

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

**DISEÑO DE UN TEST ADAPTATIVO INFORMATIZADO UTILIZANDO LA
METODOLOGÍA DE ESTIMACIÓN BAYESIANA PARA EVALUAR EL
CONOCIMIENTO DE LOS ESTUDIANTES DE LA UNIVERSIDAD TÉCNICA DEL
NORTE, APLICADO AL 1ER Y 2DO NIVEL DE LOS ESTUDIANTES DE LA FICA.**

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

Autor:

Edwin Marcelo Anrango Imbaquingo

Director:

MSc. Granda Gudiño Pedro David

Ibarra - Ecuador

2022



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

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

1. IDENTIFICACIÓN DE LA OBRA

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

| DATOS DE CONTACTO | |
|-----------------------------|---|
| CÉDULA DE IDENTIDAD: | 1004182968 |
| APELLIDOS Y NOMBRES: | ANRANGO IMBAQUINGO EDWIN MARCELO |
| DIRECCIÓN: | FEDERICO PÁEZ Y CARLOS TROYA, SAN PABLO - OTAVALO, IMBABURA |
| EMAIL: | emanrangoi@utn.edu.ec |
| TELÉFONO MÓVIL: | 0996092698 |

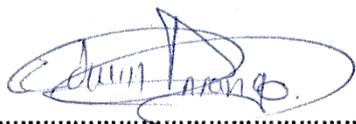
| DATOS DE LA OBRA | |
|--------------------------------|--|
| TÍTULO: | DISEÑO DE UN TEST ADAPTATIVO INFORMATIZADO UTILIZANDO LA METODOLOGÍA DE ESTIMACIÓN BAYESIANA PARA EVALUAR EL CONOCIMIENTO DE LOS ESTUDIANTES DE LA UNIVERSIDAD TÉCNICA DEL NORTE, APLICADO AL 1ER Y 2DO NIVEL DE LOS ESTUDIANTES DE LA FICA. |
| AUTOR (ES): | ANRANGO IMBAQUINGO EDWIN MARCELO |
| FECHA: | 16/03/2022 |
| PROGRAMA: | <input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO |
| TITULO POR EL QUE OPTA: | INGENIERO EN SISTEMAS COMPUTACIONALES |
| ASESOR /DIRECTOR: | MSC. PEDRO GRANDA |

2. CONSTANCIAS

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

Ibarra, a los 16 días del mes de marzo de 2022

EL AUTOR:



Nombre: Edwin Marcelo Anrango Imbaquingo

CC: 1004182968

CERTIFICADO DEL DIRECTOR DE TRABAJO DE GRADO

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICADO DEL DIRECTOR

Por medio del presente yo MSc. Pedro Granda, certifico que el Sr. Edwin Marcelo Anrango Imbaquingo, portador de la cédula de ciudadanía Nro. 1004182968. Ha trabajado en el desarrollo del proyecto de tesis **“DISEÑO DE UN TEST ADAPTATIVO INFORMATIZADO UTILIZANDO LA METODOLOGÍA DE ESTIMACIÓN BAYESIANA PARA EVALUAR EL CONOCIMIENTO DE LOS ESTUDIANTES DE LA UNIVERSIDAD TÉCNICA DEL NORTE, APLICADO AL 1ER Y 2DO NIVEL DE LOS ESTUDIANTES DE LA FICA.”**, previo a la obtención del título de Ingeniería en Sistemas Computacionales, lo cual ha realizado en su totalidad con responsabilidad.

Es todo cuanto puedo certificar en honor a la verdad.

Atentamente



Firmado digitalmente por:
**PEDRO DAVID
GRANDA GUDIÑO**

MSc. Pedro Granda
DIRECTOR DE TESIS

Dedicatoria

Dedicada a las noches de desvelo y los días de cansancio, a las 40 horas semanales de trabajo, las horas de estudio y las pocas horas de sueño que duraron algunos años, a la perseverancia y las metas por cumplir y a la motivación diaria para continuar en pie de lucha, aun después de los fracasos.

A mi madre que me demostró que no existen imposibles y mi padre que me enseñó que, ante momentos difíciles solo podemos reír y continuar, ambos quienes me enseñaron que el trabajo honesto y honrado existe.

A Katerin Valenzuela, por los años que compartimos juntos, el apoyo incondicional y la confianza brindada, y la promesa de no decepcionarte.

A las grandes amistades de años, a Maya Sánchez por el apoyo y la amistad incondicional sin importar distancias, también a Anthony y Cinthia buenas amistades que me apoyaron durante y después de la universidad.

A la memoria de todos mis abuelos: mis abuelos paternos quienes fallecieron durante la pandemia, mi abuela materna que me inculco buenos valores desde temprana edad, una persona respetada y querida por todos, siempre te extrañare.

Agradecimientos

Muy agradecido con mi familia por el apoyo brindado en todos estos años, a mis padres Fabiola y Antonio por la paciencia, los consejos, la guía ética y moral, a mi hermano Edison por sus consejos, mis hermanas Mónica y Daniela por su motivación, gracias a todos quienes me apoyaron para superarme.

A Katerin Valenzuela quien se convirtió en mi mano derecha, brindo todo su apoyo para la culminación de esta etapa, con cariño, presión y paciencia que le caracteriza.

Agradezco también la confianza brindada por quienes fueron mis jefes, en especial a la Sra. Alba Paz quien reconoció más de una vez mis esfuerzos, y me brindo muchas facilidades para trabajar y estudiar, dándome la oportunidad de crecer profesionalmente.

A mi tutor, MSc. Pedro Granda, por el apoyo y seguimiento que brindo para la culminación del presente trabajo de titulación, de la misma manera gracias al apoyo del MSc. Marcelo Jurado por las herramientas, los conocimientos y las rectificaciones que fueron pilares en el desarrollo del proyecto.

A mis opositores, MSc. Marco Pusdá y MSc. Daisy Imbaquingo, por su tiempo y observaciones que brindaron para el cumplimiento correcto de los objetivos del presente proyecto.

A todos los docentes de la Facultad de Ingeniería en Ciencias Aplicadas, por su apoyo paciencia y consejos, al club de programación, un lugar tranquilo donde estudiar, a los compañeros y amigos en el transcurso de la formación académica.

Tabla de Contenido

| | |
|---|----|
| INTRODUCCIÓN | 15 |
| El Problema | 15 |
| Antecedentes | 15 |
| Situación Actual..... | 15 |
| Planteamiento del Problema..... | 16 |
| Objetivos | 17 |
| Objetivo General | 17 |
| Alcance | 18 |
| Justificación..... | 20 |
| Capítulo I | 23 |
| 1. Marco teórico | 23 |
| 1.1. Definiciones Básicas | 23 |
| 1.1.1. Inteligencia Artificial | 23 |
| 1.1.2. Probabilidad Condicional | 24 |
| 1.1.3. Test Adaptativo Informatizado | 24 |
| 1.1.4. Teoría de Respuesta al Ítem..... | 25 |
| 1.2. TAI basado en Estimación Bayesianas..... | 29 |
| 1.2.1. Teorema de Bayes y Redes Bayesianas | 29 |
| 1.3. Relación de la TRI y el TAI en la forma de evaluar | 34 |
| 1.3.1. Componentes básicos del TAI | 34 |
| 1.3.2. Algoritmo del TAI | 35 |
| 1.3.3. ¿Cómo aporta la TRI al TAI? | 41 |
| 1.3.4. Evaluaciones por TAI..... | 42 |
| 1.4. Herramientas para la implementación | 44 |
| 1.4.1. Plataforma tecnológica | 45 |
| 1.4.2. El Gestor PostgreSQL | 49 |
| 1.4.3. Marco de trabajo..... | 50 |
| Capítulo II | 54 |
| 2. Desarrollo del prototipo..... | 54 |
| 2.1. Planteamiento del proyecto | 54 |
| 2.1.1. Implementación de Scrum | 54 |
| 2.1.2. Planificación del Sprint..... | 57 |
| 2.1.3. Análisis de Requerimientos..... | 57 |
| 2.2. Especificaciones del algoritmo adaptativo | 65 |
| 2.2.1. Casos de uso..... | 65 |
| 2.2.2. Arquitectura del sistema..... | 67 |

| | |
|---|-----|
| 2.2.3. Modelado y prototipado | 69 |
| 2.3. Desarrollo del software..... | 76 |
| 2.3.1. Sprint 0 | 77 |
| 2.3.2. Sprint 1 | 81 |
| 2.3.3. Sprint 2 | 84 |
| 2.3.4. Sprint 3 | 87 |
| 2.3.5. Sprint 4 | 90 |
| Capítulo III | 93 |
| 3. Validación de resultados | 93 |
| 3.1. Análisis de funcionamiento | 93 |
| 3.1.1. Rendimiento del TAI | 93 |
| 3.1.2. Usabilidad del prototipo | 97 |
| 3.1.3. Cumplimiento de objetivos | 105 |
| 3.2. Interpretación de resultados. | 109 |
| 3.3. Análisis de impacto | 112 |
| CONCLUSIONES | 113 |
| RECOMENDACIONES | 114 |
| BIBLIOGRAFÍA..... | 115 |

Índice de Figuras

| | |
|---|----|
| Fig. 1. Diagrama de planteamiento del problema..... | 17 |
| Fig. 2. Arquitectura de funcionamiento..... | 20 |
| Fig. 3. Ejemplo de varias CCI..... | 28 |
| Fig. 4. RB con dos nodos..... | 31 |
| Fig. 5. RB con dos nodos..... | 32 |
| Fig. 6. Diagrama de flujo TAI..... | 36 |
| Fig. 7. Herramientas seleccionadas | 45 |
| Fig. 8. Magic Quadrant para servicios de infraestructura y plataforma en la nube..... | 48 |
| Fig. 9. Tiempos de ejecución | 49 |
| Fig. 10. Procesos Scrum..... | 52 |
| Fig. 11. Responsabilidades del Product Owner..... | 55 |
| Fig. 12. Responsabilidades del Scrum Master | 55 |
| Fig. 13. Responsabilidades del Team Scrum | 56 |
| Fig. 14. Procesos Scrum..... | 56 |
| Fig. 15. Casos de uso | 66 |
| Fig. 16. Flujograma de procesos sin automatización..... | 67 |
| Fig. 17. Flujograma de procesos actual | 67 |
| Fig. 18. Arquitectura de funcionamiento..... | 68 |
| Fig. 19. Descripción de la arquitectura | 68 |
| Fig. 20. Arquitectura MVC..... | 69 |
| Fig. 21. Estructura de un ítem..... | 70 |
| Fig. 22. Prototipo presentación del ítem..... | 70 |
| Fig. 23. Modelo conceptual del sistema | 71 |
| Fig. 24. Modelo de navegación del participante | 71 |
| Fig. 25. Modelo de navegación Evaluador | 72 |
| Fig. 26. Navegación del super usuario evaluador..... | 73 |
| Fig. 27. Flujograma del TAI..... | 74 |
| Fig. 28. Flujograma TAI-APP | 74 |
| Fig. 29. Métodos Bayes | 75 |
| Fig. 30. Método inicializador Bayes..... | 75 |
| Fig. 31. Algoritmo selección de ítems..... | 76 |
| Fig. 32. Estructura del Sprint..... | 77 |
| Fig. 33. Estructura de la base de datos..... | 80 |
| Fig. 34. Configuración del Data Source..... | 80 |
| Fig. 35. Login del evaluador | 83 |
| Fig. 36. Ingreso al TAI..... | 83 |
| Fig. 37. Gestión de evaluaciones | 86 |
| Fig. 38. Banco de ítems | 87 |
| Fig. 39. Panel administración TAI..... | 89 |
| Fig. 40. Resumen respuestas | 90 |
| Fig. 41. Evaluaciones disponibles | 92 |
| Fig. 42. Análisis del sistema..... | 93 |
| Fig. 43. Modelo de respuestas | 94 |
| Fig. 44. Modelo de respuestas corregido | 95 |
| Fig. 45. Pregunta 5 análisis de los ítems..... | 95 |
| Fig. 46. Uso del CPU virtual..... | 96 |
| Fig. 47. Uso del CPU optimizado | 96 |
| Fig. 48. Tráfico en red respecto a las peticiones al servidor..... | 97 |
| Fig. 49. Resultado pregunta 1 (SUS) | 98 |

| | |
|--|-----|
| Fig. 50. Resultado pregunta 2 (SUS) | 99 |
| Fig. 51. Resultado pregunta 3 (SUS) | 99 |
| Fig. 52. Resultado pregunta 4 (SUS) | 100 |
| Fig. 53. Resultado pregunta 5 (SUS) | 100 |
| Fig. 54. Resultado pregunta 6 (SUS) | 101 |
| Fig. 55. Resultado pregunta 7 (SUS) | 101 |
| Fig. 56. Resultado pregunta 8 (SUS) | 102 |
| Fig. 57. Resultado pregunta 9 (SUS) | 102 |
| Fig. 58. Resultado pregunta 10 (SUS) | 103 |
| Fig. 59. Pregunta 6 de la encuesta..... | 103 |
| Fig. 60. Pregunta 3 estilos visuales..... | 104 |
| Fig. 61. Notas por carreras | 109 |
| Fig. 62. Gráfico escala SUS..... | 112 |

Índice de Tablas

| | |
|---|----|
| Tabla 1.1 Características de la TRI y la TCT..... | 26 |
| Tabla 1.2 Parámetros usados en APP-TAI..... | 42 |
| Tabla 1.3 Ventajas TAI-Test Tradicional | 43 |
| Tabla 1.4 Roles Scrum..... | 51 |
| Tabla 1.5 Eventos de Scrum..... | 52 |
| Tabla 1.6 Artefactos Scrum..... | 53 |
| Tabla 2. 1 Roles Scrum..... | 54 |
| Tabla 2. 2 Planificación de los Sprint | 57 |
| Tabla 2. 3 Historia de usuario Nro. 1 | 58 |
| Tabla 2. 4 Historia de usuario Nro. 2..... | 58 |
| Tabla 2. 5 Historia de usuario Nro. 3..... | 58 |
| Tabla 2. 6 Historia de usuario Nro. 4..... | 59 |
| Tabla 2. 7 Historia de usuario Nro. 5..... | 59 |
| Tabla 2. 8 Historia de usuario Nro. 6..... | 59 |
| Tabla 2. 9 Historia de usuario Nro. 7..... | 60 |
| Tabla 2. 10 Historia de usuario Nro. 8..... | 60 |
| Tabla 2. 11 Historia de usuario Nro. 9..... | 60 |
| Tabla 2. 12 Historia de usuario 10..... | 61 |
| Tabla 2. 13 Historia de usuario 11..... | 61 |
| Tabla 2. 14 Historia de usuario 12..... | 61 |
| Tabla 2. 15 Historia de usuario 13..... | 61 |
| Tabla 2. 16 Historia de usuario 14..... | 62 |
| Tabla 2. 17 Product Backlog | 62 |
| Tabla 2. 18 Requisito Funcional - RF.Apl.01 | 63 |
| Tabla 2. 19 Requisito Funcional - RF.Apl.02..... | 63 |
| Tabla 2. 20 Requisito Funcional - RF.Apl.03 | 63 |
| Tabla 2. 21 Requisito Funcional - RF.Apl.04 | 63 |
| Tabla 2. 22 Requisito Funcional - RF.Sal.01 | 64 |
| Tabla 2. 23 Requisito Funcional - RF.Com.01..... | 64 |
| Tabla 2. 24 Requisito No Funcional - RNF.Arq.01..... | 64 |
| Tabla 2. 25 Requisito No Funcional - RNF.Arq.02..... | 65 |
| Tabla 2. 26 Requisito No Funcional - RNF.Usa.01 | 65 |
| Tabla 2. 27 Requisito No Funcional - RNF.Usa.02..... | 65 |
| Tabla 2. 28 Actividades de los roles..... | 65 |
| Tabla 2. 29 Sprint 0..... | 77 |
| Tabla 2. 30 Sprint backlog | 77 |
| Tabla 2. 31 Planificación sprint | 78 |
| Tabla 2. 32 Revisión sprint..... | 78 |
| Tabla 2. 33 Revisión sprint..... | 81 |
| Tabla 2. 34 Sprint 1..... | 81 |
| Tabla 2. 35 Sprint backlog | 81 |
| Tabla 2. 36 Planificación sprint | 82 |
| Tabla 2. 37 Revisión sprint..... | 82 |
| Tabla 2. 38 Revisión sprint..... | 83 |
| Tabla 2. 39 Sprint 2..... | 84 |
| Tabla 2. 40 Sprint backlog | 84 |
| Tabla 2. 41 Planificación sprint | 85 |
| Tabla 2. 42 Revisión sprint..... | 85 |
| Tabla 2. 43 Revisión sprint..... | 87 |

| | |
|--|-----|
| Tabla 2. 44 Sprint 3..... | 88 |
| Tabla 2. 45 Sprint backlog | 88 |
| Tabla 2. 46 Planificación sprint | 88 |
| Tabla 2. 47 Revisión sprint..... | 89 |
| Tabla 2. 48 Revisión sprint..... | 90 |
| Tabla 2. 49 Sprint 4..... | 90 |
| Tabla 2. 50 Sprint backlog | 91 |
| Tabla 2. 51 Planificación sprint | 91 |
| Tabla 2. 52 Revisión sprint..... | 91 |
| Tabla 2. 53 Revisión sprint..... | 92 |
| Tabla 3. 1 Resultados SUS | 98 |
| Tabla 3. 2 Objetivo 1 | 106 |
| Tabla 3. 3 Objetivo 2..... | 106 |
| Tabla 3. 4 Objetivo 3..... | 107 |
| Tabla 3. 5 Ejemplo de resultados FECYT | 107 |
| Tabla 3. 6. Resultados de la facultad | 108 |
| Tabla 3. 7. Notas altas y bajas | 109 |
| Tabla 3. 8 Escala de puntuaciones | 110 |
| Tabla 3. 9 Resultados de las encuestas aplicando cálculo SUS | 110 |
| Tabla 3. 10 SUS-Preguntas impares..... | 111 |
| Tabla 3. 11 SUS-Preguntas pares..... | 111 |

Resumen

La obtención de resultados fiables por medio de una evaluación tradicional genera una problemática que pone en riesgo la calidad de la educación, por tal razón este proyecto tiene como objetivo el diseño de un Test Adaptativo Informatizado (TAI) utilizando la metodología de estimación Bayesiana, hace uso de una extensa base bibliográfica para la obtención de los métodos y herramientas, elementos fundamentales para el proceso de desarrollo del aplicativo web. La investigación partió del proyecto “Un modelo de evaluación adaptativa mediante razonamiento probabilístico como apoyo al proceso enseñanza aprendizaje”, siguiendo ese lineamiento este proyecto brinda una nueva propuesta en la forma de evaluar los conocimientos de los estudiantes en base a un método probabilístico condicionado y aplicando la técnica del modelo logístico de dos parámetros correspondiente a la teoría de respuesta al ítem. El aplicativo web es construido utilizando JSF PrimeFaces y PostgreSQL como base de datos, el marco de trabajo aplicado es Scrum, se publica en una máquina virtual con los servicios de Google Cloud Platform y se utilizó como medio para la obtención de los datos sobre el nivel de conocimiento de la asignatura de TIC en el primer y segundo nivel de las siete carreras de ingeniería en la Facultad de Ingeniería en Ciencias Aplicadas. Las evaluaciones aplicadas muestran variaciones mínimas respecto al conocimiento del participante y la nota promedio de este. En conclusión, el Test diseñado pudo adaptarse a las características del participante y el sistema demuestra ser eficiente para la obtención de niveles estimados sobre el conocimiento del tema, fortaleciendo los métodos de evaluación y generando altos niveles de satisfacción y confianza por parte de los participantes que miran con optimismo este tipo de evaluaciones.

Palabras clave: Test Adaptativo Informatizado, Teoría de Respuesta al Ítem, Teorema de Bayes, Tecnologías de la Información y Comunicación.

ABSTRACT

Obtaining reliable results through a traditional evaluation generates a problem that puts the quality of education at risk, for this reason, this project aims to design a Computerized Adaptive Test (CAT) using the Bayesian estimation methodology, using of an extensive bibliographic base to get the methods and tools, fundamental elements for the development process of the web application. This research work was based on the project "An adaptive evaluation model through probabilistic reasoning as a support for the teaching-learning process"; following this guideline, this project provides a new proposal in the way of evaluating the students' knowledge based on a conditioned probabilistic method and applying the two-parameters logistic model technique corresponding to Item Response Theory. The web application is built using JSF PrimeFaces and PostgreSQL as a database, the applied framework is Scrum, it is published in a virtual machine with Google Cloud Platform services and was used as a means to obtain data on the level of knowledge of the ICT subject in the first and second levels of the seven engineering careers in the Facultad de Ingeniería en Ciencias Aplicadas (Faculty of Engineering in Applied Sciences). The applied evaluations show minimal variations in regard to the participant's knowledge and their average grade. In conclusion, the designed test was able to adapt to the characteristics of the participants and the system proves to be efficient in obtaining estimated levels of knowledge of the subject, strengthening the evaluation methods and generating high levels of satisfaction and confidence by the participants who are optimistic at this type of evaluation.

Keywords: Computerized Adaptive Test, Item Response Theory, Bayes Theorem, Information and Communication Technologies.

INTRODUCCIÓN

El Problema

Antecedentes

Los Test Adaptativos Informatizados (TAI) son una de las formas más importantes de evaluación adaptativa que se desarrollan en la actualidad, su aplicabilidad varía de acuerdo con el campo de estudio y el método de inferencia para obtener un mejor resultado. Comparte una estrecha relación con la Teoría de Respuesta al Ítem (TRI), la cual sirve de base para la estimación del nivel de conocimiento del estudiante, por medio del modelo de respuesta y la estimación de máxima verosimilitud, mismas que son escogidas por su simplicidad a la hora de implementarse.

Los TAI forman parte de una variedad de sistemas inteligentes en educación que han evolucionado y se han adaptado a los requerimientos de la enseñanza, están relacionados con los sistemas de tutoriales inteligentes, y los sistemas de enseñanza probabilísticos, si bien la tecnología predictiva que este usa no es tan nueva, el uso y aplicación de estos modelos han aumentado en los últimos años.

El empleo del criterio condicionado a la probabilidad de la respuesta a un ítem mediante un TAI, han mostrado confiabilidad en los resultados, ya que permite al sistema suministrar ítems de acuerdo con el nivel de conocimiento del estudiante (Ramón, 2016). De esta forma los resultados de la evaluación brindan al evaluador datos más certeros sobre el nivel real de conocimiento del tema que aplicó a sus estudiantes, facilitando el proceso de retroalimentación.

No obstante, existe una variedad de técnicas o métodos para automatizar el proceso de evaluación adaptativa tales como: diagnóstico basado en inferencia difusa, evaluación basada en la distancia semántica, clasificación de los perfiles del estudiante usando redes neuronales, evaluación basada en mapas conceptuales y diagnóstico del nivel de conocimientos mediante redes bayesianas. (Jiménez Sánchez, 2009).

Situación Actual

Las pruebas escritas son diseñadas de manera manual los días previos a la evaluación, dependiendo del docente tendrá un nivel de dificultad y cantidad de preguntas que cumplan con sus requerimientos, los resultados obtenidos con la evaluación reflejarían

el nivel de conocimiento de los estudiantes, en procesos no automatizados toda la información obtenida llegaría en su mayoría hasta este punto, en procesos automatizados se podría tabular y obtener un promedio general, además podría facilitar los temas que generen problemas a los estudiantes, así el docente podría reforzar los temas para que exista un mejor entendimiento, pero si un estudiante demuestra conocer un tema correctamente, pero falla una evaluación, que tan probable es que conozca el tema si falla otra evaluación.

El modelo usado comúnmente al momento de evaluar sigue siendo a mano “papel y lápiz”, pese a que, los recursos tecnológicos actuales deberían brindar a la gente mejores herramientas para facilitar sus actividades, sin embargo, no existe disponible una herramienta que permita al evaluador conocer el estado real de comprensión de sus estudiantes, llegando al punto de creer que los avances tecnológicos más están enfocados al entretenimiento o el ocio.

El papel del docente a la hora de enseñar un tema tan útil como el uso de las TIC, nota un problema en los resultados: notas disparejas, conceptos fáciles con respuestas incorrectas y preguntas complejas con respuestas correctas, es posible que exista trampa, poco interés en el tema, o un elemento que genere incomodidad al estudiante que provoco que conteste de forma errada, por lo tanto, se puede decir que los test tradiciones no reflejan su conocimiento real del tema.

Planteamiento del Problema

Las evaluaciones tradicionales muestran resultados distintos a los obtenidos mediante un TAI, su amplia base de ítems no permite que el estudiante pueda memorizar todas las respuestas para realizar trampa, en tal caso estudiar los conceptos le resultaría mejor, también sería un impedimento para las respuestas al azar, mientras tanto en una prueba tradicional mediante hojas de papel con una cantidad igual de ítems, los resultados podrían devolver información incorrecta debido a que el estudiante desconocía el tema, pero contesto y acertó a la mayoría, y el test tradicional refleje que desconocía las preguntas fáciles pero conocía las preguntas difíciles.

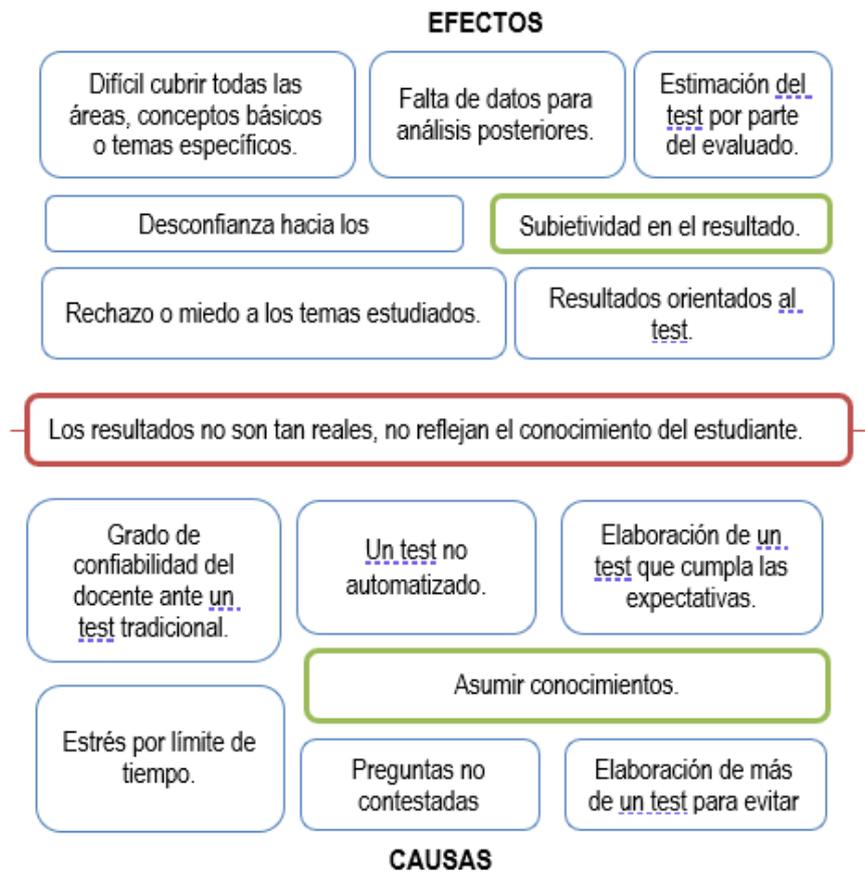


Fig. 1. Diagrama de planteamiento del problema.
Fuente: Propia.

Objetivos

Objetivo General

Diseño de un Test Adaptativo Informatizado utilizando la metodología de estimación bayesiana para evaluar el conocimiento de los estudiantes de la Universidad Técnica Del Norte, aplicado al 1er y 2do nivel de los estudiantes de la FICA.

Objetivos Específicos

1. Elaborar una base teórica que sustente el modelo aplicado para el diseño del Test Adaptativo Automatizado desde la estimación bayesiana y de una base de datos que ofrezca la interacción con los usuarios.
2. Diseñar un Test Adaptativo Informatizado utilizando la metodología de estimación Bayesiana para evaluar conocimientos en la asignatura de TIC dirigida a estudiantes de la Universidad Técnica Del Norte, aplicando SCRUM como marco de trabajo para el desarrollo.

3. Evaluar los resultados de la investigación propuesta.

Alcance

El proyecto planteado tiene como finalidad diseñar un prototipo funcional que implemente un algoritmo adaptativo, aplicativo web que integra las funciones de un Test Adaptativo Informatizado mediante el uso de técnicas de Inteligencia Artificial específicamente de la estimación probabilística bayesiana para evaluar el conocimiento en TIC, no se realizará un análisis comparativo experimental con otras técnicas de inteligencia artificial ni tendrá la capacidad para ejercer otras evaluaciones con características distintas.

El diseño del algoritmo adaptativo se implementó en un prototipo o aplicativo web que lleva el nombre de TAI-App y se enfocó en evaluar conocimientos sobre TIC a estudiantes de los primeros niveles, dentro de la Facultad de Ingeniería en Ciencias Aplicadas de la Universidad Técnica del Norte en Ibarra - Ecuador y sus resultados aportarán información confiable para la toma de decisiones, retroalimentación y mejor enseñanza del tema, siguiendo el lineamiento del proyecto del CUICYT “Un modelo de evaluación adaptativa mediante razonamiento probabilístico como apoyo al proceso enseñanza aprendizaje” propuesto por el MSc. Ing. Marcelo Jurado docente investigador dentro del instituto de postgrados de la Universidad Técnica del Norte.

El sistema TAI-App fue creado originalmente con tres módulos siguiendo los mismos pasos que investigaciones previas (Attorresi & Lozzia, 2012), la cual describe el proceso básico para diseñar un TAI: módulo inicial, módulo del test, módulo de gestión de la información.

Módulo inicial. - Cuenta con la bienvenida y las instrucciones a seguir, también administra el control de acceso, la gestión de vistas para avanzar entre pantallas, cómo responder a los ítems, imposibilidad de omitir la respuesta o de retroceder para modificarla y la retroalimentación sobre la resolución de los ítems.

Módulo de gestión del test. - La aplicación del test basado en la estimación de Bayes, el uso de la teoría de respuesta al ítem, la presentación de cada pregunta vendrá de acuerdo con la respuesta de la pregunta anterior, los resultados obtenidos definen el punto de parada para finalizar el test, cada pregunta tiene un nivel o peso estimado y se almacenará en un Banco de ítems (BI) dentro de una base de datos.

Módulo de gestión de la información. - Para generar reportes, usuarios, y creación de grupos. Además de revisar los datos de la sesión de evaluación y del test y los datos del evaluado.

El diseño del prototipo sigue los pasos del marco de trabajo que propone la metodología SCRUM, metodología elegida debido a la facilidad para resolver situaciones en que no se está entregando al cliente lo que necesita, o cuando es necesario identificar y solucionar ineficiencias sistemáticamente y se quiere trabajar utilizando un proceso especializado en el desarrollo (Molina Montero et al., 2018) , el desarrollo del prototipo usando el lenguaje de programación Java y las herramientas de Eclipse IDE, se utiliza Java para la Web (Java EE) con el framework JSF con PrimeFaces y la biblioteca de herramientas que este ofrece.

