GpiPlanificación	Frontend	Gestiona la información del plan estratégico provincial y los proyectos que conllevan
GpiOrganización	Frontend	Gestiona las organizaciones e instituciones que interactúan en el sistema
GpiProyectos	Frontend	Gestiona los procesos, programas e indicadores de la ejecución de los proyectos
GpiInventarios	Frontend	Gestiona la información de los bienes y servicios de la institución
GpiPresupuesto	Frontend	Gestiona la información presupuestaria a nivel de proyectos por departamento
GpiContratos	Frontend	Gestiona los contratos generados por la institución
GpiComponentes	Librería	Contiene los componentes personalizados y genéricos usados en la aplicación
GpiSistema	Librería	Gestiona la información de los usuarios y perfiles
GpiPlanificacionLibreria	Librería	Contiene la información del proyecto GpiPlanificación y puede ser llamada desde otros proyectos Frontend
GpiOrganizacionLibreria	Librería	Contiene la información del proyecto GpiOrganización y puede ser llamada desde otros proyectos Frontend
GpiInventariosLibreria	Librería	Contiene la información del proyecto GpiInventarios y puede ser llamada desde otros proyectos Frontend
GpiFrontend	Frontend	Contiene la estructura de la interfaz principal y la comunicación con los módulos de los proyectos

Tabla 29: Proyectos que interactúan sobre GpiGestion

Fuente: Propia

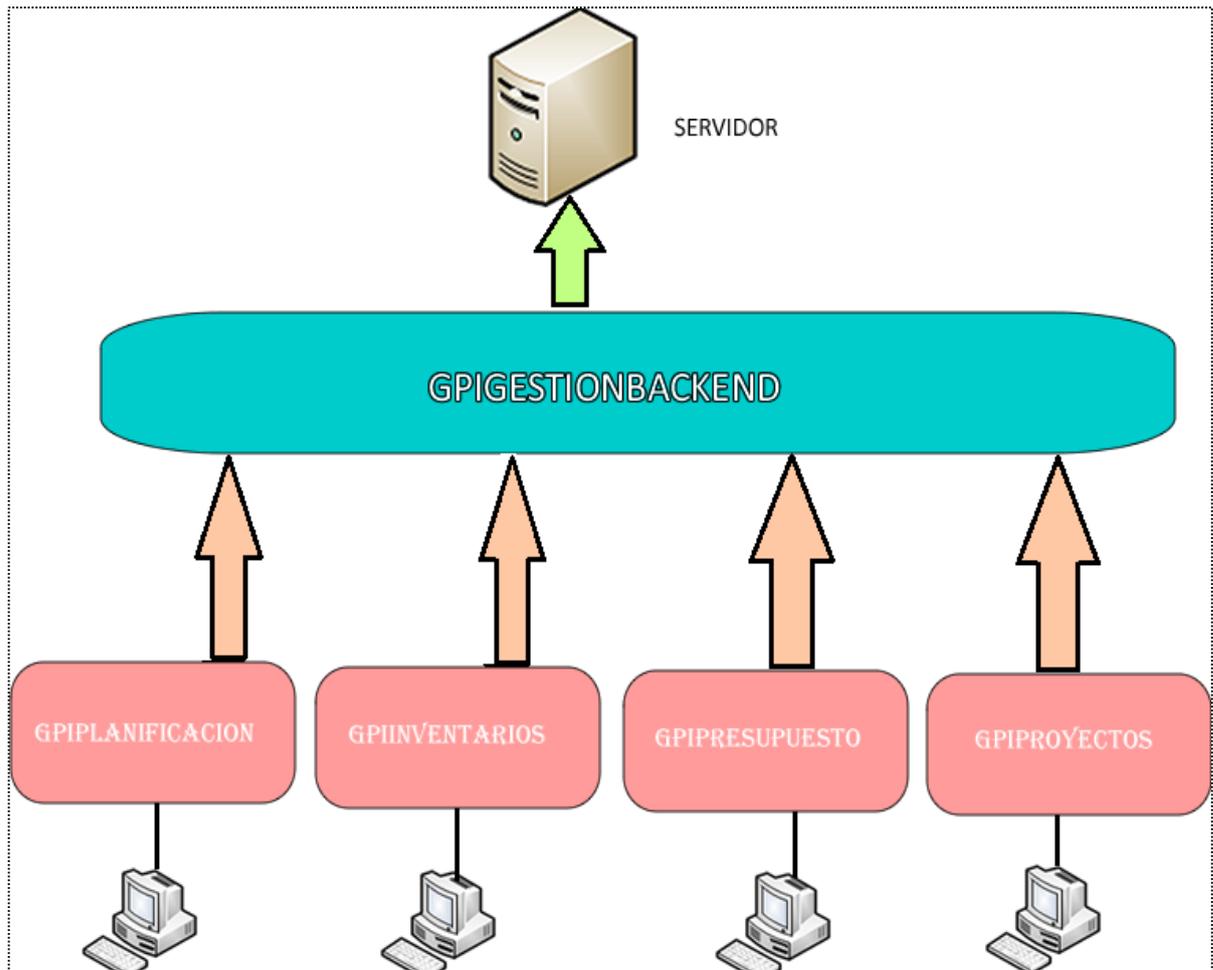


Ilustración 52: Relación Proyectos Backend y Frontend

Fuente: Propia

Relación entre Proyectos Frontend

Por ser un sistema integrado, un módulo de la arquitectura no es independiente, sino que necesita la información de otro sistema como dato de entrada para funcionar.

Así, el sistema GpiPresupuesto necesita del sistema GpiPlanificación, y GpiInventarios, en la siguiente ilustración se puede apreciar la interrelación entre los sistemas mencionados.

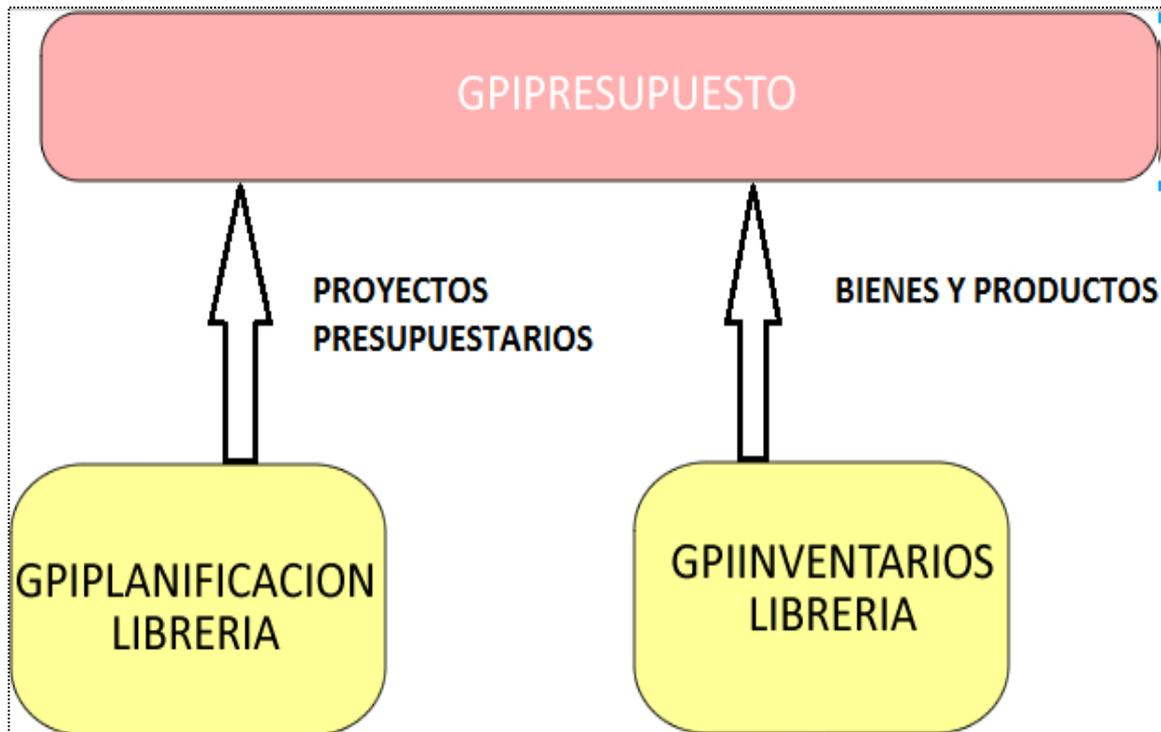


Ilustración 53: Interrelación de los Proyectos Frontend

Fuente: Propia

Relación de los Proyectos Librería y Librerías Externas:

La arquitectura para la presentación está distribuida en tres proyectos importantes:

GpiFrontend: contiene la estructura del Layout principal, el menú de opciones en la parte izquierda, el banner y datos informativos en la parte superior y el espacio de trabajo en el centro

GpiComponentes: contiene los componentes personalizados que han sido usados en este proyecto, los componentes son reutilizables

GpiSistema: contiene la información del menú de opciones y los módulos que se van a llamar en el espacio de trabajo de GpiFrontend.

En la siguiente ilustración se puede apreciar la relación existente entre estos proyectos sus dependencias con los demás proyectos.

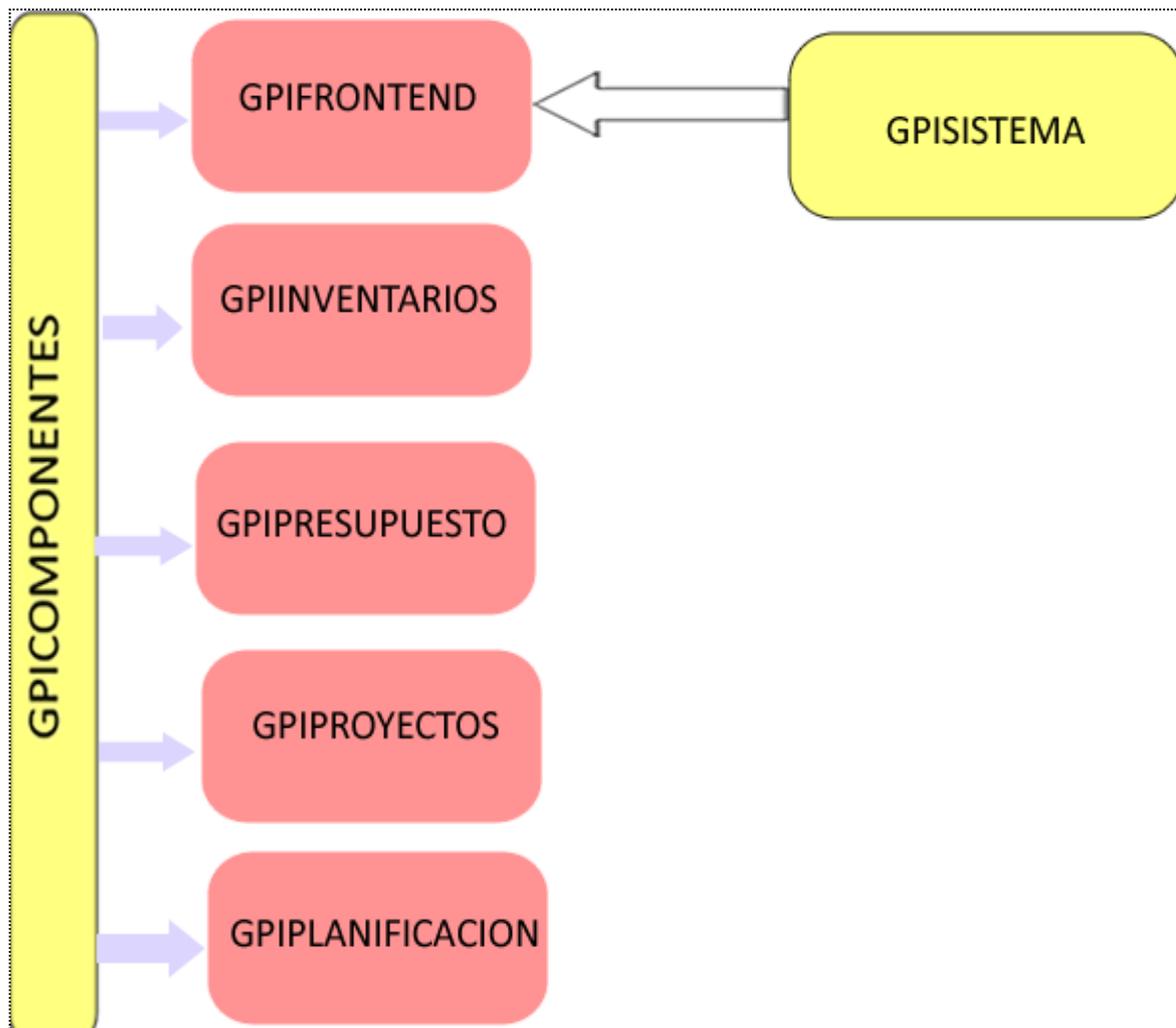


Ilustración 54: Relación de Proyectos de Librerías y Proyectos Frontend

Fuente: Propia

Relación del Proyecto GpiFrontend y los demás Proyectos

Al igual que todos los proyectos Frontend se relacionan con el proyecto Backend, así también existe un solo proyecto para la interfaz del usuario, este proyecto es GpiFrontend

Como ya se había mencionado anteriormente, el proyecto GpiFrontend contiene el espacio de trabajo en el cual se va a llamar a los módulos de los otros proyectos, esta arquitectura modular es creada dinámicamente, es decir, de acuerdo con la relación con la opción menú y el tipo de permiso que tenga el usuario.

En la siguiente ilustración se puede apreciar la relación existente entre el proyecto GpiFrontend y los demás Proyectos.