En resumen, se utilizará:

- SCRUM, Como marco de trabajo.
- Base de Datos PostgreSQL.
- Frontend: PrimeFaces, CSS, JavaScript.
- Backend: Java-JSF.
- Servidor de aplicaciones: Java EE WildFly.
- IDE de Desarrollo: Eclipse JEE 2019

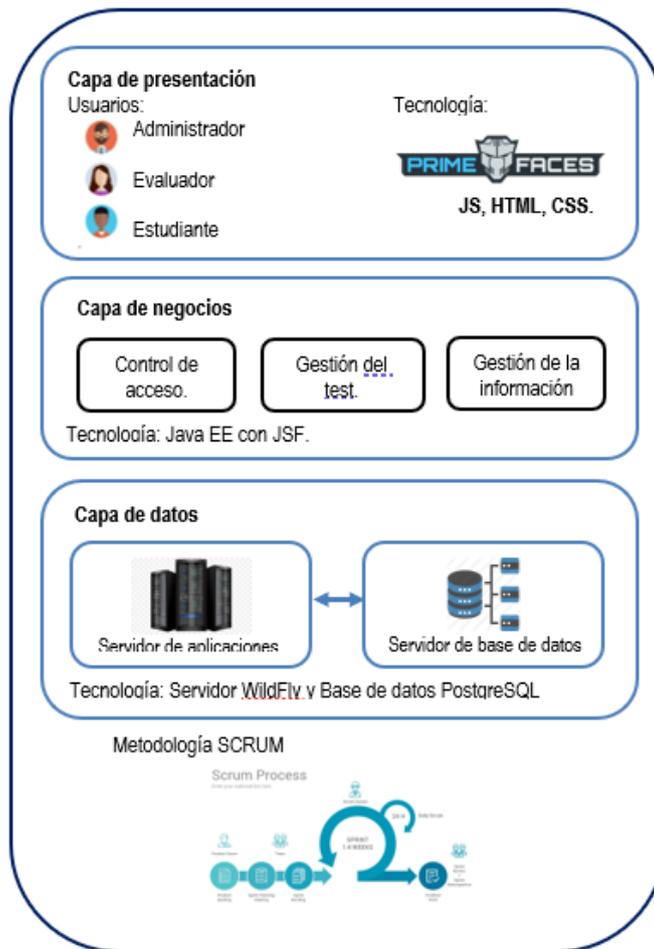


Fig. 2. Arquitectura de funcionamiento.
 Fuente Propia

Justificación

El estudio que tiene una dualidad (tecnológica - educativa) aporta a los objetivos de la agenda 2030 (Naciones Unidas, 2015 – 2030) en especial al objetivo 4 que plantea: garantizar una educación de calidad, así como promover las oportunidades de aprendizaje durante toda la vida para todos, a su vez colabora con el objetivo 9, ya que se vincula a un área temática que busca apoyar el desarrollo de tecnologías, la investigación y la innovación nacionales en los países en desarrollo.

Teórica.

Los avances tecnológicos nos facilitan la vida, aunque el uso que se le da en ocasiones solo está enfocado en el entretenimiento, por lo tanto, existe un desconocimiento por parte de los estudiantes sobre herramientas que podrían facilitar su trabajo.

La tecnología y las telecomunicaciones en todas sus formas cambiarán la forma de vivir, de trabajar, de producir, de comunicarnos, de comprar, de vender. Todo el entorno será bien distinto. El gran imperativo será él prepararnos y aprender a vivir en ese nuevo entorno. (Cardona, 2002).

La asignatura de TIC sigue estos lineamientos e intenta solucionar este inconveniente, sin embargo, las herramientas son poco conocidas y los conceptos pueden ser complejos de entender pudiendo generar un riesgo en la educación.

Investigaciones mencionan que no todas las instituciones de educación superior se encuentran en igualdad al momento de referirse a la implementación y empleo de nuevos recursos y tecnologías que puedan mantenerse a la vanguardia o seguir los pasos a los cambios permanentes respecto a las TIC (Ocaña Fernández et al., 2019).

La aplicación del TAI se enfoca en mejorar la calidad de educación, el refuerzo de temas complicados pretende identificar el conocimiento real del tema, para que sus resultados y predicciones permitan potenciar los planes de mejora en la formación académica de los estudiantes.

Tecnológica.

Mediante el uso de técnicas de inteligencia artificial se integra una idea innovadora para mejorar el estándar de las evaluaciones adaptativas informatizadas, un proceso que generalmente lleva tiempo e inversión de recursos.

Los TAI han sido aplicados en múltiples campos, desde psicología hasta medicina, un ejemplo de estos es CAT-Health (Rebollo et al., 2009) la cual describe un Test adaptativo informatizado que fue desarrollado en España para la medición de la calidad de vida relacionada con la salud, en psicología se desarrolló el proyecto Especificación del algoritmo para un Test Adaptativo Informatizado de Analogías Verbales (Atorresi & Lozzia, 2012), de la misma manera se plantea usar en dentro del entorno universitario medio al igual que el proyecto Sistema de Tests adaptativos informatizados para la unidad educativa José Julián Andrade de Dorado José (2019) que se desarrolló en la ciudad de Ibarra.

El desarrollo del TAI sigue este camino y entregará un gran aporte innovador, una herramienta fundamental que puede ser usada en posteriores investigaciones, así como adaptable a otras especialidades.

Metodológica.

La investigación corresponde con un enfoque cuantitativo, bajo un diseño de campo, de nivel explicativo. Las investigaciones bajo el enfoque cuantitativo son definidas por Rivas (2005) como: “un paradigma que en busca el análisis numérico de la investigación, a fin de conocer los porcentajes, y resultados desde un campo numérico específico” (p.419). Para efectos del presente estudio, este enfoque permitirá el análisis minucioso de los datos recolectados en la investigación, puesto que se tabularán los resultados obtenidos y se analizarán, logrando a través de cuadros estadísticos interpretar los resultados.

De igual manera, el estudio se enfoca hacia la Investigación de campo. Esta es considerada por Ramírez, citado en Martins & Palella, (2006) , como aquella que “consiste en la recolección de datos, directamente de la realidad donde ocurren los hechos, sin manipular o controlar variables. Estudia los fenómenos en su ambiente natural” (p.97). De acuerdo con esta definición las investigaciones de campo permiten obtener información en forma directa por el investigador del ambiente natural, de la situación en estudio y el sitio donde ocurren los hechos sin manipular o controlar variable alguna, es decir la información se recopila en el mismo sitio donde ocurren los hechos; en este caso en los estudiantes universitarios en carrera no afines a informática.

El diseño de investigación ejecutado fue experimental, el proceso es flexible y se mueve entre los eventos y su interpretación, permitiendo una fácil escalabilidad respecto a la calidad del TAI.

Capítulo I

1. Marco teórico

1.1. Definiciones Básicas

El presente trabajo maneja conceptos orientados a la estadística y métodos condicionales, por tal razón es necesario plantear definiciones básicas sobre dichos temas, los mismos que serán complementados en el transcurso del trabajo.

1.1.1. Inteligencia Artificial

Partiendo del concepto de Luckin et al. (2016) la Inteligencia Artificial (IA) se define como una capacidad que se les da a las máquinas similar a una habilidad humana para resolver problemas de acuerdo a las interacciones con su medio, estas son capaces de asimilar los datos de su entorno y tomar una decisión y de estas decisiones aprender y ampliar sus capacidades.

La IA resulta ser un campo muy amplio de las Ciencias de la Computación en síntesis se refiere a máquinas autónomas capaces de llevar a cabo tareas complejas sin intervención humana, León Rodríguez & Viña Brito (2017) menciona que los conceptos de IA varían de acuerdo con el modelo de la aplicación o el avance de la tecnología, entonces puede existir varios conceptos de IA que cambian con el tiempo, adaptándose a las nuevas tecnologías y el tipo de aplicación para el cual fueron diseñados.

La IA es un área del conocimiento donde convergen diferentes técnicas, los campos de aplicación son muchos, y algunos están orientados a satisfacer distintas necesidades, dependiendo de estas es posible seleccionar la técnica de IA adecuada.

Entre otras las técnicas más conocidas son:

- Machine Learning o aprendizaje automático.
- Lógica difusa.
- Sistemas expertos.
- Minería de datos.
- Redes bayesianas.
- Redes neuronales artificiales.

1.1.2. Probabilidad Condicional

La probabilidad condicionada explica de una mejor forma el teorema de Bayes. Soto (2011) menciona que: “Sí conocemos información sobre la ocurrencia de un evento A siempre que haya ocurrido un evento B, es necesario saber si es posible inferir la ocurrencia del evento B siempre que haya ocurrido A” (p.4). De esta manera la ecuación 1 toma sentido y se explica la relación de cada elemento que interviene en la formula del teorema de Bayes.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)} = \frac{P(B|A_i)P(A_i)}{\sum_{j=1}^n P(B|A_j)P(A_j)} \quad (1)$$

Entonces es posible calcular la probabilidad de un suceso A, sabiendo además que ese A cumple cierta característica que condiciona su probabilidad. López (2019) indica que el teorema de Bayes entiende la probabilidad de forma inversa al teorema de la probabilidad total, debido a que el teorema de la probabilidad total hace inferencia sobre un suceso B, a partir de los resultados de los sucesos A. Esta diferencia no parece afectar a simple vista, pero en las matemáticas un orden diferente puede no obtener el mismo resultado, por tanto, es importante el cumplimiento del Teorema de Bayes y el cálculo de la probabilidad de A condicionado a B.

1.1.3. Test Adaptativo Informatizado

El TAI o CAT por su traducción en inglés (Computerized Adaptive Testing) son un tipo de pruebas realizadas mediante el uso de un computador en las cuales, los ítems presentados cada uno de ellos son seleccionados de acuerdo con el resultado a ítems anteriores proporcionados por el examinado (Dobber, 2017), el término informatizado advierte que este tipo de test usará plataformas informáticas para su ejecución , y que en el núcleo del sistema estará un algoritmo adaptativo.

Ram et al. (2004), señalan que se trata de pruebas cuyos ítems se seleccionan según el nivel de competencia que progresivamente va manifestando la persona mediante un algoritmo aplicado en una computadora. Por lo tanto, las pruebas informatizadas muestran de acuerdo con el nivel del evaluado un ítem y según la respuesta a este ítem presentan otro ítem de menor, igual o mayor dificultad hasta obtener los datos necesarios o cumplir con el criterio de fin del test.

Es común estar familiarizado con los test tradicionales debido a su fácil uso y su aplicación en las instituciones académicas, mismos que vienen siendo aplicados constantemente desde hace tiempo atrás con el fin de comprobar el nivel de comprensión de los estudiantes, sin embargo, al 2020 el avance tecnológico ha dotado de computadores personales a la mayoría de los estudiantes, a inicios del siglo no era posible contar con esta tecnología. Cuando las primeras computadoras de escritorio fueron instaladas en las escuelas o colegios abrieron nuevas posibilidades para mejorar la educación, existieron mejores recursos a disposición y nuevas plataformas para evaluar, de acuerdo con Abad et al.,(2002) mencionan que este acontecimiento generó un cambio de enfoque, de estrategias y objetivos al momento de evaluar a los estudiantes, lo cual dio paso a la implementación de nuevos métodos de evaluación.

1.1.4. Teoría de Respuesta al Ítem

La Teoría de Respuesta al Ítem (TRI) nace de la necesidad de cubrir aspectos deficientes de la Teoría Clásica del Test (TCT), la TRI intenta dar unas bases probabilísticas al problema de la medición de rasgos no directamente observables conocidas como rasgos latentes, su nombre viene dado por la importancia que recibe el ítem como la unidad fundamental del test, en lugar de las puntuaciones totales, como es habitual en la Teoría Clásica de Test como lo menciona en Jiménez Sánchez, (2009). Cada ítem se describe como independiente de otro y precisa un nivel de dificultad de acuerdo con los fundamentos de la TRI.

Características

Aunque la TRI comprende varios modelos diferentes, todos ellos exhiben una serie de características comunes:

- Suponen la existencia de un rasgo latente o aptitud única la cual es medida por los ítems y explica el rendimiento/comportamiento del examinado en el test.
- Independencia local la cual asume que las respuestas del examinado para cualquier pareja de ítems en el test no están relacionadas al considerar un mismo nivel en el rasgo, es decir, cuando la habilidad se mantiene constante.
- Según el número de parámetros que se tengan en cuenta se hablará de un modelo u otro dentro de la TRI. Actualmente, los modelos más utilizados en la TRI son el modelo logístico de un parámetro, el logístico de dos parámetros y el logístico de tres

parámetros. El modelo logístico de un parámetro o modelo de Rasch es el más popular dentro de los modelos de la TRI, debido principalmente a su sencillez.

Además, como característica importante a considerar en la Tabla 1.1, la TRI proporciona estimaciones invariantes de las propiedades psicométricas de los ítems, así como de las características de los sujetos, es decir, que los parámetros que caracterizan al ítem y al test son menos dependientes de la muestra particular de sujetos utilizada y que los parámetros que caracterizan al sujeto no dependen de la muestra particular de ítems utilizada. (Hidalgo Montesinos & French, 2016). Esta propiedad descrita es la principal diferencia del TCT el cual es dependiente de la muestra.

TABLA 1.1
CARACTERÍSTICAS DE LA TRI Y LA TCT

| Característica | TRI | TCT |
|---|---|---|
| Influencia de un ítem sobre otro. | No influyen, se evalúa cada ítem como si fuese por separado. | |
| Invarianza de los parámetros de los ítems respecto a la muestra que se calcula. | Los parámetros no cambian debido a que se evalúa los resultados de cada ítem. | Los parámetros cambian debido a que se evalúa el test completo. |
| Invarianza del parámetro del rasgo del sujeto respecto al instrumento utilizado para estimarlo. | La dificultad del test no depende de la muestra. Cada ítem maneja su propia dificultad. | Cada test maneja su dificultad y sus resultados dependen de la muestra a la que se aplique. |

Fuente: (Hidalgo Montesinos & French, 2016).

Como lo muestra la tabla 1.1, si bien la TCT tiene argumentos bien planteados y han sido usados por muchos años, la TRI brinda un nuevo enfoque más fresco, capaz de adaptarse a los métodos usados por las tecnologías que actualmente desarrollan tests, ofrecen la capacidad de implementar nuevas herramientas capaces de mostrar de forma precisa, las deficiencias que el docente necesita cubrir o a su vez, al tratarse de un sistema con retroalimentación permitirán al mismo sistema poder cubrir que conceptos no han sido comprendidos por el evaluado.

Beneficios

En 2016, Moya Ricardo et al. Realizaron un trabajo en el cual al familiarizar tanto a estudiantes como a docentes con la Teoría de Respuesta al Ítem y sus beneficios, se realizó una encuesta con el fin de buscar el nivel de satisfacción de ambas partes y se obtuvieron resultados en los cuales remarcan cuatro puntos:

- **TRI para el control de la formación y desarrollo de habilidades:** facilita la ejecución de este proceso en los estudiantes y la planeación de las estrategias de intervención educativa.
- **Los métodos y técnicas investigativas:** permiten al profesor dirigir eficazmente el proceso de enseñanza-aprendizaje.
- **La evaluación con ayuda de la TRI:** permite la comunicación profesor-estudiante y estudiante-estudiante en el proceso evaluativo, aporta en la motivación y el protagonismo de los estudiantes en las tareas.
- **La evaluación mediante la TRI:** ayuda a identificar con mayor calidad errores, lo cual permite mejorar el aprendizaje.

Modelos de la TRI

La Teoría de Respuesta al Ítem cuenta con 3 modelos principales, como lo resalta García et al. (2013), estos establecen una relación matemática entre la probabilidad de emitir una determinada respuesta a un ítem (respuesta elegida) y otras características del sujeto (rasgo latente) y del ítem (discriminación). Estos tres elementos trabajan dentro de la TRI y se operan de acuerdo con las necesidades es decir cuando se requiere comprobar que un ítem depende de un único rasgo latente se habla de modelos unidimensionales; cuando el ítem depende de dos o más rasgos se habla de modelos multidimensionales cada uno de estos modelos están vinculados a una función matemática, gráfico de dicha función matemática se denomina Curva Característica del Ítem (CCI).

Estos modelos se utilizan dependiendo de sus requisitos y cada una incorpora funciones diferentes de esta manera se presentan tres:

- a) El **modelo logístico de 1 parámetro (1P)**, también conocido como el modelo de Rasch, es uno de los modelos más sencillos de los modelos dicotómicos. Es representada en la Ecuación (2) y consta de las siguientes propiedades:
 - $P(\theta)$, es la probabilidad de acertar el ítem j cuando la persona tiene un nivel de rasgo θ , este parámetro normalmente asume valores entre -4 y $+4$.
 - b_j , es el parámetro de dificultad del ítem (normalmente asume valores entre -4 y $+4$, pues se mide en la misma escala que θ),
 - D , es un valor constante (si $D=1$ se habla de escala logística; si $D=1.7$, de escala normal)

- La variable “e”, es la base de los logaritmos naturales (e=2.718).

$$P(\theta) = \frac{1}{1 + e^{-D(\theta - b_j)}} \quad (2)$$

b) El **modelo logístico de 2 parámetros (2P)**, incorpora a la función logística un parámetro de discriminación del ítem (a_j) como se muestra la ecuación (3), además de las anteriores propiedades descritas esta cuenta con:

- a_j , es proporcional a la pendiente de la CCI en el valor $\theta = b_j$, este parámetro de discriminación, que suele oscilar entre 0 y 3, indica el grado en que el ítem discrimina entre los niveles θ superiores e inferiores a la dificultad del ítem.
- Si la CCI tiene poca pendiente en b_j (a_j cercano a 0), el ítem resulta poco discriminativo; si su pendiente es elevada, el ítem sirve para diferenciar los niveles de rasgo por encima y por debajo de su dificultad.

$$P(\theta) = \frac{1}{1 + e^{-Da_j(\theta - b_j)}} \quad (3)$$

La Figura 3 muestra la forma que adoptan las curvas características de tres ítems de diferente dificultad ($b = -1$; $b = 0$; y $b = 1$) según el modelo de Rasch. Nótese que todas las CCI tienen la misma pendiente (son paralelas), similar al modelo de un parámetro, en los cuales todos los ítems tienen la misma discriminación.

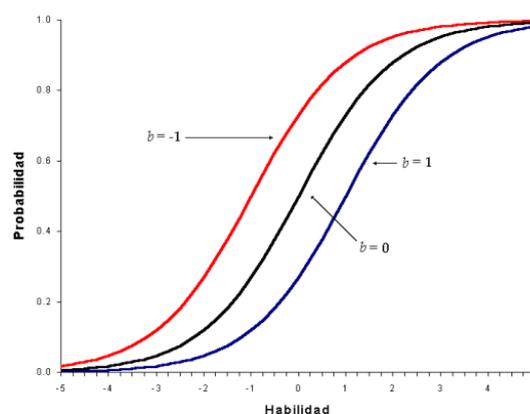


Fig. 3. Ejemplo de varias CCI.
Fuente:(López Cuadrado, 2008)

- c) El **modelo logístico de 3 parámetros (3P)**, añade al anterior un nuevo parámetro (c) como lo muestra la ecuación (4), a este parámetro se lo denomina parámetro de pseudo-azar y se caracteriza por que puede oscilar aproximadamente entre 0 y 0.5, se representa como la asíntota inferior de la CCI y representa la probabilidad que tienen de acertar el ítem personas con nivel θ extremadamente bajo (límite de $P(\theta)$ cuando q tiende a $-\infty$):

$$P(\theta) = c_j + (1 - c_j) \frac{1}{1 + e^{-Da_j(\theta - b_j)}} \quad (4)$$

1.2. TAI basado en Estimación Bayesianas.

1.2.1. Teorema de Bayes y Redes Bayesianas

Thomas Bayes (1702-1761) fue un matemático británico que aportó mucho al campo de la probabilidad, particularmente, el teorema que lleva su nombre y que refiere a la probabilidad de un evento condicionado por la ocurrencia de otro, a esta operación se le da el nombre de operación condicionada.

En una operación matemática del teorema de Bayes intervienen 3 clases de probabilidades, que son las siguientes:

- $P(A_i)$ o probabilidad a priori de un suceso "A".
- $P(A_i|B)$ o probabilidad a posteriori de un suceso "A", (cuando se obtiene la información de que ha ocurrido un suceso B).
- $P(B|A_i)$ o verosimilitudes del suceso "B" son supuestos que habrían de ocurrir a cada suceso A_i .

El Teorema de Bayes

Matemáticamente se la representa con la formula (5) en la que describe que el teorema de Bayes es igual al cociente del producto de la probabilidad "B" dados (A_i), $P(B|A_i)$ (siendo B el suceso conocido y " A_i " los sucesos condicionados) por la probabilidad $P(A_i)$ entre la sumatoria de cada probabilidad que contenga el suceso conocido por cada suceso conocido. (Riquelme, n.d.). En síntesis, el numerador es la probabilidad condicionada y el denominador es la probabilidad total.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)} = \frac{P(B|A_i)P(A_i)}{\sum_{j=1}^n P(B|A_j)P(A_j)} \quad (5)$$

Las redes que utilizar este teorema en la actualización de sus probabilidades son conocidas como Redes Bayesianas, del mismo teorema se utilizan métodos de propagación eficientes en redes donde existen muchas variables (R. S. García, 2007). El uso del teorema de Bayes ha demostrado tener una variedad de aplicaciones, muchas de ellas enfocadas en predecir el comportamiento humano, remarcando la importancia de los métodos estadísticos y probabilísticos en campos de la Inteligencia Artificial.

Redes Bayesianas

(Conejo et al., 2001), recomienda que, el primer paso es definir los elementos básicos: variables, enlaces entre ellas y parámetros. Enfocando prioritariamente al diagnóstico basado en preguntas tipo test, aunque en principio sería posible considerar cualquier tipo de preguntas mientras el sistema compruebe si la solución propuesta por el estudiante es o no correcta, es decir, para facilidad y pruebas del sistema solamente usar preguntas con dos opciones, posteriormente aplicar cualquier tipo de ítem mientras esta se pueda comprobar su respuesta.

Las redes bayesianas influyen en la construcción de sistemas expertos, y muchos de los sistemas con IA implementan en sus funciones. Una RB (Red Bayesiana) es considerada un grafo acíclico dirigido con las siguientes características:

- Los nodos representan variables
- Los arcos representan la dependencia probabilística, estas contienen especificaciones de la probabilidad condicional de cada variable dados sus padres.
- La variable a la que apunta el arco es dependiente (causa-efecto) de la que está en el origen de éste.
- La estructura de la red nos da información sobre las dependencias probabilísticas entre las variables, pero también sobre las independencias condicionales de una o unas variables dadas, simplificando la representación del conocimiento y el razonamiento.

Definición General

Un concepto más generalizado de RB lo representa como un conjunto de nodos y arcos. En el cual a cada nodo le corresponde una variable, esta variable puede ser una

cantidad o valor en este campo, los arcos relacionan a los nodos indicando influencia causal entre las variables.

Partiendo de la red causal de ejemplo (Fig. 4), es posible establecer las siguientes relaciones:

- C es padre P, debido al arco dirigido que existe de C a P, P es hijo de C.
- C es antecesor de P de existir un camino, de tal manera que desde C se puede acceder a P, en términos analógicos es posible decir que P es descendiente de C.

La RB no trivial simple se representa con dos variables, y un arco dirigido entre ellas. De esta forma la Figura 4, representa al nodo C como el conocimiento del estudiante sobre cierto concepto y P1 su capacidad de resolver correctamente cierta pregunta relativa a dicho concepto (Millán & Pérez-de-la-Cruz, 2004). Entonces, que el estudiante sepa el concepto C tiene influencia causal en que sea capaz de responder bien a la pregunta P1, esta capacidad se representa mediante el arco dirigido.



Fig. 4. RB con dos nodos
Fuente: Propia

Parámetros de una Red Bayesiana

Partiendo del ejemplo de (Castelló Pascual, 2018), que a la vez hace referencia al trabajo de (Millán, 2000) sobre el modelado de redes bayesianas, a la figura 4 del ejemplo anterior se agregan nuevos parámetros con el fin de explicar la funcionalidad, la notación usada es:

- X es una variable binaria, es denotada por +x la presencia de aquello a lo que representa y por $\neg x$ a su ausencia.
- +c significara que el estudiante conoce el tema o concepto C.
- $\neg c$ indica que el estudiante no conoce el tema o concepto C.
- +p1 significa que el estudiante es capaz de resolver correctamente la pregunta P1.

- $\neg p1$ indica que el estudiante no es capaz de resolver correctamente la pregunta P1.

La información cuantitativa de una Red Bayesiana viene dada por:

- La probabilidad a priori de los nodos que no tienen padres.
- La probabilidad condicionada de los nodos con padres.

Por tanto, en el ejemplo los datos que debemos conocer son $P(c)$ y $P(p1/c)$. Así, la Red Bayesiana completa de este ejemplo sería la que se muestra en la Figura 5.



Fig. 5. RB con dos nodos
Fuente: Ejemplo de (Millán & Pérez-de-la-Cruz, 2004)

- $P(+c) = 0.3$ indica que el 30% de los estudiantes del grupo en estudio conocen el concepto.
- $P(+p1/+c) = 0.9$ indica que el 90% de los estudiantes que conocen el concepto C responden correctamente a la pregunta P1. Esto quiere decir que incluso los estudiantes que conocen el concepto pueden tener un despiste y contestar mal a la pregunta (en una proporción del 10%).
- $P(+p1/-c) = 0.01$ significa que sólo el 1% de los estudiantes que no conocen el concepto C son capaces de contestar correctamente a la pregunta P1. Este parámetro indica por tanto qué estudiantes que no conocen el concepto pueden adivinar la respuesta correcta a la pregunta P1.

Con los datos obtenidos se puede calcular:

- a) La probabilidad a priori de que un estudiante cualquiera conteste correctamente a la pregunta P1:

$$P(+p1) = P(+p1/+c) \cdot P(+c) + P(+p1/-c) \cdot P(-c) = 0.277$$

$$P(-p1) = P(-p1/+c) \cdot P(+c) + P(-p1/-c) \cdot P(-c) = 0.723$$

- b) Las probabilidades a posteriori dada una evidencia observada e, $P^*(c) = P(c/e)$.

En el caso de que la evidencia observada es de cierto estudiante que ha contestado correctamente a la pregunta P1. ¿Qué probabilidad hay ahora de que conozca el concepto C?, Si no existiese posibilidad ninguna de que un estudiante que no conozca el concepto C responda bien a la pregunta P1, esa probabilidad sería 1, pero como no es así es necesario calcular $P^*(+c) = P(+c/+p_1)$. Para ello se aplica el teorema de Bayes y así obtener que:

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)} \quad (6)$$

$$P^*(+c) = P\left(\frac{+c}{+p_1}\right) = \frac{P(+c)P\left(+\frac{p_1}{+c}\right)}{P(+p_1)} = \frac{0.3 * 0.9}{0.277} = 0.97473 \quad (7)$$

Definición Técnica de RB

Una red Bayesiana es un par (D, P) , donde D es un grafo dirigido acíclico, $P = p(x_1|\pi_1, \dots, p(x_n|\pi_n))$ es un conjunto de n funciones de probabilidad condicionada, una para cada variable, y $\prod i$ es el conjunto de padres del nodo X_i en D (Castelló Pascual, 2018). El conjunto P define una función de probabilidad asociada mediante la factorización (Ecuación 8).

$$p(x) = \prod_{i=1}^n p(x_i|\pi_i) \quad (8)$$

Para el TAI la fórmula 8 se adapta reemplazando el valor de π_i y se interpreta de la siguiente forma:

$$p(x) = \prod_{i=1}^n p(x_i|X_{n(i)}) \quad (9)$$

También conocida como regla de la cadena, donde $X_{n(i)}$ son las variables predecesoras inmediatas de la variable X_i .

Este teorema permite describir una Red Bayesiana partiendo del concepto de probabilidad condicionada respecto a cada nodo (o la probabilidad a priori en el caso de nodos sin padres), debido a que requiere un número de parámetros exponencial en el número de

nodos no aplica la probabilidad conjunta, de esta manera que el número de parámetros requerido para dar las probabilidades condicionadas es menor (Ramón, 2016).

Al relacionar el Test Adaptativo Informatizado con la Inteligencia Artificial se intuye que este test aplica métodos de IA en su algoritmo, es aquí donde interviene la red Bayesiana, este método de IA fortalece el algoritmo y permite la integración de IA en el sistema, los complejos cálculos matemáticos y la forma del algoritmo requieren de potentes elementos de desarrollo que soporten métodos de IA y posean experiencia en el campo además de una librería extensa de fácil acceso a plugins o complementos.

1.3. Relación de la TRI y el TAI en la forma de evaluar

Los métodos condicionados forman un aporte esencial para el diseño de la evaluación adaptativa, sin embargo, el TAI requiere de un conjunto de componentes para su correcto funcionamiento, estos componentes deben tener una estrecha relación para trabajar con métodos probabilísticos. Aparte del algoritmo basado en métodos condicionados del teorema de Bayes, requiere un conjunto de preguntas y un modelo logístico de la teoría de respuesta al ítem, entre otros componentes.

1.3.1. Componentes básicos del TAI

El TAI está compuesto de un Banco de ítems (BDÍ), cada ítem tiene un valor asignado de acuerdo con la TRI, estos ítems son seleccionados por medio de un algoritmo operado con redes bayesianas o como en este caso por un algoritmo adaptativo mediante estimación bayesiana. El primer ítem se presenta al examinado de acuerdo con el criterio de inicio detallado en el algoritmo, y según su respuesta se le presenta un segundo ítem de igual o superior valor; la respuesta al segundo ítem se usa para seleccionar un tercer ítem y así sucesivamente. Cuando ha logrado cumplir con el criterio de parada establecido de antemano (cierta cantidad de ítems o nivel de precisión), se finaliza la presentación de ítems y se da por terminado el test.

De acuerdo con las especificaciones para la creación de un TAI en trabajos posteriores (Attorresi & Lozzia, 2012) y trabajos recientes (Jiménez López, 2019) resaltan en común que todo TAI requiere para su funcionamiento de dos componentes importantes puesto que la selección adecuada de estos procedimientos depende que el test arroje información relevante sobre la habilidad del individuo:

- a) **Un banco de ítems calibrados desde un modelo de la TRI.** Parte importante del sistema pues se trata de un conjunto de reactivos debidamente calibradas con técnicas de la teoría de respuesta al ítem.
- b) **Un algoritmo adaptativo informatizado.** Todos los tests son administrados siguiendo una serie de reglas que determinan los ítems por responder y su orden de presentación establecida por un algoritmo adaptativo, su característica distintiva es la selección dinámica de los ítems en función del nivel de rasgo que va manifestando el evaluado.

El procedimiento para iniciar y finalizar el test así como el método estadístico para la estimación del rasgo son elementos necesarios e importantes para el TAI los mismos están detallados dentro del algoritmo, en ocasiones el método de presentación de los ítems es considerado como un elemento básico del TAI, sin embargo, siguiendo el modelo de Attorresi & Lozzia (2012), se definió únicamente los dos elementos mencionados debido a que todos los elementos mencionados serían agrupados dentro del algoritmo adaptativo.

1.3.2. Algoritmo del TAI

El algoritmo del TAI cumple con todos los pasos descritos en el diagrama de flujo (Fig. 6), cada una de sus funciones descritas son esenciales para su funcionamiento.

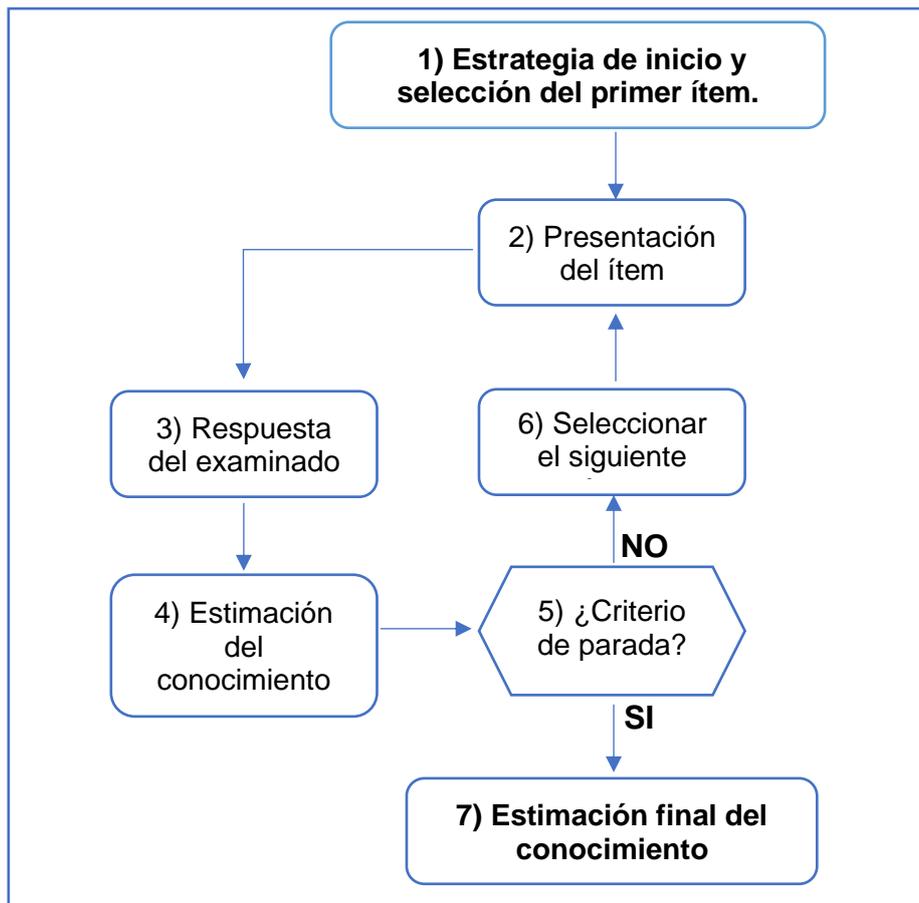


Fig. 6. Diagrama de flujo TAI
Fuente: Propia

Estrategia de inicio y selección del primer ítem.