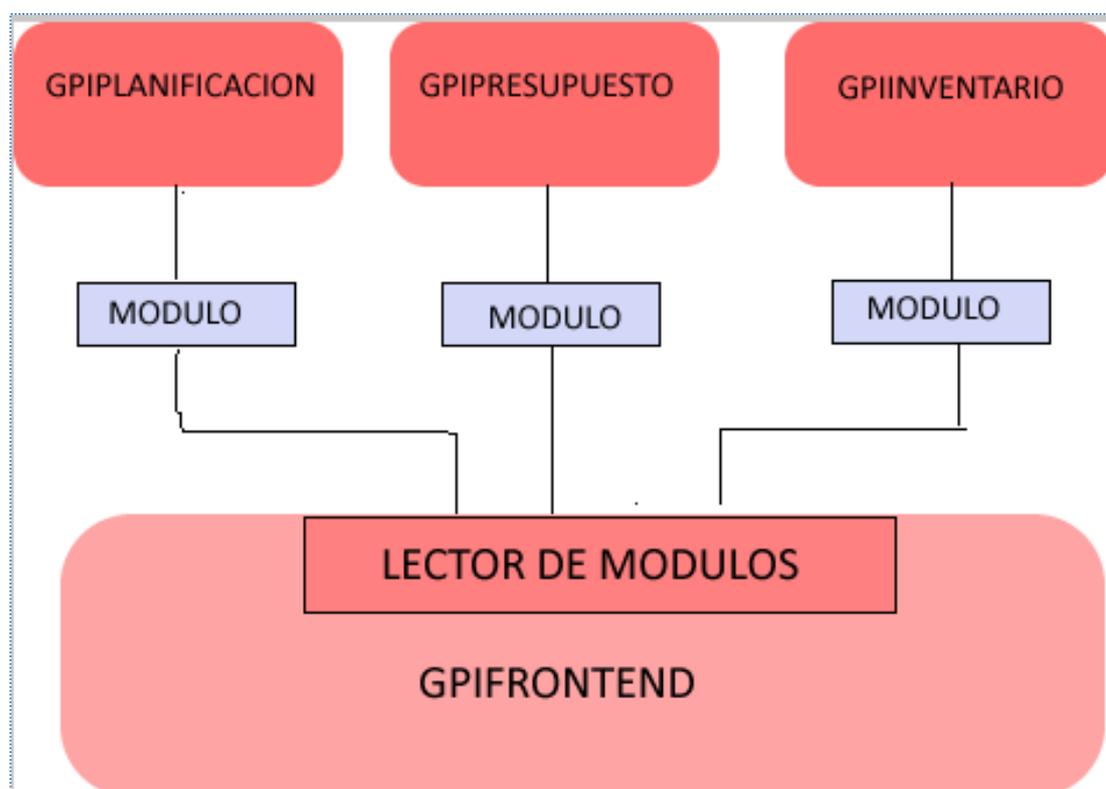


Ilustración 55: Relación GpiFrontend con otros proyectos

Fuente: Propia

4.2 DEFINICIÓN DE LOS PROCESOS DEL SISTEMA

4.2.1 RELACIONES Y DEPENDENCIAS

La suite GPI GESTION involucra y compromete a toda la institución con el logro de los objetivos estratégicos y establece la integración de la planificación con todo el proceso operativo y de gestión del Gobierno Provincial.

Como se observa en el grafico los sistemas son integrados y dependientes, quiere decir que un sistema depende de otro.

La arquitectura propuesta en la anterior ilustración está en desarrollo, es decir ninguno de los módulos o subsistemas está en producción.

Solamente algunos están en desarrollo, es por eso que no se podrá completar todo el alcance planificado, ya que no se implementó un sistema que está estrechamente relacionado con el sistema GpiPresupuesto

En la siguiente ilustración se puede apreciar la relación del sistema GpiPresupuesto con los sistemas de Gestión Financiera.

Además el Sistema de Presupuesto tiene la dependencia directa con los sistemas de la Gestión Estratégica que comprenden: El sistema de planificación, y el sistema de proyectos.

Ya que un sobre los proyectos se realiza la planificación y ejecución presupuestaria. Así también depende el sistema de Inventarios puesto que de ahí se toma la información de los ítems que se requiera presupuestar.

Dependencias de Planificación y Proyectos:

El departamento de planificación del GPI, es el departamento encargado de planificar, coordinar el Plan Estratégico de la institución, este Plan es elaborado en base a los objetivos estratégicos y las necesidades de los habitantes de la provincia de Imbabura.

El plan estratégico es desglosado hasta el nivel de Actividades programáticas y Proyectos según la siguiente estructura.

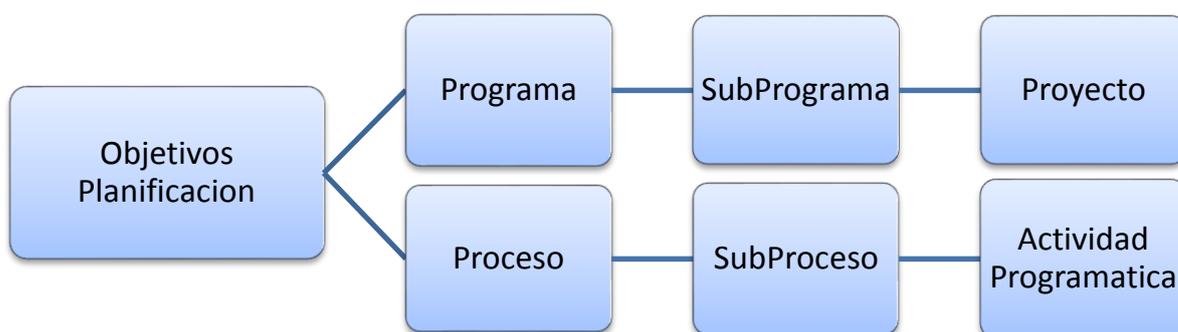


Ilustración 56: Organización de los Objetivos Estratégicos

Fuente: Propia

La planificación presupuestaria se realizara a nivel de Actividades Programáticas y Proyectos.

A su vez este departamento será el encargado de asignar los presupuestos a cada departamento de la institución.

Dependencias de Inventarios:

El departamento de inventarios es el encargado de la administración de los bienes y productos de la institución.

La organización de los catálogos de los productos debe regirse al INCOP, que se encuentra en la página web de compras públicas.

Para la planificación presupuestaria se tomara los productos o servicios de la categoría del CPC a nivel 8.

Ej.: **48122.00.1**

4.2.2 PERMISOS DE USUARIOS

El GPI está organizado en departamentos o áreas según su función.

Los departamentos de GPI son:

- ADMINISTRACION GENERAL
- COMUNICACIÓN
- PLANIFICACION
- GESTION SOCIAL
- DESARROLLO ECONOMICO
- INFRAESTRUCTURA FISICA
- TECNOLOGIAS DE INFORMACION
- PRESUPUESTO

Cada departamento cuenta con un usuario o director de área.

Existe también un departamento de Presupuesto que es el administrador del sistema y es el encargado de administrar los datos presupuestarios que son los necesarios para el funcionamiento del sistema.

El sistema GpiPresupuesto administrará los siguientes usuarios:

- Usuario de Presupuesto
- Usuario de Departamento
- Director de Departamento

Usuario de Presupuesto:

Es el encargado de la administración de los datos presupuestarios, necesarios para el funcionamiento del proceso presupuestario.

Los datos presupuestarios son:

- Ejercicios Presupuestarios
- Periodos Presupuestarios
- Partidas Presupuestarias

Además también puede obtener reportes del proceso presupuestario.

Usuario de Departamento:

A este usuario le corresponde realizar la planificación presupuestaria a nivel de ítems presupuestarios en el periodo de planificación, también podrá realizar la consolidación presupuestaria y obtener reportes del proceso presupuestario.

Director de Departamento:

Es el encargado de aprobar los requerimientos realizados por los usuarios de su departamento, así también podrá realizar la consolidación presupuestaria y generar reportes del proceso presupuestario.

4.2.3 PROCESO PRESUPUESTARIO

Proceso Presupuestario

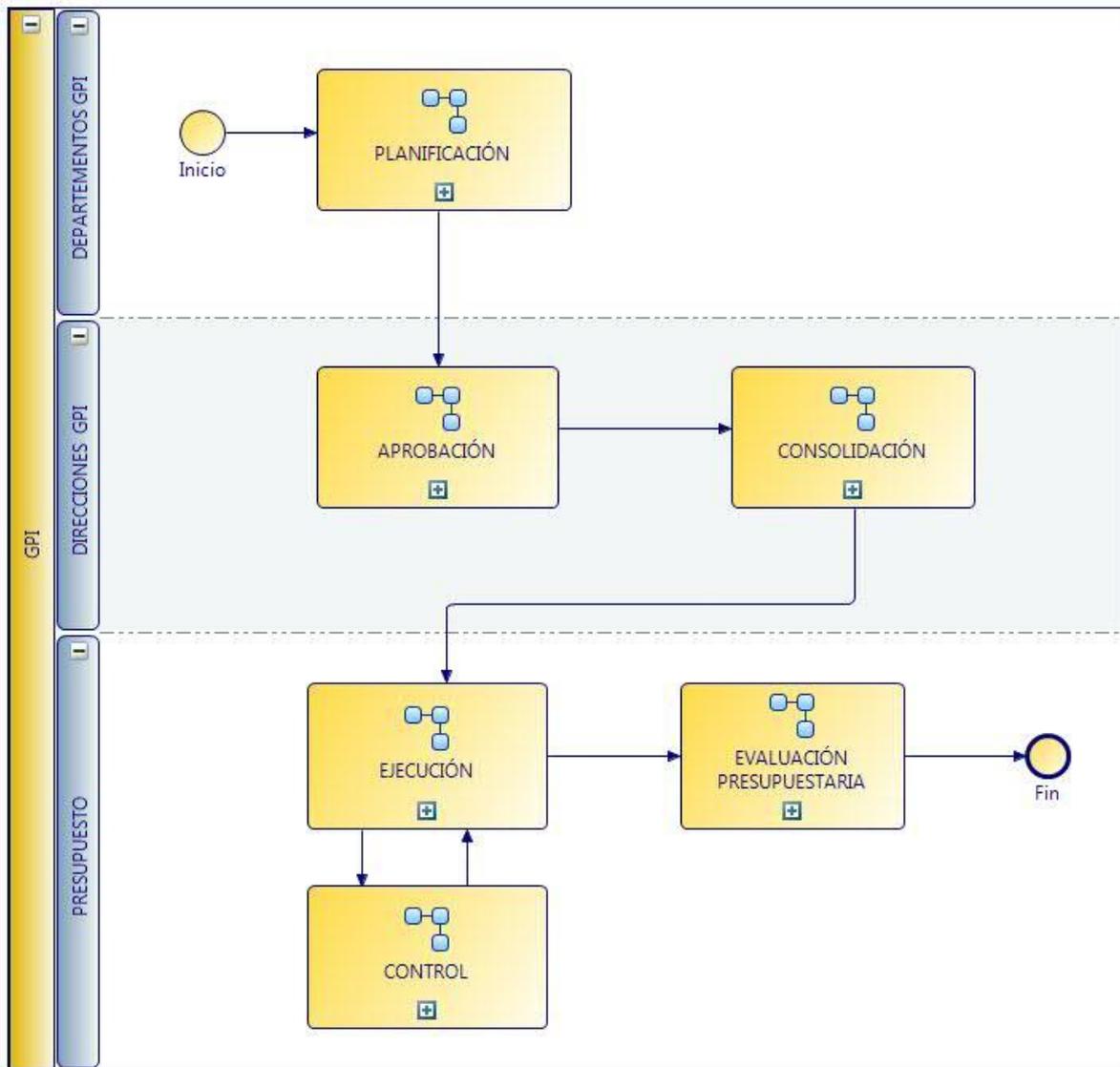


Ilustración 57: Proceso Presupuestario

Fuente: Propia

El sistema GpiPresupuesto realizará la gestión solamente de una parte del proceso presupuestario.

- Planificación
- Aprobación
- Consolidación

El sistema GpiPresupuesto recibirá la información de los proyectos y actividades programáticas entregadas por el departamento de Planificación, estos proyectos deben constar de un monto estimado, y el monto presupuestado.

Y finaliza con la generación del Plan Anual de Compras y la Proforma Presupuestaria.

4.2.4 DATOS PRESUPUESTARIOS:

Ejercicios y Periodos

El proceso presupuestario se realiza dentro de un lapso de tiempo, a este lapso se le denomina Ejercicio Presupuestario

Un ejercicio presupuestario está comprendido entre septiembre del año en curso y diciembre del siguiente año.

Los subprocesos que se ejecutan en el proceso presupuestario se desarrollan en lapsos de tiempo más pequeños. Estos lapsos de tiempo son denominados Periodos Presupuestarios.

Solamente el usuario de Presupuesto puede iniciar y terminar los ejercicios y periodos presupuestarios.

Los periodos presupuestarios gestionará en el ejercicio presupuestario son:

- Planificación
- Aprobación
- Consolidación

Y estarán serán activados en secuencia según el proceso presupuestario.

Solamente el usuario de presupuesto será el encargado de gestionar los ejercicios y periodos presupuestarios.

4.2.5 PLANIFICACIÓN

En este subperiodo se define los objetivos, programas y procesos necesarios para llevar a cabo la misión del GPI y cumplir con el Plan Nacional del buen vivir.

Estos objetivos son definidos en el departamento de planificación a nivel de Proyecto y Actividades Programáticas.

Además serán los responsables de la asignación del presupuesto para cada departamento y también de la asignación del monto estimado en cada proyecto.

El periodo de planificación inicia cuando el usuario de presupuesto activa este periodo, entonces se comunica a los usuarios de los diferentes departamentos del GPI que inicien con el registro de requerimientos.

Registro de Requerimientos

Es la actividad mediante la cual los usuarios de los departamentos deberán asignar los recursos necesarios para que se lleve a cabo la ejecución de los objetivos previamente planificados.

El recurso necesario para llevar a cabo un objetivo se denomina Ítem Presupuestario.

Un Ítem Presupuestario básicamente consiste en un producto, servicio, rubro, cargo.

Para el sistema GpiPresupuesto se contemplara la asignación de productos, estos deberán ser tomados de la fuente del Catálogo Central de Productos (CPC), a nivel de código 8, es decir un tipo de producto.

Ejemplo:

Código: 44216.15.1

Descripción: Maquinas - herramientas que trabajan mediante la remoción de metales, cargueros metálicos sinterizados o materiales metalocericos, n.c.p, no controladas por clave numérica: máquinas de grabar, máquinas de limar (excepto las de mano), etc.

Además el usuario deberá ingresar el monto unitario de cada producto o servicio asignado y la cantidad de ítems que va a presupuestar en su proyecto.

El monto total de los ítems presupuestarios asignados al proyecto no debe pasar el monto estimado de este.

Después de un trimestre el departamento de Presupuesto termina el periodo de planificación, en ese momento se comunica a los departamentos de la institución que ya no pueden registrar requerimientos.

Seguidamente el departamento de presupuesto inicia el periodo de Aprobación.

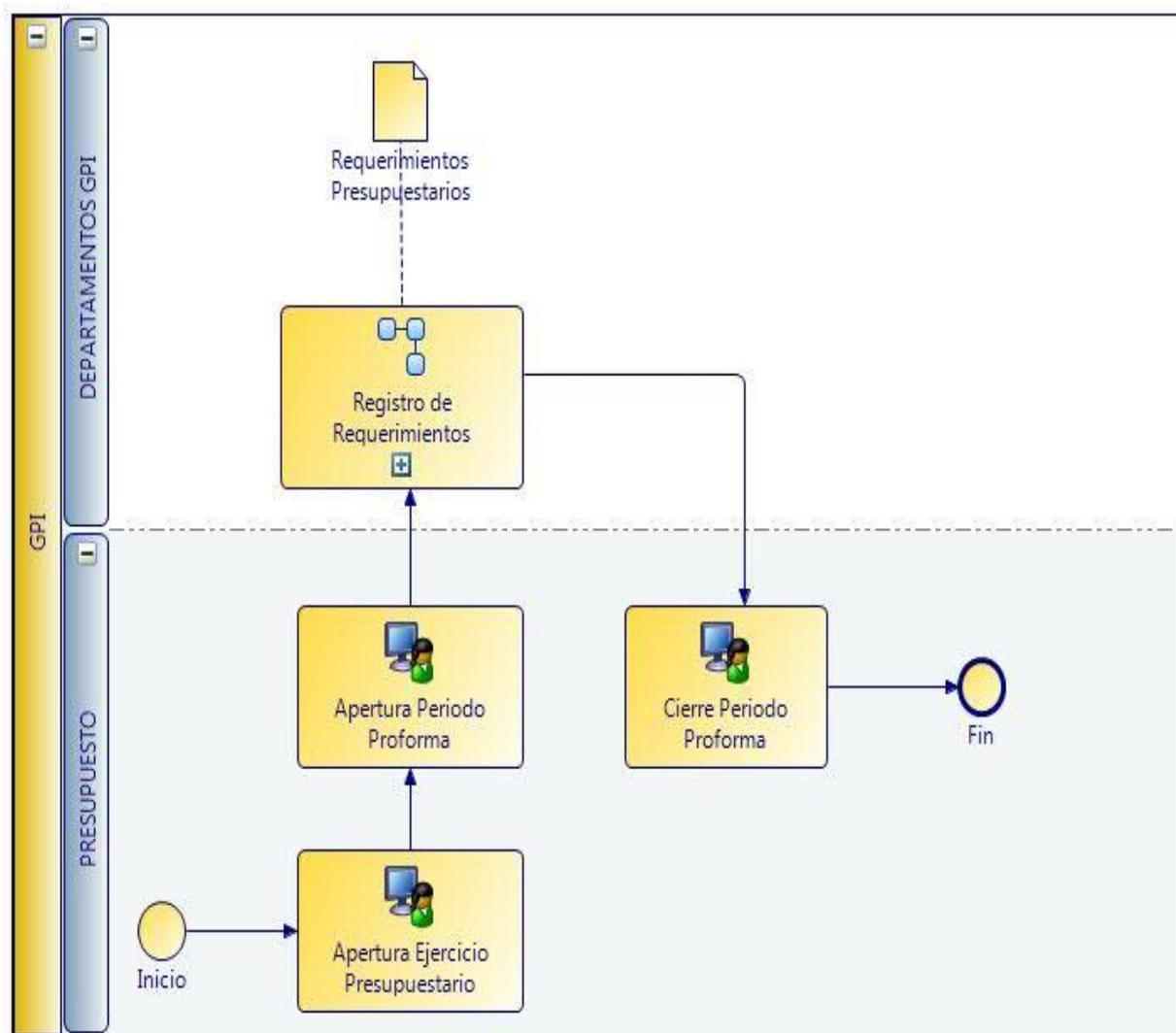


Ilustración 58: Planificación Presupuestaria

Fuente: Propia

4.2.6 APROBACIÓN:

Iniciado el periodo de aprobación por el departamento de Presupuesto el usuario de cada departamento revista los requerimientos planificados; si el director encuentra que la información requerimiento esta congruente y el monto presupuestado es válido, entonces aprueba el requerimiento. Pero si encuentra alguna inconsistencia el director tiene la opción de eliminar o modificar dicho requerimiento.

Una vez finalizado el periodo de aprobación los directores ya no tienen permiso para modificar o eliminar los requerimientos, solamente para visualizar. Y como resultado se obtiene los requerimientos aprobados. La aprobación finaliza cuando todos los proyectos hayan sido revisados y aprobados.

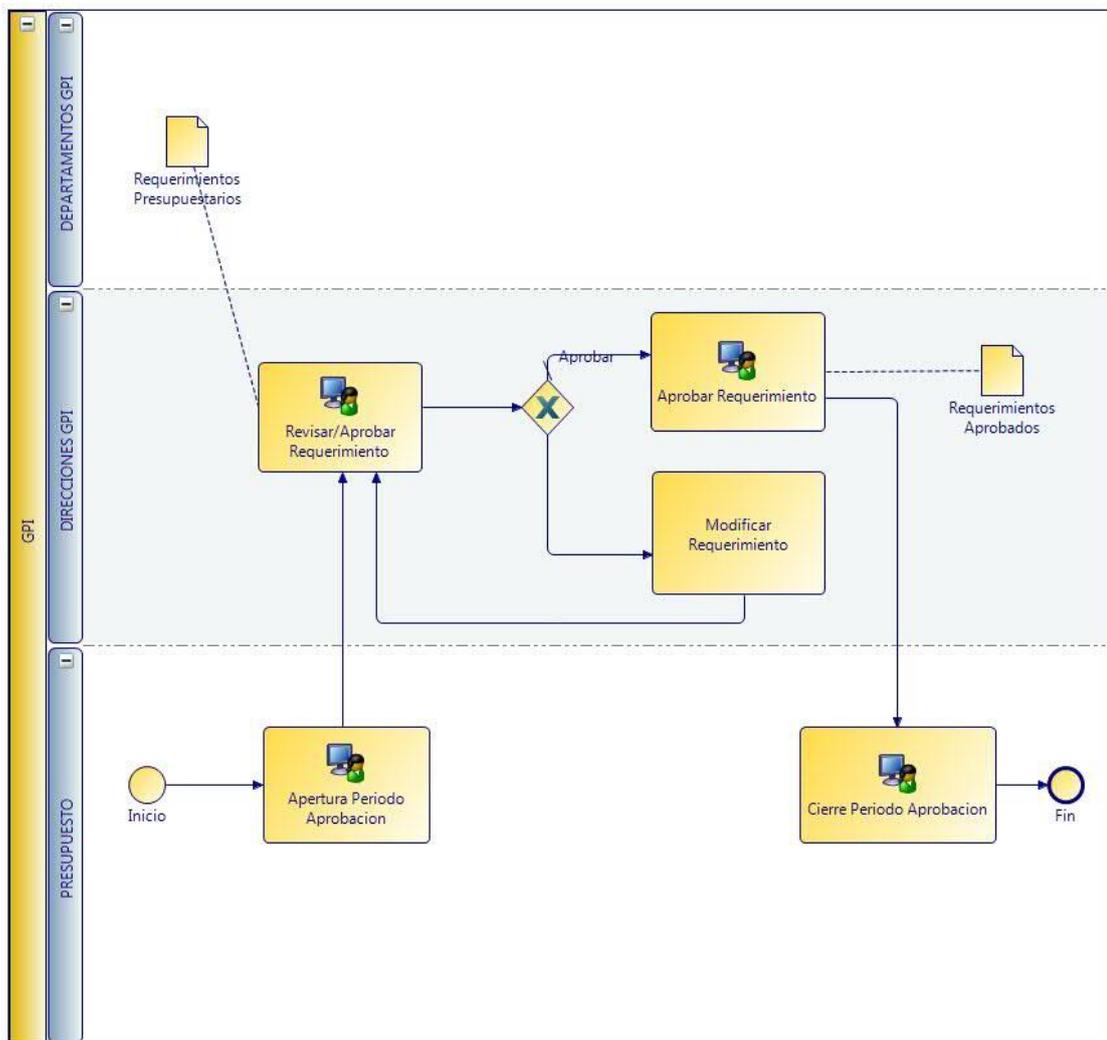


Ilustración 59: Aprobación Presupuestaria

Fuente: Propia

4.2.7 CONSOLIDACIÓN PRESUPUESTARIA:

El proceso inicia cuando el usuario de presupuesto activa el periodo de consolidación.

Es el proceso de agrupar o consolidar los requerimientos aprobados por los Departamentos del GPI en el periodo de aprobación, esta agrupación puede ser a nivel de ítems presupuestarios y a nivel de partidas presupuestarias.

Para poder agrupar a nivel de partida presupuestaria es necesario asignar una partida presupuestaria a cada uno de los ítems presupuestarios consolidados, cuando se haya asignado una partida presupuestaria a todos los ítems consolidados el proyecto ha sido consolidado.

Cuando todos los proyectos han sido consolidados se puede generar el Plan Anual de Compras y la Proforma Presupuestaria por cada departamento del GPI, y los datos de la proforma presupuestaria son los que serán ingresados al sistema Olympos para la ejecución presupuestaria.

El periodo de consolidación termina cuando el usuario de presupuesto cierra el periodo de consolidación.

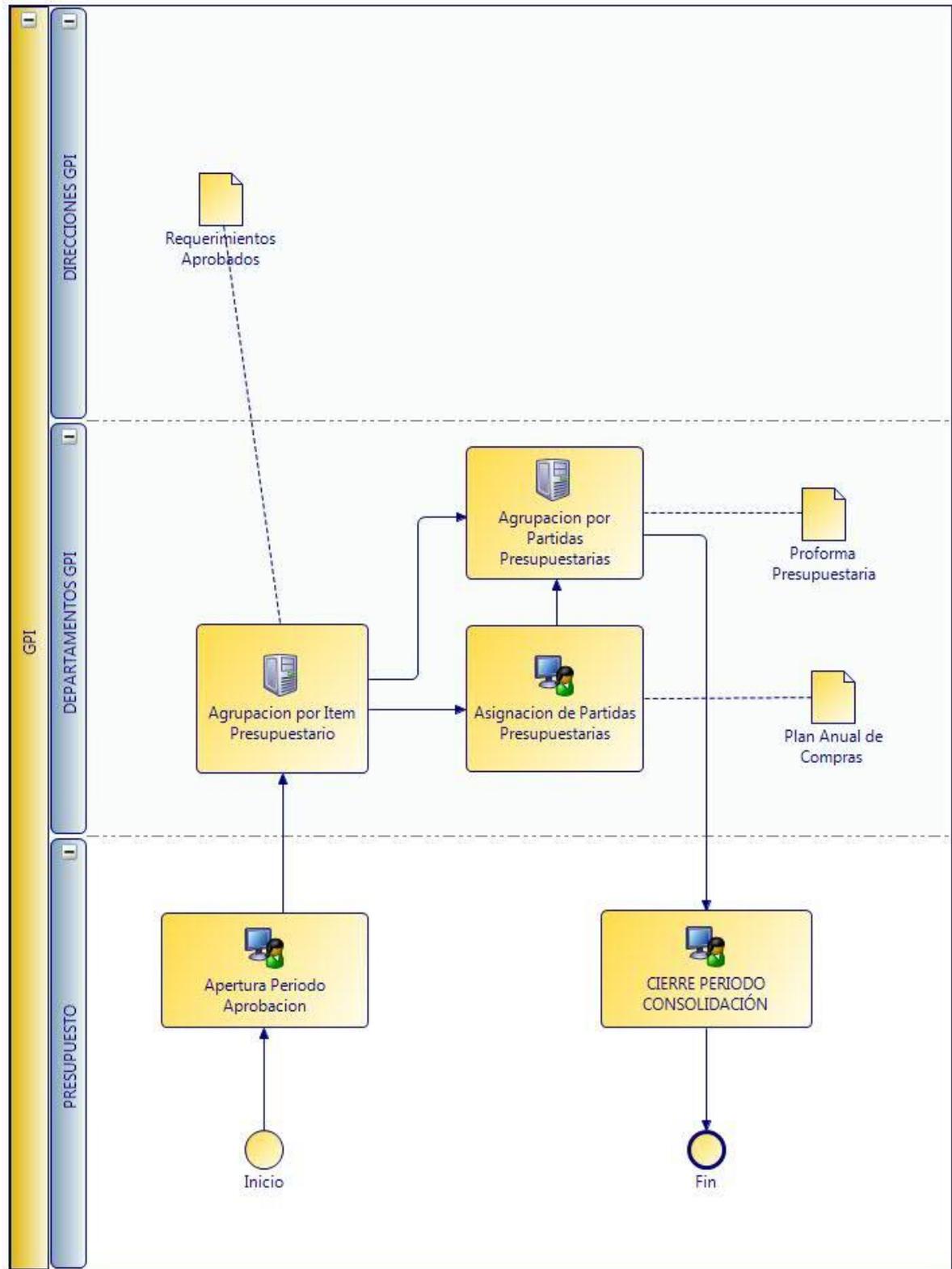


Ilustración 60: Consolidación Presupuestaria

Fuente: Propia

CAPÍTULO V

5 IMPLEMENTACIÓN Y PRUEBAS

Según las políticas y el acuerdo con la institución para la cual se desarrolló el sistema de evaluación presupuestaria, el sistema transaccional y bpm se implementó en un servidor de la institución para el uso de los usuarios de la misma.

La institución adquirió una ip pública y un dominio, con el cual se podrá ingresar directamente desde el internet

- Ip: 172.3.16.23
- Dominio: <http://gpigestion.gob.ec/>

Para acceder al sistema transaccional GpiPresupuesto ingresamos a la siguiente dirección url

<http://gpigestion.gob.ec/GpiGestionBackend/public/bin-debug/GpiGestionFrontend.php>

Y la dirección para la administración de los procesos bpm es la siguiente

<http://gpigestion.gob.ec:8080/sysworkflow/en/neoclassic/login/authentication.php>

5.1 IMPLEMENTACIÓN

5.1.1 IMPLEMENTACIÓN DEL SISTEMA TRANSACCIONAL

Para la implementación del sistema transaccional GpiPresupuesto es necesario configurar e instalar php, la base de datos, el servidor de aplicaciones y ZendFramework.

Se lo puede instalar por separado y luego configurarlos entre ellos, otra opción también es usar xampp server que contiene el lenguaje de programación, y el servidor de aplicaciones.

Para la instalación manual de ZendFramework es necesario agregar en las variables del sistema la ruta en donde se encuentra el framework

Path: c:\ZendFramework\bin

Otra alternativa es instalar Zend Server, que contiene el servidor de aplicaciones, el lenguaje de programación (php) y ZendFramework instalado y configurado.