También conocida como estrategia de arranque, este es el primer proceso y su función principal es la estimación de un nivel inicial de habilidad del evaluado para poder elegir el ítem correcto, la información previa disponible del evaluado podrá ser usada como punto de partida para seleccionar el primer ítem.

Para la selección de la primera pregunta existe una variedad de métodos que se pueden aplicar, tomando en cuenta la disponibilidad de elementos a priori del evaluado, entre otros como lo detalla en (Herrera Rojas et al. 2016):

- Fijar un criterio estándar sobre el nivel de habilidad para cada evaluado, siendo indispensable información de evaluaciones previas.
- Selección de un ítem inicial como punto de partida para todos los evaluados, por lo general de dificultad media.
- La selección de un ítem inicial de manera aleatoria de un nivel de dificultad baja.

La información disponible sobre el evaluado o grupo de examinados es un elemento básico para rendir el test que está estructurado por métodos probabilísticos que usan estimación bayesiana, el conocimiento a priori mejorará el rendimiento del test. Jiménez Sánchez (2009) plantea que, de existir ausencia de información, se considera asignar una distribución uniforme a los conceptos elementales o el uso de métodos diferentes para estimar el nivel de conocimiento inicial.

Presentación del Ítem

Una vez seleccionado el ítem adecuado el sistema trae los datos del ítem del Banco de Ítems y se presenta al evaluador, tratándose de un sistema virtual puede incorporarse fácilmente recursos visuales o audios en cualquiera de los ítems, en pantalla se mostrará únicamente el ítem y las opciones disponibles para su respuesta.

Al encontrarse el sistema en un servidor conectado a internet los usuarios requerirán conexión fija a la red, la conexión se realizará como a cualquier sitio web, el framework en la que se desarrolla es responsive y no dificultará la manipulación por parte de los evaluados.

Respuesta del examinado

Para este proceso es importante la intervención del evaluado, se obtiene a través de él la respuesta contestada de forma correcta o incorrecta, el resultado de esta formará parte de la estimación del rasgo latente y también como un elemento a priori en el teorema de Bayes, el tiempo también es un elemento importante en la evaluación, sin embargo al permanecer oculto a los ojos del evaluado disminuye la presión psicológica que podría tener. Se evalúa la posibilidad de que ciertos ítems verifiquen la respuesta a un ítem anterior, pero al aplicar este método, irrespetaría una de las propiedades importantes de los TAI.

Estimación del conocimiento por Máxima Verosimilitud

Este es un método de puntuación, su tarea es la estimación del nivel de habilidad o el valor del rasgo θ , se aplica por primera vez después que el evaluado responda el primer ítem, los resultados son muy relevantes para el procedimiento de estimación Bayesiana de la máxima verosimilitud y poder estimar de manera provisional el primer nivel de rasgo θ (Rojas et al., 2016). El método de máxima verosimilitud permite cargar una variable que opera el valor del conocimiento estimado, y al finalizar el test la variable provisional no sea necesaria.

Cada pregunta tiene un valor específico para su respuesta, para las redes bayesianas los algoritmos de propagación de probabilidades proporcionan un método consistente para valorar las respuestas es decir, para estimar el nivel de conocimiento de los conceptos que intervienen en las preguntas de acuerdo con las respuestas dadas por el estudiante (Millán & Pérez-de-la-Cruz, 2004).

Los primeros resultados no brindarán suficiente información por lo tanto la estimación será muy imprecisa pero mientras más resultados revise, la precisión será mejor, este procedimiento se ejecutará a la par de la selección y presentación de ítems, dicho de otra manera, cada vez que se requiera presentar un nuevo ítem al evaluado. Cada puntuación será almacena en la base de datos al igual que en el nodo del algoritmo, el proceso termina al cumplir con el criterio de parada y el test se dé por finalizado.

Criterio de parada.

El criterio de parada permite al sistema la aplicación de un método condicional para finalizar el test, estos procedimientos descritos por Olea Díaz & Ponsoda (2015) pueden cumplir uno de los cuatro criterios:

- **Criterio fijo.** - Es decir estableciendo un número prefijo de ítems para aplicarse a los evaluados, esto causaría que diferentes estimaciones tendrán distinta precisión.
- **Criterio variable.** – Consiste en fijar nivel de error típico para todos los evaluados, el test finaliza cuando el error típico desciende del valor predeterminado.
- **Criterio mixto.** - Combina el criterio fijo y el variable de esta manera la aplicación se detiene cuando se presentan el numero predeterminado de ítems o cuando el error típico desciende del valor preestablecido.
- **Criterio variable ascendente.** - Consiste en detener la aplicación del test cuando el nivel de rasgo estimado se aleja del valor máximo establecido.

Ramón (2016) incluye en la lista de criterios de parada un modelo en el que el test finaliza únicamente cuando el test haya evaluado todos los conceptos y el elemento de probabilidad condicionada reconozca que domina el concepto. La aplicación de este método podría generar un test demasiado largo por lo cual Ramon describe la implementación de un criterio fijo, la combinación de ambos criterios refuerza la estimación de un mejor criterio de parada.

Se opta entonces por seleccionar un criterio de parada mixto, un método que determine un número máximo de ítems que el evaluado debe responder, junto con un método que aplique el criterio variable ascendente y descendente el cual el evaluado al contestar un número determinado de preguntas erróneas seguidas el test se por finalizado de la misma manera si contesta todas correctas, a estos métodos integramos el factor tiempo para no extender el test debido a que no se busca evaluar para conseguir una nota como resultado.

Selección de ítems

Tras la estimación del conocimiento o la asignación de un valor de rasgo θ provisional, se analizan probabilidades para la presentación de los posibles ítems, el algoritmo elige el siguiente ítem que refleje máxima utilidad para dicho nivel de rasgo, diferenciando y apartando a los ítems que ya le fueron presentados. El algoritmo se ejecuta durante la duración del test realizando los cálculos oportunos para determinar cuál será el siguiente ítem para presentar si el evaluado acierta o falla el ítem actual. El tiempo que transcurre entre la respuesta a un ítem y la presentación del siguiente ítem resulta nulo e imperceptible para el evaluado.

De acuerdo con (Olea Díaz & Ponsoda,2015), actualmente cualquier procedimiento de selección de ítems sigue una estrategia de ramificación variable con múltiples niveles, permitiendo actualizar progresivamente el nivel de habilidad estimado (variable de rasgo θ) luego de contestar los ítems que se le van presentando.

Millán & Pérez de-la-Cruz,(2004) Clasifica los métodos para la selección del ítem adecuado en dos grupos que a su vez se dividen en subgrupos cada una con un criterio diferente:

- a) **El criterio aleatorio.**- Fals & Piloto (2009) considera como el más sencillo y no muy pedagógico, la selección de ítems la realiza por azar, este generalmente se lo confunde con un test tradicional ya que no posee características de adaptativo. Sin embargo, usa el método Bayesiano para la evaluación y posterior diagnóstico, pese a esto queda en desventaja frente a un test adaptativo.
- b) **El criterio Adaptativo.** - Selecciona el siguiente ítem en base a la respuesta del anterior ítem para obtener el rasgo o nivel de conocimiento, existen dos criterios:
 - Criterio basado en la información. – El ítem más informativo será aquel de más utilidad.

- Criterio condicionado. - El próximo ítem a elegir dependerá del comportamiento del estudiante a preguntas previas, estas se clasifican de dos formas:
 - i. Criterio condicionado a la probabilidad del concepto. – Selecciona el ítem enfocándose en demostrar el conocimiento del evaluado ante un concepto definido.
 - ii. Criterio condicionado a la probabilidad de la pregunta. - Se selecciona en función a la probabilidad de ser contestada la pregunta.

Las dificultades que presentan los métodos bayesianos, tales como que el nivel de habilidad estimado es independiente al desempeño del evaluado, pero es dependiente de la distribución a priori del nivel de habilidad en la población, es decir, de los valores de la media y la varianza de esta; otro problema presente en los TAI es la longitud del test, si esta resulta ser pequeña, el sesgo en las estimaciones aumenta debido a que cuenta con un número reducido de ítems (Herrera Rojas et al., 2016) . Entonces dependiendo del tamaño del test si son más de 30 ítems aplicados el sesgo será bajo y de pocas diferencias en los resultados, caso contrario el sesgo será alto.

Debido a que el método de Máxima Verosimilitud no proporciona estimaciones finitas cuando un individuo tienen un patrón regular de respuestas ya sean solo de aciertos o de desaciertos, se han desarrollado estrategias para corregir dicho error, las cuales utilizan métodos bayesianos para la selección del ítem.(Herrera Rojas et al., 2016). Posterior a las respuestas del evaluado a los ítems presentados se dispondrá de la distribución posterior del rasgo θ , en este punto se realizan los cálculos de precisión, tal como la varianza de la distribución posterior la cual se conseguirá si se acierta o se falla cualquiera de los ítems disponibles en el banco de ítems, seleccionando aquel que posea una menor varianza lo que es igual a una mayor precisión. Olea Díaz & Ponsoda (2015), recalca que este no busca el ítem que más contribuye a la precisión de la estimación de un nivel concreto de rasgo, sino el más apropiado para toda una distribución de estimaciones.

Banco de Ítems

Un banco de ítems no es más que una base de datos sobre los ítems, conformada por los elementos básicos de un ítem: Instrucciones, pregunta, opción correcta, opciones incorrectas, también puede incluir información psicométrica como parámetros de estimación de la TRI o el contenido que mide cada ítem. La calidad del banco de ítems es independiente al algoritmo del TAI, de tal manera que un buen BI no debe presentar problemas de

representación, tamaño o calidad psicométrica, el banco de ítems es la base sustantiva fundamental de un TAI.

Olea Díaz & Ponsoda (2015), comparte el punto de vista, además detalla que el número de ítems que deben elaborarse tiene mucho que ver con otras consideraciones prácticas como la cantidad de restricciones que se establecen en el algoritmo de selección de ítems, con el número previsible de aplicaciones del TAI en un futuro inmediato o con el propósito de diseñar bancos de ítems diferentes.

1.3.3. ¿Cómo aporta la TRI al TAI?

El desarrollo del TAI es un trabajo complejo que requiere de la colaboración de expertos en el área de informática y expertos en el área de Psicometría, el trabajo conjunto de estos garantiza el correcto funcionamiento del sistema.

Importancia de la TRI en el TAI

Una vez elaborado el banco de ítems, la aplicación de la TRI es fundamental para calibrar el banco de ítems, exige obtener respuestas de muestras numerosas de evaluados en los diferentes sub-tests establecidos (Ram et al., 2004), para de esta manera asegurar su correcta calibración, caso contrario se recomienda calibrar nuevamente el test, pero esto dependerá del método de IA usado en el algoritmo.

Sin una Base de ítems correctamente calibrados usando un modelo de la TRI el algoritmo adaptativo no tendría forma de seleccionar un ítem que de acuerdo con el nivel de rasgo del participante requiera, entonces el TAI no funcionaría correctamente y no cumpliría con la cualidad de ser adaptativo, es decir, la TRI y los modelos logísticos son de gran importancia para el TAI.

El modelo adecuado para el APP-TAI

Los elementos que intervienen en cualquier de los modelos logísticos y representados en la CCI, vienen dados por la necesidad de determinar los valores de los parámetros (a_j, b_j, c_j) para la valides del comportamiento del ítem, de estos depende la selección del modelo adecuado.

Es evidente la cantidad de parámetros que influyen en un determinado modelo, por su parte APP-TAI busca por medio de sus ítems diferenciar al evaluado respecto a su nivel de conocimiento pose obtenidas fuera de las aulas o por acciones formativas, entonces es

necesario el uso de la variable a_j para definir el nivel de habilidad en el rasgo latente θ . Solamente los modelos 2P y 3P poseen este parámetro, sin embargo el modelo 3P estima el valor c_j el cual calcula la probabilidad del coeficiente del azar, es decir que probabilidad existe en que el evaluado acierte una pregunta sin tener el conocimiento de esta, ambos modelos incluyen el índice de dificultad c_j también denominado punto de escala de habilidad así pudiendo definir la tabla 1.2.

TABLA 1.2
PARÁMETROS USADOS EN APP-TAI

| APP-TAI | Define | Parámetro |
|--------------------------|--------|-----------|
| Índice de discriminación | Si | a |
| Índice de dificultad | Si | b |
| Coeficiente del azar | No | c |

Fuente: Propia

La aplicación del modelo logístico de 2P para la calibración de los ítems diseñará una Base de Ítems con la cualidad de diferenciar las habilidades de los evaluados y estimar la dificultad de los ítems, clasificando de esta manera 5 niveles de dificultad, el índice de azar es descartado para el desarrollo de esta investigación puesto que los cálculos son complejos y no se dio importancia en el requerimiento de la funcionalidad del sistema.

1.3.4. Evaluaciones por TAI

Los sistemas que utilizan Test Adaptativos Informatizados para realizar evaluaciones previamente poseen una lista de los usuarios que están autorizados para rendir el test, inician el procedimiento con credenciales (Usuario y contraseña), los sistemas TAI cuentan con un Banco de ítems de mínimo 200 ítems, los cuales se calibran utilizando modelos logísticos, cada ítem con un nivel de dificultad, cuando el evaluado inicia el test los ítems presentados no seguirán un orden específico, tampoco se repetirán los ítems con otro evaluado si se encuentra rindiendo el mismo test en el mismo momento, cada test es único y dependerá de las capacidades del evaluado la presentación de los ítems.

Ventajas

De acuerdo con Sierra et al. (2007) las ventajas de la aplicación de un TAI se reflejan desde la optimización del tiempo, así como la aplicación de métodos que brinden un mayor nivel de seguridad, estos beneficios son mayores cuando se aplica la Teoría de Respuesta al Ítem para la calibración del Banco de Ítems. La selección de ítems aleatorios y su vez adaptativos resultan ser más eficientes y precisos, los resultados obtenidos de acuerdo a

Tseng (2016) muestran su viabilidad y efectividad al tratar de identificar un rasgo, los TAI pueden resultar cortos con el uso aproximado del 1/3 de ítems del banco de ítems y pueden obtener los mismos resultados que un test tradicional. La aplicación del TAI no resuelve el problema de la copia ni la fuga de información, pero es significativamente menor que cualquier test convencional.(Olea Díaz & Ponsoda, 2015).

Desventajas

El problema en los TAI se ve reflejada más en el ámbito financiero, debido a los altos costos de desarrollo de los sistemas adaptativos, ya que generalmente estos necesitan de expertos desde, para la calibración de ítems hasta la verificación del correcto funcionamiento del sistema (Dobber, 2017), la institución o el evaluador deberá realizar una significativa inversión para la adquisición de estos sistemas de test siendo este la principal desventaja, Dobber también indica puntos negativos referentes a la reacción psicológica de algunos encuestados, sin embargo, este no sería un problema al tratarse del APP-TAI ya que la finalidad del test no es la obtención de una nota si no la retroalimentación.

La tabla 1.5 señala la ventaja notable entre las dos formas de evaluar y detalla como esta ventaja podría ser conveniente.

TABLA 1.3
VENTAJAS TAI-TEST TRADICIONAL

| Ventaja | TAI | Test común | Descripción |
|---|-----|------------|---|
| Seguridad ante fugas de información. | x | | Requiere un número muy alto de preguntas, si alguien tuviera acceso a ellas tendría que estudiar todas, en cuyo caso se merecería buena nota. (Olea Díaz & Ponsoda, 2015) |
| Tiempo de ejecución del test por parte de los examinados. | x | | Pueden responder las preguntas a su ritmo siendo el tiempo de respuesta de cada ítem información útil en la evaluación. |
| Forma de ejecución del test | x | | Se presentan desafíos durante el examen manteniendo la dificultad para no desalentar al examinado. |
| Forma de calificar. | x | | Los exámenes se pueden puntuar inmediatamente, dándole la retroalimentación de forma inmediata al estudiante. |
| Mejorar el test | x | | Permite modificar el test, ya que se pueden introducir nuevos ítems de manera no intrusiva. |
| Correcciones del test | x | | Los ítems defectuosos pueden eliminarse inmediatamente. |
| Contenido variado. | x | | Se pueden incluir varios tipos de preguntas al examen. Puede incluir preguntas con ficheros de audio y vídeo. |

| | | |
|---|---|---|
| Confiabilidad del test. | x | Los test de tipo TAI pueden acortarse hasta un 50% referente a los test tradicionales, manteniendo el mismo nivel de precisión. (Tseng, 2016) |
| Facilidad de elaboración. | x | No requieren un gran conjunto de preguntas, ni la calibración de los ítems. |
| Pruebas tras modificaciones en el test. | x | No requiere experto para realizar pruebas tras alguna modificación en el test. (Dobber, 2017). |
| Desarrollo de un sistema que gestione el TAI. | x | No requieren de sistemas complejos para su desarrollo. (Dobber, 2017). |

Fuente: Propia

Es evidente que la aplicación de un TAI posee mejores ventajas, la más importante y la razón de su aplicación es la entrega de mejores resultados en comparación de un test tradicional, además del banco de ítems utilizado que brinda un respaldo al evaluador por su complejidad, por su parte los costos de elaboración es una principal desventaja, sin embargo la forma de aplicar el test cuenta como factor variable, y este será ventajoso si el test es desarrollado correctamente al igual que su banco de ítems y que tan amigable es el sistema que lo gestiona.

1.4. Herramientas para la implementación

Para el proceso de implementación se opta por el uso de herramientas de código abierto y servicios que brinden versiones de prueba para estudiantes, esto con el fin de amenorar costos en el desarrollo y pruebas posteriores. La Figura 7 muestra las herramientas que se requieren para el desarrollo del aplicativo cumpliendo con los modelos de arquitectura.



Fig. 7. Herramientas seleccionadas
Fuente: Propia

1.4.1. Plataforma tecnológica

Es la estructura tecnológica sobre la cual se ejecuta un sistema o aplicativo, brinda compatibilidad y herramientas para la ejecución para frameworks, arquitecturas e incluso de lenguajes de programación específicos.

Servidor de aplicaciones J2EE

Valarezo Pardo et al.(2018), describe como un programa informático o sitio Web que funciona exclusivamente con acceso a Internet, sin necesidad de instalación en el ordenador, mantienen comunicaciones cliente-servidor mediante protocolos HTML. Por tal razón, aunque no se instalará el aplicativo es indispensable el uso de un navegador web.

Un servidor de aplicaciones es un software integrado en un servidor web, se encarga de realizar el despliegue de aplicaciones web en internet. Se aloja en el servidor físico (Hardware) junto con la base de datos de la misma aplicación, mediante una configuración generada por el servidor de aplicaciones se conecta con la base de datos y realiza la integración de la información, posteriormente el servidor de aplicaciones publica la información en la dirección URL específica (Otálora, 2018), por lo tanto, un servidor web no cumple con las funciones de un servidor de aplicaciones, pero un servidor de aplicaciones

cumple las funciones de un servidor web, siendo el servidor de aplicaciones un tipo de servidor web más complejo.

J2EE es la definición de Java 2 Enterprise Edition, los servidores J2EE son servidores de aplicaciones que integra la tecnología de J2EE con el objetivo de usar un modelo de desarrollo para aplicaciones empresariales escalables de arquitectura multicapa¹. Permite integrar el modelo de trabajo Modelo-Vista-Controlador (MVC), y diferentes estándares de tecnología como: Java Persistence API (JPA), Enterprise Java Beans (EJB), Java Server Faces (JSF) entre otras (Godoy et al., 2017), J2EE hace referencia a la versión 2 de Java empresarial, que crea aplicaciones Java más complejas, que integra módulos que tienen la capacidad de generar servicios web.

JBoss Enterprise Application Platform (EAP) es un servidor de aplicaciones J2EE bajo licencia LGPL², por su parte WildFly Application Server es un sucesor del JBoss AS un servidor de código abierto discontinuado, actualmente WildFly sigue siendo gratuito y de código abierto, se mantiene en constantemente actualizado por una red mundial de, a diferencia de otros servidores web gratuitos, para WildFly existe gran cantidad de documentación, y debido a su amplia comunidad de colaboradores cuenta con actualizaciones y soporte colaboradores (Balbuca Ramones & Ortiz Ramírez, 2017) obtenido de (WildFly.org, n.d.).

JSF y PrimeFaces

Java Server Faces (JSF) es el marco de interfaz de usuario (UI) orientado a componentes estándar para la plataforma Java EE. Java y Java Server Faces son marcas comerciales registradas de Oracle (JavaServerFaces.org, 2020), JSF es considerado como un Framework, Pagés Chacón (2010) lo describe como un Framework que brinda un acercamiento a los componentes de interfaz y las clases que lo representan, capaz de manejar sus propios eventos.

Para facilitar la construcción de aplicaciones web, se desarrollaron librerías de componentes como PrimeFaces o RichFaces, las cuales son ricas en interfaz de usuario. PrimeFaces es una suite de componentes Open Source³, además de contar con mucha documentación de apoyo, provee un conjunto de componentes enriquecidos, Ajax

¹ Capacidad de integrar más de una capa: capa de presentación, capa de negocios, capa de datos.

² En inglés GNU Lesser General Public License (Licencia Pública General para Bibliotecas de GNU),

³ Código abierto, que puede acceder al código fuente de manera gratuita.

incorporado, ligero, etc. Entre las dos opciones más conocidas del mercado Palacios et al. (2018) concluye que PrimeFaces ofrece un mejor rendimiento que RichFaces en páginas web desarrolladas bajo JSF, pero recalca que con una mínima diferencia entre las dos y en ambientes controlados.

Para seleccionar un Entorno de Desarrollo Integrado (IDE⁴) para el desarrollo del sistema se toman varios puntos, el tamaño del producto a desarrollar, la familiarización con las herramientas del IDE, la documentación disponible y la disponibilidad de plugins o complementos, entre otras cosas. Las opciones más comunes para el desarrollo de aplicaciones JEE son: NetBeans y Eclipse. Ponce Briones (2016) menciona que el IDE eclipse cuenta con un mayor número de usuarios en el ámbito académico como en el laboral, su amplio kit de herramientas y plugins lo colocan en ventaja.

Computación en la nube

De acuerdo con Rodríguez (2019) , la computación en la nube o Cloud Computing es conocida por analistas como un nuevo modelo de informática, la evolución de un conjunto de tecnologías orientadas directamente al enfoque de la infraestructura de TIC de las empresas, no solo es internet. Son un nuevo modelo tecnológico que brindan soluciones empresariales a través del internet, no requiere de instalación de ninguna herramienta en el escritorio, ni pagar por tecnologías sin usarlas

Los conceptos de computación en la nube son diferentes dependiendo de cada autor ya que no existe una definición universal, sin embargo, el NITS⁵ define a la computación en la nube como:

Un modelo que permite el acceso bajo demanda a través de la red a un conjunto compartido de recursos de computación configurables (redes, servidores, almacenamiento, aplicaciones y servicios) que se pueden aprovisionar rápidamente con el mínimo esfuerzo de gestión o interacción del proveedor del servicio. Mell & Grance (citado por Aguilar, 2012).

Sin embargo Aguilar L.J. , 2012, p.5 menciona que: “La nube es un conjunto de hardware y software, almacenamiento, servicios e interfaces que facilitan la entrada de la información como un servicio”. Una definición aceptada y compartida hasta el momento de la

⁴ Del inglés Integrated Development Environment

⁵ El NIST (National Institute of Standards and Technology) es una Agencia del Departamento de Comercio de los Estados Unidos. Dentro del NIST, el Computer Security Resource Center (CSRC) se encarga de los estándares de las Tecnologías de la Información.

redacción de esta investigación, los proveedores de servicios en la nube cumplen con esta definición.

Proveedores de servicios en la nube

Ofrecen soluciones que incluyen redes e infraestructuras de computación mediante plataformas y soluciones, son empresas que operan con centros de datos propios, apoyados en servicios de virtualización y su experiencia en alojamiento de datos y aplicaciones. Ofrecen a las empresas soluciones a menor costo de lo que valdría implementar un hardware completo de forma local, ofrecen también múltiples herramientas dentro de la plataforma para que los desarrolladores puedan acceder a ellas, desarrollar aplicaciones y alojarlas dentro de la plataforma.

Entre las plataformas más conocidas de acuerdo con el Cuadrante de Gartner, están (Figura 8):

- Amazon Web Services (AWS).
- Microsoft Azure.
- Google Cloud.



Fig. 8. Magic Quadrant para servicios de infraestructura y plataforma en la nube
Fuente: (Gartner, 2021)

1.4.2. El Gestor PostgreSQL

PostgreSQL es un Sistema Gestor de Bases de Datos (SGBD), es relacional y Open Source, compite con Oracle y MySQL como los SGBD más usados, se le atribuye características de estabilidad, confianza, rendimiento y capacidad de almacenamiento (Flores Castro, 2018).

Pérez Román (2020) menciona que PostgreSQL se ha convertido en una de las bases de datos relacionales más usadas debido a que cuenta con diversas ventajas tales como:

- Es una herramienta gratuita.
- Es multiplataforma.
- Presenta un sistema de alta disponibilidad.
- Implementa la mayor parte de las funcionalidades principales del estándar SQL.
- Incorpora una herramienta gráfica para la fácil administración de las bases de datos de manera intuitiva.
- Es bastante robusta y fiable.
- Cumple con el protocolo ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad de los datos)
- Cuenta con un gran soporte y una infinidad de foros y páginas donde consultar.

Otro aspecto importante que Pérez Román (2020), toma en cuenta es el rendimiento, de acuerdo a la figura 9, muestra los tiempos de ejecución (ms) de PostgreSQL frente a MongoDB una base de datos no relacional, los resultados muestran trabajos a la par sin embargo PostgreSQL es relacional.

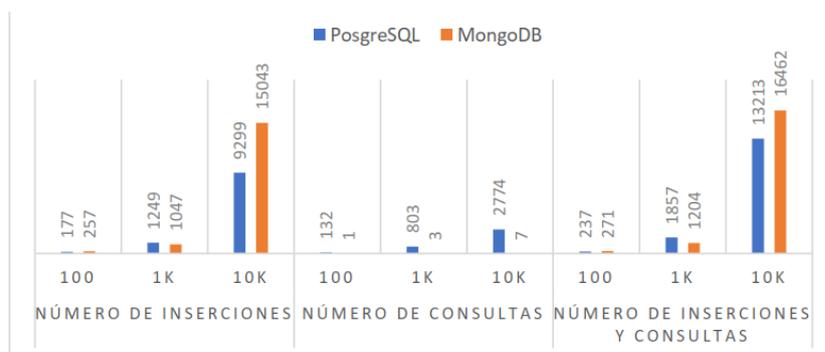


Fig. 9. Tiempos de ejecución
Fuente: Pérez Román (2020)

Es compatible con Eclipse IDE ya que cuenta con plugins para la conexión con PostgreSQL, de igual manera es compatible con WildFly. Entre muchos beneficios y debido a los requerimientos de librerías para el software de desarrollo es determinada como la mejor opción.

1.4.3. Marco de trabajo

Las Metodologías de Desarrollo de Software como objetivo principal presentan un conjunto de técnicas tradicionales, modernas y ágiles de modelado de sistemas que permitirían desarrollar software con calidad, incluyendo heurísticas de construcción y criterios de comparación de modelos de sistemas. La ingeniería de software por su parte aplica un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software (Molina Montero et al., 2018).

(Egas & Játiva, 2008) señala que no existe una metodología universal que garantice el éxito a cualquier proyecto de desarrollo de software, debido a que toda metodología se debe adaptar al contexto del proyecto. Las metodologías tradicionales han intentado abordar la mayor cantidad de situaciones de contexto del proyecto, exigiendo un esfuerzo considerable para ser adaptadas, sobre todo en proyectos pequeños y con requisitos muy cambiantes, por otra parte, las metodologías ágiles ofrecen una solución a los proyectos con esas características, además que su metodología es sencillez, tanto en el aprendizaje como en la aplicación, reduciendo los costos.

¿Qué es Scrum?

Scrum no es un acrónimo, su nombre fue tomado en referencia a una formación estratégica del deporte rugby. En su lugar Scrum es un marco de trabajado iterativo e incremental que se define dentro de las metodologías ágiles, para desarrollar, entregar y mantener productos complejos. Scrum mediante sus procesos brinda la capacidad de mejorar continuamente el producto, equipo y el entorno de trabajo. Diariamente la utilidad de scrum está a prueba contra la complejidad tecnológica.(Schwaber & Sutherland, 2020).

Schwaber & Sutherland también detalla que el marco de trabajo Scrum consiste en Equipos Scrum y sus roles, eventos, artefactos y reglas asociadas. Cada componente sirve a un propósito específico y del funcionamiento de cada uno de ellos dependerá el éxito de Scrum.

La tabla 1.4 detalla los roles de Scrum de acuerdo con la definición de Kuz et al. (2018):

TABLA 1.4
ROLES SCRUM

| Rol | Descripción | Función |
|----------------------|---------------------------------------|---|
| Product Owner | Dueño del producto | Decide qué trabajo deberá ser realizado. |
| Scrum Master | El líder del equipo | Ayuda al equipo y a la organización a hacer el mejor uso de scrum. |
| Equipo de desarrollo | Los miembros del equipo de desarrollo | Construyen el producto en forma incremental, en una serie de sprints. |

Fuente: propia.

Posterior a la definición de los roles se definen los eventos scrum siendo este el punto de partida para la ejecución de la metodología en el proyecto, el trabajo constante de los grupos, garantiza el éxito en la entrega del objeto en desarrollo.

Procesos

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija conocidos como iteraciones, cada iteración entrega un incremento del producto final como resultado, la figura 10 describe el ciclo de vida Scrum: los procesos, las iteraciones y los artefactos, se muestra un diseño estándar que se adapta de acuerdo a los requerimientos (Kuz et al., 2018). El punto positivo de esta metodología y que beneficia al proyecto es el control que ejerce, y la forma de actualizarse en caso de existir cambios.