Una vez configurado el servidor de aplicaciones ubicamos en la carpeta hdocs el proyecto Backend

Este contiene los servicios php y los archivos swf generados por cada uno de los módulos de los proyectos y componentes, así como también los archivos de configuración de la conexión con la base de datos y zend framework;

El sistema transaccional antes de la implantación paso por pruebas y revisiones tanto con el personal de informática como también con el usuario del departamento de presupuesto.

5.1.2 IMPLEMENTACIÓN DEL GESTOR DE PROCESOS DE NEGOCIO BPM

ProcessMaker tiene versiones para Linux y Windows, la diferencia es que en el primero se instala manualmente, mientras que para el segundo sistema operativo la empresa Colosa, desarrollo un wizard, el cual facilita mucho la instalación.

Debido a que ProcessMaker está desarrollado en php, es necesario que en la maquina en donde se va a instalar, este correctamente instalado y configurado php y también un servidor de aplicaciones esté disponible, adicionalmente como cualquier sistema desarrollado se necesita un servidor de base de datos en donde se almacenaran los datos necesarios para que ProcessMaker funcione.

En resumen los requisitos para el funcionamiento de ProcessMaker son.

- Apache 2.2.3 o superior.
- MySQL 4.1.20 o superior.
- Php 5.1.6 o superior

Como se mencionó anteriormente en Linux estos tres requerimientos se instalan y configuran por separado y es necesario descargar el código de ProcessMaker de la página oficial <http://sourceforge.net/projects/processmaker/files>, y configurar según lo que indica la documentación.

En Windows es mucho más fácil la instalación ya que en el wizard que también se descarga de la página oficial esta contenido todos los requerimientos necesarios, es decir, el lenguaje de programación, la base de datos, el servidor de aplicaciones y ProcessMaker obviamente; una gran ventaja de esta instalación es que ya están configurados entre sí, y en el momento de la finalización de la instalación, ProcessMaker ya está implementado en el servidor y está listo para su uso.

5.2 INTEGRACIÓN DEL SISTEMA TRANSACCIONAL CON EL GESTOR DE PROCESOS BPM

La interfaz que proporciona Flash Builder 4.5 nos brinda la funcionalidad de conectarnos con diferentes tipos de servicios, entre ellos el de conectarnos directamente con un servicio web publicado pero existe un pequeño problema, la versión de la herramienta mencionada soporta un webservices soap 1.1 mientras que el servicio web de processmaker se publica mediante el protocolo 1.2 de manera que no es posible consumir el servicio web de processmaker desde flex.

Otra alternativa también es consumir el servicio web desde ZendFramework y php usando el servidor de aplicaciones establecido apache.

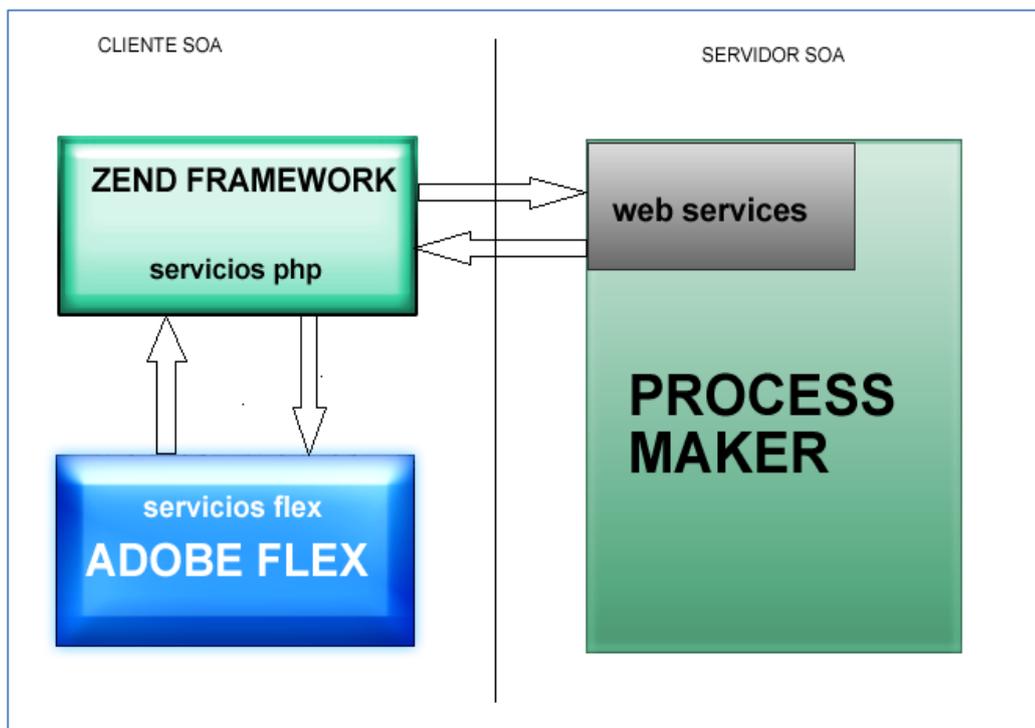


Ilustración 61: Flujo de la petición y respuesta del servicio web

Fuente: Propia

Como podemos observar en la ilustración el flujo se realiza de la siguiente manera:

1. El servicio flex realiza una petición sobre una operación del servicio soap.
2. El servicio php que se comunica mediante el protocolo AMF con el servicio flex recibe la petición y realiza una nueva petición a la operación requerida del servicio web.
3. El webservices de processmaker recibe la petición y la procesa

4. El webservices de processmaker devuelve los datos de la petición en forma de un objeto Responde, conforme está establecido en el soap.
5. El servicio php captura el objeto response y lo devuelve en forma de objeto php, o variable según este estructurado el servicio php.
6. El servicio flex captura el valor devuelto del servicio php y lo dispone para la funcionalidad del sistema transaccional.

5.2.1 SERVICIO PHP:

Para el servicio php es necesario declarar un cliente soa, y en el parámetro del constructor ingresar la url del webservices, y luego se puede invocar a cualquier operación del servicio web, un punto muy importante es tomar en cuenta la estructura de la operación, es decir los parámetros se deben ingresar en la estructura que determine el servicio web, de otro modo no funcionará.

A continuación un pequeño ejemplo sobre la manera de invocar la operación login y obtener como resultado el id de la session bpm.

```
$url="http://gpigestion.gob.ec/sysworkflow/en/classic/services/wsdl2";
```

```
$this->client = new SoapClient ($URL);
```

```
$params = array(array('userid'=>$user, 'password'=>$pass));
```

```
$resultLogin = $this->client->__SoapCall('login', $params);
```

```
$sessionId = $resultLogin->message;
```

5.2.2 SERVICIO FLEX

Para implementar el servicio flex se crea un nuevo servicio usando el wizard que nos proporciona Flash Builder, y seleccionamos la opción: *PHP by Zend*

Inmediatamente se crea los archivos en actionsript que corresponden tanto al servicio como a los valueObjects (objetos) que devuelve el servicio php.

FlashBuilder nos proporciona la funcionalidad del Test Operation, es decir, podemos realizar una operación de prueba al servicio creado.

Ejemplo:

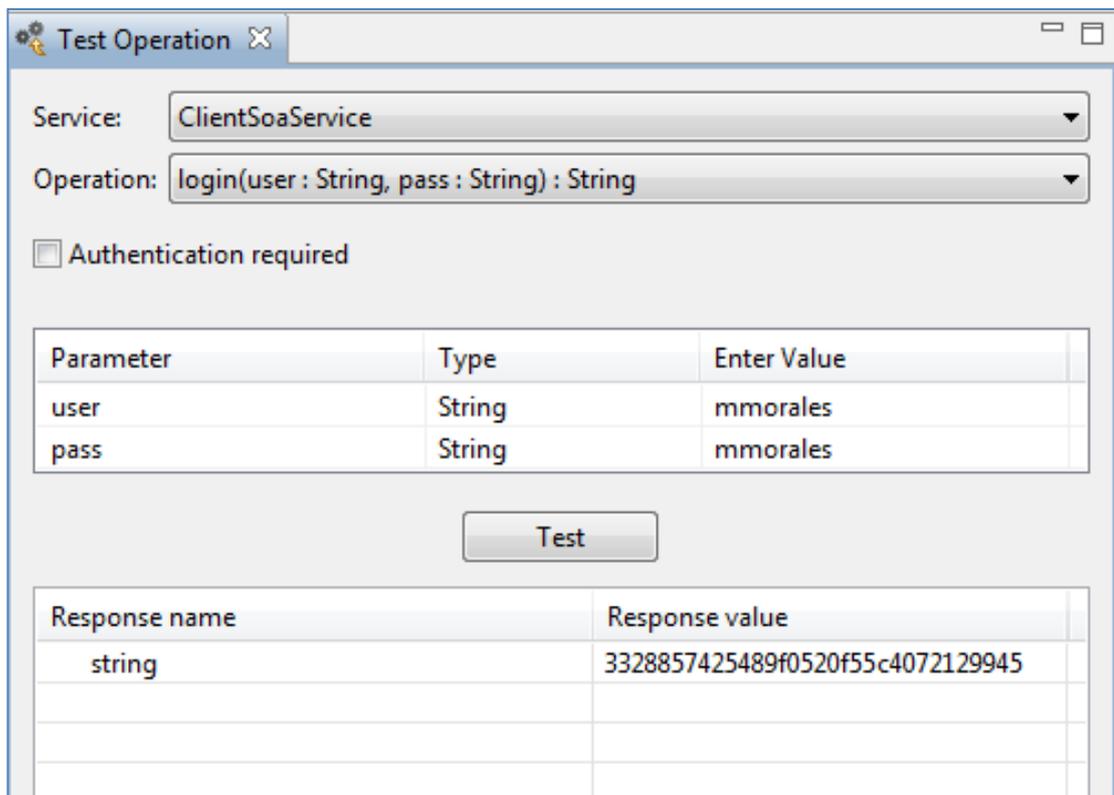


Ilustración 62: Test de una operación del servicio web

Fuente: Propia

5.3 CINCROIZACIÓN DEL PROCESO BPM CON EL SISTEMA TRANSACCIONAL

Una aspecto muy importante es que el proceso bpm fluya paralelamente con el sistema transaccional, es decir, si realizo una operación transaccional mover el flujo del proceso a otra tarea, y así también el sistema transaccional no debe permitir realizar alguna transacción sino le corresponde en el proceso bpm.

Para explicarlo de mejor manera realizaremos un ejemplo piloto desde el inicio del ejercicio presupuestario hasta la finalización del mismo.

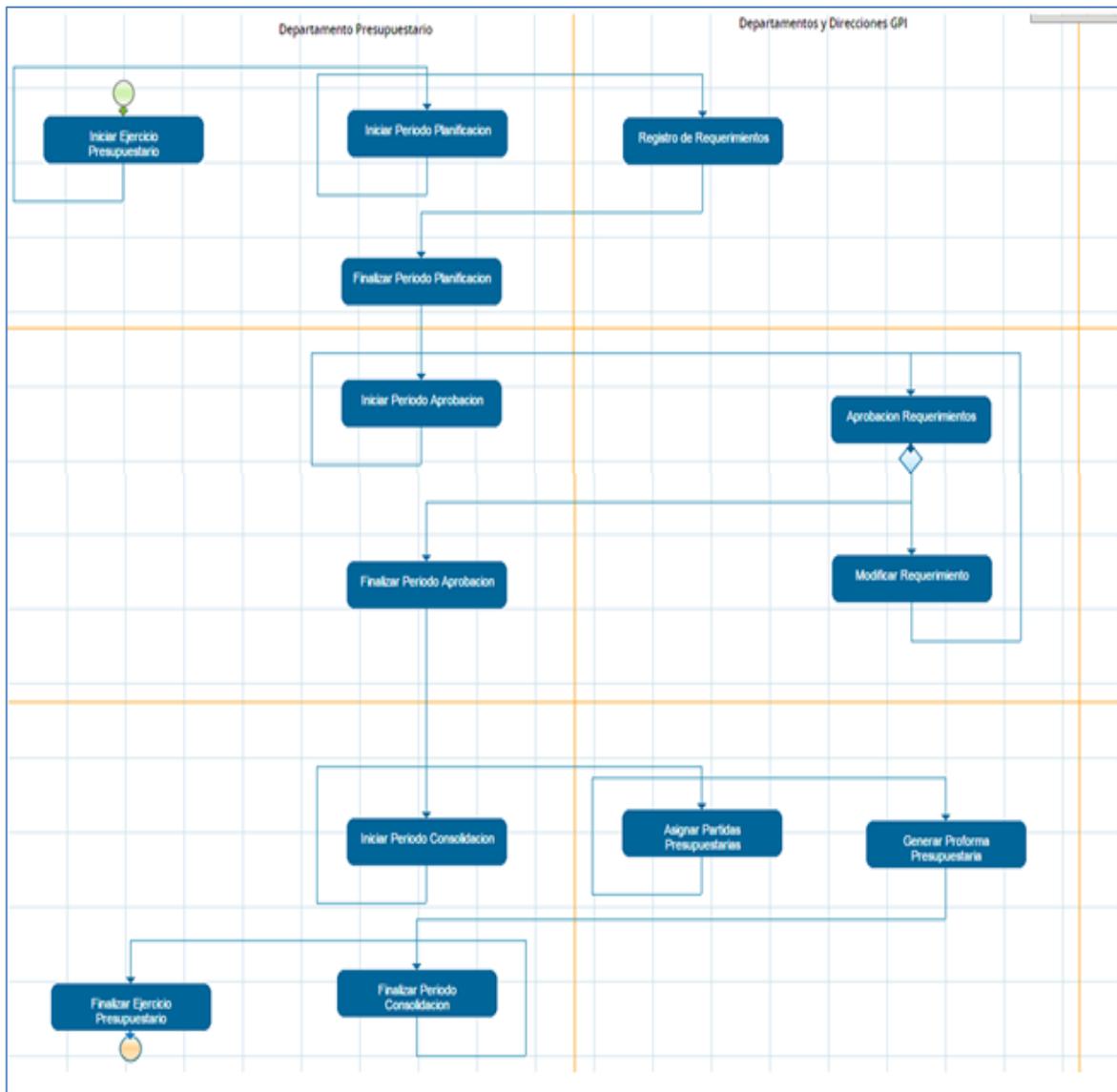


Ilustración 63: Flujo del proceso presupuestario

Fuente: Propia

5.4 FLUJO DEL PROCESO TRANSACCIONAL Y BPM

Se realiza login mediante el sistema transaccional e internamente se inicia sesión en el proceso bpm.