Las actividades que se llevan a cabo en Scrum son:

- La planificación de la iteración
- La ejecución de la iteración
- La inspección y adaptación

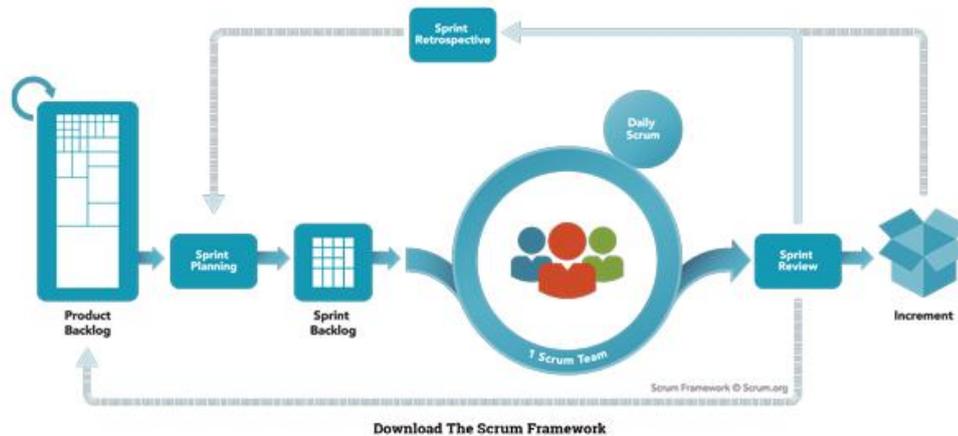


Fig. 10. Procesos Scrum
Fuente: scrum.org

Eventos de Scrum

Los eventos en Scrum están predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Los eventos finalizan siempre que se alcance el objetivo del evento dentro del límite de tiempo del sprint. Una vez que comienza un Sprint, su duración es fija y no puede acortarse o alargarse. (Schwaber & Sutherland, 2020). La tabla 1.5 detalla los eventos de scrum:

TABLA 1.5
EVENTOS DE SCRUM

| Evento | Descripción |
|---|---|
| Sprint | Bloque de tiempo en el cual se crea un incremento del producto a entregarse, por lo tanto, es el contenedor para el resto de los eventos del Scrum. |
| Sprint Planning (Planificación del Sprint) | Reunión que realiza todo el equipo scrum al inicio de cada Sprint, con el fin de planear las actividades siguientes y sincronizar lo realizado. |
| Daily Scrum (Scrum diario) | Reunión diaria de 15 minutos del equipo de desarrollo para revisar los avances del sprint. |
| Sprint Review (Revisión del Sprint) | Reunión al finalizar el Sprint que lleva a cabo una revisión para verificar el cumplimiento del sprint. |
| Sprint Retrospective (Retrospectiva de Sprint) | Oportunidad para revisar decadencias e identificar mejoras para el siguiente sprint. |

Fuente: (Schwaber & Sutherland, 2017).

Definiendo los roles se identifican quienes formarán parte del equipo Scrum, posteriormente la planificación de los eventos scrum marca otro punto fundamental de la metodología Scrum, su correcto cumplimiento, la planificación y el trabajo en equipo recaerá desde este punto en el Scrum Master.

Artefactos de Scrum

De acuerdo con Schwaber & Sutherland (2017), los artefactos de Scrum son elementos que están diseñados para mantener la transparencia y el registro de la información clave del proceso Scrum. Los artefactos son detallados en la tabla 1.6. además de adjuntar sus requisitos para la obtención:

TABLA 1.6
ARTEFACTOS SCRUM

| Artefacto | Descripción | Requisitos |
|--|---|--|
| Product Backlog (Lista de producto) | Lista ordenada de todos los requisitos necesarios en el producto, los elementos de esta lista pueden sufrir cambios al igual que el producto. | Historias de usuario y especificaciones de requerimientos tomadas del Product Owner. |
| Sprint Backlog (Lista de tareas) | Conjunto de tareas del Product Backlog para ser ejecutado por los miembros del equipo de desarrollo durante el Sprint. | Debe haberse realizado el Sprint Planning para definir el Sprint Backlog |
| Incremento | Resultado de todos los elementos del Product Backlog que se han dado por finalizado durante el Sprint. | La suma de todos los elementos de la Product Backlog completados durante un Sprint |

Fuente:(Schwaber & Sutherland, 2017)

Los artefactos forman parte de las evidencias del correcto funcionamiento del Scrum, es importante que no falte un artefacto; pues de pasar significaría un incumplimiento de una tarea, y por lo tanto de Scrum. La obtención de los artefactos se obtiene en el mismo orden de la tabla 1.4, siguiendo la orden marcada en la figura 9.

Capítulo II

2. Desarrollo del prototipo

Para desarrollo del prototipo Web que gestione el test adaptativo informatizado (APP-TAI) se aplica la metodología Scrum, este marco de trabajo define las principales actividades en cuatro grupos de tareas:

- Aplicación del marco de trabajo
- Definición de las especificaciones
- Desarrollo del sistema
- Verificación del funcionamiento.

Cada actividad cuenta con varias tareas que generan anexos, estos son resultados de tareas fundamentales que forman o formaron parte de planificación y la organización del proyecto y su respectiva documentación, el proyecto parte desde la definición de roles de la metodología para el desarrollo.

2.1. Planteamiento del proyecto

La fase inicial para la implementación de scrum parte con la definición de roles y la visión del proyecto para posteriormente definir la lista de prioridades que se deben tener en cuenta a la hora del desarrollo, así también fijar los sprints es considerada en la fase de iniciación.

2.1.1. Implementación de Scrum

Definición de roles

La Tabla 2.1. muestra los miembros del equipo Scrum y sus roles, también señala las actividades esenciales que deberán cumplir para el funcionamiento del desarrollo del sistema.

TABLA 2. 1
ROLES SCRUM

| Rol | Nombre | Responsabilidades |
|---------------|---------------------|--|
| Product Owner | Marcelo Jurado MSc. | Emitir los requisitos del software. |
| Scrum Máster | Marcelo Jurado MSc. | Controlar la aplicación de la metodología. |
| Team Scrum | Edwin Anrango | Codificación del sistema. |

Fuente: Propia

Responsabilidades SCRUM

Los roles son asignados conociendo las actividades que cada miembro va a cumplir dentro del proceso de desarrollo y sobre los cuales tiene responsabilidad.

a) Product Owner.

Quien ocupa este rol hará uso directo el proyecto final y por lo tanto conoce las funciones que el producto debe tener, de él se obtiene el listado de tareas a realizar durante la elaboración del proyecto a partir previamente de la creación de las historias de usuario en las que detallan los requisitos (Figura 11).

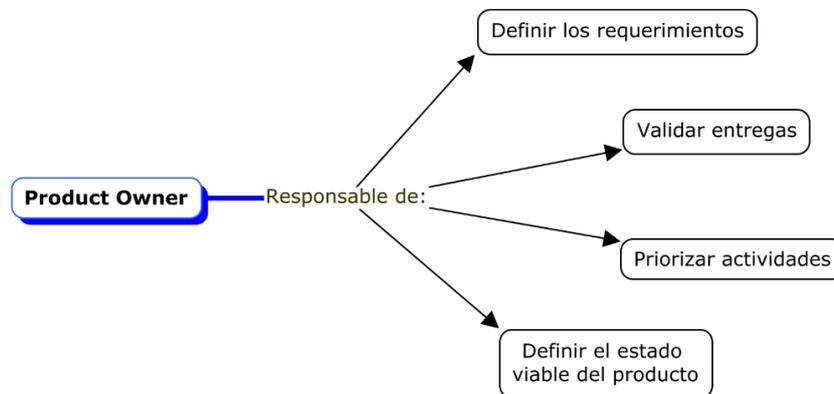


Fig. 11. Responsabilidades del Product Owner
Fuente: propia

b) Scrum Master

Es el líder del equipo y ayuda en la gestión más ágil y la superación de obstáculos que entorpezcan el trabajo del equipo, es responsable de que se cumplan todas las prácticas Scrum en el tiempo correcto (Figura 12).

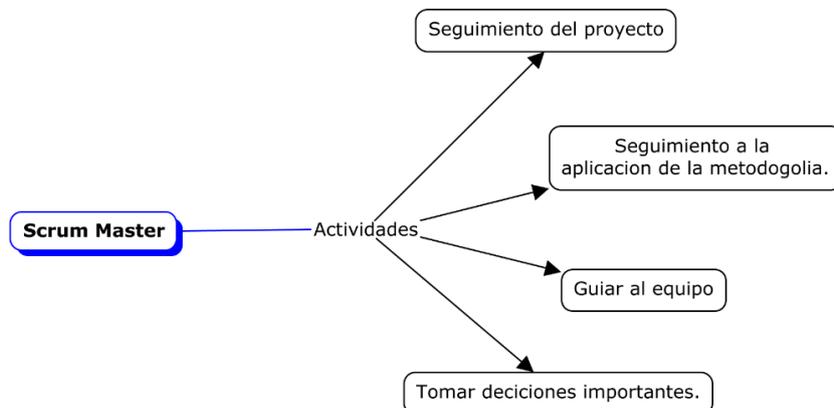


Fig. 12. Responsabilidades del Scrum Master
Fuente: propia

c) Equipo de desarrollo

Conformado para el desarrollo de esta aplicación por un integrante es el encargado de la entrega del producto final, como lo muestra la Figura 13, entre sus actividades están el análisis, diseño y desarrollo de software, así como también de las pruebas de funcionamiento.

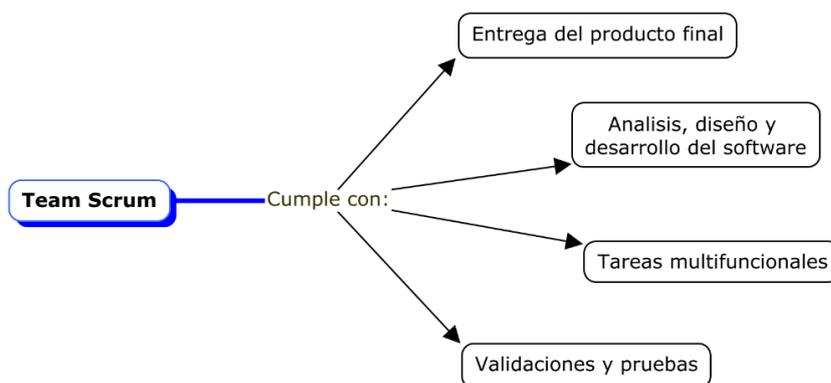


Fig. 13. Responsabilidades del Team Scrum
Fuente: propia

Estructura de trabajo

Las actividades cumplen con la aplicación de la metodología Scrum, sin existir excepciones de ningún tipo, cada actividad realizada se lo hace cumpliendo con la metodología de trabajo de acuerdo con la figura 14.

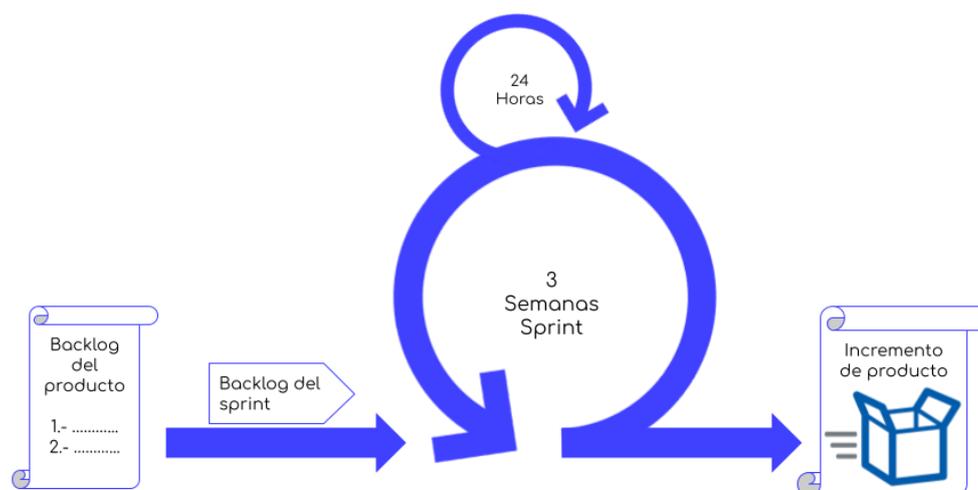


Fig. 14. Procesos Scrum
Fuente: propia

2.1.2. Planificación del Sprint

La siguiente tabla 2.2 se muestra la planificación de los sprints declarando la fecha de inicio y la fecha de finalización, con el tiempo que se demoró en realizar la actividad.

TABLA 2. 2
PLANIFICACIÓN DE LOS SPRINT

| Nro. Sprint | Semanas 2020-2021 | Fecha inicio | Fecha fin | Horas Disponibles |
|-------------|-------------------|--------------|------------|-------------------|
| Sprint 0 | Semana 1 | 30/11/2020 | 11/12/2020 | 80 |
| | Semana 2 | | | |
| Sprint 1 | Semana 3 | 14/12/2020 | 25/12/2020 | 80 |
| | Semana 4 | | | |
| Sprint 2 | Semana 5 | 28/12/2020 | 15/01/2021 | 120 |
| | Semana 6 | | | |
| | Semana 7 | | | |
| Sprint 3 | Semana 8 | 18/01/2021 | 29/01/2021 | 80 |
| | Semana 9 | | | |
| Sprint 4 | Semana 10 | 01/02/2021 | 12/02/2021 | 80 |
| | Semana 11 | | | |

Fuente: Propia

2.1.3. Análisis de Requerimientos

Cumpliendo con el marco de trabajo Scrum el desarrollo del proyecto parte del levantamiento de los requisitos del software mediante historias de usuario, las cuales posteriormente formarán parte del Product Backlog siendo el primer artefacto que genera Scrum. Los requisitos son emitidos por el Product Owner durante varias reuniones, a su vez las historias de usuario fueron aprobadas por el mismo y el equipo de desarrollo.

Historias de usuario

Las historias de usuario fueron emitidas por parte de los usuarios del sistema y se registraron en las siguientes tarjetas que detallan los requerimientos que el usuario plantea, con la validación respectiva, junto con el resto de los detalles importantes para la elaboración del Product Backlog, los requisitos para el usuarios participante fueron obtenidos al azar del grupo de estudiantes de la facultad de educación Ciencia y Tecnología, mientras los requisitos del usuario evaluador fue brindada por el experto del proyecto MSc. Marcelo Jurado. Las historias de usuario obtenidas se detallan en las tablas de la 2.3 a la 2.16.

TABLA 2. 3
HISTORIA DE USUARIO NRO. 1

Historia de Usuario

| | | |
|--|---------------------------|----------------------------|
| Numero: H1 | Usuario: Evaluador | |
| Nombre: Presentación del sistema | | |
| Prioridad: Media | Riesgo: Bajo | Puntos estimados: 2 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador y Product Owner quiero acceder a una pantalla de bienvenida al test, además que las colores sean claras y en tono azul. | | |
| Validación: Un index dará la bienvenida y posteriormente recorrerá las funciones de la aplicación de acuerdo con el usuario, el tono a usar será azul con tonos sin color. | | |

Fuente: Propia

TABLA 2. 4
HISTORIA DE USUARIO NRO. 2

Historia de Usuario

| | | |
|---|------------------------------|----------------------------|
| Numero: H2 | Usuario: Participante | |
| Nombre: Disponibilidad del servidor. | | |
| Prioridad: Alta | Riesgo: Alto | Puntos estimados: 8 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como participante quiero que no se cierre la sesión, que exista disponibilidad del servidor, y no se cuelgue el sitio. | | |
| Validación: El sistema controlará las sesiones con los métodos propios de Java, y trabajará con el servidor de aplicaciones WildFly montado en un servidor en línea que ofrezca buena disponibilidad. | | |

Fuente: Propia

TABLA 2. 5
HISTORIA DE USUARIO NRO. 3

Historia de Usuario

| | | |
|---|---------------------------|-----------------------|
| Numero: H3 | Usuario: Evaluador | |
| Nombre: Método de selección de preguntas. | | |
| Prioridad: Alta | Riesgo: Alto | Estimación: 10 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador requiero que los participantes respondan un test distinto adaptado a cada uno, con el fin de obtener una puntuación más real que refleje sus conocimientos. | | |
| Validación: El test contará con la implementación métodos de estimación Bayesiana para la selección de preguntas, esto generará un test adaptativo diferente para cada participante. | | |

Fuente: Propia

TABLA 2. 6
HISTORIA DE USUARIO NRO. 4

Historia de Usuario

| | | |
|---|---------------------------|----------------------|
| Numero: H4 | Usuario: Evaluador | |
| Nombre: Control de usuarios. | | |
| Prioridad: Media | Riesgo: Alto | Estimación: 8 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador quiero un control de participantes mediante una cuenta de usuario que me entrega información básica del participante, previo a la evaluación. | | |
| Validación: El participante se registrará sus datos en el sistema de forma obligatoria antes de realizar el test. | | |

Fuente: Propia

TABLA 2. 7
HISTORIA DE USUARIO NRO. 5

Historia de Usuario

| | | |
|--|---------------------------|-----------------------|
| Numero: H5 | Usuario: Evaluador | |
| Nombre: Ingreso al test. | | |
| Prioridad: Alta | Riesgo: Alto | Estimación: 10 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador requiero que los participantes ingresen al test con un código de acceso, además de controlar el número de oportunidades. | | |
| Validación: El ingreso al test se realizará mediante un código de acceso o clave de autorización definida por el evaluador. | | |

Fuente: Propia

TABLA 2. 8
HISTORIA DE USUARIO NRO. 6

Historia de Usuario

| | | |
|--|---------------------------|----------------------|
| Numero: H6 | Usuario: Evaluador | |
| Nombre: Reportes. | | |
| Prioridad: Alta | Riesgo: Medio | Estimación: 6 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador deseo acceder al listado de estudiantes que rindieron el test, además de un reporte individual y general de manera numérica y gráfica. | | |
| Validación: El evaluador estará en la capacidad de crear reportes con el listado de los participantes, gráficos del resultado general de su evaluación, gráficos individuales de los participantes, reportes de la evaluación. | | |

Fuente: Propia

TABLA 2. 9
HISTORIA DE USUARIO NRO. 7
Historia de Usuario

| | | |
|---|---------------------------|----------------------|
| Numero: H7 | Usuario: Evaluador | |
| Nombre: Base de datos acorde a especificaciones. | | |
| Prioridad: Alta | Riesgo: Medio | Estimación: 8 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como Evaluador requiero modificar información de la base de ítems además de contar con una base de datos segura. | | |
| Validación: El test contará con la base de datos PostgreSQL la cual se conectará por medio de un driver al aplicativo. Se garantiza la integridad de la información por lo cual se controlará los errores que pueda generar al poder modificar información de la Base de ítems. | | |

Fuente: Propia

TABLA 2. 10
HISTORIA DE USUARIO NRO. 8
Historia de Usuario

| | | |
|---|---------------------------|----------------------|
| Numero: H8 | Usuario: Evaluador | |
| Nombre: Registro de participantes. | | |
| Prioridad: Media | Riesgo: Medio | Estimación: 6 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como Evaluador requiero un registro previo de los participantes que rendirán el test. | | |
| Validación: Registro previo de participantes, para lo cual se permitirá crear aulas privadas con previo registro. | | |

Fuente: Propia

TABLA 2. 11
HISTORIA DE USUARIO NRO. 9
Historia de Usuario

| | | |
|---|---------------------------|----------------------|
| Numero: H9 | Usuario: Evaluador | |
| Nombre: Carga de recursos. | | |
| Prioridad: Media | Riesgo: Medio | Estimación: 3 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador quiero que el sistema pueda mostrar recursos digitales como fotografías o imágenes. | | |
| Validación: El sistema permitirá gestionar recursos digitales mismos que se mostrará al participante al momento de presentarle un ítem durante el test. | | |

Fuente: Propia

TABLA 2. 12
HISTORIA DE USUARIO 10
Historia de Usuario

| | | |
|---|------------------------------|----------------------------|
| Numero: H10 | Usuario: Participante | |
| Nombre: Interfaz amigable. | | |
| Prioridad: Alta | Riesgo: Medio | Puntos estimados: 4 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como participante quiero una Interfaz amigable, simple, que no genere confusión al momento de realizar el test. | | |
| Validación: El proceso del test será sencillo, no tendrá trabas, explicará el funcionamiento y no se diferenciará de otros test informatizados. | | |

Fuente: Propia

TABLA 2. 13
HISTORIA DE USUARIO 11
Historia de Usuario

| | | |
|--|---------------------------|----------------------------|
| Numero: H11 | Usuario: Evaluador | |
| Nombre: Autenticación de los usuarios | | |
| Prioridad: Alta | Riesgo: Alta | Puntos estimados: 4 |
| Programador responsable: Edwin Anrango | | |
| Descripción: El evaluador requiere autenticación de los usuarios, además de validar el uso del correo electrónico para todo usuario. | | |
| Validación: El ingreso al sistema será mediante un Login o verificación de credenciales dependiendo del usuario. | | |

Fuente: Propia

TABLA 2. 14
HISTORIA DE USUARIO 12
Historia de Usuario

| | | |
|---|---------------------------|----------------------------|
| Numero: H12 | Usuario: Evaluador | |
| Nombre: Gestión de ítems | | |
| Prioridad: Alta | Riesgo: Alta | Puntos estimados: 4 |
| Programador responsable: Edwin Anrango | | |
| Descripción: Como evaluador quiero gestionar las preguntas del test (crear y modificar, eliminar). | | |
| Validación: El evaluador podrá gestionar las unidades de la asignatura y sus temas (crear y modificar, eliminar). | | |

Fuente: Propia

TABLA 2. 15
HISTORIA DE USUARIO 13
Historia de Usuario

| | |
|--------------------|---------------------------|
| Numero: H13 | Usuario: Evaluador |
|--------------------|---------------------------|

Nombre: Gestión de temas y unidades.
Prioridad: Alta **Riesgo:** Alta **Puntos estimados:** 4
Programador responsable: Edwin Anrango
Descripción:
Como evaluador quiero gestionar las unidades de la asignatura y sus temas, entre las funciones quiero que me permita crear y modificar, eliminar.

Validación:
El evaluador podrá gestionar las preguntas del test (crear y modificar, eliminar).

Fuente: Propia

TABLA 2. 16
HISTORIA DE USUARIO 14
Historia de Usuario

Numero: H14 **Usuario:** Evaluador
Nombre: Bloquear información sensible.
Prioridad: Media **Riesgo:** Medio **Puntos estimados:** 4
Programador responsable: Edwin Anrango
Descripción:
Como evaluador quiero que la gestión de información sensible se encuentre bloqueada con clave de acceso.

Validación:
La gestión de unidades, temas e ítems estarán disponibles bajo un perfil de administración dentro del usuario evaluador.

Fuente: Propia

Definición del Product Backlog

Posterior al registrar las historias de usuario se realiza una recopilación de estas con el fin de generar un registro en el cual se listen las características del producto, a esta lista se lo denomina Product Backlog (Tabla 2.17) de acuerdo con el manual de Scrum.

TABLA 2. 17
PRODUCT BACKLOG

| ID | Descripción | Prioridad | Riesgo | Esfuerzo |
|-----|--|-----------|--------|----------|
| H1 | Presentación del sistema. | Media | Bajo | Bajo |
| H2 | Disponibilidad del servidor. | Alta | Alto | Medio |
| H3 | Método de selección de preguntas. | Alta | Alto | Alto |
| H4 | Control de usuarios. | Alta | Alto | Alto |
| H5 | Ingreso al test. | Alta | Alto | Alto |
| H6 | Reportes. | Alta | Medio | Medio |
| H7 | Base de datos acorde a especificaciones. | Alta | Medio | Alto |
| H8 | Registro de participantes. | Media | Medio | Medio |
| H9 | Carga de recursos. | Media | Medio | Medio |
| H10 | Interfaz amigable. | Alta | Medio | Medio |
| H11 | Autenticación de los usuarios. | Alta | Alto | Medio |
| H12 | Gestión de ítems. | Alta | Alto | Alto |
| H13 | Gestión de temas y unidades. | Alta | Alto | Alto |
| H14 | Bloquear información sensible. | Media | Medio | Bajo |

Fuente: Propia

Requisitos Funcionales

Especifica los requisitos que necesita el sistema APP-TAI para su correcto funcionamiento.

✓ Aplicación

TABLA 2. 18
REQUISITO FUNCIONAL - RF.APL.01

| | |
|-------------------|---|
| Numero: | RF. 01 |
| Nombre: | Test Adaptativo con técnicas de IA |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Integración de un estimador bayesiano y la base de ítems es calibrada usando la teoría de respuesta al ítem. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

TABLA 2. 19
REQUISITO FUNCIONAL - RF.APL.02

| | |
|-------------------|---|
| Numero: | RF. 02 |
| Nombre: | Tipos de usuarios |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Existirán 2 tipos de usuarios: Evaluador y participante. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

TABLA 2. 20
REQUISITO FUNCIONAL - RF.APL.03

| | |
|-------------------|---|
| Numero: | RF. 03 |
| Nombre: | Gestión del test |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | El evaluador gestiona las evaluaciones. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

TABLA 2. 21
REQUISITO FUNCIONAL - RF.APL.04

| | |
|-------------------|---|
| Numero: | RF. 04 |
| Nombre: | Gestión del banco de ítems. |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | El evaluador puede gestionar el banco de ítems mientras tenga acceso al panel de administración. |
| Prioridad: | <input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

✓ **Datos de Salida**

TABLA 2. 22
REQUISITO FUNCIONAL - RF.SAL.01

| | |
|-------------------|--|
| Numero: | RF. 05 |
| Nombre: | Entrega de reportes |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | El sistema debe entregar reportes de la nómina de estudiantes, resultados del test general, por grupo e individual de manera numérica. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

✓ **Comportamientos.**

TABLA 2. 23
REQUISITO FUNCIONAL - RF.COM.01

| | |
|-------------------|---|
| Numero: | RF. 06 |
| Nombre: | Diseño de prototipado. |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Presentación del respectivo prototipado del sistema |
| Prioridad: | <input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

Requisitos no funcionales

Especifica funciones que no están directamente asociadas a una propiedad o característica del sistema APP-TAI.

✓ **Arquitectura**

TABLA 2. 24
REQUISITO NO FUNCIONAL - RNF.ARQ.01

| | |
|-------------------|---|
| Numero: | RNF. 01 |
| Nombre: | Herramientas. |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Usará Java con IDE Eclipse JEE con WildFly como servidor de aplicaciones y el uso de PostgreSQL como gestor de base de datos. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

TABLA 2. 25
REQUISITO NO FUNCIONAL - RNF.ARQ.02

| | |
|-------------------|---|
| Numero: | RNF. 02 |
| Nombre: | Tecnologías para el desarrollo. |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | La aplicación web APP-TAI usara el framework JSF con la librería de PrimeFaces 8.0. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

✓ **Usabilidad**

TABLA 2. 26
REQUISITO NO FUNCIONAL - RNF.USA.01

| | |
|-------------------|---|
| Numero: | RNF. 03 |
| Nombre: | Estándares de usabilidad |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Diseño amigable y que no permita que el usuario genere errores. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

TABLA 2. 27
REQUISITO NO FUNCIONAL - RNF.USA.02

| | |
|-------------------|---|
| Numero: | RF. 04 |
| Nombre: | Control de sesiones |
| Tipo: | <input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción |
| Detalle: | Desarrollo de login seguro, claves y estándares de acceso para facilitar el test y manejo de sesiones. |
| Prioridad: | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseada <input type="checkbox"/> Baja/Opcional |

Fuente: Propia

2.2. Especificaciones del algoritmo adaptativo

2.2.1. Casos de uso

Para la obtención y creación de los casos de uso primeramente hay que definir los roles, la Tabla 2.28 muestra los roles que hacen referencia a la persona o grupo de personas quienes hacen uso del sistema y las actividades que estos realizan.

TABLA 2. 28
ACTIVIDADES DE LOS ROLES

| Rol | Actividad |
|-----|-----------|
|-----|-----------|

| | |
|---------------|--|
| Participante | Accede a los test. Contesta las evaluaciones. Visualiza los resultados. |
| Evaluador | Crea, modifica o elimina aulas o grupos de trabajo. Habilita, modifica o deshabilita evaluaciones adaptativas. Crea, modifica o elimina participantes dentro de un aula. Obtiene los resultados de los estudiantes evaluados. |
| Evaluador | Crea, modifica o elimina unidades de una asignatura. |
| Administrador | Crea, modifica o elimina temas de una unidad. Crea, modifica o elimina preguntas de un tema para los test. Crea, modifica o elimina usuarios evaluadores. |

Fuente: Propia

Posterior a la definición de roles y actividades se realiza el análisis de los casos de uso y se los representa como muestra la Figura 15, detallando las actividades a las que cada uno de los usuarios puede acceder.

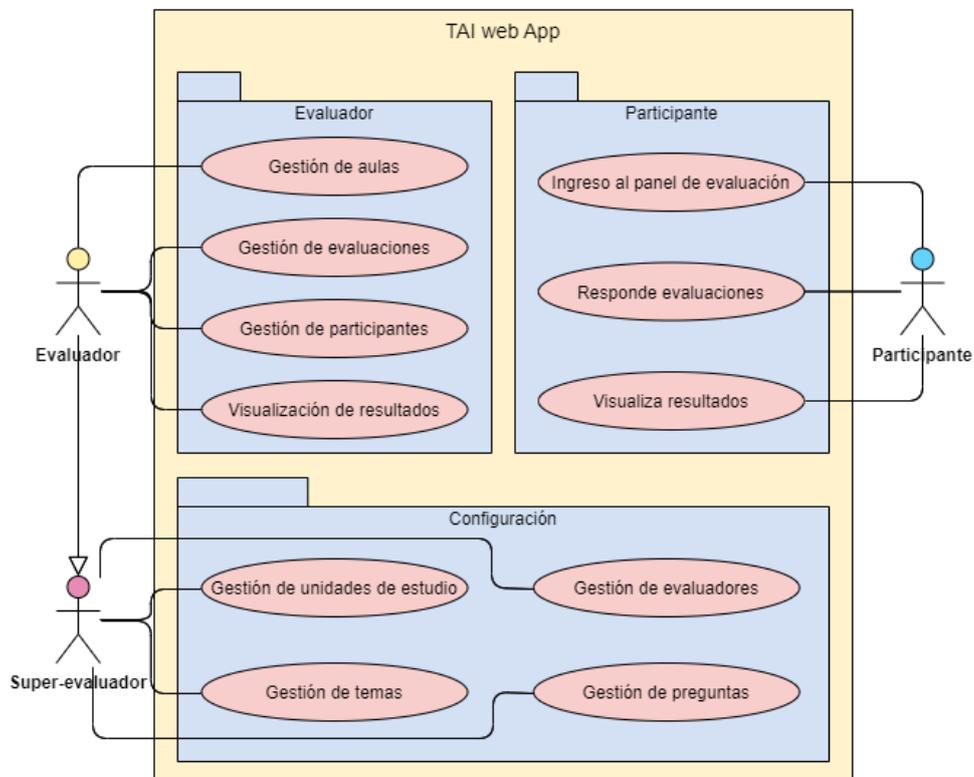


Fig. 15. Casos de uso
Fuente: propia

2.2.2. Arquitectura del sistema

Flujograma de procesos

La figura 16 indica un flujograma de procesos el cual es usado para la aplicación de las evaluaciones tradicionales, en ocasiones se realiza dos evaluaciones con diferentes ítems para obtener un conocimiento verdadero.

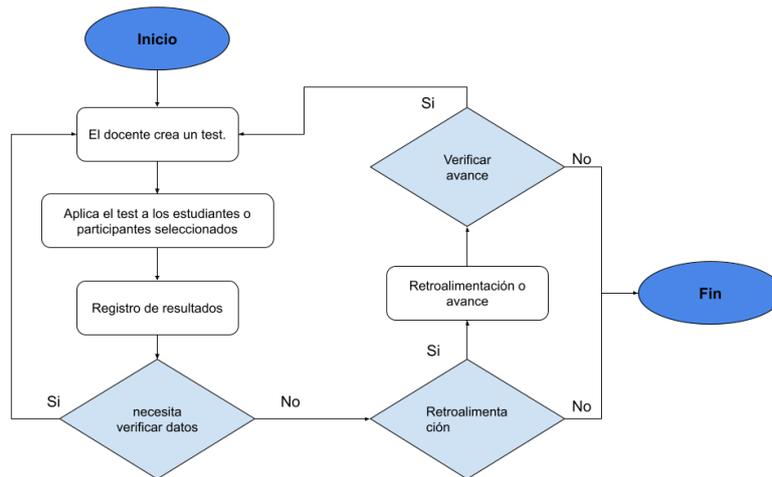


Fig. 16. Flujograma de procesos sin automatización
Fuente: propia

La figura 17 propone un modelo en la cual los procesos se llevan a cabo con la implementación del TAI, en la figura incluye la retroalimentación un proceso posterior a la evaluación la cual no se incluye en este proyecto.

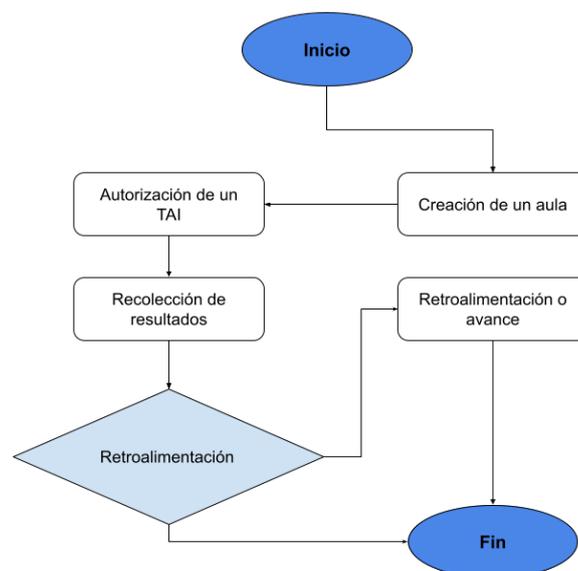


Fig. 17. Flujograma de procesos actual
Fuente: propia

Arquitectura del sistema

El funcionamiento del sistema va distribuido en 3 capas, como se muestra la Figura 18; capa de presentación, capa de negocios y capa de datos.

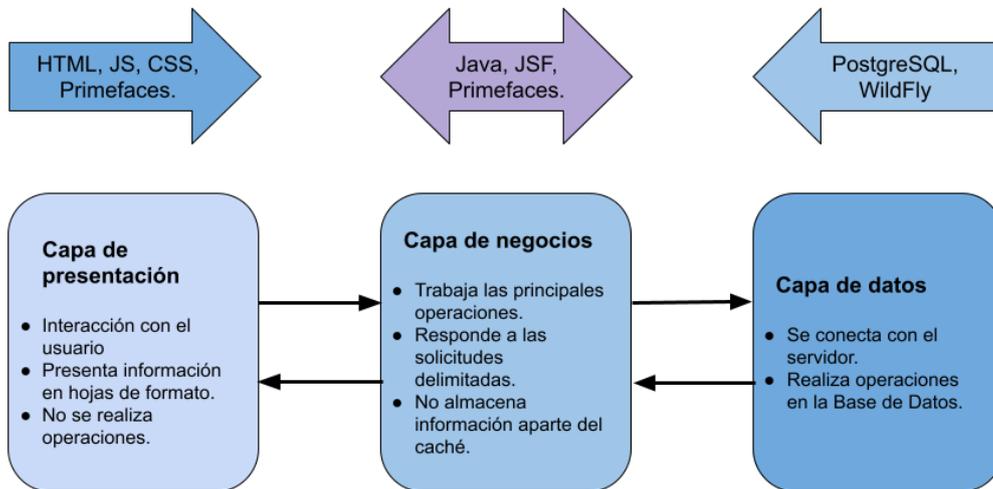


Fig. 18. Arquitectura de funcionamiento
Fuente: Propia

La figura 19 describe a detalle el funcionamiento de la arquitectura con la que trabaja el TAI-APP, y la herramienta que predomina en cada área de trabajo.

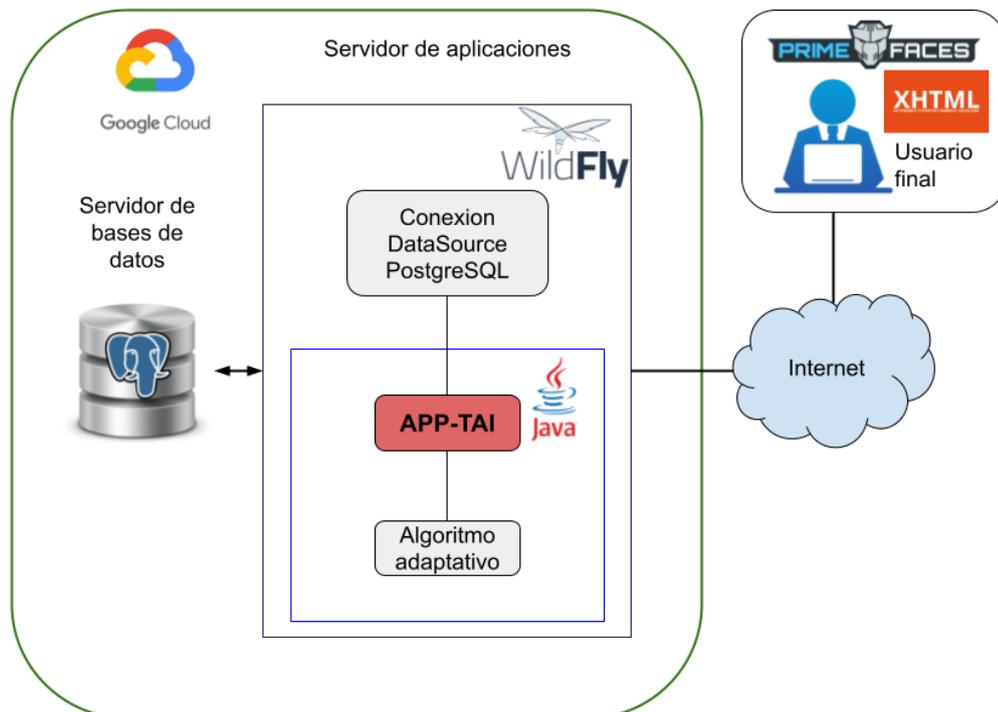


Fig. 19. Descripción de la arquitectura
Fuente: Propia

La tecnología usada permite trabajar con la arquitectura MVC (Modelo Vista Controlador) representada en la figura 20, mediante interacciones con el usuario por medio de una página XHTML que forman parte de la vista o capa de presentación se obtiene la información necesaria para realizar operaciones con java en los controladores, la tecnología JPA permite interactuar y operar a la capa de datos y la base de datos cuando esta lo requiera.

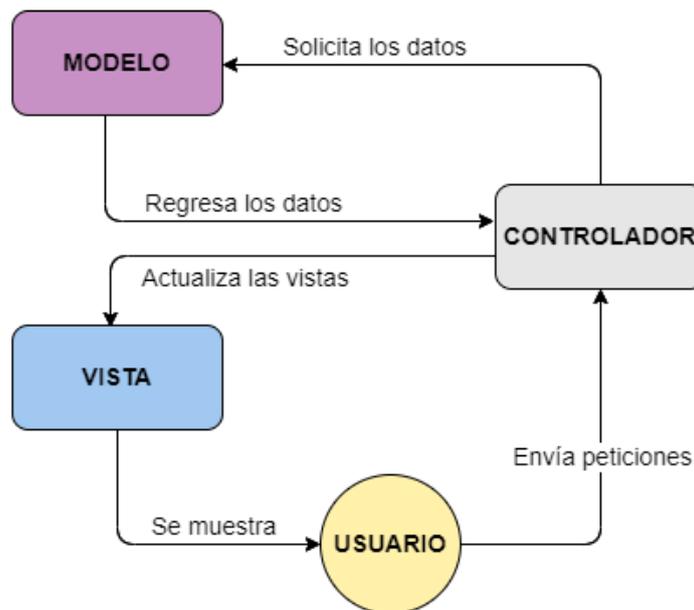


Fig. 20. Arquitectura MVC
Fuente: Propia

2.2.3. Modelado y prototipado

Prototipo del ítem

Definimos un ítem a una pregunta con un nivel de dificultad, que cumpla con un concepto, y que pueda ser útil de acuerdo con la Teoría de Respuesta al Ítem, registrada en el Banco de ítems. La figura 21 indica la estructura básica de un ítem, basada en un test tradicional, cumple con tres elementos básicos: Instrucciones, descripción de la pregunta, y opciones de respuesta para el ítem.

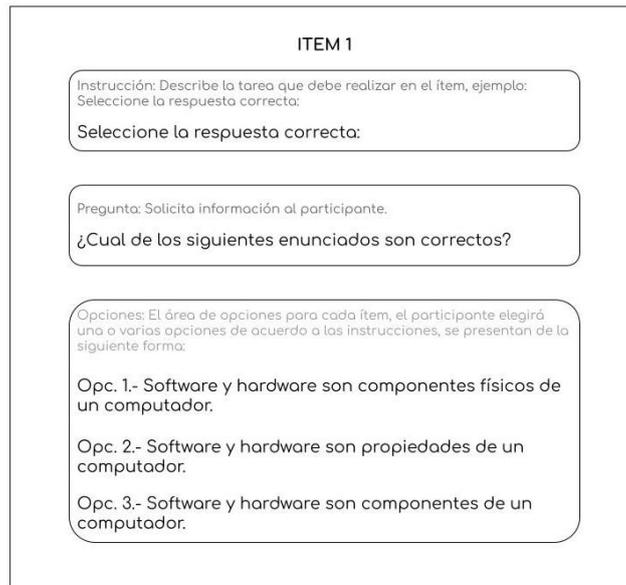


Fig. 21. Estructura de un ítem
 Fuente: propia

La figura 22 representa el estilo que tendrá la evaluación, en este prototipo se plantea representar únicamente la instrucción y las opciones.

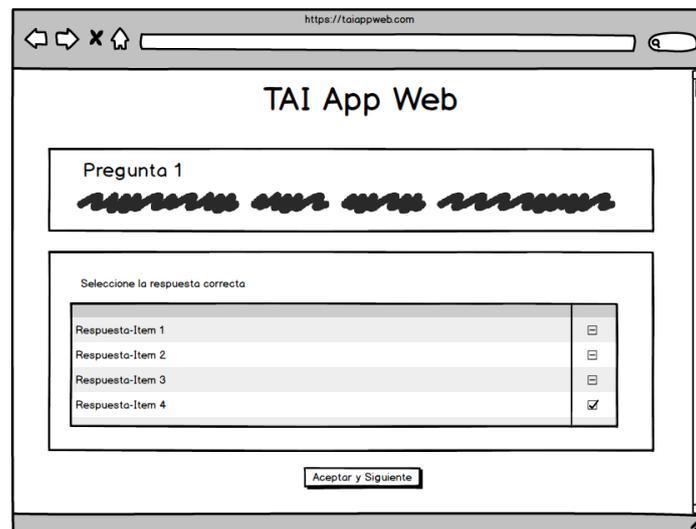


Fig. 22. Prototipo presentación del ítem
 Fuente: propia

Modelo conceptual

Modelo conceptual del sistema para mostrar las interacciones tanto del Evaluador como del participante en el cual se puede ubicar las principales clases para su funcionamiento (Figura 23), está diseñado de acuerdo con los requisitos obtenidos al realizar el análisis de los requerimientos y cumple con lo sugerido en las historias de usuario.

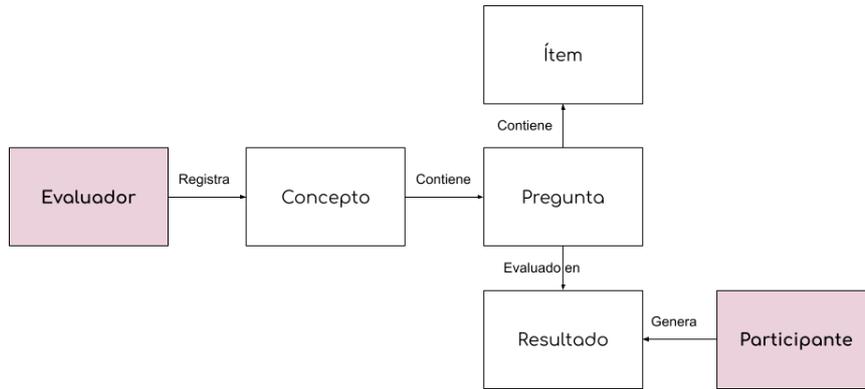


Fig. 23. Modelo conceptual del sistema
Fuente: propia

Modelo de navegación del participante

El presente modelo de navegación en la figura 24, describe el mecanismo que el participante usará para registrarse y rendir la evaluación, este proceso viene especifica de forma más detallada el proceso que va a realizar durante la navegación o interacción con el sistema.

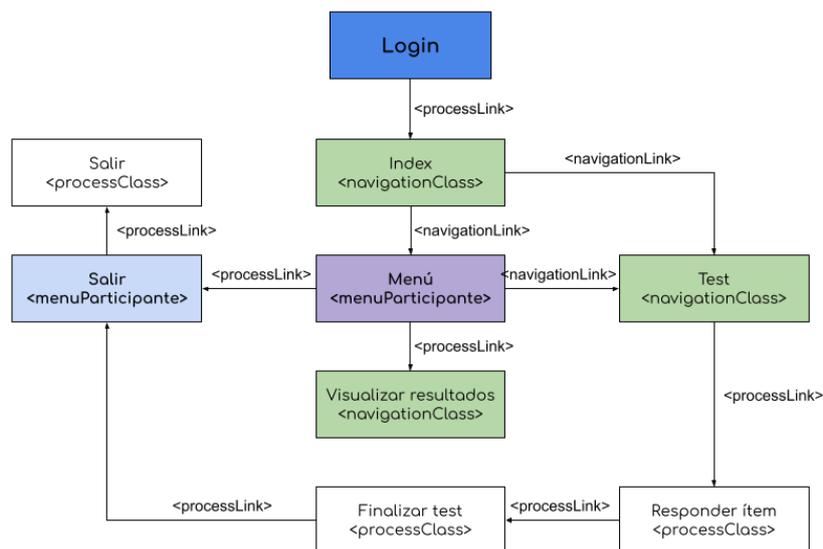


Fig. 24. Modelo de navegación del participante
Fuente: propia

También indica los elementos con los que interactuará el participante, no manipulará una entidad, solamente registrará su información y aporta con la generación de resultados al dar respuesta de una evaluación adaptativa.

Modelo de navegación del Evaluador

De acuerdo con los requerimientos del usuario, el Evaluador interactúa con tres entidades principales para su correcta funcionalidad, el diseño del modelo de navegación del evaluador se muestra de la forma en la figura 25.

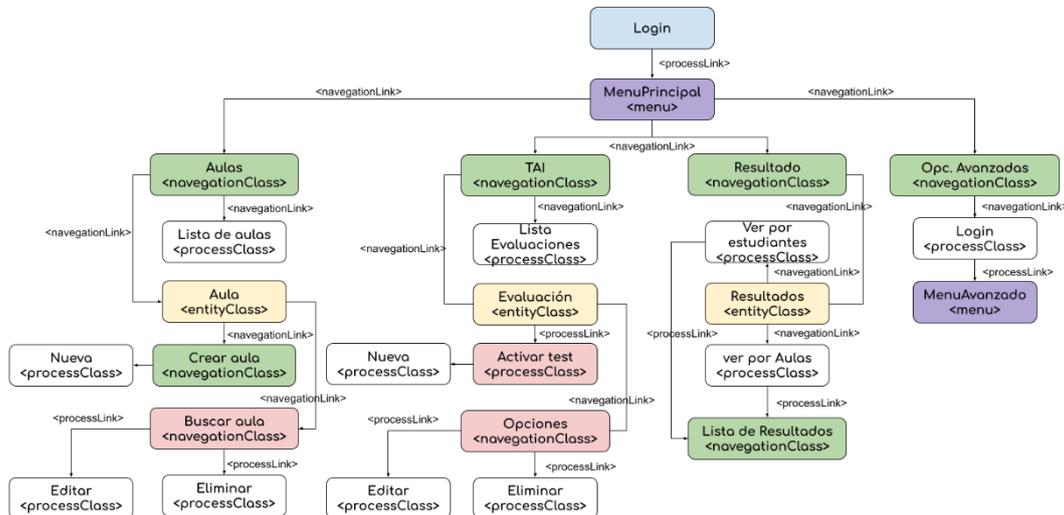


Fig. 25. Modelo de navegación Evaluador
Fuente: propia

Modelo de navegación del Evaluador con credenciales administrativas

Para la protección de los datos o la mala manipulación del banco de ítems, se aplica la opción de colocar un segundo panel bajo contraseña (Figura 26), en este panel el evaluador realiza un número superior de operaciones respecto a los demás roles dentro del sistema TAI, sus cambios afectan directamente a la evaluación final, por este sentido solamente ciertos usuarios evaluadores lo podrán usar.

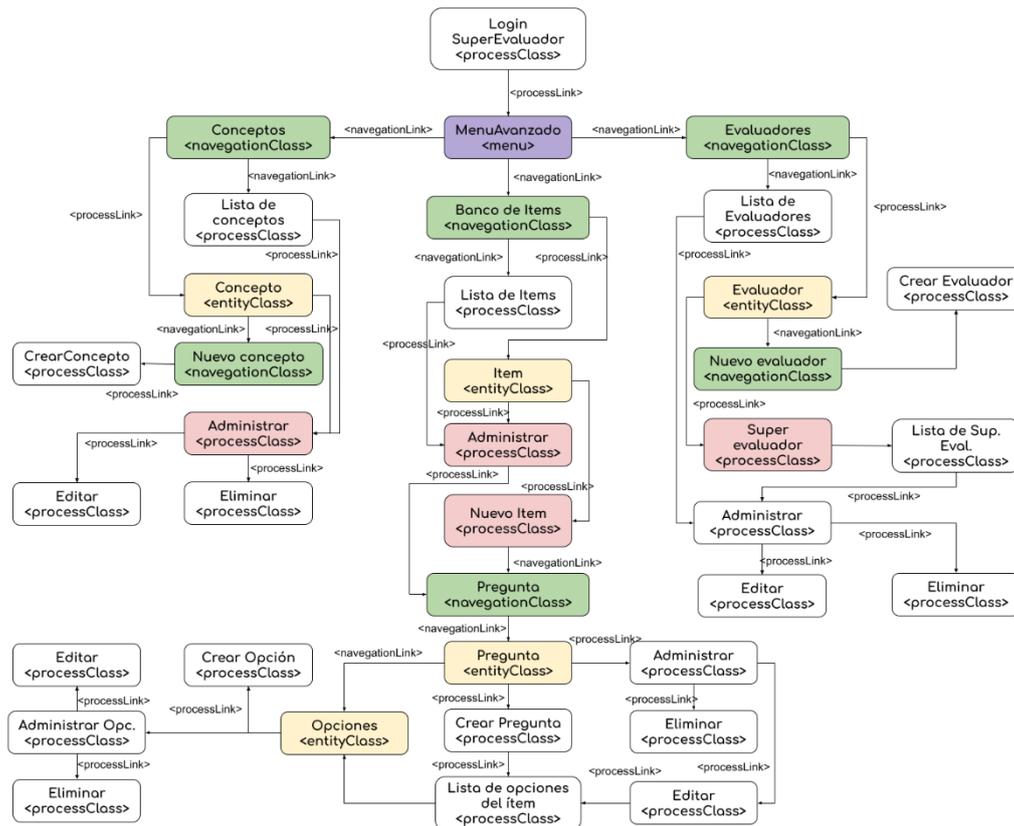


Fig. 26. Navegación del super usuario evaluador
Fuente: propia

La manipulación incorrecta de estos valores podría afectar la calibración de los ítems del TAI, las funciones principales son el registro de nuevas preguntas, modificar conceptos o agregar nuevos evaluadores, esta no es considerada como un rol, solamente se considera una medida de seguridad para evitar modificaciones o que se pierda la información en caso registrarse una vulnerabilidad.

Algoritmo Adaptativo

Este algoritmo es presentado como el núcleo de este sistema, la figura 27 muestra los procesos que debe cumplir el algoritmo del TAI para, de esta forma mantener la característica de adaptabilidad tal como se lo vino planteando.

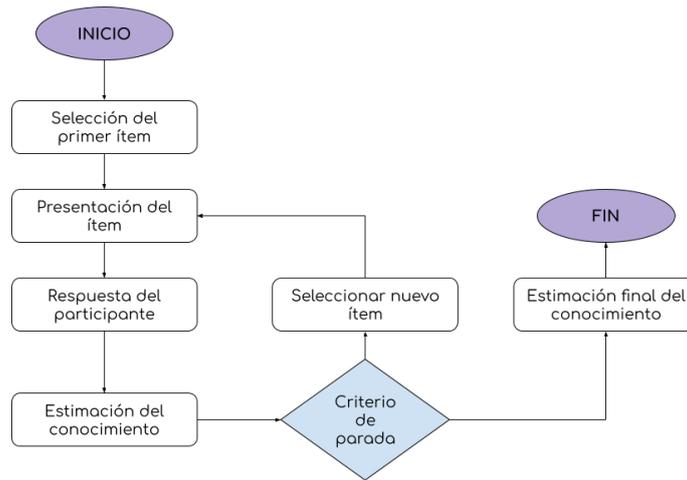


Fig. 27. Flujograma del TAI
Fuente: propia

En este proceso se realiza modificaciones en el proceso, entre las cuales se implementan estructuras de control para validar el funcionamiento, sin modificar la esencia, se mantiene la estructura principal y se representa en la figura 28.

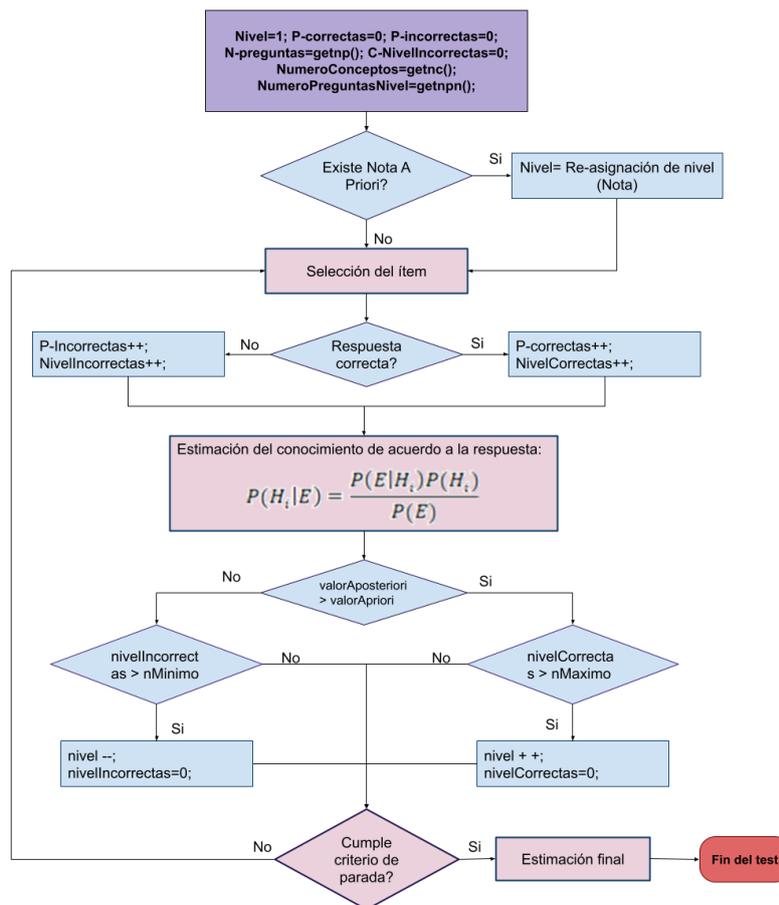


Fig. 28. Flujograma TAI-APP
Fuente: propia

El diseño del estimador parte del análisis del teorema de bayes, esta probabilidad condicional trabaja con el uso de variables independientes y variables dependientes, estas son presentadas como clases Figura 29:

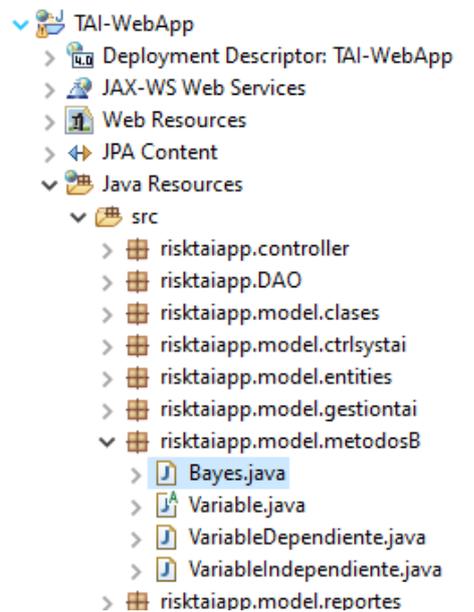


Fig. 29. Métodos Bayes
Fuente: propia

La variable independiente más importante hace referencia al valor a priori, es decir a los resultados de las estimaciones previas, para su funcionamiento se hace uso de mapas Hash (Figura 30):

```
//inicializar metodo
public void iniciarInferenciaB(float val,String nivel, float triValSi, float triValNo) {
    //inicaliza lalista de variables independientes
    notasApriori = new ArrayList<>();
    String index="item "+notasApriori.size();
    notasApriori.add(index);
    result = new VariableIndependiente("Estimaciones", notasApriori);

    //carga el primer valor; nota a priori o 0.
    Map<String, Float> estimacionProb = new HashMap<>();
    estimacionProb.put(index, (float) (val));
    result.setProbabilidades(estimacionProb);

    //inicializa la lista de variables dependientes
    List<String> itemsValores = new ArrayList<>();
    itemsValores.add("si"); //correcto
    itemsValores.add("no"); //incorrecto
    List<Variable> RespuestasItems = new ArrayList<>();
    RespuestasItems.add(result);
    items = new VariableDependiente("mantenimiento", RespuestasItems, itemsValores);

    //carga el primer valor de la variable
    ResultadosItemProbable = new HashMap<>();
    List<String> listaSoluciones = new ArrayList<>();
    listaSoluciones.add(nivel);
    MapaValoresProbables = new HashMap<>();
    mapaValoresProbables.put("si", triValSi);
    mapaValoresProbables.put("no", triValNo);
    ResultadosItemProbable.put(listaSoluciones, mapaValoresProbables);
}
}
```

Fig. 30. Método inicializador Bayes
Fuente: propia

Los valores son cargados en memoria y al finalizar el test son eliminados, sin embargo cada respuesta se guarda en la base de datos, posteriormente otro método llama al estimador y hace uso de los datos cargados para generar la estimación de conocimiento y continuar con el proceso, que dependiendo si cumple con el criterio de parada detiene el test o continúa seleccionando el siguiente ítem de acuerdo con su estimación, la figura 31 describe el método de selección usado, el ítem de que brinde un mejor aporte o este dentro del rango de probabilidad de acuerdo a la TRI será el seleccionado, el TAI se adapta para ampliar el rango de selección.

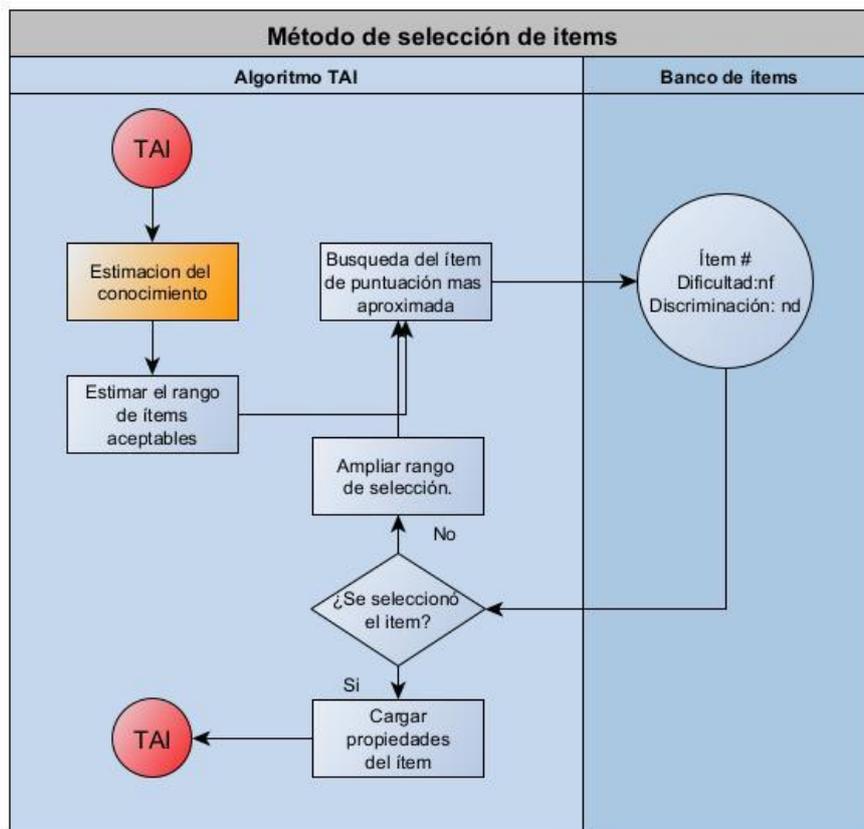


Fig. 31. Algoritmo selección de ítems
Fuente: propia

2.3. Desarrollo del software

El proceso de desarrollo de la aplicación APP-TAI se la realizó tal cual fue planificada, aplicando la metodología Scrum, de esta forma el único objetivo buscado fue cumplir con los objetivos de cada sprint determinaba, cada Sprint tenía una duración aproximada de tres semanas, cada sprint detalla las actividades cumplidas durante su proceso y se estructura de acuerdo con el manual Scrum.

2.3.1. Sprint 0

Se pone en marcha las iteraciones definidas, y arranca el proceso Scrum con la planificación del Sprint, la definición de requerimientos, el registro de historias de usuario y el análisis de la arquitectura, casos de uso y herramientas tecnológicas continuando con el orden de acuerdo con el marco de trabajo (Figura 32):



Fig. 32. Estructura del Sprint
Fuente: Propia

Cumpliendo con la estructura del sprint se obtiene la tabla 2.29 que muestra los resultados de la reunión de planificación del sprint y su sprint backlog se define con la tabla 2.30 que representa las historias de usuario que se buscan cumplir, posteriormente en la tabla 2.31 se detalla la planificación del sprint con las actividades de acuerdo con Scrum, las actividades cumplidas son detalladas en la tabla 2.32 misma que es obtenida de la reunión de revisión.

a) Reunión de planificación.

TABLA 2. 29
SPRINT 0

| Sprint 0 | |
|------------------------------|---|
| Fecha de la reunión: | 30/11/2020 |
| Asistentes: | Scrum Master, Product Owner y Development Team. |
| Fechas de inicio del sprint: | 30/11/2020 |
| Fecha de fin del sprint: | 11/12/2020 |
| Objetivos del Sprint: | <ul style="list-style-type: none"> • Recolección de requisitos y planificación del proyecto • Definición de modelo de base de datos. • Casos de uso y prototipado. |

Fuente: Propia

Sprint Backlog:

TABLA 2. 30
SPRINT BACKLOG

| ID | Historia de usuario |
|-----------|-----------------------------|
| H1 | Presentación del sistema |
| H2 | Disponibilidad del servidor |

Fuente: Propia

Planificación:

TABLA 2. 31
PLANIFICACIÓN SPRINT

| Nro. H.U. | Responsable | Fase de desarrollo | Tarea | Tipo | Tempo estimado (Horas). |
|-----------|---------------|--------------------|--|-------|-------------------------|
| H1 | Equipo Scrum | Análisis | Levantamiento de requisitos | Nuevo | 4 |
| | Desarrollador | Análisis | Definición del Backlog. | Nuevo | 4 |
| | Desarrollador | Análisis | Planificación del proyecto | Nuevo | 4 |
| | Desarrollador | Análisis | Definición de los casos de uso y diagrama de clase. | Nuevo | 6 |
| | Desarrollador | Análisis | Definición de la arquitectura del sistema. | Nuevo | 6 |
| | Desarrollador | Análisis | Documentación de la arquitectura de software | Nuevo | 8 |
| | Desarrollador | Análisis | Prototipo del área visual del sistema. | Nuevo | 5 |
| H2 | Desarrollador | Análisis | Diseño del modelo de la Base de datos. | Nuevo | 4 |
| | Desarrollador | Análisis | Creación de la base de datos preliminar | Nuevo | 2 |
| H1, H2. | Desarrollador | Análisis | Adecuación del ambiente de desarrollo, aplicaciones y librerías. | Nuevo | 3 |
| | Desarrollador | Desarrollo | Creación del proyecto APP-TAI | Nuevo | 1 |
| H2 | Desarrollador | Desarrollo | Instalación y configuración del servidor WildFly | Nuevo | 3 |
| H1, H2. | Desarrollador | Desarrollo | Creación y configuración del Data Source. | Nuevo | 4 |

Fuente: Propia

b) Reunión de revisión

TABLA 2. 32
REVISIÓN SPRINT

| Responsable | Tarea | Tempo estimado (Horas). | Tempo Real (Horas). | Estado |
|---------------|---|-------------------------|---------------------|-----------|
| Equipo Scrum | Levantamiento de requisitos | 4 | 3 | Realizado |
| Desarrollador | Definición del Backlog. | 4 | 5 | Realizado |
| Desarrollador | Planificación del proyecto | 4 | 6 | Realizado |
| Desarrollador | Definición de los casos de uso y diagrama de clase. | 6 | 6 | Realizado |
| Desarrollador | Definición de la arquitectura del sistema. | 6 | 4 | Realizado |
| Desarrollador | Documentación de la arquitectura de software | 6 | 10 | |
| Desarrollador | Prototipo del área visual del sistema. | 5 | 8 | Realizado |

| | | | | |
|---------------|--|----|----|-----------|
| Desarrollador | Diseño del modelo de la Base de datos. | 4 | 6 | Realizado |
| Desarrollador | Creación de la base de datos preliminar | 2 | 2 | Realizado |
| Desarrollador | Adecuación del ambiente de desarrollo, aplicaciones y librerías. | 3 | 2 | Realizado |
| Desarrollador | Creación del proyecto APP-TAI | 1 | 2 | Realizado |
| Desarrollador | Instalación y configuración del servidor WildFly | 3 | 3 | Realizado |
| Desarrollador | Creación y configuración del Data Source. | 4 | 3 | Realizado |
| Equipo Scrum | Planificación | 1 | 1 | Realizado |
| | Revisión | 1 | 1 | Realizado |
| | Retrospectiva | 1 | 1 | Realizado |
| Total | | 55 | 63 | |

Fuente: Propia

c) Incremento

La ejecución de las actividades genera entregables respecto a la arquitectura del software entre otras pruebas de la ejecución de estas tareas:

- ✓ Casos de usos. - los cuales son detallados dentro del actual capítulo.
- ✓ Modelado. - Modelo del sistema, tanto como el modelo físico y lógico del sistema.
- ✓ Flujogramas de trabajo que muestran los procesos del TAI y procesos de funcionamiento del sistema.
- ✓ Prototipo del sistema. - incluido dentro de los archivos de arquitectura del software los cuales se detallan en los anexos.
- ✓ Instalación del ambiente de desarrollo, IDE, Base de Datos y Servidor.
- ✓ Creación de la base de datos y configuración de la conexión con el proyecto (Figura 33).