SUB DIRECCIÓN DE GESTIÓN TECNOLÓGICA

© Copyright GPI
2014

Ilustración 64: Login con el usuario mmorales

Fuente: Propia

Se inicia un nuevo ejercicio presupuestario e inmediatamente se crea un nuevo caso en el proceso bpm, y se mueve la tarea a la siguiente, según el flujo la tarea actual es Iniciar Periodo Planificación.

Ejercicios Presupuestarios		Periodos Presupuestarios		
CODIGO	NOMBRE	INICIO	FIN	ESTADO
E-2014	Ejercicio 2014-2015	2014-09-01	2015-08-31	⊖
E-2013	Ejercicio 2013-2014	2013-09-02	2014-08-29	⊕
E-2012	Ejercicio 2012-2013	2012-09-03	2013-08-30	⊖

EXITO Se inicio correctamente un nuevo proceso BPM

Ilustración 65: Inicio del ejercicio presupuestario

Fuente: Propia

Como paso siguiente vamos a ingresar en el proceso bpm con el mismo usuario



The login form is titled "Login" and contains the following fields and controls:

- User: Input field containing "cortega"
- Password: Input field with masked characters (dots)
- Language: Dropdown menu set to "English"
- Login: Submit button

Ilustración 66: Login en el proceso bpm, usuario cortega

Fuente: Propia

Se puede observar que en la bandeja de entrada del usuario llega una notificación indicando que se ha creado un nuevo caso y tiene asignado una tarea

The screenshot shows the "HOME" dashboard with a sidebar on the left and a main content area. The sidebar includes "Cases" with sub-items: "New case", "Inbox (10)", "Draft (2)", "Participated (26)", "Unassigned (0)", and "Paused (0)". The main content area displays a table of cases with filters for "Open Actions", "Read", "Unread", "All", "Category", and "Process".

#	Summ...	Case N...	Case	Process	Task	Sent By
167			#167	Gestion Presupuesto	Iniciar Periodo Planificacion	morales, maria (m...
166			#166	Gestion Presupuesto	Finalizar Ejercicio Presupuestario	morales, maria (m...
119			#119	Gestion Presupuesto	Iniciar Ejercicio Presupuestario	
117			#117	Gestion Presupuesto	Iniciar Ejercicio Presupuestario	
116			#116	Gestion Presupuesto	Iniciar Ejercicio Presupuestario	

Ilustración 67: Buzón de entrada en el proceso bpm

Fuente: Propia

Procedemos a iniciar un nuevo periodo planificación en el sistema transaccional.

Periodos Presupuestarios				
Ejercicio Presupuestario				
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2014-12-09				
PERIODO		INICIO	FIN	ESTADO
	Planificacion	2014-01-02	2014-11-20	✔
	Aprobacion	2014-07-01	2015-01-15	✘
	Consolidacion	2014-08-13	2014-11-16	✘

Ilustración 68: Inicio del periodo planificación

Fuente: Propia

Internamente el flujo proceso se modifica y la tarea actual es la de Registrar Requerimientos.

Esta tarea está asignada al usuario cortega (director de departamento), y se visualizara en la bandeja de entrada del proceso la tarea a realizar con el mismo número de caso.

Tenemos que tomar en cuenta que el proceso no se perdió para el usuario anterior, solamente cambio de usuario, esta es una característica muy importante de processmaker, que nos facilita la participación de varios usuarios en un proceso.

HOME								
Cases		Open	Actions	Read	Unread	All	Category	Process
#	Summ...	Case N...	Case	Process	Task	Sent By		
167		#167	Gestion Presupuesto	Registro de Requerimientos	morales, maria (m...		All Categories	All Processes
165		#165	Gestion Presupuesto	Generar Proforma Presupuestaria	ortega, cosme (cor...			
144		#144	Planificacion	Registrar Requerimientos	, Administrator (a...			
143		#143	Planificacion	Registrar Requerimientos	, Administrator (a...			
142		#142	Planificacion	Registrar Requerimientos	, Administrator (a...			
141		#141	Planificacion	Registrar Requerimientos	, Administrator (a...			

Ilustración 69: Buzón de entrada en el proceso bpm

Fuente: Propia

Una característica muy importante de bpm es que se puede enviar notificaciones personalizadas a los demás usuarios informando que se ha cumplido la tarea.

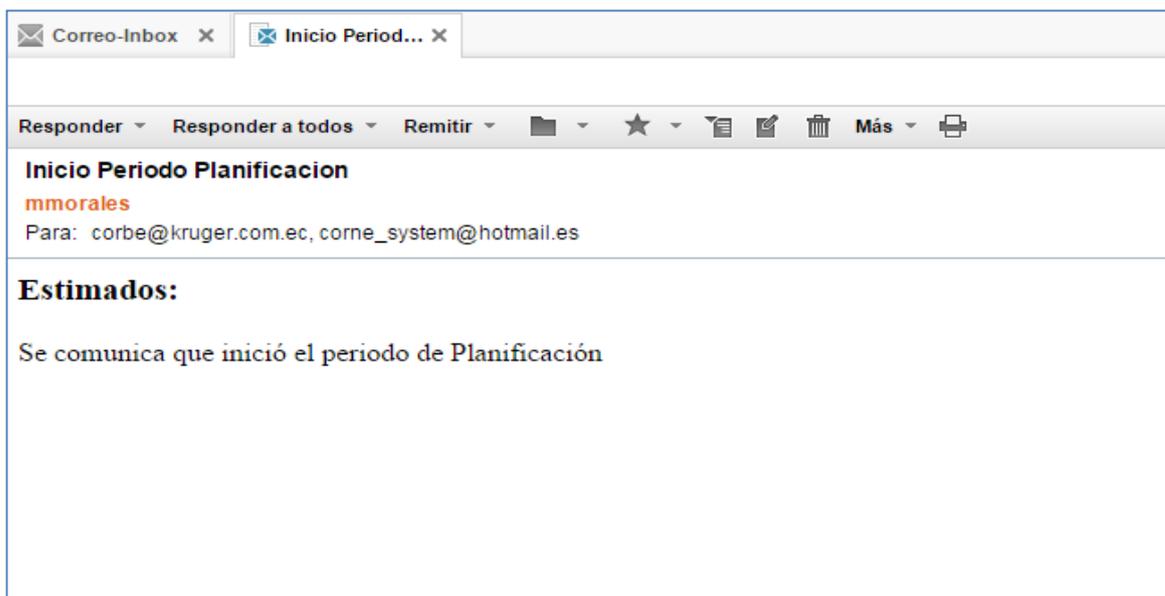


Ilustración 70: Notificación del inicio del periodo

Fuente: Propia

A continuación el usuario cortega ingresa al sistema transaccional y se desplegará el menú correspondiente al periodo planificación.

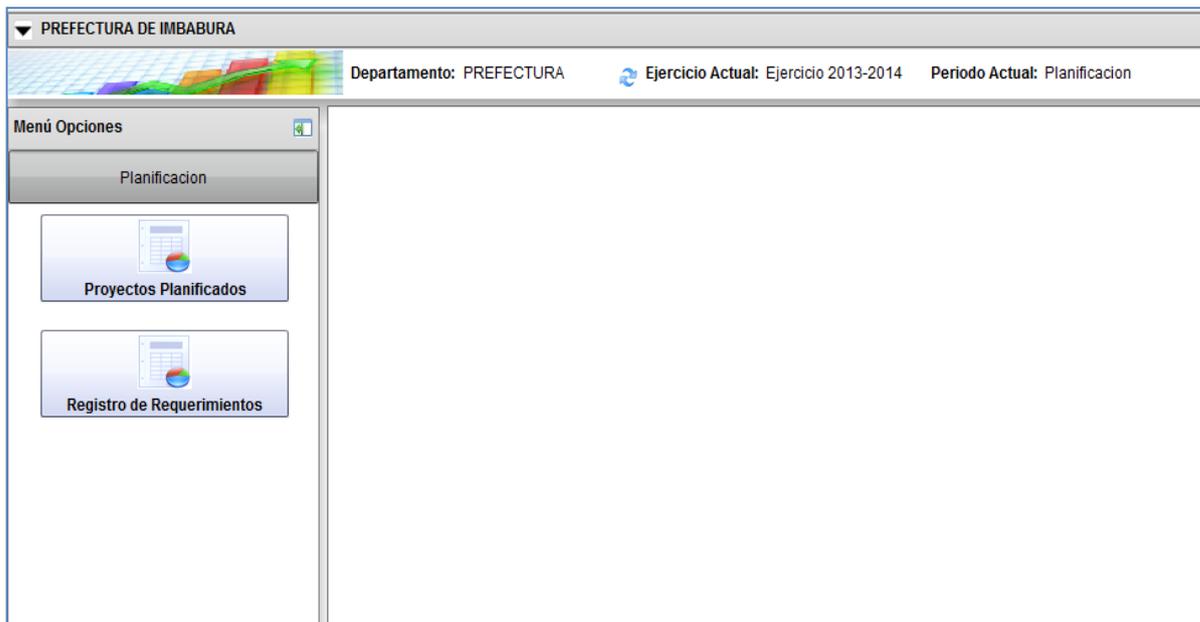


Ilustración 71: Espacio de trabajo del sistema transaccional en el periodo planificación

Fuente: Propia

Posteriormente el mismo usuario realiza el registro de requerimientos.

Proyectos Planificados		TRANSACCIONES	NUEVA TRANSACCION	
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> </div>				
<p>▼ Transacción</p>				
PROYECTO:	Nombre de Programa...			
DESCRIPCION:	compra de servicios			
FECHA REGISTRO:	2014-12-09	Monto Transaccion:	30	
		Monto Presupuestado Proyecto:	230	
		Monto Estimado Proyecto:	600	
<p>Detalle</p> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <input type="text"/> + ... </div>				
Codigo	Item Presupuestario	Cantidad	Monto Unitario (\$)	Monto Total (\$)
623940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10
625940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10
622940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10

Ilustración 72: Registro de Requerimientos

Fuente: Propia

Inmediatamente después de esta operación le llega la notificación al usuario mmorales sobre la tarea asignada: Finalizar Periodo Aprobación

HOME							
Cases		Open	Actions	Read	Unread	All	
#	Summ...	Case N...	Case	Process	Task	Sent By	
167			#167	Gestion Presupuesto	Finalizar Periodo Planificacion	ortega, cosme (cor...	
166			#166	Gestion Presupuesto	Finalizar Ejercicio Presupuestario	morales, maria (m...	
119			#119	Gestion Presupuesto	Iniciar Ejercicio Presupuestario		
117			#117	Gestion Presupuesto	Iniciar Ejercicio Presupuestario		
116			#116	Gestion Presupuesto	Iniciar Ejercicio Presupuestario		
115			#115	Gestion Presupuesto	Iniciar Ejercicio Presupuestario		

Ilustración 73: Buzón de entrada en el proceso bpm

Fuente: Propia

Seguidamente el usuario mmorales ingresa al sistema transaccional, finaliza el periodo planificación y consecuentemente inicia el periodo aprobación como siguiente tarea asignada.

Periodos Presupuestarios				
Ejercicio Presupuestario				
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2014-12-09				
PERIODO		INICIO	FIN	ESTADO
	Aprobacion	2014-07-01	2015-01-15	✓
	Consolidacion	2014-08-13	2014-11-16	✗
	Planificacion	2014-01-02	2014-11-20	✗

Ilustración 74: Inicio del periodo aprobación

Fuente: Propia

Y le llega la notificación al usuario cortega que debe realizar la aprobación de requerimientos, esta llega tanto al buzón de entrada del processmaker como al mail asignado.

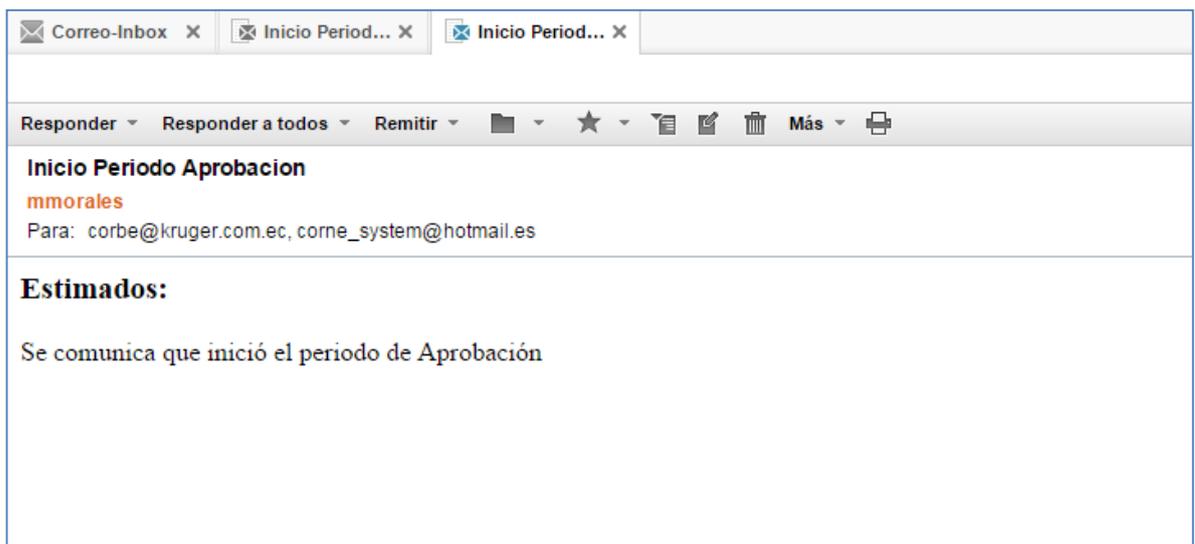


Ilustración 75: Notificación sobre el periodo de aprobación

Fuente: Propia

El usuario cortega ingresa al sistema transaccional y aprueba los requerimientos aprobados.

Proyectos Aprobados		Aprobacion Requerimientos		
  <input type="checkbox"/> Filtros				
Proyecto Presupuestario : programa de Rescate				
Monto Presupuestado : 2 \$ Monto Estimado : 800 \$				
Descripcion	Fecha Registro	Monto	Aprobada	Detalle
compra de materiales	2014-09-10 11:47:21	0		
adquisicion de servicios basicos	2014-11-05 11:41:36	2		
compra de sillas	2014-11-05 11:46:00	0		
compra de escritorios	2014-11-05 11:47:29	0		

Ilustración 76: Aprobación de Requerimientos

Fuente: Propia

Según el proceso bpm, existe una regla de decisión que determina a cual tarea elegir, esta regla de enrutamiento funciona a través del valor de una variable.