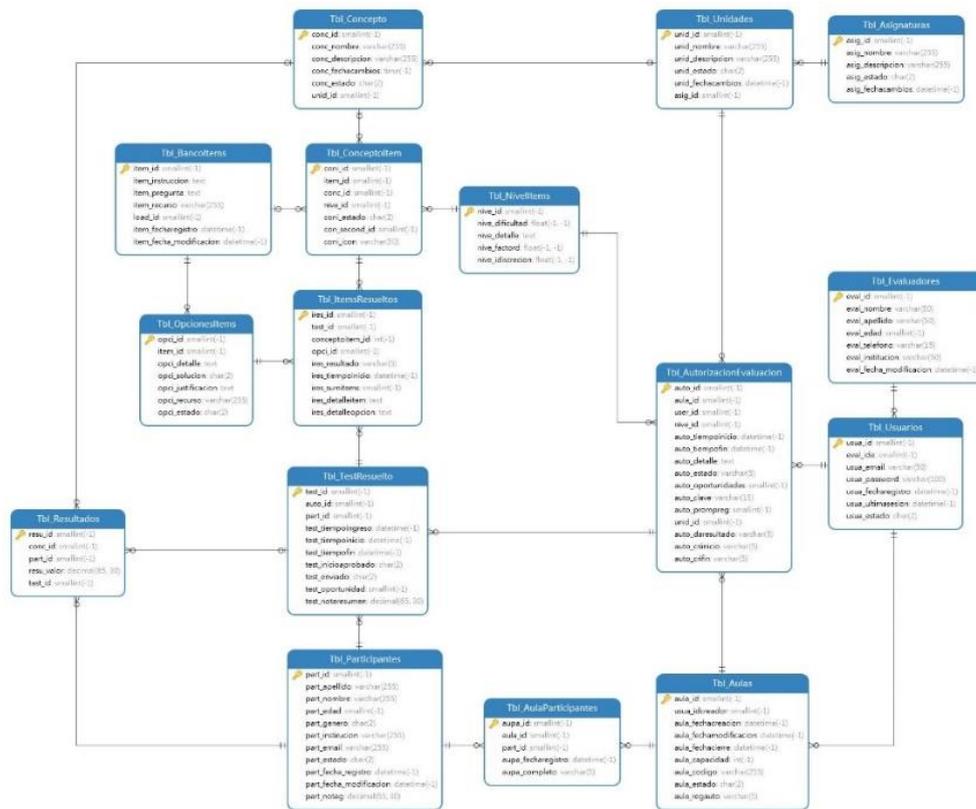


Fig. 33. Estructura de la base de datos
Fuente: Propia

✓ Configuración del Data Source desde WildFly Servidor (Figura 34).

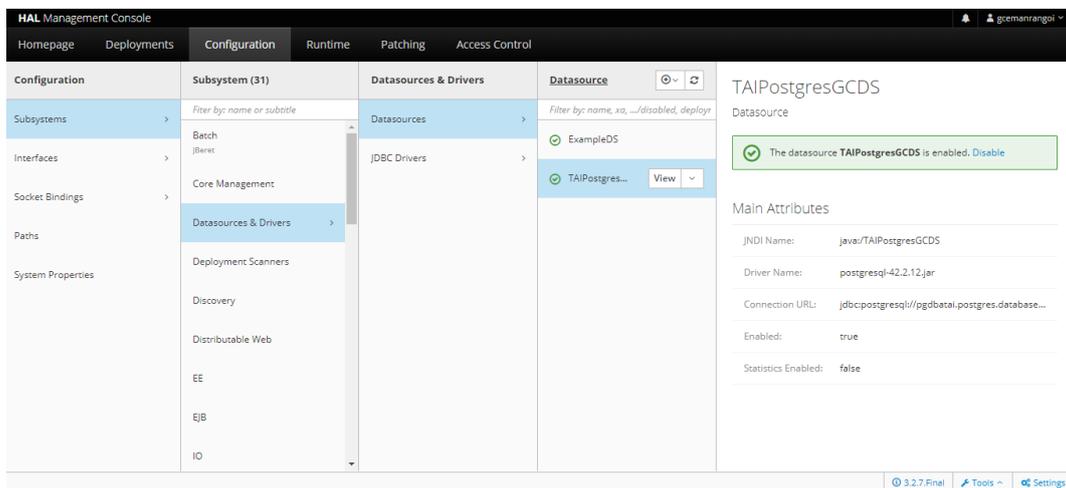


Fig. 34. Configuración del Data Source
Fuente: Propia

d) Reunión retrospectiva

Posterior a la revisión y el análisis del aporte detallado en el incremento se realiza la reunión de retrospectiva con el fin de analizar el funcionamiento del primer sprint, los resultados se representan en la tabla 2.33.

TABLA 2. 33
REVISIÓN SPRINT

| Puntos positivos de la iteración | Puntos negativos de la iteración | Mejoras para la próxima iteración |
|----------------------------------|---|---|
| Creación del proyecto | Librerías obsoletas las cuales fueron remplazadas por otras más actuales. | Revisar actualizaciones de las librerías. |
| Definición del plan del proyecto | Depender de la documentación. | Aumentar los tiempos de cada sprint. |
| Finalización del sprint | Tiempos muy prolongados | |

Fuente: Propia

2.3.2. Sprint 1

El sprint 1 se ejecuta de la misma forma que el sprint anterior, la estructura del sprint se muestra con la tabla 2.34 de acuerdo a la reunión de planificación del sprint y su Sprint Backlog se define con la tabla 2.35 que representa las historias de usuario que se buscan cumplir, posteriormente en la tabla 2.36 se detalla la planificación del sprint con las actividades de acuerdo con Scrum, las actividades cumplidas son detalladas en la tabla 2.37 misma que es obtenida de la reunión de revisión.

a) Reunión de planificación.

Busca cumplir con las historias de usuario definidas.

TABLA 2. 34
SPRINT 1

| Sprint 1 | |
|------------------------------|--|
| Fecha de la reunión: | 14/12/2020 |
| Asistentes: | Scrum Master, Product Owner y Development Team. |
| Fechas de inicio del sprint: | 14/12/2020 |
| Fecha de fin del sprint: | 25/12/2020 |
| Objetivo del Sprint: | <ul style="list-style-type: none"> Elaboración del prototipo funcional para la gestión de ítems y usuarios. Crear el método de selección de preguntas. Versión preliminar de métodos para gestionar credenciales. |

Fuente: Propia

Sprint Backlog:

TABLA 2. 35
SPRINT BACKLOG

| ID | Historia de usuario |
|-----|-----------------------------------|
| H3 | Método de selección de preguntas. |
| H4 | Control de usuarios. |
| H5 | Ingreso al test. |
| H11 | Autenticación de los usuarios |

Fuente: Propia

Planificación:

TABLA 2. 36
PLANIFICACIÓN SPRINT

| Nro. H.U. | Responsable | Fase de desarrollo | Tarea | Tipo | Tempo estimado (Horas). |
|-------------|---------------|--------------------|--|-------|-------------------------|
| Todos | Desarrollador | Desarrollo | Creación de elementos de navegación web para el módulo del Evaluador e inicio de sesión. | Nuevo | 20 |
| H3 | Desarrollador | Análisis | Modelo de selección de ítems | Nuevo | 12 |
| | Desarrollador | Desarrollo | Algoritmo para seleccionar preguntas. | Nuevo | 12 |
| | Desarrollador | Desarrollo | Algoritmo para la gestión del módulo de la presentación del test. | Nuevo | 3 |
| H4 | Desarrollador | Desarrollo | Algoritmo para gestionar usuarios. | Nuevo | 4 |
| H5, H11 | Desarrollador | Desarrollo | Diseño y configuración del login | Nuevo | 6 |
| H4, H5, H11 | Desarrollador | Desarrollo | Control de sesiones y verificación del usuario. | Nuevo | 4 |

Fuente: Propia

b) Reunión de revisión

TABLA 2. 37
REVISIÓN SPRINT

| Responsable | Tarea | Tempo estimado (Horas). | Tempo Real (Horas). | Estado |
|---------------|--|-------------------------|---------------------|--------------|
| Desarrollador | Creación de elementos de navegación web para el módulo del Evaluador e inicio de sesión. | 20 | 21 | Realizado |
| Desarrollador | Modelo de selección de ítems | 12 | 20 | No realizado |
| Desarrollador | Algoritmo para seleccionar preguntas. | 12 | 12 | Realizado |
| Desarrollador | Algoritmo para la gestión del módulo de la presentación del test. | 3 | 6 | Realizado |
| Desarrollador | Algoritmo para gestionar usuarios. | 4 | 2 | Realizado |
| Desarrollador | Diseño y configuración del login | 6 | 4 | Realizado |
| Desarrollador | Control de sesiones y verificación del usuario. | 4 | 4 | Realizado |
| Equipo Scrum | Planificación | 1 | 1 | Realizado |
| | Revisión | 1 | 1 | Realizado |
| | Retrospectiva | 1 | 1 | Realizado |
| | Total | 64 | 72 | |

Fuente: Propia

c) Incremento

- ✓ Páginas de navegación web para el módulo del evaluador

- ✓ Selección de un mejor método para el escogimiento de las preguntas.
- ✓ Diseño del login e implementación en las principales paginas básica.
- ✓ Control de sesiones y autenticación de usuarios (Figura 35).

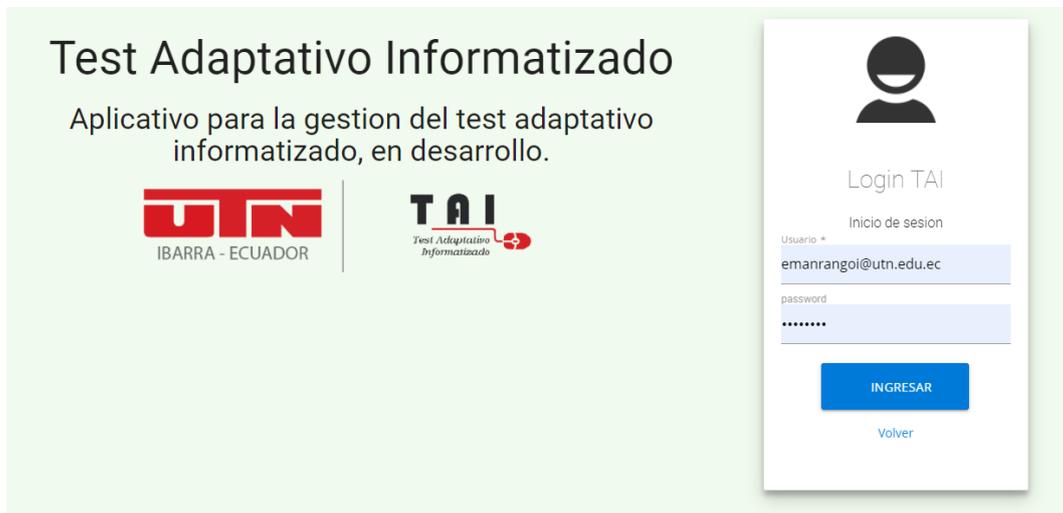


Fig. 35. Login del evaluador
Fuente: Propia

- ✓ Control de sesión del participante (Figura 36).

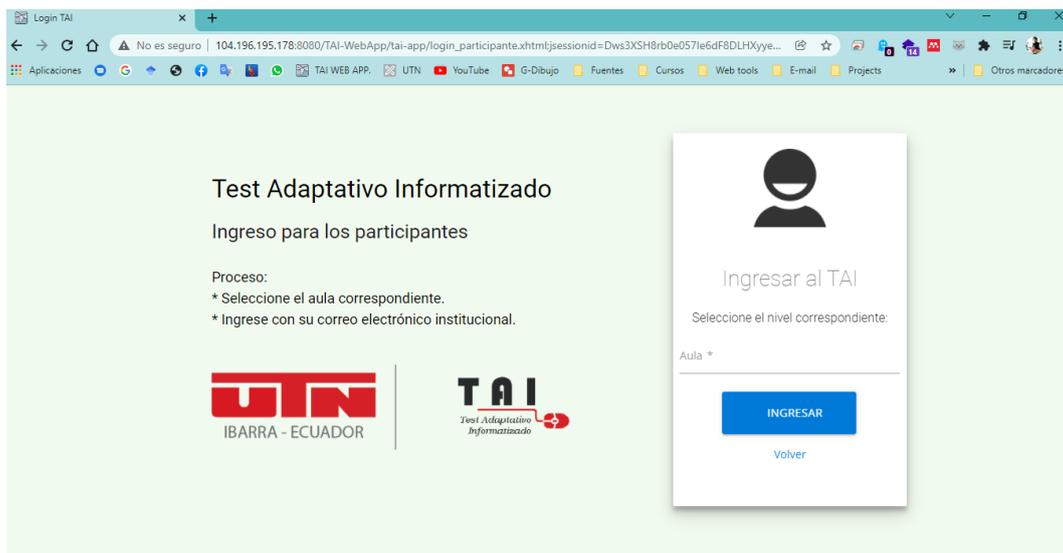


Fig. 36. Ingreso al TAI
Fuente: Propia

d) Reunión retrospectiva

TABLA 2. 38
REVISIÓN SPRINT

| Puntos positivos de la iteración | Puntos negativos de la iteración | Mejoras para la próxima iteración |
|---------------------------------------|---|---|
| Gestión del usuario. | Librerías obsoletas para la selección del ítem. | Desarrollo del algoritmo a cuenta propia. |
| Reestructuración de la base de datos. | No se completó todas las tareas del sprint | Estudio de nuevo método para la selección del ítem. |

Tiempos muy prolongados

Aumentar los tiempos de cada sprint.

Fuente: Propia

2.3.3. Sprint 2

Este sprint busca generar los módulos básicos del aplicativo, los respectivos artefactos de scrum al respecto de este sprint se detallan de la tabla 2.39 a la 2.43.

a) Reunión de planificación.

Busca cumplir con las historias de usuario definidas.

TABLA 2. 39
SPRINT 2

| Sprint 2 | |
|------------------------------|---|
| Fecha de la reunión: | 28/12/2020 |
| Asistentes: | Scrum Master, Product Owner y Development Team. |
| Fechas de inicio del sprint: | 28/12/2020 |
| Fecha de fin del sprint: | 15/01/2021 |
| Objetivo del Sprint: | <ul style="list-style-type: none">• Desarrollo de la plataforma del módulo de presentación del test al participante.• Desarrollo de la plataforma para el módulo de gestión del TAI. |

Fuente: Propia

Sprint Backlog:

TABLA 2. 40
SPRINT BACKLOG

| ID | Historia de usuario |
|-----------|--|
| H1 | Presentación del sistema |
| H3 | Método de selección de preguntas. |
| H7 | Base de datos acorde a especificaciones. |
| H13 | Gestión de temas y unidades. |

Fuente: Propia

Planificación:

TABLA 2. 41
PLANIFICACIÓN SPRINT

| Nro. H.U. | Responsable | Fase de desarrollo | Tarea | Tipo | Tempo estimado (Horas). |
|-------------|---------------|--------------------|---|-----------------|-------------------------|
| H1 | Desarrollador | Desarrollo | Modelamiento de la plantilla para la navegación web respecto al usuario Participante. | Nuevo | 8 |
| | Desarrollador | Desarrollo | Mejoramiento de la apariencia estética. | Nuevo | 6 |
| | Desarrollador | Desarrollo | Creación de carpetas y estructura de navegación. | Nuevo | 2 |
| H1, H3, H13 | Desarrollador | Desarrollo | Diseño de la presentación de la prueba. | Nuevo | 8 |
| | Desarrollador | Desarrollo | Creación del algoritmo para manejo de los temas y unidades de estudio. | Nuevo | 6 |
| | Desarrollador | Desarrollo | Creación de las funciones de cada usuario respecto a la evaluación. | Nuevo | 12 |
| | Desarrollador | Desarrollo | Algoritmo para seleccionar preguntas. | Sprint anterior | 20 |
| H7 | Desarrollador | Desarrollo | Modelado del Banco de ítems | Nuevo | 8 |
| | Desarrollador | Análisis | Gestión de los resultados que genera la evaluación. | Nuevo | 8 |
| | Desarrollador | Desarrollo | Gestión de información respecto a la base de datos. | Nuevo | 4 |
| | Desarrollador | Desarrollo | Creación del algoritmo para procesar la información generada. | Nuevo | 6 |
| Todos | Desarrollador | Desarrollo | Presentación del prototipo funcional versión 1. | Nuevo | 25 |
| | Desarrollador | Desarrollo | Pruebas de funcionamiento del prototipo versión 1. | Nuevo | 5 |

Fuente: Propia

b) Reunión de revisión

TABLA 2. 42
REVISIÓN SPRINT

| Responsable | Tarea | Tempo estimado (Horas). | Tempo Real (Horas). | Estado |
|---------------|---|-------------------------|---------------------|-----------|
| Desarrollador | Modelamiento de la plantilla para la navegación web respecto al usuario Participante. | 8 | 12 | Realizado |
| Desarrollador | Mejoramiento de la apariencia estética. | 6 | 4 | Realizado |
| Desarrollador | Creación de carpetas y estructura de navegación. | 2 | 2 | Realizado |
| Desarrollador | Diseño de la presentación de la prueba. | 8 | 12 | Realizado |
| Desarrollador | Creación del algoritmo para manejo de los conceptos. | 6 | 4 | Realizado |

| | | | | |
|---------------|---|-----|-----|-----------|
| Desarrollador | Creación de las funciones de cada usuario respecto a la evaluación. | 12 | 16 | Realizado |
| Desarrollador | Algoritmo para seleccionar preguntas. | 20 | 22 | Realizado |
| Desarrollador | Modelado del Banco de ítems | 8 | 5 | Realizado |
| Desarrollador | Gestión de los resultados que genera la evaluación. | 8 | 6 | Realizado |
| Desarrollador | Procesamiento de información recolectada. | 4 | 6 | Realizado |
| Desarrollador | Presentación del prototipo funcional versión 1. | 20 | 23 | Realizado |
| Desarrollador | Pruebas de funcionamiento del prototipo versión 1. | 5 | 4 | Realizado |
| Equipo Scrum | Reunión de planificación | 1 | 1 | Realizado |
| | Reunión de revisión | 1 | 1 | Realizado |
| | Retrospectiva | 1 | 1 | Realizado |
| Total | | 110 | 119 | |

Fuente: Propia

c) Incremento

- ✓ Pantallas de navegación del usuario Evaluador y Participante.
- ✓ Presentación de los ítems al participante y recolección de datos para el informe del evaluador.
- ✓ presentación y análisis de la primera versión funcional.
- ✓ Construcción del módulo para la gestión del test (Figura 37).

| Nombre | Aula | Unidad | Finaliza | Estado | OPCIONES |
|--------------|---------------|--------|-----------------------|------------|----------|
| taicsoft1-36 | csoft1-20 | TIC 1 | 2021-12-18 14:14:26.0 | Expirado | |
| test116-35 | test1-16 | TIC 1 | 2022-01-22 19:18:13.0 | Disponible | |
| test2-34 | publicidad-19 | TIC 1 | 2021-11-28 12:59:55.0 | Expirado | |

Fig. 37. Gestión de evaluaciones
Fuente: Propia

- ✓ Modelado y construcción del banco de ítems (Figura 38).

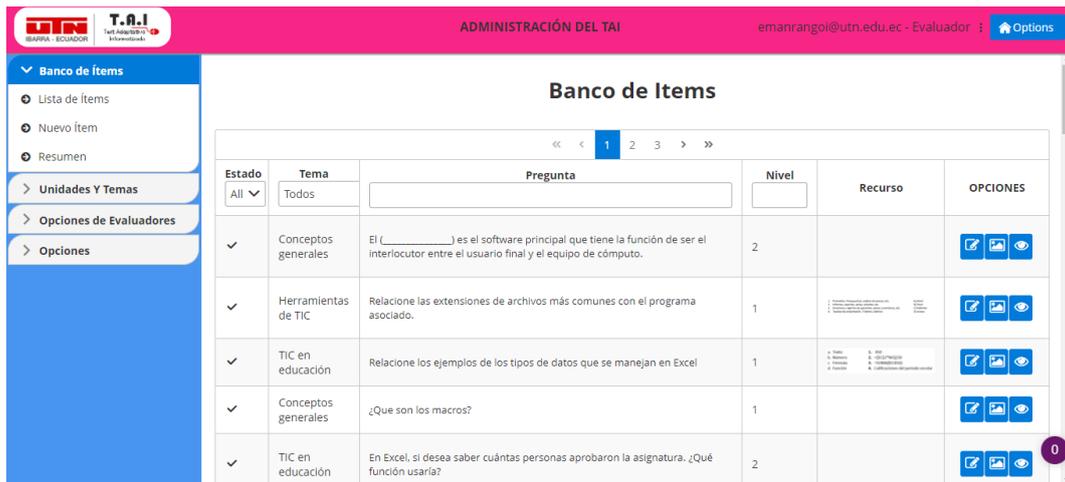


Fig. 38. Banco de ítems
Fuente: Propia

d) Reunión retrospectiva

TABLA 2. 43
REVISIÓN SPRINT

| Puntos positivos de la iteración | Puntos negativos de la iteración | Mejoras para la próxima iteración |
|---|--|-----------------------------------|
| Prueba de funcionamiento del módulo del que presenta el test con novedades para su respectiva corrección. | Método de selección de preguntas debe ser modificado. | Revisar la parte visual. |
| Pruebas de funcionamiento del módulo de gestión del test, con novedades para su respectiva corrección. | Necesita mejor control de credenciales para evitar fallas en el sistema. | Corregir errores y validaciones. |
| Construcción de la base de ítems, sin novedades. | La base de datos es preliminar y requiere modificaciones de ser posibles para el almacenamiento de nuevos valores. | |

Fuente: Propia

2.3.4. Sprint 3

Este sprint busca generar un modelo funcional básico que incluya la gestión del TAI, los respectivos artefactos de scrum al respecto de este sprint se detallan de la tabla 2.44 a la 2.48.

a) Reunión de planificación.

Busca cumplir con las historias de usuario definidas.

TABLA 2. 44
SPRINT 3

| Sprint 3 | |
|------------------------------|---|
| Fecha de la reunión: | 18/01/2021 |
| Asistentes: | Scrum Master, Product Owner y Development Team. |
| Fechas de inicio del sprint: | 18/01/2021 |
| Fecha de fin del sprint: | 29/01/2021 |
| Objetivo del Sprint: | <ul style="list-style-type: none"> • Construcción del módulo de gestión del TAI. |

Fuente: Propia

Sprint Backlog:

TABLA 2. 45
SPRINT BACKLOG

| ID | Historia de usuario |
|-----|--------------------------------|
| H4 | Control de usuarios. |
| H8 | Registro de participantes. |
| H12 | Gestión de ítems. |
| H14 | Bloquear información sensible. |

Fuente: Propia

Planificación:

TABLA 2. 46
PLANIFICACIÓN SPRINT

| Nro. H.U. | Responsable | Fase de desarrollo | Tarea | Tipo | Tempo estimado (Horas). |
|-----------|---------------|--------------------|--|-------|-------------------------|
| Todos | Desarrollador | Desarrollo | Corrección del prototipo versión 1 | Nuevo | 8 |
| H4, H8 | Desarrollador | Desarrollo | Control de sesión del usuario evaluador. | Nuevo | 4 |
| H8, H14 | Desarrollador | Desarrollo | Proceso de registro de participantes | Nuevo | 4 |
| | Desarrollador | Análisis | Capa de seguridad para los parámetros que no podrán ser modificados. | Nuevo | 6 |
| | Desarrollador | Desarrollo | Algoritmo para la gestión de participantes. | Nuevo | 12 |
| | Desarrollador | Desarrollo | Creación de las vistas y aulas. | Nuevo | 6 |
| | Desarrollador | Desarrollo | Creación de las Aulas y relación con los participantes. | Nuevo | 12 |
| Todos | Desarrollador | Desarrollo | Presentación del modelo funcional versión 2 | Nuevo | 6 |

Fuente: Propia

b) Reunión de revisión

TABLA 2. 47
REVISIÓN SPRINT

| Responsable | Tarea | Tempo estimado (Horas). | Tempo Real (Horas). | Estado |
|---------------|--|-------------------------|---------------------|-----------|
| Desarrollador | Corrección del prototipo versión 1 | 8 | 12 | Realizado |
| Desarrollador | Control de sesión del usuario evaluador. | 4 | 5 | Realizado |
| Desarrollador | Control de seguridad bajo rol de Super Evaluador. | 4 | 4 | Realizado |
| Desarrollador | Parámetros que se podrán ser modificados. | 6 | 4 | Realizado |
| Desarrollador | Algoritmo para la gestión de participantes. | 12 | 8 | Realizado |
| Desarrollador | Diseño web para la gestión del TAI. | 6 | 12 | Realizado |
| Desarrollador | Algoritmo que genere el resumen de los ítems contestados | 4 | 5 | Realizado |
| Desarrollador | Desarrollo funcional del módulo de gestión del TAI | 8 | 7 | Realizado |
| Desarrollador | Presentación del modelo funcional versión 2 | 6 | 13 | Realizado |
| Equipo Scrum | Planificación | 1 | 1 | Realizado |
| | Revisión | 1 | 1 | Realizado |
| | Retrospectiva | 1 | 1 | Realizado |
| | Total | 61 | 73 | |

Fuente: Propia

c) Incremento

- ✓ Creación de la versión preliminar del módulo de gestión del TAI, se integra todas las funciones a excepción de los reportes (Figura 39).



Fig. 39. Panel administración TAI
Fuente: Propia

- ✓ Registro del ítem en el banco de ítems.
- ✓ Resumen análisis de ítems (Figura 40).

| Item ID | Tema | Respuestas correctas | Respuestas erroneas | Respuestas totales | Dificultad | OPCIONES |
|---------|---------------------|----------------------|---------------------|--------------------|------------|----------|
| 2 | Conceptos generales | 85 | 27 | 112 | Basico | |
| 3 | Herramientas de TIC | 48 | 50 | 98 | Bajo | |
| 4 | TIC en educación | 62 | 55 | 117 | Bajo | |
| 5 | Conceptos generales | 13 | 19 | 32 | Bajo | |
| 6 | TIC en educación | 46 | 77 | 123 | Basico | |
| 7 | Comunicación | 78 | 23 | 101 | Bajo | |
| 8 | Herramientas de TIC | 16 | 14 | 30 | Bajo | |

Fig. 40. Resumen respuestas
Fuente: Propia

d) Reunión retrospectiva

TABLA 2. 48 REVISIÓN SPRINT

| Puntos positivos de la iteración | Puntos negativos de la iteración | Mejoras para la próxima iteración |
|---|---|---|
| Creación del módulo de gestión del TAI. | Tiempo muy extendido para el desarrollo. Modificación de la base de datos y las entidades lo cual complico finalizar rápido una tarea. | Optimizar más el tiempo. Describir mejor las tareas. |

Fuente: Propia

2.3.5. Sprint 4

Este Sprint es el final del desarrollo y genera un modelo funcional completo con todos los módulos previos a la implementación, los respectivos artefactos de scrum al respecto de este sprint se detallan de la tabla 2.49 a la 2.53.

a) Reunión de planificación.

Busca cumplir con las historias de usuario definidas.

TABLA 2. 49
SPRINT 4

| Sprint 4 | |
|------------------------------|--|
| Fecha de la reunión: | 01/02/2021 |
| Asistentes: | Scrum Master, Product Owner y Development Team. |
| Fechas de inicio del sprint: | 01/02/2021 |
| Fecha de fin del sprint: | 12/02/2020 |
| Objetivo del Sprint: | corrección de errores, validaciones y criterios de visión para interfaces amigables. |

Fuente: Propia

Sprint Backlog:

TABLA 2. 50
SPRINT BACKLOG

| ID | Historia de usuario |
|-----|---------------------|
| H6 | Reportes. |
| H9 | Carga de recursos. |
| H10 | Interfaz amigable |

Fuente: Propia

Planificación:

TABLA 2. 51
PLANIFICACIÓN SPRINT

| Nro. H.U. | Responsable | Fase de desarrollo | Tarea | Tipo | Tempo estimado (Horas). |
|-----------|---------------|--------------------|---|-------|-------------------------|
| H6 | Desarrollador | Desarrollo | Creación de métodos para entregar reportes al evaluador. | Nuevo | 8 |
| | Desarrollador | Desarrollo | Métodos para entrega de resultados al participante | Nuevo | 4 |
| H9 | Desarrollador | Desarrollo | Método para cargar recursos visuales para los ítems | Nuevo | 6 |
| H10 | Desarrollador | Desarrollo | Corrección de características visuales del sistema. | Nuevo | 8 |
| | Desarrollador | Desarrollo | modificación de funciones para adaptarse a una interfaz más amigable. | Nuevo | 8 |
| Todos | Desarrollador | Desarrollo | Evaluación a la versión final. | Nuevo | 6 |
| | Desarrollador | Desarrollo | Corrección de errores de la versión final. | Nuevo | 12 |
| | Desarrollador | Desarrollo | Elaboración de documentación | Nuevo | 20 |

Fuente: Propia

b) Reunión de revisión

TABLA 2. 52
REVISIÓN SPRINT

| Responsable | Tarea | Tempo estimado (Horas). | Tempo Real (Horas). | Estado |
|---------------|--|-------------------------|---------------------|-----------|
| Desarrollador | Creación de métodos para entregar reportes al evaluador. | 8 | 12 | Realizado |
| Desarrollador | Métodos para entrega de resultados al participante | 4 | 9 | Realizado |
| Desarrollador | Método para cargar recursos visuales para los ítems | 6 | 4 | Realizado |
| Desarrollador | Corrección de características visuales del sistema. | 8 | 8 | Realizado |

| | | | | |
|---------------|---|----|----|-----------|
| Desarrollador | Modificación de funciones para adaptarse a una interfaz más amigable. | 8 | 12 | Realizado |
| Desarrollador | Evaluación a la versión final. | 6 | 4 | Realizado |
| Desarrollador | Corrección de errores de la versión final. | 12 | 16 | Realizado |
| Desarrollador | Elaboración de documentación e informe final. | 20 | 25 | Realizado |
| Equipo Scrum | Planificación | 1 | 1 | Realizado |
| | Revisión | 1 | 1 | Realizado |
| | Retrospectiva | 1 | 1 | Realizado |
| Total | | 75 | 93 | |

Fuente: Propia

c) Incremento

- ✓ Diseño y aplicación de reportes de acuerdo con los casos de uso.
- ✓ Documentación respecto al desarrollo.
- ✓ Versión final del sistema APP-TAI (Figura 41).