Esta variable podrá ser enviada desde el sistema transaccional o desde un mail gerencial, en el cual el usuario tome la decisión directamente desde su correo.

Buscar en el correo 

Carpetas 

Bandeja de entrada 955

Correo no deseado 6

Borradores 8

Enviados

Eliminados

[Nueva carpeta](#)

Aprobacion Requerimientos

 GpiPresupuesto (cornetaleroo@gmail.com) [Agregar a contactos](#) 15:17 ▶

Para: corne_system@hotmail.es ▼

Se han bloqueado partes del mensaje por tu seguridad.

[Mostrar el contenido](#) | cornetaleroo@gmail.com es de confianza. [Mostrar siempre el contenido.](#)

Estimado:

Desea aprobar los siguientes proyectos:

Proyecto	Monto
programa de Rescate	2

[Aprobar](#)

[No Aprobar](#)

Ilustración 77: Mail de aprobación del proyecto

Fuente: Propia

Según el flujo si selecciona *No Aprobar* el proceso se direccionara a la tarea: Modificar Requerimiento, pero si selecciona *Aprobar* el proceso se direcciona a la tarea finalizar periodo aprobación.

El usuario mmorales ingresa al sistema transaccional a finalizar el periodo aprobación y seguidamente a iniciar el periodo consolidación.

Periodos Presupuestarios				
Ejercicio Presupuestario				
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2014-12-09				
PERIODO		INICIO	FIN	ESTADO
	Consolidacion	2014-08-13	2014-11-16	✔
	Planificacion	2014-01-02	2014-11-20	✘
	Aprobacion	2014-07-01	2015-01-15	✘

Ilustración 78: Inicio del periodo consolidación

Fuente: Propia

La notificación sobre el inicio de un nuevo periodo llega al usuario cortega y la nueva tarea asignada que es la de asignación de Partidas Presupuestarias.

Código	Nombre
1.3.03	Tasas Diversas
1.3.04	Contribuciones
1.4	VENTADE BIENES Y SERVICIOS
1.4.01	Ventas de Derivados de Petróleo
1.4.02	Ventas de Productos y Materiales
1.4.03	Ventas no Industriales
1.4.09	Débitos por Impuesto al Valor Agregado
1.7	RENTAS DE INVERSIONES Y MULTAS
1.7.01	Rentas de Inversiones
1.7.02	Rentas por Arrendamientos de Bienes
1.7.03	Intereses por Mora
1.7.04	Multas

Ilustración 79: Asignación de partidas presupuestarias

Fuente: Propia

Una vez completada esta operación el proceso avanza a la siguiente tarea que está asignada al mismo usuario, y esta es Generar Proforma Presupuestaria.

Proforma Presupuestaria		
Departamento		
Proforma Departamento: PREFECTURA		Monto Total Presupuestado: 0 \$
		<input type="radio"/> General <input checked="" type="radio"/> Resumido
Generar		
Código	Partida Presupuestaria	Monto
1.0	INGRESOS CORRIENTES	0
1.1	IMPUESTOS	0
5.0	GASTOS CORRIENTES	0
5.1	GASTOS EN PERSONAL	0
7.0	GASTOS DE INVERSIÓN	0
7.3	BIENES Y SERVICIOS PARA INVERSIÓN	0

Ilustración 80: Generación de la proforma presupuestaria

Fuente: Propia

Una vez completada esta acción se notificara al usuario mmorales sobre la siguiente tarea a realizar, y esta es la de finalizar el periodo de consolidación.

El usuario mmorales ingresa al sistema transaccional y finaliza el periodo de consolidación.

Periodos Presupuestarios			
Ejercicio Presupuestario			
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2014-12-09			
PERIODO		INICIO	FIN
Planificacion		2014-01-02	2014-11-20
Aprobacion		2014-07-01	2015-01-15
Consolidacion		2014-08-13	2014-11-16

Ilustración 81: Finalización del periodo consolidación

Fuente: Propia

CAPÍTULO VI

6 CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

- 1) Con la implementación del sistema GpiPresupuesto se automatizó la elaboración de la proforma presupuestaria y el plan anual de compras a nivel de departamentos y a nivel de institución.
- 2) Mediante BPM se pudo realizar una administración, seguimiento y control sobre el proceso presupuestario
- 3) A través de los servicios web que proporciona la herramienta bpm se integraron el sistema transaccional y el gestor de procesos bpm
- 4) Con el uso de la metodología Scrum se desarrolló mucho más rápido y eficiente el sistema GpiPresupuesto ya que el objetivo de una metodología ágil es terminar un producto.
- 5) Mediante el uso de la herramienta RIA es posible diseñar una interfaz amigable e interactiva, y su desarrollo es mucho más rápido y eficiente que con otras herramientas.

6.2 RECOMENDACIONES

- 1) Es necesario que se implemente un módulo de administración de usuarios y asignación de perfiles para tener un mayor control sobre las opciones del sistema
- 2) Se recomienda implementar un módulo de administración de departamentos internos, complementario del módulo propuesto de administración de usuarios.
- 3) Se recomienda el uso de una metodología ágil para el desarrollo de los sistemas y/o de las tesis.
- 4) Se recomienda el uso de tecnología RIA para el desarrollo de la interfaz de los sistemas a desarrollar.

BIBLIOGRAFÍA

AJBDSOFT. (2013). *AJBDSOFT*. Recuperado el 2013, de AJBDSOFT:
<http://www.ajpdsoft.com/>

Calderon A, S. D. (2010). *Metodologia Agil*. Universidad de Trujillo.

Carrizo, F. (2011). *Mundo Flex*. AlfaOmega.

Club-BPM. (16 de 01 de 2010). *Club del BPM*. Recuperado el 16 de 01 de 2010, de Club del BPM: <http://www.club-bpm.com>

Colosa, I. (s.f.). *Process Maker*. Recuperado el 11 de 12 de 2014, de <http://www.processmaker.com/advanced-dashboards>

Foundation, T. A. (21 de 01 de 2014). *The Apache Software Foundation*. Recuperado el 21 de 01 de 2014, de The Apache Software Foundation: <http://www.apache.org/>

Gabriellebet. (11 de 01 de 2014). *Desarrollo de Aplicaciones*. Recuperado el 21 de 01 de 2014, de Desarrollo de Aplicaciones:
<http://gabriellebet.wordpress.com/2014/01/11/unidad-i-aplicaciones-ria-rich-internet-applications/>

Gonzales, B. (2009). *Zend Framework Manual en Español*.

Gonzales, B. (2009). *Zend Framework Manual en Español*.

GPI. (20 de 11 de 2013). *Gobierno Provincial de Imbabura*. Recuperado el miercoles de noviembre de 2013, de Gobierno Provincial de Imbabura: <http://www.imbabura.gob.ec/>

Group, T. P. (21 de 01 de 2014). *PHP Hypertext Preprocessor*. Recuperado el 21 de 01 de 2014, de PHP Hypertext Preprocessor: <http://php.net/>

Guanotasig J, S. E. (2010). *Roles de Scrum*.

Hidalgo L, T. J. (2010). *Metodologias Agiles*.

Ireland, A. S. (21 de 01 de 2014). *Adobe Flash Builder 4.7 Premium*. Recuperado el 21 de 01 de 2014, de Adobe Flash Builder 4.7 Premium:
<http://www.adobe.com/es/products/flash-builder.html>

Ireland, A. S. (21 de 01 de 2014). *Flex*. Recuperado el 21 de 01 de 2014, de Flex: <http://www.adobe.com/es/products/flex.html>

KIRAN GARIMELLA, M. L. *BPM*.

Management. (2010). *Management* , 36.

Martinez, R. (21 de 01 de 2014). *Portal en español sobre PostgreSQL*. Recuperado el 21 de 01 de 2014, de Portal en español sobre PostgreSQL: <http://www.postgresql.org.es>

PF, P. (2012). *Diseño y Dimensaiconamiento de plantas de tratamiento convencional de aguas residuales con utilizacion de software libre*. Ibarra - Ecuador: UTN.

PHP. (2013). *PHP Oficial Page*. Recuperado el 2013, de PHP Oficial Page: <http://www.php.net/manual/es/>

Pyme, E. (15 de 01 de 2013). *Easy Pyme*. Recuperado el 15 de 01 de 2013, de Easy Pyme: <http://www.easypyme.es/tecnologiaRIA>

Robison Elijah, C. A. (2011). *Learning Flex 4*.

Rouse, M. (2013). *techtarget*. Obtenido de techtarget: <http://searchcio-midmarket.techtarget.com>

Software, D. (21 de 01 de 2014). *Toad Data Modeler Data Modeling Tool for Database Desing*. Recuperado el 21 de 01 de 2012, de Toad Data Modeler Data Modeling Tool for Database Desing: <http://www.quest.com/toad-data-modeler/>

Solutions Bista. (s.f.). *Bista Solutions*. Recuperado el 10 de 12 de 2014, de <http://www.bistasolutions.com/business-solutions/bpm/overview>

Technologies, Z. (21 de 01 de 2014). *Zend Framework*. Recuperado el 21 de 01 de 2014, de Zend Framework: <http://framework.zend.com/>

6.3 ANEXOS

6.3.1 MANUAL DE USUARIO

Manual de Usuario

1 LOGIN

Para acceder al sistema transaccional GpiPresupuesto ingresamos a la siguiente dirección url

<http://gpigestion.gob.ec/GpiGestionBackend/public/bin-debug/GpiGestionFrontend.php>

La primera pantalla que se despliega es la del login, el usuario debe ingresar el nombre y la contraseña.

Acceso a GPI GESTION

gpi Gobierno Provincial de Imbabura
gpi gestión
Suite de Aplicaciones Institucionales

Usuario:

Clave:

INGRESAR

SUB DIRECCIÓN DE GESTIÓN TECNOLÓGICA
© Copyright GPI
2014

El menú de las funcionalidades del sistema se mostrara dependiendo del rol que tenga el usuario logeado.

Los roles se asignan a los usuarios dependiendo de su rango y su cargo.

En este sistema manejaremos tres tipos de roles.

- Usuario de Departamento
- Director de Departamento
- Usuario de Presupuesto

El usuario de presupuesto será el encargado de administrar los datos presupuestarios, necesarios para que los usuarios puedan realizar las transacciones presupuestarias.

2 DEPARTAMENTOS:

En esta pantalla se podrá visualizar los departamentos y organizaciones internas del gobierno provincial.

Menú Opciones		Departamentos		
Datos del Sistema				
 Departamentos		CODIGO	NOMBRE	INSTITUCION
		001	PREFECTURA	Prefectura de Imbabura
		201	Planificacion	Prefectura de Imbabura
		CGB001	Gobernacion	Prefectura de Imbabura
		PRE001	Presidencia de la República del Ecuador	Prefectura de Imbabura
		GADCI001	Ilustre Municipio de Ibarra	Prefectura de Imbabura
		0	Ninguna	Prefectura de Imbabura
		243assd	adasd	Prefectura de Imbabura
		VPRE001	Vicepresidencia de la República del Ecuador	Prefectura de Imbabura
		SPRE001	Secretaría de la Presidencia de la República	Prefectura de Imbabura
		SNPD001	Secretaría Nacional de Planificación y Desarrollo	Prefectura de Imbabura
		SGAP001	Secretaría General de la Administración Pública	Prefectura de Imbabura
		MCDS001	Ministerio de Coordinación y Desarrollo Social	Prefectura de Imbabura

3 ÍTEMS PRESUPUESTARIOS

Son los productos que se van a presupuestar, estos son administrados desde el sistema de inventario.

Código	Nombre
3526074419	ACCESORIOS NO METALICOS Y CON RECUBRIMIENTO METALICO PARA GORRAS DEL EJERCITO NACIONAL
35260434	ACCESORIOS PARA ALUMBRADO DE PLASTICO
441901021	ACCESORIOS PARA APILADORAS
412780431	ACCESORIOS PARA SOLDADO A TOPE, DE ACERO INOXIDABLE
381402011	ACCESORIOS PARA SOLDADO A TOPE, DE ACERO INOXIDABLE
375700017	ACCESORIOS PARA TUBERIA
412780011	ACCESORIOS PARA TUBOS O CANOS DE HIERRO O ACERO, DE FUNDICION NO MALEABLE: EMPALMES, CODOS O COLLARES.
381401917	ACCESORIOS PARA TUBOS O CANOS DE HIERRO O ACERO, DE FUNDICION NO MALEABLE: EMPALMES, CODOS O COLLARES.
4111590110	ACCESORIOS Y TUBERIA DE COBRE PARA REFRIGERACION Y AIRE ACONDICIONADO
48243019	ACCESS POINTS
461110111	ACCONDICIONADORES DE VALVULA ELECTRICOS DE CORRIENTE CONTINUA
04110003212	ACEITE DE SOYA (SOJA) EN BRUTO, SIN GOMA O NO
36111001173	ACEITE DE SOYA REFINADO Y SUS FRACCIONES, PERO SIN MODIFICAR QUIMICAMENTE
352607811	ACEITE DE ALCANFOR BLANCO
216503011	ACEITE DE ALMENDRA DE PALMA O BABASU EN BRUTO
352607416194	ACEITE DE ALMENDRA DE PALMA O BABASU REFINADO Y SUS FRACCIONES, PERO SIN MODIFICAR QUIMICAMENTE
2799400348	ACEITE DE ALMENDRAS

4 PARTIDAS PRESUPUESTARIAS

Las partidas presupuestarias representan el conjunto de campos, compuestos por códigos y descripciones que se utiliza para ordenar sistemáticamente la información presupuestaria de ingresos y gastos, normalmente es la unión o interrelación de los catálogos y clasificadores presupuestarios.

Las partidas presupuestarias estarán clasificadas de manera jerárquica en base al CLASIFICADOR PRESUPUESTARIO DE INGRESOS Y GASTOS DEL SECTOR PÚBLICO generado por el Ministerio de Finanzas de Ecuador.

Código	Nombre
1.0	INGRESOS CORRIENTES
1.1	IMPUESTOS
1.1.01	Sobre la Renta, Utilidades y Ganancias de Capital
1.1.01.01	A la Renta Global
1.1.01.02	A la Utilidad por la Venta de Predios Urbanos
1.1.01.03	A la Utilidad por la Venta de Predios Rurales
1.2	SEGURIDAD SOCIAL
1.2.01	Aportes a la Seguridad Social
1.2.01.01	Aportes al Seguro de Pensiones
1.3	TASAS Y CONTRIBUCIONES
1.3.01	Tasas Generales
1.3.02	Tasas Portuarias y Aeroportuarias
1.3.03	Tasas Diversas
1.3.04	Contribuciones

Para crear una nueva partida presupuestaria se selecciona una partida (partida presupuestaria padre) y seleccionamos la primera opción del menú superior.



Y nos despliega la siguiente ventana.

Nueva Partida Presupuestaria x

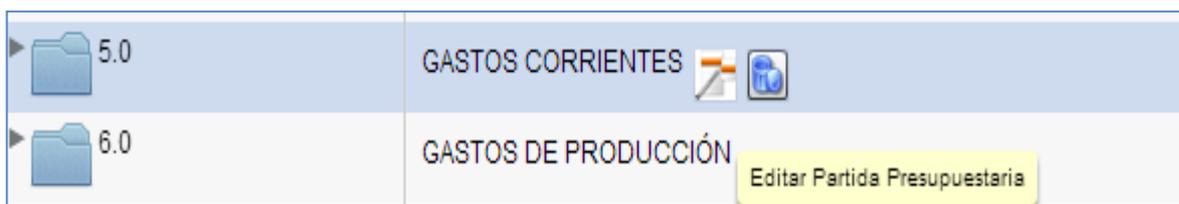
CODIGO:

NOMBRE:

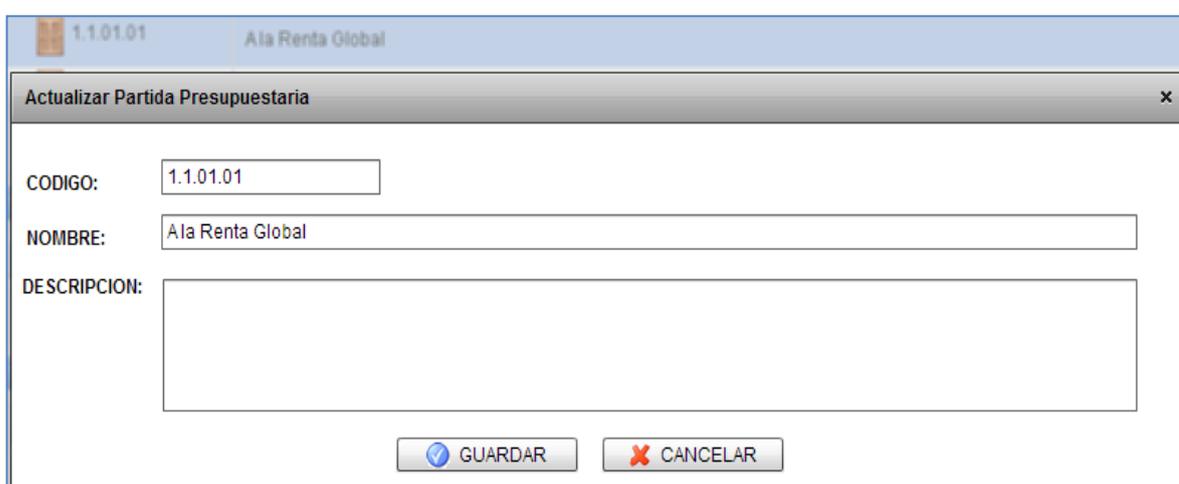
DESCRIPCION:

Seleccionamos la opción de guardar y la nueva partida 1.1.01.04 se añade a la colección de partidas presupuestarias hijas de la partida 1.1.01

Para editar una partida presupuestaria seleccionamos la opción de edición que se muestra cuando pasamos el mouse sobre una partida presupuestaria.

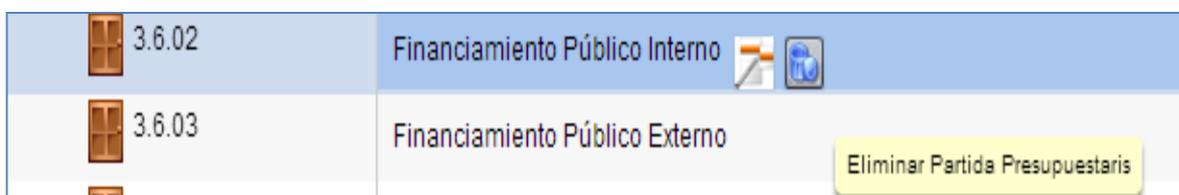


Y nos despliega la siguiente ventana.

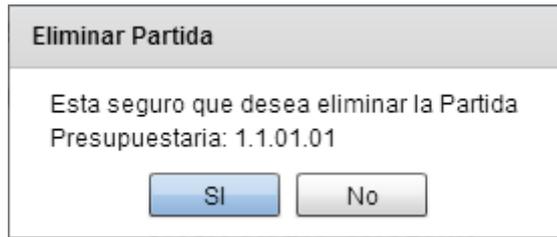


Modificamos la información y seleccionamos la opción guardar.

Para eliminar seleccionamos la opción de borrar partida, que se hace visible cuando movemos el mouse sobre la partida presupuestaria.



Y se desplegará la siguiente opción.



Seleccionamos la opción SI, y la partida presupuestaria se removerá.

Existe la funcionalidad de expandir y contraer la partida presupuestaria padre y poder visualizar las partidas hijas, esto se lo consigue seleccionando la partida presupuestaria y presionando sobre el botón Expandir Registro



3.6	FINANCIAMIENTO PÚBLICO
3.6.01	Colocación de Títulos y Valores
3.6.02	Financiamiento Público Interno
3.6.03	Financiamiento Público Externo
3.6.04	Descuentos en Títulos y Valores
3.6.05	Créditos de Proveedores Internos

También existe la funcionalidad de expandir y contraer todos los registros, esto se lo consigue seleccionando la opción del menú superior Expandir Todo



Si esta contraído los registros se expandirán, pero si están expandidos todos o algunos registros se contraerán todos los registros.

Código	Nombre
▶ 1.0	INGRESOS CORRIENTES
▶ 2.0	INGRESOS DE CAPITAL
▶ 3.0	INGRESOS DE FINANCIAMIENTOS
▶ 5.0	GASTOS CORRIENTES
▶ 6.0	GASTOS DE PRODUCCIÓN
▶ 7.0	GASTOS DE INVERSIÓN
▶ 8.0	GASTOS DE CAPITAL
▶ 9.0	APLICACIÓN DEL FINANCIAMIENTO

5 EJERCICIOS PRESUPUESTARIOS

Un ejercicio presupuestario es un lapso de tiempo en el cual se realiza un proceso presupuestario, consta de una fecha de inicio y una de fin

Existen dos visualizaciones del menú de opciones, la de lectura de registros y la de edición; en la primera solamente se podrá visualizar los ejercicios presupuestarios y su información.

Ejercicios Presupuestarios				
CODIGO	NOMBRE	INICIO	FIN	ESTADO
E-2012	Ejercicio 2012-2013	2012-09-03	2013-08-30	⊖
E-2013	Ejercicio 2013-2014	2013-09-02	2014-08-29	⊖
E-2014	Ejercicio 2014-2015	2014-09-01	2015-08-31	⊖

Para acceder al modo de edición seleccionamos la opción del menú superior y nos mostrara la siguiente pantalla.

Ejercicios Presupuestarios					
  					
CODIGO	NOMBRE	INICIO	FIN	ESTADO	
E-2012	Ejercicio 2012-2013	2012-09-03	2013-08-30	Inactivo	
E-2013	Ejercicio 2013-2014	2013-09-02	2014-08-29	Inactivo	
E-2014	Ejercicio 2014-2015	2014-09-01	2015-08-31	Inactivo	

En este modo de visualización podremos crear, editar y activar/inactivar ejercicios presupuestarios.

Para activar un ejercicio presupuestario se presiona en la descripción del estado, si está inactivo se activa y si este se encuentra activo se inactiva, como regla de negocio solamente puede haber un ejercicio presupuestario activo.

Para crear un ejercicio presupuestario seleccionamos en la opción de añadir y nos mostrará la siguiente ventana.

Nuevo Ejercicio Presupuestario x

NOMBRE:

FECHA INICIO:  FECHA FIN: 

Asignamos la fecha de inicio, la fecha de fin; y seleccionamos la opción guardar.

El nuevo ejercicio presupuestario se añade a la lista de registros, pero no está registrado en la base de datos, para ello seleccionamos la opción de guardar del menú superior y la vista cambia a modo visualización, si no estamos seguros de registrar cancelamos los últimos cambios, y los registros aparecen como inicialmente se mostró.

Para editar un ejercicio presupuestario seleccionamos la opción de edición que se encuentra en la parte derecha del registro y nos aparece la siguiente ventana.

Editar Ejercicio Presupuestario ×

NOMBRE:

FECHA INICIO:  FECHA FIN: 

6 PERIODOS PRESUPUESTARIOS

Un periodo presupuestario es un lapso de tiempo en el cual se realiza una fase del proceso presupuestario, forma parte encuentra dentro de un ejercicio presupuestario.

La administración de los periodos presupuestarios es similar a la de los ejercicios presupuestarios, teniendo dos modos de pantalla, visualización y edición.

Periodos Presupuestarios			
Ejercicio Presupuestario			
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2015-01-08			
PERIODO	INICIO	FIN	ESTADO
Planificacion	2014-01-02	2014-11-20	-
Aprobacion	2014-07-01	2015-01-15	-
Consolidacion	2014-09-13	2014-12-26	-

La activación o inactivación de un periodo presupuestario se lo realiza activando el modo de edición y seleccionando la descripción del estado del ejercicio presupuestario.

Periodos Presupuestarios				
Ejercicio Presupuestario				
Ejercicio Presupuestario Actual: Ejercicio 2013-2014 Fecha Actual: 2015-01-08				
PERIODO	INICIO	FIN	ESTADO	
Planificacion	2014-01-02	2014-11-20	Inactivo	 
Aprobacion	2014-07-01	2015-01-15	Inactivo	 
Consolidacion	2014-09-13	2014-12-26	Inactivo	 

En este modo de edición podemos crear, actualizar y remover registros.

Para agregar un nuevo periodo presupuestario seleccionamos la opción de añadir que está ubicada en el menú superior y para la edición de un periodo presupuestario seleccionamos la opción de edición que se encuentra en la parte derecha del registro.

Editar Periodo Presupuestario x

NOMBRE: Planificacion

FECHA INICIO:  **FECHA FIN:** 

Para la eliminación del registro seleccionamos la opción de remover que está ubicada en la parte derecha del registro

ELIMINAR

Esta Seguro que desea Eliminar el Periodo Presupuestario: Planificacion

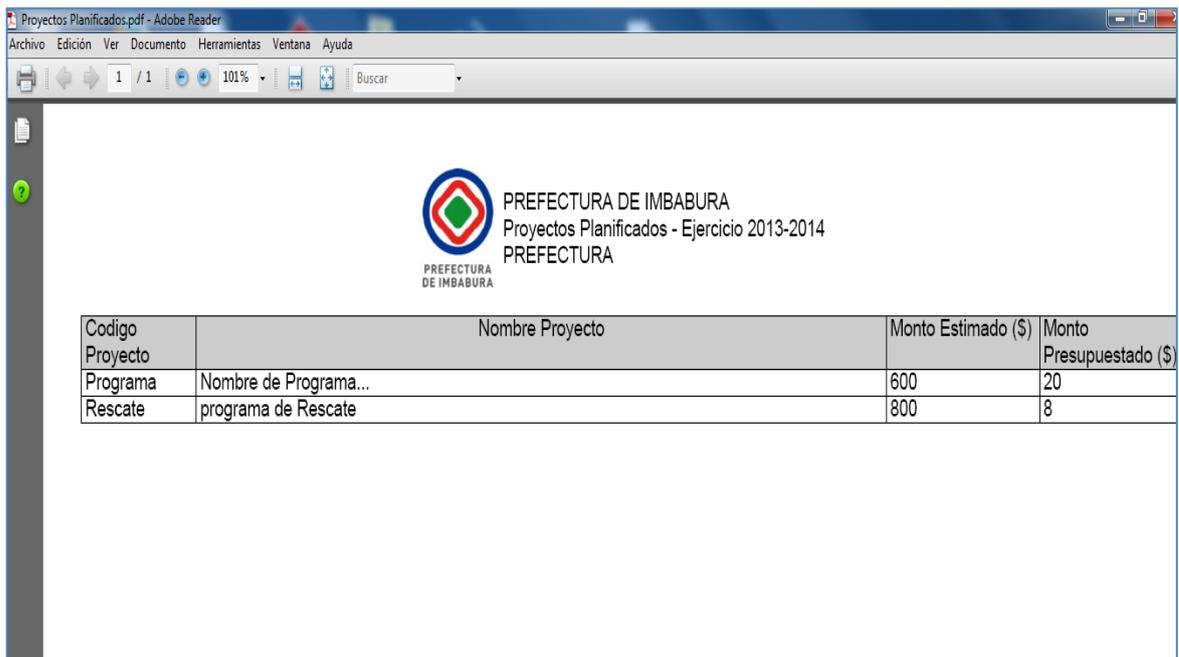
Y para guardar en base todos los cambios realizados seleccionamos la opción de guardar si estamos seguros de los cambios. Caso contrario seleccionamos la opción cancelar cambios y los registros volverán a su estado inicial.

7 PLANIFICACIÓN PRESUPUESTARIA

En el proceso de planificación tendremos dos opciones, un reporte de los proyectos planificados y el registro de requerimientos.

Codigo	Nombre	TIPO	Monto Estimado	Monto Presupuestado	Transacciones
Programa	Nombre de Programa...	Programa	600	20	 
Rescate	programa de Rescate	Programa	800	8	 

El reporte nos permite exportar la información en formato pdf o en gráficos estadísticos.