Fig. 41. Evaluaciones disponibles
Fuente: Propia

- ✓ Publicación del sistema mediante el uso de una máquina virtual por Google Cloud.

d) Reunión retrospectiva

TABLA 2. 53 REVISIÓN SPRINT

| Puntos positivos de la iteración | Puntos negativos de la iteración | Procesos a continuación |
|--|--|--|
| Creación del Prototipo final para la evaluación. | No existió pruebas de funcionamiento a gran escala. Implementación sin éxito, | Pruebas de funcionamiento Implementación en servicios de nube. Aplicación de la evaluación |

Fuente: Propia

Capítulo III

3. Validación de resultados

3.1. Análisis de funcionamiento

Para el análisis del funcionamiento se usó el procedimiento descrito en el gráfico 42, se aplicó la evaluación adaptativa a participantes de prueba y se recolectó dicha información para analizar los resultados mediante la comparación de tablas, con el objetivo de conocer si cumple o no con los requisitos de un Test Adaptativo, posteriormente mediante el uso de encuestas se analiza la usabilidad del sistema, y el cumplimiento de los objetivos, el proceso sigue con la interpretación de los resultados, y el análisis de impacto.



Fig. 42. Análisis del sistema.
Fuente: propia

3.1.1. Rendimiento del TAI

Se realizaron dos evaluaciones con diferentes grupos dentro de la Universidad Técnica del Norte: la primera fue presentada a tres carreras correspondientes a segundo nivel de la Facultad de Educación, Ciencia Y Tecnología mismas que recibían la asignatura de TIC, posteriormente se presentó la evaluación a los estudiantes de primer y segundo nivel de la Facultad de Ingeniería en Ciencias Aplicadas.

Durante las primeras pruebas se utilizó una base de ítems compuesta por 72 preguntas que evaluaban el conocimiento en cinco niveles y en dos conceptos: Computación básica y ofimática. Estas preguntas se encontraban orientadas a las TIC, estos ítems formaron parte de los reactivos de las evaluaciones pertenecientes al MSc. Ing. Marcelo Jurado también docente en la facultad de Educación, Ciencia Y Tecnología.

Se realizó el análisis de las respuestas obtenidas con el fin de calibrar los ítems y reasignar los niveles de dificultad, así como también retirar o modificar las preguntas que por no estar bien planteadas no aportaban información precisa, los ítems se encontraban distribuidos de la siguiente manera (Figura 43):

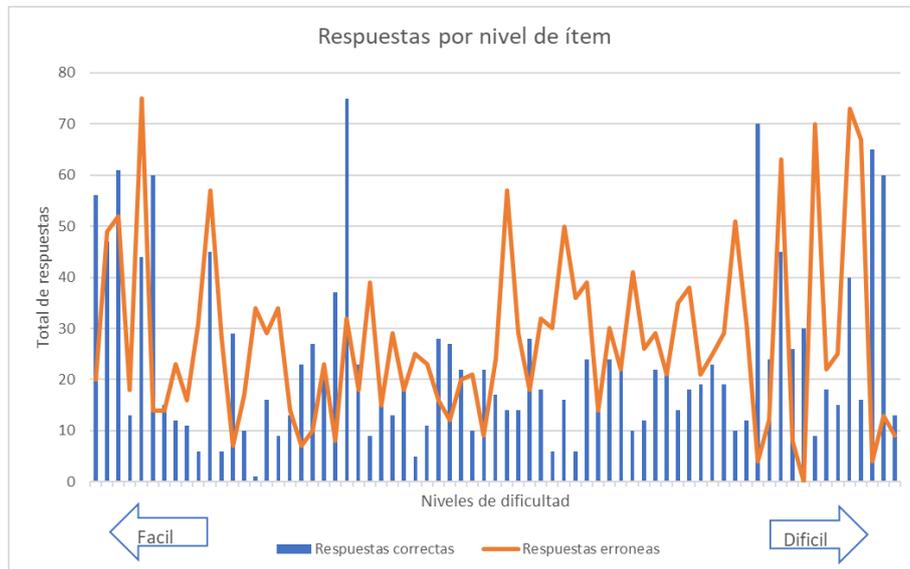


Fig. 43. Modelo de respuestas
Fuente: propia

La figura 43 muestra la cantidad de veces que fue suministrado un ítem durante las evaluaciones realizadas, y las veces en las que se obtuvo respuestas positivas y negativas (correctas e incorrectas), usando la herramienta JMetrik se realizó el análisis de estas preguntas y se calibró la dificultad de cada una de estas, así como su valor de importancia de acuerdo con la teoría de respuesta al ítem.

La figura 44 muestra un análisis de las respuestas de los ítems similar a la Figura 43, este es el resultado posterior a la evaluación en la Facultad de Ingeniería en Ciencias Aplicadas.

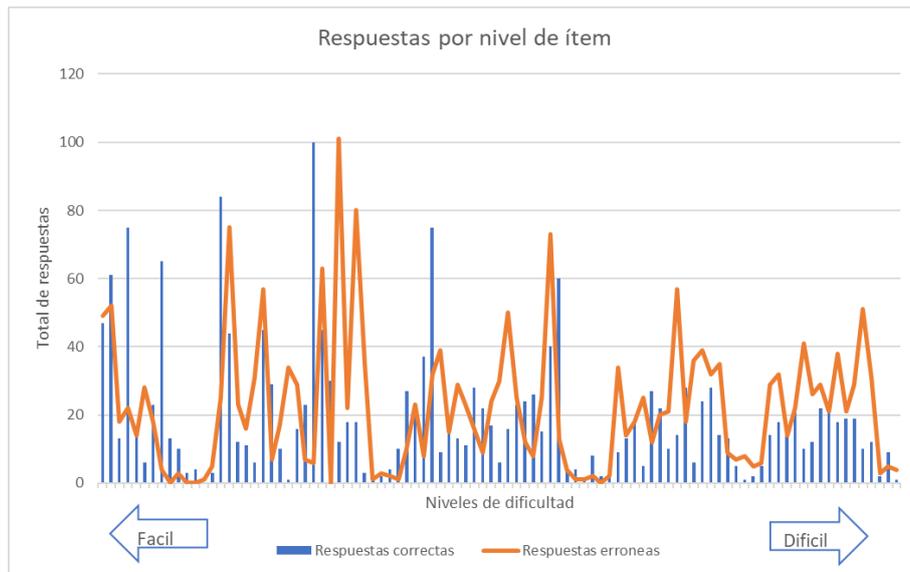


Fig. 44. Modelo de respuestas corregido
Fuente: propia

En este caso posterior a la calibración de los ítems, como se puede evidenciar, las respuestas erróneas van aumentando en relación con la dificultad, de la misma manera las respuestas correctas disminuyen mientras más difícil sea el ítem, aunque esta no es una regla en la que se debe confiar si es un indicador del correcto funcionamiento.

Análisis del Banco de ítems

Se aplica el TAI en las carreras seleccionadas dentro de la Facultad de Ingeniería en Ciencias Aplicadas, para esta evaluación se prepara una base de ítems de 100 preguntas de 5 niveles de dificultad y divididos en cuatro temas o conceptos. Para evaluar los ítems previamente calibrados se incluye un bloque de preguntas dentro de la encuesta (Figura 45), los resultados son:

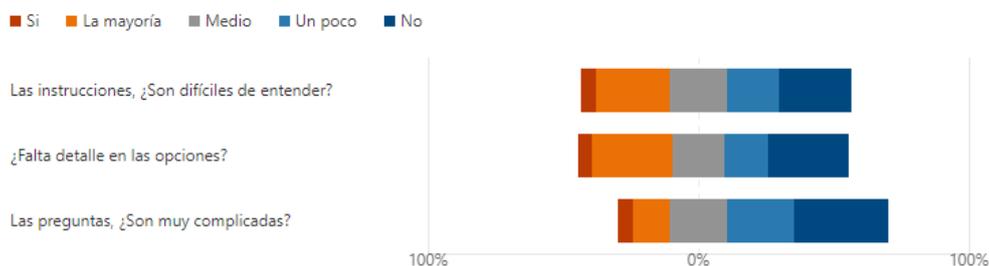


Fig. 45. Pregunta 5 análisis de los ítems
Fuente: propia

Para el análisis se presentan las siguientes cuestiones:

- ¿Los ítems están bien planteados?

Más del 30% de los encuestados indicaron tener algún tipo de problema al entender las preguntas, sin embargo, el resto pese a tener problemas entendieron la pregunta, entre ellos un 27% que no tuvo ningún inconveniente.

- **¿Están correctamente calibrados los ítems?**

Podemos decir que los ítems se encuentran calibrados con el análisis de la figura 36, además la figura 37 la tercera pregunta sobre la complicidad de las preguntas menos del 20% de los participantes cree que todas las preguntas son complicadas mientras un 35% no cree que sean complicadas, estos resultados responden a los niveles de dificultad planteados dentro del test.

Por consecuencia, de acuerdo con el análisis previo es recomendable una nueva calibración de ítems, pero esta debe llevarse a cabo posterior a una retroalimentación de los temas, con el fin de cubrir los temas que más dificultad presentaron los estudiantes.

Funcionalidad en la nube

Google Cloud Platform permite monitorizar en tiempo real el uso del CPU como indica la Figura 46, sobre el uso de la memoria RAM, entre otras herramientas que su plataforma ofrece para monitorizar el funcionamiento del sistema.

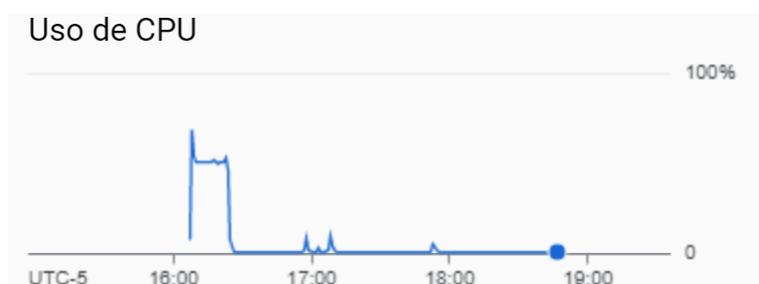


Fig. 46. Uso del CPU virtual
Fuente: propia

El gráfico representa el uso del CPU con un grupo de 30 estudiantes, posterior a este inconveniente se decidió mejorar la máquina virtual con un mejor CPU, los resultados se muestran en la Figura 47:

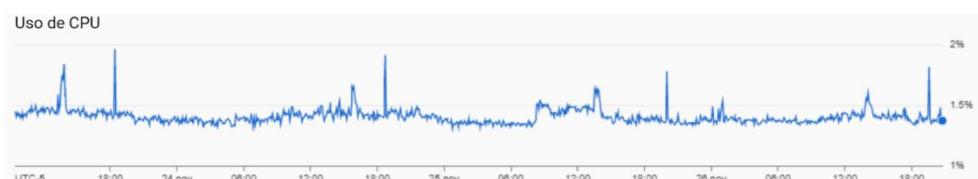


Fig. 47. Uso del CPU optimizado
Fuente: propia

El uso del CPU durante el proceso de evaluación a 10 participante simultáneamente produjo un uso de apenas el 2 % de su CPU tal como lo muestra la gráfica, dando a entender que se optimizó la plataforma.

Rendimiento del aplicativo

El sistema se ejecutó en la plataforma de Google Cloud mediante una máquina virtual con un servidor de WildFly, en relación con el costo-servicio se eligió el servidor de us-east4-b a la par se ejecutó la conexión a la base de datos la misma que se encontraba alojada en una instancia SQL PostgreSQL de la misma plataforma de Google, la Figura 48 muestra el monitor de transacciones de Google Cloud.



Fig. 48. Tráfico en red respecto a las peticiones al servidor.
Fuente: propia

Durante la evaluación con 10 participantes simultáneos rindiendo la evaluación y con el panel de administración operando por el administrador, se registró un 10% en la capacidad de transacciones, siendo valores bajos además que de ser necesaria el mismo servidor amplia la capacidad de red aumentando el número de transacciones.

3.1.2. Usabilidad del prototipo

Para el análisis de la usabilidad se utilizó el Sistema de Usabilidad Escalable (SUS) ya que presenta resultados rápidos y fiables al momento de evaluar Sistemas o tecnologías. Este método plantea 10 preguntas con 5 escalas cada una con un valor del 1 al 5 respectivamente con el fin de identificar si está o no de acuerdo con el enunciado. El nivel de usabilidad es determinado con una puntuación entre 0 a 100, las puntuaciones altas indican que el sistema o tecnología posee una buena usabilidad.(Binyamin et al., 2016). Aplicando el concepto con un total de 37 encuestados la Tabla 3. 1 muestra los resultados obtenidos:

TABLA 3. 1
RESULTADOS SUS

| Preguntas | Totalmente en desacuerdo | En desacuerdo | Ni en acuerdo, ni desacuerdo | De acuerdo | Totalmente de acuerdo |
|-------------|--------------------------|---------------|------------------------------|------------|-----------------------|
| Pregunta 1 | 0 | 1 | 0 | 26 | 10 |
| Pregunta 2 | 17 | 7 | 5 | 3 | 5 |
| Pregunta 3 | 4 | 3 | 6 | 10 | 14 |
| Pregunta 4 | 13 | 9 | 6 | 7 | 2 |
| Pregunta 5 | 0 | 3 | 6 | 20 | 8 |
| Pregunta 6 | 18 | 6 | 6 | 4 | 3 |
| Pregunta 7 | 1 | 4 | 0 | 21 | 11 |
| Pregunta 8 | 10 | 13 | 8 | 4 | 2 |
| Pregunta 9 | 3 | 6 | 9 | 13 | 6 |
| Pregunta 10 | 13 | 9 | 8 | 5 | 2 |

Fuente: Propia

Análisis de cada pregunta

Para entender los resultados se analizan y detallan cada una de las preguntas de la tabla 3.1., realizando la tabulación de los datos y representando cada una de las preguntas de manera gráfica, desde la Figura 49 a la Figura 58 respectivamente.

Pregunta 1.- Creo que usaría este sistema frecuentemente.

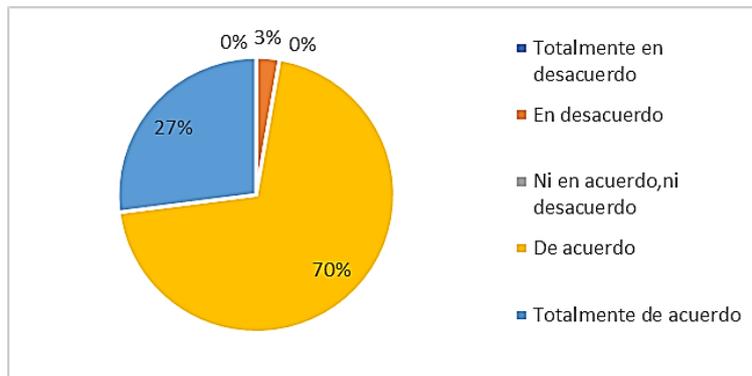


Fig. 49. Resultado pregunta 1 (SUS)
Fuente: propia

Para el estudiante el sistema es innovador, y la nueva forma propuesta de evaluación ha cumplido con sus expectativas, ya que los resultados de aceptación de acuerdo con la Figura 49 indica un alto número de agrado y aprobación para usarlo de forma frecuente, sin presentar resultados preocupantes.

Pregunta 2.- Encuentro este sistema innecesariamente complejo.

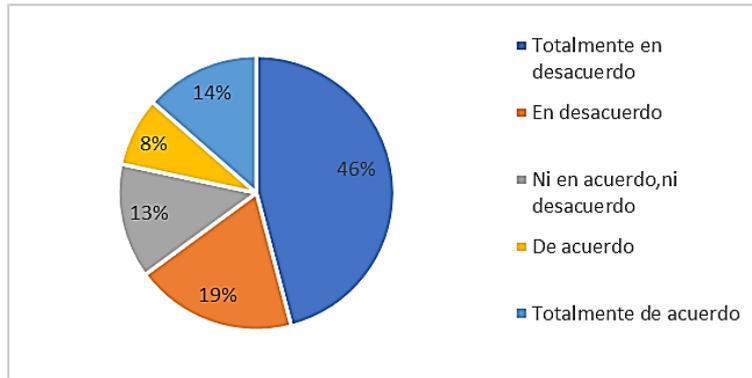


Fig. 50. Resultado pregunta 2 (SUS)
Fuente: propia

El sistema sostiene un formato de evaluación similar a cualquier otra plataforma de evaluación, el proceso de evaluación es simple y el ingreso a la evaluación busca ser cómoda y sencilla, es decir, el sistema no busca ser innecesariamente complejo, por la cual, recibe la aprobación de los encuestados de acuerdo con lo que se indicó en la figura 50, la mala interpretación de la idea de adaptativo en ciertos casos muestra desconfianza, por lo tanto, existen señales de complejidad innecesaria.

Pregunta 3.- Creo que el sistema fue fácil de usar.

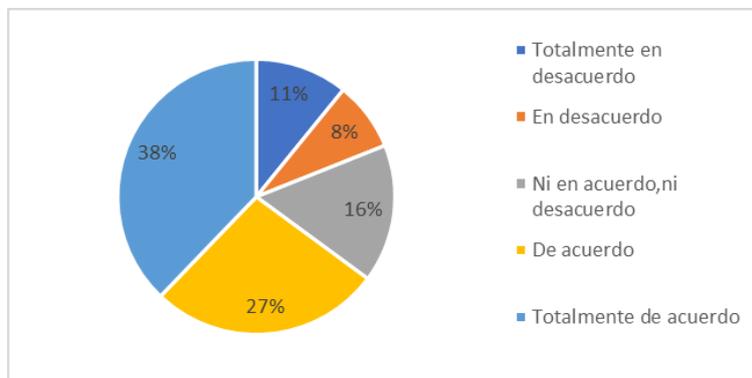


Fig. 51. Resultado pregunta 3 (SUS)
Fuente: propia

Un módulo simple para la evaluación y la experiencia previa manejando equipos de cómputo permiten al evaluado desenvolverse sin inconvenientes, es decir el evaluado encontró un sistema fácil de usar, sin embargo, la mala interpretación de los resultados de la evaluación generó que algunos evaluados creen que existe algún tipo de dificultad, y se generen resultados negativos en la encuesta, aunque estos sean mínimos.

Pregunta 4.- Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema.

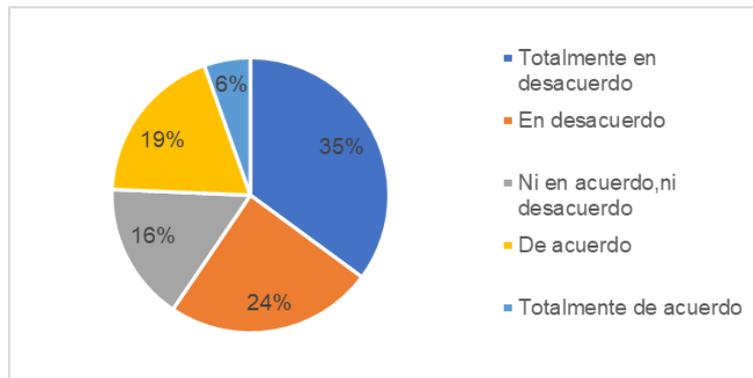


Fig. 52. Resultado pregunta 4 (SUS)
Fuente: propia

El sistema detalla en todo punto las funciones que puede realizar el usuario, tanto el evaluador como el evaluado no requerían de mucha ayuda para manejar el sistema, se colocaron indicaciones y se emiten mensajes antes de cualquier operación que pueda generar errores, se agregaron iconos de acuerdo con la función a realizar con el fin de no requerir ayuda técnica, sin embargo dichos elementos o no funcionaron correctamente o ayudaron a creer que el sistema es complejo, dado a que los resultados negativos de la encuesta son relativamente altos.

Pregunta 5.- Las funciones de este sistema están bien integradas.

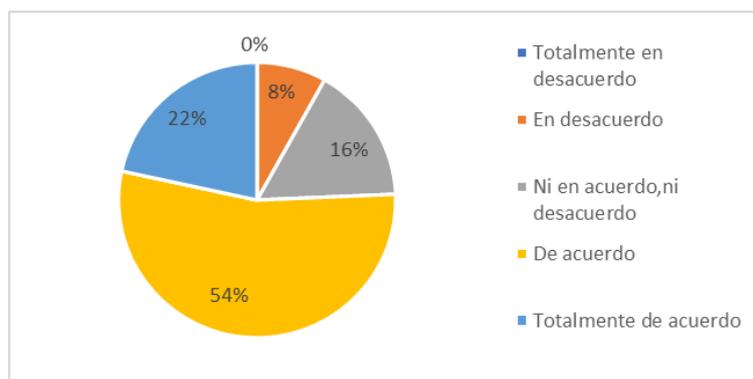


Fig. 53. Resultado pregunta 5 (SUS)
Fuente: propia

La integración de las funciones de forma correcta permite un fácil manejo del sistema por parte del usuario, las funciones no hacen más de lo que se necesita y responden de acuerdo con lo que se solicita, sin que generen problemas o agregar funciones innecesarias, lo que permite una mejor interacción con el usuario que muestra estar de acuerdo.

Pregunta 6.- Creo que existe demasiada inconsistencia en este sistema.

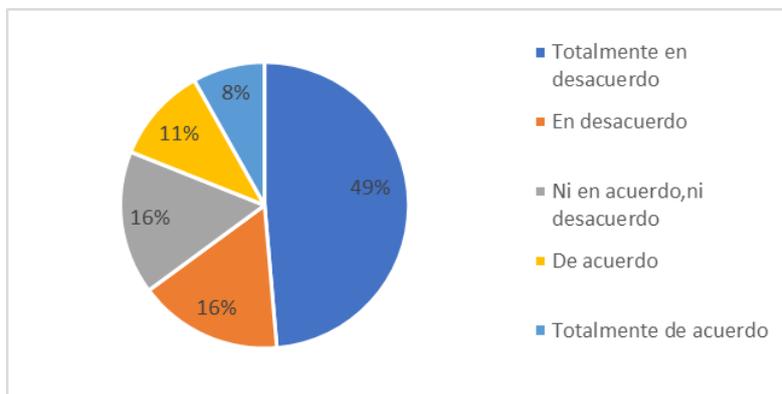


Fig. 54. Resultado pregunta 6 (SUS)
Fuente: propia

Las inconsistencias generan errores, mismos que pueden presentarse en cualquier sistema, por su parte se buscó eliminar dichas inconsistencias lo cual se realizó con éxito debido a que un pequeño porcentaje reporto tener inconsistencias las cuales estaban sujetas a factores externos mayormente relacionados con la conectividad a internet.

Pregunta 7.- Imagino que la mayoría de la gente aprendería a usar este sistema en forma muy rápida.

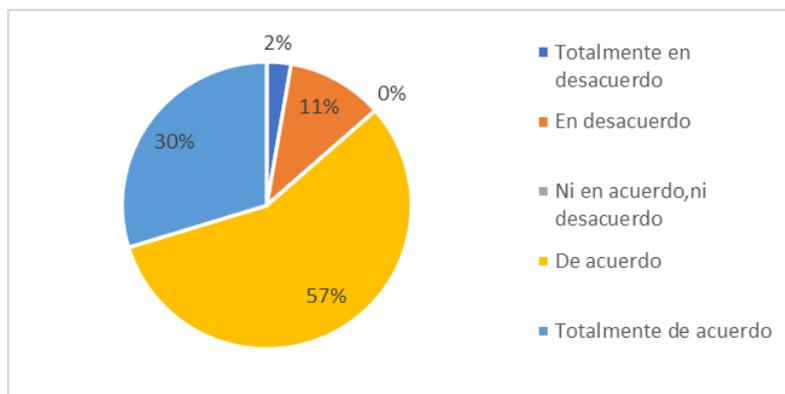


Fig. 55. Resultado pregunta 7 (SUS)
Fuente: propia

El sistema busca ser fácil de usar y mediante esta pregunta, busca conocer el punto de vista del participante respecto al resto de posibles usuarios, esta percepción es positiva en la mayoría de los casos y negativa respecto a un grupo de participantes, en otras palabras, el participante cree que las personas en su entorno podrán usar el sistema sin inconvenientes.

Pregunta 8.- Encuentro que el sistema es muy difícil de usar

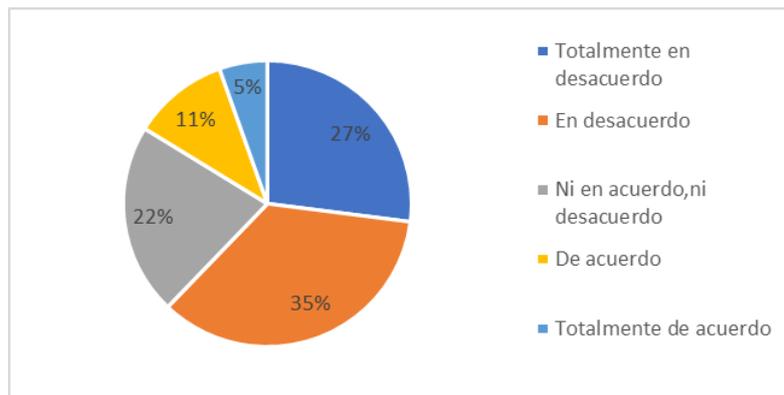


Fig. 56. Resultado pregunta 8 (SUS)
Fuente: propia

EL sistema no se considera difícil de usar, ya que por parte del participante cumple con las expectativas de un test y por parte del evaluador cumple con los requerimientos por el descritos, los resultados de la encuesta apoyan la fácil usabilidad junto a las pocas respuestas negativas que se vienen dando debido a la mala interpretación de lo que significa un test adaptativo.

Pregunta 9.- Me siento confiado al usar este sistema.

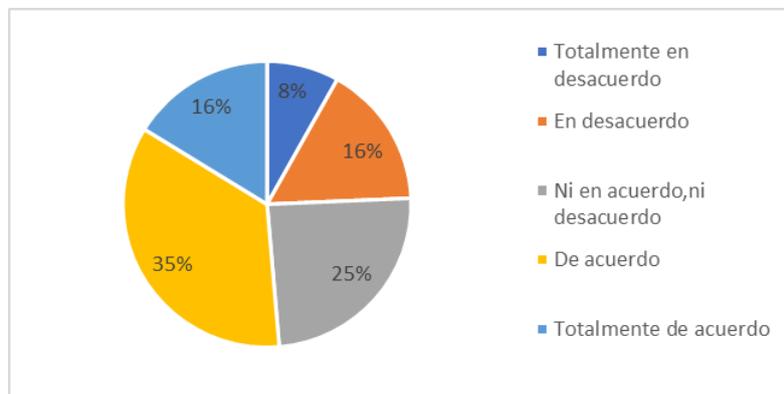


Fig. 57. Resultado pregunta 9 (SUS)
Fuente: propia

El TAI busca obtener los resultados más reales del participante, lo cual genera confianza en este tipo de evaluaciones, sin embargo, ya sea por la costumbre, resistencia al cambio o alguna otra razón el participante no confía del todo en este tipo de evaluaciones junto a él una relativa cantidad de participantes demuestran también no tener confianza en este método de evaluar.

Pregunta 10.- Necesité aprender muchas cosas antes de ser capaz de usar este sistema.

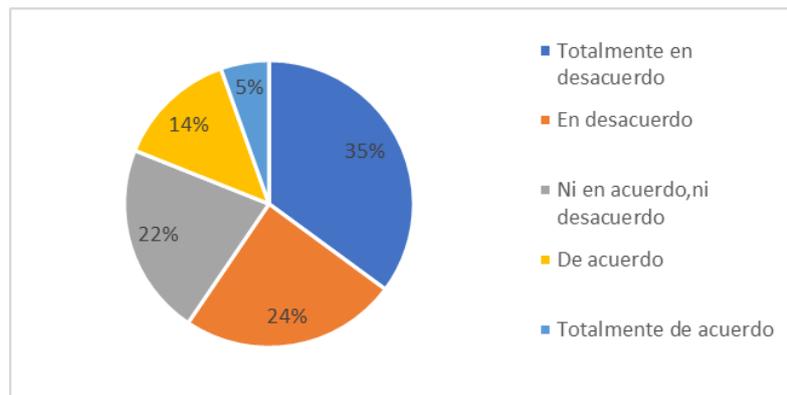


Fig. 58. Resultado pregunta 10 (SUS)
Fuente: propia

Aunque un TAI es un tema nuevo para muchos de los participantes, entender el funcionamiento del sistema les parece complejo, por tal razón las explicaciones son básicas o simplemente se presenta como una evaluación adaptativa sin más detalles para no generar confusión, es decir, los participantes no requieren una capacitación de cómo funciona el test, los resultados negativos se dan porque a principio se trató de explicar el funcionamiento del sistema, lo cual se presta para mal interpretar la idea de una capacitación rápida.

Análisis del TAI

Mediante la misma encuesta se recolecta la opinión respecto a la evaluación y con los resultados se pueda realizar un análisis hipotético deductivo, los resultados se muestran en la figura 59.

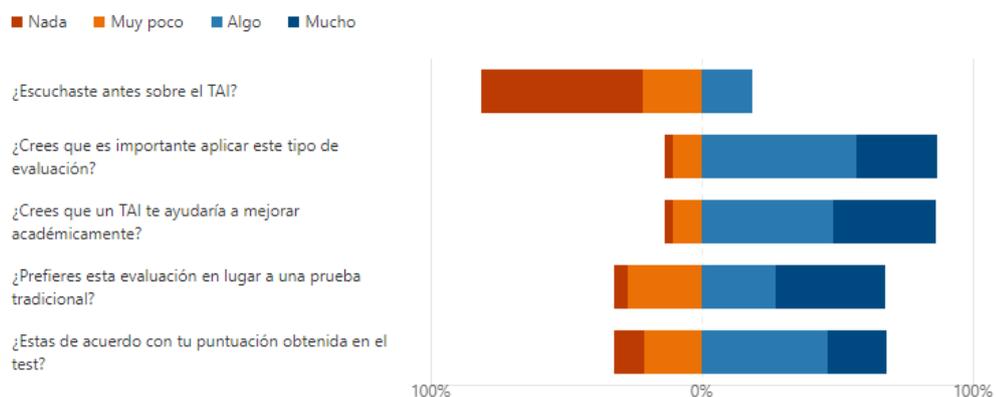


Fig. 59. Pregunta 6 de la encuesta
Fuente: propia

Como es evidente el desconocimiento sobre los TAI es relativamente alto con un 59.5% de los encuestados, sin embargo, se refleja optimismo en este modelo de evaluaciones ya que hay una preferencia del 86.5% aunque en su contra un 32.4% aun dudan en cambiarse del modelo tradicional de evaluaciones.

Por lo tanto, se puede decir que existe un grado de optimismo por parte de los estudiantes ya sea que no les gusta la actual forma de evaluar o el mismo sistema académico, por su parte también se muestra una ligera resistencia al cambio.