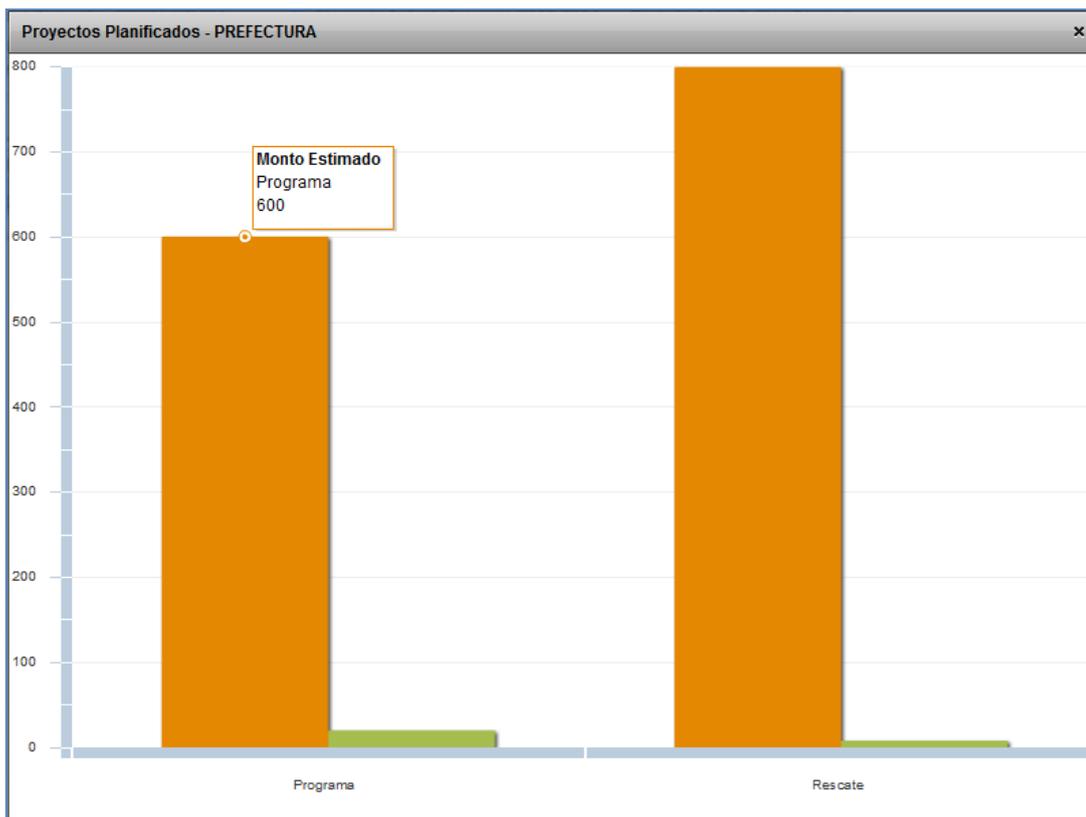
Projectos Planificados.pdf - Adobe Reader

Archivo Edición Ver Documento Herramientas Ventana Ayuda

1 / 1 101% Buscar

 **PREFECTURA DE IMBABURA**
Proyectos Planificados - Ejercicio 2013-2014
PREFECTURA

Codigo Proyecto	Nombre Proyecto	Monto Estimado (\$)	Monto Presupuestado (\$)
Programa	Nombre de Programa...	600	20
Rescate	programa de Rescate	800	8



Desde esta pantalla podremos crear una nueva transacción presupuestaria relacionada al proyecto presupuestario o también visualizar las transacciones relacionadas al mismo.

Proyectos Planificados		TRANSACCIONES		
<input style="float: left;" type="button" value="+"/> <input style="float: left; margin-left: 20px;" type="button" value="↻"/> <input style="float: left; margin-left: 20px;" type="checkbox" value="Filtros"/>				
Proyecto Presupuestario: Nombre de Programa...				
Monto Presupuestado: 200 \$ Monto Estimado: 600 \$				
Descripción	Fecha Registro	Monto	Detalle	
compra de mesas	2014-11-05 12:08:09	20	<input type="button" value="Detalle"/>	
compra de materiales	2014-11-05 12:07:08	0	<input type="button" value="Detalle"/>	
adquisición de materiales	2014-11-05 12:06:21	180	<input type="button" value="Detalle"/>	

Para visualizar del detalle de una transacción presupuestaria seleccionamos la opción derecha del registro, y para crear una nueva transacción seleccionamos la opción de añadir ubicada en la parte superior de la ventana.

Proyectos Planificados | TRANSACCIONES | NUEVA TRANSACCION

▼ Transacción

PROYECTO: Nombre de Programa...

DESCRIPCION: compra de servicios

FECHA REGISTRO: 2014-12-09 Monto Transaccion: 0 Monto Presupuestado Proyecto: 200 Monto Estimado Proyecto: 600

Detalle +

Codigo	Item Presupuestario	Cantidad	Monto Unitario (\$)	Monto Total (\$)

En la nueva transacción debemos ingresar la descripción y la fecha de registro, además debemos se agregan los ítems presupuestarios, es decir los productos de inventario que se van a requerir para la ejecución del proyecto;

Si conocemos el código del producto ingresamos y se añadirá a la lista de productos. Pero si no conocemos, seleccionamos la opción de buscar productos y se abrirá la lista de productos de inventario.

Productos x

Filtros

papel

<input type="checkbox"/>	CODIGO	NOMBRE	CATEGORIA
<input type="checkbox"/>	622630012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL DE EMPAPELAR PAREDES	62263.00.1
<input type="checkbox"/>	612940012	SERVICIOS COMERCIALES AL POR MAYOR PRESTADOS A COMISION O POR CONTRATO, DE PAPEL	61294.00.1
<input type="checkbox"/>	612630012	SERVICIOS COMERCIALES AL POR MAYOR PRESTADOS A COMISION O POR CONTRATO, DE PAPEL DE EMPAPEL	61263.00.1
<input type="checkbox"/>	611940011	SERVICIOS COMERCIALES AL POR MAYOR, EXCEPTO LOS PRESTADOS A COMISION O POR CONTRATO, DE PAP	61194.00.1
<input type="checkbox"/>	611630012	SERVICIOS COMERCIALES AL POR MAYOR, EXCEPTO LOS PRESTADOS A COMISION O POR CONTRATO, DE PAP	61163.00.1
<input type="checkbox"/>	623630012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL DE EMPAPELAR PAREDES	62363.00.1
<input type="checkbox"/>	621940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL EN TIENDAS NO ESPECIALIZADAS	62194.00.1
<input checked="" type="checkbox"/>	623940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	62394.00.1
<input checked="" type="checkbox"/>	625940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	62594.00.1
<input checked="" type="checkbox"/>	622940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	62294.00.1
<input type="checkbox"/>	621630013	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL DE REVESTIMIENTO DE SUELOS EN TIENDAS NO ESPEC	62163.00.1
<input type="checkbox"/>	625630012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL DE EMPAPELAR PAREDES	62563.00.1
<input type="checkbox"/>	621630012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL DE EMPAPELAR PAREDES EN TIENDAS NO ESPECIALIZAI	62163.00.1
<input type="checkbox"/>	547500211	SERVICIOS DE COLOCACION DE PAPEL Y OTROS REVESTIMIENTOS FLEXIBLES DE PAREDES	54750.02.1
<input type="checkbox"/>	854000013	SERVICIOS DE EMPAQUETADO EN BOLSAS Y SACOS DE PAPEL	85400.00.1
<input type="checkbox"/>	881400013	SERVICIOS DE LA INDUSTRIA DE PRODUCTOS DEL PAPEL	88140.00.1
<input type="checkbox"/>	881400012	SERVICIOS DE LA INDUSTRIA DEL PAPEL	88140.00.1

Seleccionamos los productos que se requieran y seleccionamos la opción de aceptar, y estos registros se añadirán a la lista de productos requeridos

Proyectos Planificados		TRANSACCIONES	NUEVA TRANSACCION																																																						
<div style="border: 1px solid #ccc; padding: 5px;"> </div>																																																									
<p>▼ Transacción</p> <p>PROYECTO: Nombre de Programa...</p> <p>DESCRIPCION: compra de servicios</p> <p>FECHA REGISTRO: 2014-12-09 <input type="text"/> <input type="calendar"/></p> <p>Monto Transaccion: 30 Monto Presupuestado Proyecto: 230 Monto Estimado Proyecto: 600</p>																																																									
<p>Detalle <input type="text"/> <input type="button" value="+"/> <input type="button" value="..."/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Codigo</th> <th style="width: 55%;">Item Presupuestario</th> <th style="width: 10%;">Cantidad</th> <th style="width: 10%;">Monto Unitario (\$)</th> <th style="width: 10%;">Monto Total (\$)</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td>623940012</td> <td>SERVICIOS COMERCIALES AL POR MENOR DE PAPEL</td> <td style="text-align: center;">1</td> <td style="text-align: center;">10</td> <td style="text-align: center;">10</td> <td style="text-align: center;"></td> </tr> <tr> <td>625940012</td> <td>SERVICIOS COMERCIALES AL POR MENOR DE PAPEL</td> <td style="text-align: center;">1</td> <td style="text-align: center;">10</td> <td style="text-align: center;">10</td> <td style="text-align: center;"></td> </tr> <tr style="background-color: #e6f2ff;"> <td>622940012</td> <td>SERVICIOS COMERCIALES AL POR MENOR DE PAPEL</td> <td style="text-align: center;">1</td> <td style="text-align: center;">10</td> <td style="text-align: center;">10</td> <td style="text-align: center;"></td> </tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table>				Codigo	Item Presupuestario	Cantidad	Monto Unitario (\$)	Monto Total (\$)		623940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10		625940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10		622940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10																															
Codigo	Item Presupuestario	Cantidad	Monto Unitario (\$)	Monto Total (\$)																																																					
623940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10																																																					
625940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10																																																					
622940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	1	10	10																																																					

Para cada ítem presupuestado se ingresan la cantidad y el monto unitario para cada registro y el monto total se calcula automáticamente para cada registro.

El monto de la transacción se calcula sumando los montos totales de cada uno de los ítems presupuestados, y sumados todos los montos de las transacciones pertenecientes a un proyecto se calcula el monto presupuestado del proyecto; este último monto no de superar el monto estimado del proyecto.

En el periodo de planificación cada departamento deberá realizar el registro de los requerimientos necesarios para la ejecución de los proyectos planificados

8 APROBACIÓN PRESUPUESTARIA

En la aprobación presupuestaria el director de departamento puede aprobar y/o modificar los requerimientos presupuestados en el periodo de planificación, para la ejecución de este proceso el periodo de aprobación debe estar activo y el usuario logeado debe tener asociado el rol de director de departamento.

En este proceso encontramos dos opciones: proyectos aprobados y aprobación de requerimientos

Productos Agrupados								
Código	Nombre	Cantidad	Monto Unitario (\$)	Monto Total (\$)		Estado A. P.	A. P.	Partida Presupuestaria
429940023	BANDEJAS PARA PAPELES	7	6	42	<input type="checkbox"/>	✓		7.3.07.01
3526040175	CARDURAN 4 MG 28 COMPRIMIDOS	5	5	25	<input type="checkbox"/>	✓		6.3.03
334200111	SILLAS DE RUEDAS	10	6	60	<input type="checkbox"/>	✓		6.3.01
371130011	SILLAS DE VRIDGE	12	6	72	<input type="checkbox"/>	✓		6.3.01
4391300133	CAJA TERMICA CON CAPACIDAD NETA	32	16	512	<input type="checkbox"/>	✓		6.3.04
333500011	ANILLO DE PISTON (RIN)	10	10	100	<input type="checkbox"/>	✓		7.5.01
3811102140	BUTACA, APOYABRAZOS FIJOS, EN PO	5	12	60	<input type="checkbox"/>	✓		7.3.07.01

Es necesario asociar cada ítem presupuestado a una partida presupuestaria, esta operación se la puede realizar individualmente o masivamente, mediante la opción ubicada en la parte derecha del registro o seleccionando la opción asignación múltiple ubicada en la parte superior de la pantalla.

Código	Nombre
1.3.03	Tasas Diversas
1.3.04	Contribuciones
1.4	VENTA DE BIENES Y SERVICIOS
1.4.01	Ventas de Derivados de Petróleo
1.4.02	Ventas de Productos y Materiales
1.4.03	Ventas no Industriales
1.4.09	Débitos por Impuesto al Valor Agregado
1.7	RENTAS DE INVERSIONES Y MULTAS
1.7.01	Rentas de Inversiones
1.7.02	Rentas por Arrendamientos de Bienes
1.7.03	Intereses por Mora
1.7.04	Multas

Cuando todos los ítems presupuestados han sido asociados una partida presupuestaria el proyecto presupuestario pasa a estado de consolidado.

Internamente se agrupan los registros a nivel de partida presupuestaria, según las asignaciones y se puede obtener un reporte seleccionando la opción C.P. (Consolidación Partida)

Consolidación por Proyecto		Consolidación por Partida	
Proyecto Presupuestario			
Proyecto Presupuestario: Administracion de Licencias			
Partidas Presupuestarias			
Código	Partida Presupuestaria	Monto (\$)	
7.3.07.01	Desarrollo de Sistemas Informáticos	102	
6.3.03	Traslados, Instalaciones, Viáticos y Subsistencias	25	
6.3.01	Servicios Básicos	132	
6.3.04	Instalación, Mantenimiento y Reparaciones	512	
7.5.01	Obras de Infraestructura	100	

En la consolidación el director de departamento puede obtener un reporte de ítems presupuestarios asignados la partida presupuestaria a nivel de departamento y a nivel general.

Este reporte se lo denomina PLAN ANUAL DE COMPRAS, y este consta de la partida presupuestaria, el código y el nombre del ítem presupuestario, la cantidad de unidades asignadas, el monto unitario y total de cada uno de los registros

Datos Generales					
Pac Departamento: PREFECTURA		Monto Total Presupuestado: 1567 \$			
Partida Presupuestaria	Codigo	Item Presupuestario	Cantidad	Monto Unitario (\$)	Monto Total (\$)
7.3.07.01	429940023	BANDEJAS PARA PAPELES	7	6	364
6.3.03	3526040175	CARDURAN 4 MG 28 COMPRIMIDOS	5	5	260
6.3.01	334200111	SILLAS DE RUEDAS	10	6	520
6.3.01	371130011	SILLAS DE VRIDGE	12	6	624
6.3.04	4391300133	CAJA TERMICA CON CAPACIDAD NETA DE 6 L	32	16	1664
7.5.01	333500011	ANILLO DE PISTON (RIN)	10	10	520
7.3.07.01	3811102140	BUTACA, APOYABRAZOS FIJOS, EN POLIURETANO, ESPALDA	5	12	260
6.1.01	439490013	ANILLOS DE JUNTAS	2	20	104
5.3.02.06	623940012	SERVICIOS COMERCIALES AL POR MENOR DE PAPEL	2	110	104
5.3.02.06	961420013	SERVICIOS AUXILIARES N.C.P. PARA LA PRODUCCION DE P	1	100	52
8.4.04.01	3526010279	APARATOS DE FREGAR SUELOS	9	20	468
7.3.08.03	352600312	ALOPURINOL	3	52	156