Estilos visuales

Sobre la experiencia visual y el diseño del test se recolecta información dentro de la encuesta aplicada, los resultados se muestran en la Figura 60:

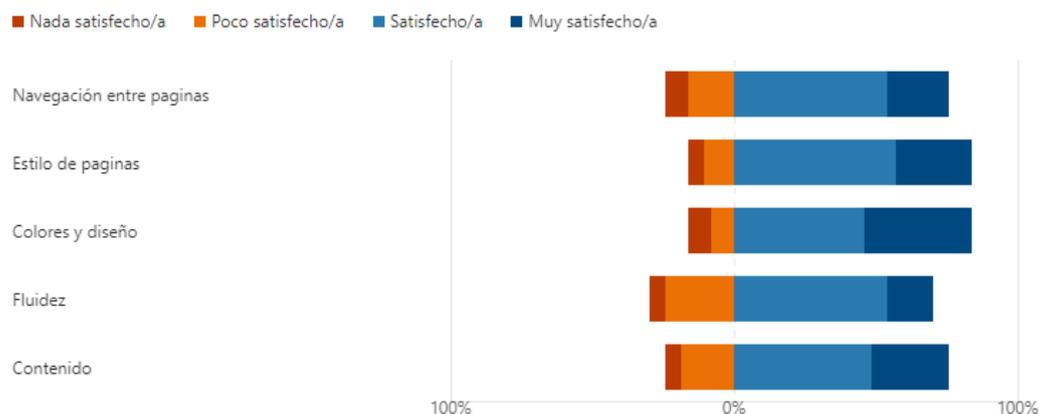


Fig. 60. Pregunta 3 estilos visuales
Fuente: propia

Para el análisis de la experiencia visual se plantea cinco aspectos mismos que se presentan en la gráfica con sus respectivos porcentajes de respuestas:

- **La navegación entre paginas** muestra un 8.1% de insatisfacción, un 16.2% de poco satisfecho, un 54.1% de satisfactorio y un 21.6% de muy satisfecho. Es decir, el porcentaje de satisfacción es relativamente alto, siendo muy aceptable.
- **Estilo de página** muestra un 5.4% de insatisfacción, un 10.8% de poco satisfecho, un 56.8% de satisfactorio y un 27% de muy satisfecho. Es decir, el porcentaje de satisfacción del usuario es alto y los de insatisfacción bajos, siendo muy aceptable el estilo.
- **Colores y diseño** muestra un 8.1% de insatisfacción, un 8.1% de poco satisfecho, un 45.9% de satisfactorio y un 37.8% de muy satisfecho. Es decir,

el porcentaje de satisfacción es alto, siendo muy aceptable la paleta de colores y el diseño.

- **Fluidez** muestra un 5.4% de insatisfacción, un 24.3% de poco satisfecho, un 54.1% de satisfactorio y un 16.2% de muy satisfecho. Es decir, no todos estuvieron satisfechos con la fluidez del sitio, sin embargo, se toma en cuenta el tipo de computador usado y la conexión a internet.
- Para el **Contenido** que la pagina muestra el nivel de insatisfacción indica un 5.4%, un 18.9% de poco satisfecho, un 48.60% de satisfactorio y un 27% de muy satisfecho, al no existir mucho contenido ya sea en información o multimedia no todos estuvieron satisfechos.

Ajustes sugeridos

Posterior al análisis de los resultados y pruebas con los estudiantes de prueba de la Facultad de Educación, Ciencia y Tecnología se notaron inconvenientes que a continuación se detallan:

- **Optimización del sistema**, desde la capa de negocios se realizó cambios en la estructura de los controladores y las funciones, para facilitar la carga rápida.
- **Modificación de la base de ítems**, retiro de los ítems extensos o con instrucciones largas, con el fin de facilitar la comprensión de estos.
- **Adaptación del reporte de resultados**, para que genere un resumen completo de los resultados del TAI.

3.1.3. Cumplimiento de objetivos

El presente trabajo de titulación nombrado como “**Diseño de un Test Adaptativo Informatizado utilizando la metodología de estimación bayesiana para evaluar el conocimiento de los estudiantes de la Universidad Técnica Del Norte, aplicado al 1er y 2do nivel de los estudiantes de la Facultad de Ingeniería en Ciencias Aplicadas.**” usa herramienta APP-TAI, misma que se obtiene como resultado del presente trabajo y forma parte para la solución de los problemas planteados en el presente trabajo de grado. Este trabajo cumple con tres objetivos importantes para su finalización.

Objetivo 1

Elaborar una base teórica que sustente el modelo aplicado para el diseño del Test Adaptativo Automatizado desde la estimación bayesiana y de una base de datos que ofrezca

la interacción con los usuarios. Para el cumplimiento del objetivo se realizaron varias tareas detalladas en la Tabla 3.2:

TABLA 3. 2
OBJETIVO 1

| Objetivo 1 | | |
|---------------------------------------|-----------------------------------|---------------|
| Actividades | Resultados | Estado |
| Búsqueda y recolección de información | Investigación de la base teórica. | Cumplido |
| Búsqueda de metodologías | Estructura de la base teórica | Cumplido |
| Búsqueda de herramientas | Base teórica. | Cumplido |
| Documentación de la investigación | Cumplimiento del objetivo | Cumplido |

Fuente: Propia

El objetivo es cumplido sin novedades, la información recolectada permitió la creación de todo el marco teórico y la selección de las herramientas adecuadas para el desarrollo del presente proyecto.

Objetivo 2

Diseñar un Test Adaptativo Informatizado utilizando la metodología de estimación Bayesiana para evaluar conocimientos en la asignatura de TIC dirigida a estudiantes de la Universidad Técnica Del Norte, aplicando SCRUM como marco de trabajo para el desarrollo. Para el cumplimiento del objetivo se crearon varias actividades detalladas en la Tabla 3.3, desarrollo se realizó utilizando el tablero camban y la metodología Scrum:

TABLA 3. 3
OBJETIVO 2

| Objetivo 2 | | |
|-------------------------------|---------------------------|---------------|
| Actividades | Resultados | Estado |
| Aplicación de scrum | Artefactos Scrum | Cumplido |
| Levantamiento de requisitos | Historias de usuario | Cumplido |
| Diseño de la base de datos | Base de datos | Cumplido |
| Diseño del algoritmo base | Algoritmo adaptativo | Cumplido |
| Diseño del Banco de Ítems | Banco de ítems | Cumplido |
| Desarrollo del aplicativo web | Aplicativo web | Cumplido |
| Implementación en la nube | Resultados del aplicativo | Cumplido |

Fuente: Propia

Como resultado de este objetivo se presenta la herramienta TAI-APP un prototipo funcional que simula una plataforma web desarrollada con Java y JSF, usa los estilos visuales y herramientas que brinda PrimeFaces y publicada en la web por medio de un servidor WildFly

con las herramientas de Google Cloud Platform. Herramienta fundamental para el cumplimiento del objetivo 3.

Objetivo 3

Evaluar los resultados de la investigación propuesta y describe el resultado del cumplimiento de los objetivos y las soluciones a los problemas planteados, también hace un análisis de la herramienta usada para lo cual se aplica encuestas y se tabulan datos, de la misma manera se divide en actividades (Tabla 3.4), el propósito final de este objetivo culmina con el análisis e interpretación de los resultados.

TABLA 3. 4
OBJETIVO 3

| Objetivo 3 | | |
|--|----------------------------------|----------|
| Actividades | Resultados | Estado |
| Análisis de funcionamiento. | Análisis de las encuestas | Cumplido |
| Análisis de los objetivos | Análisis de los objetivos | Cumplido |
| Análisis e Interpretación de resultados. | Interpretación de los resultados | Cumplido |
| Análisis de impacto. | Impacto | Cumplido |

Fuente: Propia

Para el análisis e interpretación de resultados se utiliza gráficos estadísticos con los resultados de las evaluaciones, además de la aplicación de una encuesta a los evaluados con el fin de recolectar información relevante que ayude a la interpretación de resultados.

Problema de estudio

Uno de los principios del TAI es que mientras más ítems se suministre al participante, los resultados pueden ser más exactos, siguiendo esa primicia detallada en la base bibliográfica (1.3.2), se suministró a los siguientes evaluados para este ejemplo 17 preguntas, los resultados obtenidos se reflejan en la tabla 3.5:

TABLA 3. 5
EJEMPLO DE RESULTADOS FECYT

| N. | Nota parcial | Estimación TAI | Estimación Alta |
|----|--------------|----------------|-----------------|
| 1 | 8.1 | 79.88% | No |
| 2 | 9.3 | 75.31% | No |
| 3 | 9.3 | 93.00% | No |
| 4 | 7.7 | 66.75% | No |
| 5 | 8.4 | 95.81% | Si |
| 6 | 8.7 | 46.94% | No |
| 7 | 9.1 | 48.88% | No |
| 8 | 7.2 | 69.81% | No |

| | | | |
|-----------|------|--------|----|
| 9 | 8.8 | 83.00% | No |
| 10 | 5.2 | 44.06% | No |
| 11 | 9.3 | 85.44% | No |
| 12 | 9.3 | 82.19% | No |
| 13 | 8.5 | 97.19% | Si |
| 14 | 8.5 | 83.00% | No |
| 15 | 8.4 | 40.56% | No |
| 16 | 9.6 | 92.88% | No |
| 17 | 8.7 | 61.13% | No |
| 18 | 9.4 | 97.50% | Si |
| 19 | 7.4 | 35.25% | No |
| 20 | 8.8 | 73.38% | No |
| Promedio: | 8.49 | 72.60% | |

Fuente: Propia

En la tabla 3.5 podemos hacer un análisis para el cual se incluyó en una columna la nota respecto a la primera parcial de los estudiantes, y el porcentaje de estimación del conocimiento respecto al participante. Durante el análisis se encontraron distintos inconvenientes, uno de ellos es la falta de interés de los participantes al momento de resolver el test, al no tratarse de una nota se dieron casos que respondieron al azar por lo tanto tuvieron resultados muy bajos que no se reflejaba en el promedio de ellos, por lo tanto, se desechó dichos resultados, por otro lado, existen casos en los cuales los participantes con bajos promedios obtuvieron una mejor estimación como también se muestran en la tabla.

De la tabla 3.5 se obtiene un promedio general de la estimación de conocimientos de los participantes de cada carrera, la tabla 3.6 muestra todas las estimaciones en promedio.

TABLA 3. 6.
RESULTADOS DE LA FACULTAD

| Carrera | Promedio estimación TAI |
|--------------------|--|
| Software | 51.53% |
| Mecatrónica | 49.09% |
| Automotriz | 37.26% |
| Textil | 43.05% |
| Electricidad | 66.32% |
| Industrial | 67.27% |
| Telecomunicaciones | 37.43% |

Fuente: Propia

La tabla 3.7 muestra los resultados con las estimaciones más altas y bajas, se tomaron las dos primeras y las dos últimas del total de estimaciones.

TABLA 3. 7.
NOTAS ALTAS Y BAJAS

| | Carrera | Estimación |
|-------------------|---------|------------|
| 1ra Nota más alta | CIME | 95.71% |
| 2da Nota más alta | CSOFT | 88.50% |
| 1ra Nota más baja | CITEL | 16.29% |
| 2da Nota más baja | CIMANAU | 16.79% |

Fuente: Propia

La figura 61 representa la escala de estimaciones en orden ascendente y el grupo de participantes representando a la carrera a la que pertenecen, en la misma la Carrera de Ingeniería en Software y Carrera de Ingeniería Mecatrónica lideran el grupo de estimaciones altas.



Fig. 61. Notas por carreras
Fuente: Propia

3.2. Interpretación de resultados.

Para la interpretación de los resultados de la encuesta se debe cumplir con la metodología SUS, misma que indica que las preguntas no deben ser alteradas, ni su contenido, ni su orden, de la misma manera no se debe manejar porcentajes en los resultados el incumplimiento de esta regla podría generar datos erróneos.

La valoración de la puntuación va de 1 a 5 como se muestra en la Tabla 3.8, las cuales son asignadas a las respuestas de la encuesta.

TABLA 3. 8
ESCALA DE PUNTUACIONES

| Opciones | Puntuación |
|------------------------------|------------|
| Totalmente en desacuerdo | 1 |
| En desacuerdo | 2 |
| Ni en acuerdo, ni desacuerdo | 3 |
| De acuerdo | 4 |
| Totalmente de acuerdo | 5 |

Fuente: Propia

Para continuar con el proceso se debe crear una tabla cuyos valores son calculados de la siguiente forma:

(Nro. de personas * Valor de Opción)

Ejemplo:

- Para la pregunta 1 hubo 10 participantes que eligieron la opción “Totalmente de acuerdo” que tiene un valor de “5”, el cálculo sería:

$$(10 * 5) = 50,$$

- 26 participantes eligieron la opción “De acuerdo” que tiene un valor de “4” el cálculo sería:

$$(26 * 4) = 104.$$

Este cálculo debe ser aplicado con cada opción de cada una de las preguntas, seleccionando el valor de la respectiva opción elegida por el encuestado, de esta manera se crea la Tabla 3.9:

TABLA 3. 9
RESULTADOS DE LAS ENCUESTAS APLICANDO CÁLCULO SUS

| Preguntas | Totalmente en desacuerdo | En desacuerdo | Ni en acuerdo, ni desacuerdo | De acuerdo | Totalmente de acuerdo |
|-------------|--------------------------|---------------|------------------------------|------------|-----------------------|
| Pregunta 1 | 0 | 2 | 0 | 104 | 50 |
| Pregunta 2 | 17 | 14 | 15 | 12 | 25 |
| Pregunta 3 | 4 | 6 | 18 | 40 | 70 |
| Pregunta 4 | 13 | 18 | 18 | 28 | 10 |
| Pregunta 5 | 0 | 6 | 18 | 80 | 40 |
| Pregunta 6 | 18 | 12 | 18 | 16 | 15 |
| Pregunta 7 | 1 | 8 | 0 | 84 | 55 |
| Pregunta 8 | 10 | 26 | 24 | 16 | 10 |
| Pregunta 9 | 3 | 12 | 27 | 52 | 30 |
| Pregunta 10 | 13 | 18 | 24 | 20 | 10 |

Fuente: Propia

Con los datos de la tabla 3.9 se realiza un cálculo de promedio por cada pregunta y posteriormente dividir la tabla en dos, una con las preguntas impares y otra con las pares.

El cálculo se realiza sumando el total de puntos y dividirlos para el número de encuestados, además el sistema de usabilidad SUS dice que para cada una de las preguntas impares (1,3,5,7,9) se debe restar 1 del puntaje total de cada pregunta.

Ejemplo Pregunta 1:

$$(2 + 104 + 50) / 37$$

$$156 / 37 = 4.22 - 1 = 3.22$$

TABLA 3. 10
SUS-PREGUNTAS IMPARES

| Preguntas | Promedio |
|------------------|-----------------|
| Pregunta 1 | 3.22 |
| Pregunta 3 | 2.73 |
| Pregunta 5 | 2.89 |
| Pregunta 7 | 3.00 |
| Pregunta 9 | 2.35 |
| Total | 14.19 |

Fuente: Propia

En el caso de las preguntas pares (2,4,6,8,10) se debe restar su valor de 5, como se muestra en el ejemplo de la pregunta 2:

$$5 - \sum (\text{Promedio preguntas pares})$$

$$5 - ((17 + 14 + 15 + 12 + 25) / 37)$$

$$5 - 2.24 = 2.76$$

TABLA 3. 11
SUS-PREGUNTAS PARES

| Preguntas | Promedio |
|------------------|-----------------|
| Pregunta 2 | 2.76 |
| Pregunta 4 | 2.65 |
| Pregunta 6 | 2.86 |
| Pregunta 8 | 2.68 |
| Pregunta 10 | 2.70 |
| Total | 13.65 |

Fuente: Propia

Finalmente, el método SUS indica que se deben sumar los dos resultados parciales y multiplicarlos por 2.5, el resultado convierte el rango de valores de 0 a 40 a una escala percentil de 0 a 100, que sirve para ubicarnos dentro de la tabla de rangos y comprobar si el sistema cumple con la escala mínima de calificación.

$$\begin{aligned}
 &(\text{Total Impares} + \text{Total pares}) * 2.5 \\
 &(14.19 + 13.65) * 2.5 = \mathbf{69,9}
 \end{aligned}$$

3.3. Análisis de impacto

Posterior al análisis realizado utilizando la metodología “SUS” se estima que el puntaje mínimo que un sistema web debe alcanzar debe superar el 68%, si el porcentaje es menor significaría que el sistema web posee fallas de usabilidad y requiera corregir ciertos errores.

De acuerdo con el análisis anterior se pudo evidenciar una puntuación de 96.9% como se muestra en la figura 62.

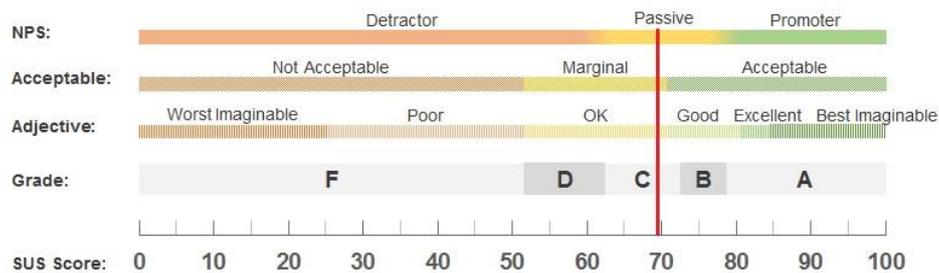


Fig. 62. Gráfico escala SUS
Fuente: (<https://measuringu.com/interpret-sus-score/>)

La puntuación ubica al sistema web dentro del grado C, el cual se encuentra en un rango aceptable, la puntuación tan cerca del 70% indica que, de existir errores al momento de generar la encuesta, el sistema podría estar superando dicho valor.

CONCLUSIONES

- De acuerdo con la base bibliográfica y los resultados recopilados se determina que los métodos condicionados han resultado ser más asertivos para determinar los criterios de selección de preguntas de acuerdo con el nivel de conocimiento del estudiante.
- El Test diseñado pudo adaptarse a las características del participante, el orden de preguntas e incluso de opciones son diferentes dependiendo del evaluado, el sistema demuestra ser eficiente para la obtención de niveles estimados sobre el conocimiento del tema, fortalece los métodos de evaluación y genera altos niveles de satisfacción y confianza por parte de los participantes que miran con optimismo este tipo de evaluaciones.
- La aplicación de la evaluación dentro de la facultad de ingeniería generó resultados que promediando por carreras, las estimaciones se aproximan sin embargo, se tiene sus excepciones, individualmente los datos son dispersos y se nota la diferencia de conocimiento, por tanto se demuestra existen estudiantes con bajo y alto conocimiento del tema en cada y se encuentran dispersos.
- El análisis del aplicativo indica que el participante se deja llevar mucho por el diseño de los sistemas y muy poco por el funcionamiento, pese al desconocimiento del funcionamiento del TAI más del 70% aprobaron su uso en evaluaciones, el otro porcentaje son estudiantes indiferentes y apenas un 10% de estudiantes que se resisten al cambio y que miran con desconfianza este tipo de sistemas de evaluación.
- El diseño del Banco de ítems necesitó grandes cantidades de reactivos sobre el tema de estudio a evaluar, para la calibración de estos se optó por aplicar los ítems con grupos de prueba y en orden indistinto y en forma de encuesta, la finalidad fue obtener valores que solicita la TRI, esto sin que los participantes se sientan presionados y generen datos incorrectos, con el análisis posterior se optó por modificar o eliminar diferentes ítems.

RECOMENDACIONES

- Disponer de más ítems sobre el tema, dependiendo del grupo a evaluar consultar el temario con el docente a cargo, posteriormente para la calibración se requiere un grupo de prueba que tengan similares características. Usar herramientas de análisis para calcular el nivel de dificultad, excluir evaluaciones de estudiantes rebeldes que no quieran cooperar ya que podría suministrar información incorrecta.
- Para optimizar costos durante el desarrollo es recomendable el uso de herramientas de software libre, de ser posible usar servicios web y trabajar con java en backend y en frontend usar un lenguaje diferente con un framework que permitan consumir servicios web, para la publicación de la aplicación se puede usar las pruebas gratuitas en las plataformas de Cloud Computing, tener en cuenta que múltiples plataformas ofrecen descuentos o créditos de prueba para estudiantes, para evitar altos costos.
- En trabajos futuros se recomienda continuar con la investigación con métodos diferentes para la estimación del nivel de rasgo y usando librerías externas para los cálculos avanzados, comparar resultados y evaluar el funcionamiento del algoritmo con el fin de obtener un mejor aporte.
- Se recomienda implementar en el sistema un módulo de retroalimentación con el fin de cubrir áreas en las cuales obtuvo puntajes bajos, sumados a la explicación de los docentes generarían un mejor aporte a la comprensión del tema.

BIBLIOGRAFÍA

- Abad, F. J., Olea Díaz, J., Ponsoda, V., & Real, E. (2002). Estimación de habilidad y precisión en tests adaptativos informatizados y tests óptimos: Un caso práctico. *Revista Electrónica de Metodología Aplicada*, 7(1), 1–20.
- Aguilar, L. J. (2012). COMPUTACIÓN EN LA NUBE Notas para una estrategia española en cloud computing. *Revista Del Instituto Español de Estudios Estratégicos*, 2, 89–112.
<https://revista.ieee.es/article/view/406>
- Attorresi, H. F., & Lozzia, G. S. (2012). Especificación del algoritmo para un Test Adaptativo Informatizado de Analogías Verbales. *Summa Psicológica*, 9(2), 15–23.
<https://doi.org/10.18774/448x.2012.9.90>
- Balbuca Ramones, D. M., & Ortiz Ramírez, J. M. (2017). *Análisis comparativos de servidores de aplicaciones open source para la plataforma Java EE*. [UNIVERSIDAD NACIONAL DE CHIMBORAZO].
<http://dspace.unach.edu.ec/bitstream/51000/3397/1/UNACH-EC-ING-SIS-COM-2017-0001.pdf>
- Binyamin, S., Rutter, M., & Smith, S. (2016). the Utilization of System Usability Scale in Learning Management Systems: a Case Study of Jeddah Community College. *ICERI2016 Proceedings*, 1(December), 5314–5323.
<https://doi.org/10.21125/iceri.2016.2290>
- Castelló Pascual, P. (2018). *Simulación de redes Bayesianas*. Universidad Politécnica de Madrid.
- Conejo, R., Millán, E., Perez, J. L., & Trella, M. (2001). Modelado del alumno: un enfoque bayesiano. *INTELIGENCIA ARTIFICIAL*, 5(12), 50–58.
<https://doi.org/10.4114/ia.v5i12.707>
- Díaz, J., & Ponsoda, V. (2002). *TEST ADAPTATIVOS INFORMATIZADOS*.
https://www.researchgate.net/publication/265040034_TEST_ADAPTATIVOS_INFORMATIZADOS
- Dobber, S. (2017). MODELOS DE TEST ADAPTATIVOS PARA EVALUACIÓN AUTOMÁTICA. In *Universidad Autonoma de Madrid*. Universidad Autonoma de Madrid.
- Egas, L., & Játiva, J. (2008). *Evolución de las Metodologías de Desarrollo de la Ingeniería de Software en el Proceso la Ingeniería de Sistemas Software y Determinación de una metodología adaptable orientada a una organización pequeña*.
- Fals, R., & Piloto, Y. (2009). ALGORITMO PARA LA CLASIFICACIÓN DE PREGUNTAS Y EVALUACIÓN DEL NIVEL DE CONOCIMIENTO EMPLEANDO REDES BAYESIANAS Y LA TEORÍA DE RESPUESTA AL ÍTEM. *UCI, Cuba*.

- Flores Castro, E. G. (2018). Implementación de una base de datos heterogénea distribuida entre los SGBDs ORACLE, MySQL y PostgreSQL con replicación, mediante un script bash implementado en el sistema operativo CentOS usando software libre. *INNOVA Research Journal*, 3(2.1), 68–77. <https://doi.org/10.33890/innova.v3.n2.1.2018.668>
- García, P. E., Abad, F. J., Olea Díaz, J., & Aguado, D. (2013). Un nuevo método de standard setting basado en la TRI: Aplicación a eCat-Listening. *Psicothema*, 25(2), 238–244. <https://doi.org/10.7334/psicothema2012.147>
- García, R. S. (2007). *ANÁLISIS DE SENSIBILIDAD EN REDES BAYESIANAS GAUSSIANAS*. UNIVERSIDAD COMPLUTENSE DE MADRID.
- Gartner. (2021). *Magic Quadrant para servicios de infraestructura y plataforma en la nube*. Gartner. <https://www.gartner.com/technology/media-products/reprints/AWS/1-271W1OTA-ESP.html>
- Godoy, F. J., Cabrera, I. J., Rocha, J., González, J. J., & Rodríguez, C. (2017). Aplicación Web Para Autocorrección De Pruebas. *CISCI 2017 - Decima Sexta Conferencia Iberoamericana En Sistemas, Cibernética e Informática, Decimo Cuarto Symposium Iberoamericano En Educacion, Cibernética e Informática, SIECI 2017 - Memorias, Cisci*, 300–305.
- Hidalgo Montesinos, M., & French, B. (2016). Una introducción didáctica a la Teoría de Respuesta al Ítem para comprender la construcción de escalas. *Revista de Psicología Clínica Con Niños y Adolescentes*, 3(2), 13–21.
- JavaServerFaces.org. (2020). *JavaServer Faces.org*. JavaServerFaces.Org.
- Jiménez López, G. J. (2019). *TEST ADAPTATIVOS INFORMATIZADOS PARA INVIDENTES : UNA ALTERNATIVA PARA EVALUAR CON EQUIDAD*. Universidad Nacional de Colombia.
- Jiménez Sánchez, M. (2009). *MODELO DE EVALUACIÓN ADAPTATIVA DEL NIVEL DE CONOCIMIENTOS DEL ESTUDIANTE PARA SISTEMAS TUTORIALES INTELIGENTES*. *Universidad Nacional de Colombia*. <http://www.bdigital.unal.edu.co/913/>
- Kuz, A., Falco, M., & Giandini, R. S. (2018). Comprendiendo la Aplicabilidad de Scrum en el Aula: Herramientas y Ejemplos. *Revista Iberoamericana de Tecnología En Educación y Educación En Tecnología*, 21, e07. <https://doi.org/10.24215/18509959.21.e07>
- León Rodríguez, G. de la C., & Viña Brito, S. M. (2017). La inteligencia artificial en la educación superior. Oportunidades y amenazas. *INNOVA Research Journal*, 2(8.1), 412–422. <https://doi.org/10.33890/innova.v2.n8.1.2017.399>
- López Cuadrado, J. (2008). *Evaluación mediante test adaptativos informatizados en el contexto de un sistema adaptativo para el aprendizaje de la lengua vasca*. Universidad del País Vasco.

- López, J. (n.d.). *Teorema de Bayes*. Economipedia.
<https://economipedia.com/definiciones/teorema-de-bayes.html>
- Luckin, R., Holmes, W., Griffiths, M., & Forcier, L. (2016). *Intelligence unleashed: An argument for AI in education*. Pearson.
- Martins, F., & Palella, S. (2006). Metodología de la investigación cuantitativa. *FEDUPEL*.
- Mell, P., & Grance, T. (2009). *Effectively and Securely Using the Cloud Computing Paradigm*.
- Millán, E. (2000). Sistema bayesiano para modelado del alumno. In *System and Computer Thesis*. UNIVERSIDAD DE MÁLAGA.
- Millán, E., & Pérez-de-la-Cruz, J. L. (2004). Un algoritmo de diagnóstico para modelado del alumno basado en test adaptativos y redes bayesianas. *Universidad de Málaga*, 1–24.
<http://www.lcc.uma.es/~eva/doc/materiales/millane.pdf>
- Molina Montero, B., Vite Cevallos, H., & Dávila Cuesta, J. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales*, 2(15), 113–121.
<https://doi.org/https://doi.org/10.31876/re.v2i17.269>
- Moya Ricardo, D., Pérez Gómez, Y., & Ruiz Cordovéz, R. (2016). La Teoría de Respuesta al Ítem para la evaluación del aprendizaje en Matemática The Item Response Theory for the evaluation of Mathematics learning. In *dialnet.unirioja.es* (Vol. 16, Issue 55).
<https://dialnet.unirioja.es/servlet/articulo?codigo=5678501>
- Ocaña Fernández, Y., Valenzuela Fernández, A., Gálvez Suárez, E., Aguinaga Villegas, D., Nieto Gamboa, J., & López Echevarria, T. I. (2019). Gestión del conocimiento y tecnologías de la información y comunicación (TICs) en estudiantes de ingeniería mecánica. *Apuntes Universitarios*, 10(1), 77–88.
<https://doi.org/10.17162/revapuntes.v10i1.195>
- Otálora, C. V. H. (2018). Servidor de Aplicaciones. *OneDrive Institucional*.
- Pagés Chacón, D. (2010). *Cartucho de AndroMDA para JSF interpretando nueva estrategia de modelado*. D - Instituto Superior Politecnico Jose Antonio Echeverria. CUJAE.
- Palacios, D., Guamán, J., & Contento, S. (2018). Análisis del rendimiento de librerías de componentes Java Server Faces en el desarrollo de aplicaciones web. *NOVASINERGIA REVISTA DIGITAL DE CIENCIA, INGENIERÍA Y TECNOLOGÍA*, 1(2), 54–59. <https://doi.org/10.37135/unach.ns.001.02.06>
- Pérez Román, A. (2020). *Comparación de rendimiento entre bases de datos Relacionales, NoSQL y Blockchain Comparación de rendimiento entre PostgreSQL, MongoDB y Kaleido*. UNIVERSIDAD DE MÁLAGA.
- Ponce Briones, D. K. (2016). *ANÁLISIS COMPARATIVO DE LOS ENTORNOS DE DESARROLLO INTEGRADOS (IDE): ECLIPSE, NETBEANS Y JDEVELOPER PARA EL DESARROLLO DE APLICACIONES JAVA ENTERPRISE EDITION*. Universidad De

Guayaquil.

- Ram, J., Olea, J., & Ponsoda, V. (2004). *Tests Adaptativos Informatizados*. 55–61.
- Ramón, K. (2016). *Desarrollo de un sistema web utilizando Redes Bayesianas para enseñanza del idioma Inglés en la escuela Zoila Alvarado de Jaramillo*.
<https://doi.org/10.13140/RG.2.2.33486.05449>
- Rebollo, P., García Cueto, E., Zardaín, P. C., Cuervo, J., Martínez, I., Alonso, J., Ferrer, M., & Muñiz, J. (2009). Desarrollo del CAT-Health, primer test adaptativo informatizado para la evaluación de la calidad de vida relacionada con la salud en España. *Medicina Clínica*, 133(7), 241–251. <https://doi.org/10.1016/j.medcli.2008.09.045>
- Riquelme, M. (n.d.). *Teorema de Bayes (fórmula e importancia)*. Web y Empresas.
Retrieved August 23, 2020, from <https://www.webyempresas.com/teorema-de-bayes/>
- Rodríguez, G. S. (2019). Computación en la nube: algunas consideraciones técnico-jurídicas. *LEX*, 17(23), 145. <https://doi.org/10.21503/lex.v17i23.1674>
- Rojas, A. N. H., Sierra, R. B., & López, G. J. J. (2016). Validity in computerized adaptive testing: An alternative to assess visually impaired individuals. *Revista Avaliação Psicológica*, 14(3), 299–307. <https://doi.org/10.15689/ap.2015.1403.01>
- Schwaber, K., & Sutherland, J. (2020). La Guía de Scrum. La Guía Definitiva de Scrum: Las Reglas del Juego. *Scrum.Org*, 16.
- Sierra, F., Valdelamar, J., Hernández, F., & Sarmiento, L. (2007). TEST ADAPTATIVOS INFORMATIZADOS. *Avances En Medición*, 5, 157–162.
- Soto, G. (2011). El teorema de Bayes. *Revista De Educación Matemática*, 26(3).
- Tseng, W. T. (2016). Measuring English vocabulary size via computerized adaptive testing. *Computers and Education*, 97, 69–85. <https://doi.org/10.1016/j.compedu.2016.02.018>
- Valarezo Pardo, M. R., Honores Tapia, J. A., Gómez Moreno, A. S., & Vines Sánchez, L. F. (2018). COMPARACIÓN DE TENDENCIAS TECNOLÓGICAS EN APLICACIONES WEB. *3C Tecnología. Glosas de Innovación Aplicadas a La Pyme.*, 7(3), 28–49.
<https://doi.org/http://dx.doi.org/10.17993/3ctecno.2018.v7n3e27.28-49>
- WildFly.org. (n.d.). *WildFly*.