

Development of Management Software for Accreditation experience in CAPSTON course

Freddy Gonzales-Saji, Msc¹, Lady Concha-Diaz¹, Lisbeth Ortiz-Huarachi¹, Edwar Saire-Peralta¹, Giovanni Cabrera-Málaga¹

¹Universidad Nacional de San Agustín de Arequipa, Perú

fgonzales@unsa.edu.pe, lconcha@unsa.edu.pe, lordizh@unsa.edu.pe, esaire@unsa.edu.pe, gcabrerama@unsa.edu.pe

Abstract – The work describes the experience of students in the last semester of the CAPSTONE course called Software Engineering, which involves the implementation of software to optimize the accreditation process applied to the Professional School of Industrial Relations of the National University of San Agustín de Arequipa, in materials and methods, the SCRUM methodology was adapted to comply with the planned activities, the developed system demonstrated effective compliance with the quality standards required for accreditation, significantly improving document management, the lessons learned about the importance of communication stand out. effectiveness of the team, the flexible and scalable design of the software, and interdisciplinary collaboration, from the study it can be concluded that the experience reinforces the viability of adapting agile methodologies in different contexts to respond to changing and complex needs.

Keywords-- Software, development, accreditation process, document system, SCRUM, CAPSTONE

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

Desarrollo de Software de Gestión para Acreditación experiencia en curso CAPSTONE

Freddy Gonzales-Saji, Msc¹, Lady Concha-Diaz¹, Lisbeth Ortiz-Huarachi¹, Edwar Saire-Peralta¹, Giovanni
Cabrerá-Málaga¹

¹Universidad Nacional de San Agustín de Arequipa, Perú

fgonzales@unsa.edu.pe, lconcha@unsa.edu.pe, lordizh@unsa.edu.pe, jvaldiviar@utp.edu.pe

Resumen—El trabajo describe la experiencia de estudiantes de último semestre del curso CAPSTONE denominado Ingeniería de Software, la misma que involucra la implementación de un software para optimizar el proceso de acreditación aplicado a la Escuela Profesional de Relaciones Industriales de la Universidad Nacional de San Agustín de Arequipa, en materiales y métodos se adaptó la metodología SCRUM para cumplir con las actividades planificadas, el sistema desarrollado demostró el cumplimiento efectivo de los estándares de calidad requeridos para la acreditación mejorando significativamente la gestión documental, se destacan las lecciones aprendidas sobre la importancia de una comunicación efectiva del equipo, el diseño flexible y escalable del software, y la colaboración interdisciplinaria, del estudio se puede concluir que la experiencia refuerza la viabilidad de adaptar metodologías ágiles en distintos contextos para responder a necesidades cambiantes y complejas.

Palabras clave: Software, desarrollo, proceso de acreditación, sistema documental, SCRUM, CAPSTONE.

I. INTRODUCCIÓN

La escuela profesional de Ingeniería de Sistemas [1] de la Universidad Nacional de San Agustín de Arequipa acreditada por ABET [2], dentro de sus asignaturas CAPSTONE cuenta con Proyecto de Ingeniería de Software el cual busca integrar competencias anteriormente adquiridas para el desarrollo de un producto [3].

En el marco del curso de Proyecto de Ingeniería de Software, que se desarrolla a lo largo de un año académico, enfrentamos el desafío de desarrollar un software a medida para una entidad o empresa que solicite el apoyo de la escuela profesional de Ingeniería de Sistemas. En nuestro caso particular, colaboramos con la Escuela Profesional de Relaciones Industriales, que se encuentra en el proceso de obtener la acreditación. Donde su principal obstáculo era la ausencia de un sistema automatizado para la generación de informes y gestión de los 34 estándares de calidad establecidos por el SINEACE [4], lo que complicaba significativamente su proceso de acreditación.

Ante esta necesidad, nuestra tarea como equipo consistió en diseñar un sistema web de gestión, que permita automatizar los procesos de gestión y flujo de información interna, para lograr así una mejor organización y búsqueda documentaria [5], el desarrollo de este sistema centralizará y automatizará estos procesos esenciales, enfrentando así la problemática de falta de eficiencia y disminuyendo la probabilidad de errores en la documentación crucial para la acreditación. El presente estudio tiene como objetivo compartir el proceso de desarrollo de nuestro proyecto, destacando y describiendo nuestra experiencia en cada una de las etapas de desarrollo de software. Mostraremos cómo se ha aplicado aplicado y fusionado las habilidades y conocimientos que se han adquirido a lo largo de la carrera.

Este artículo se organiza de acuerdo a la siguiente estructura: La Sección II aborda los trabajos relacionados, proporcionando un marco de referencia y situando nuestro estudio en el contexto de investigaciones previas. En la Sección III, se expone la descripción de los materiales y métodos utilizados, detallando las herramientas y procedimientos empleados en el desarrollo del proyecto. La Sección IV se dedica al proceso de desarrollo, explicando las fases y estrategias implementadas para la construcción del sistema web de gestión. Los resultados obtenidos se presentan en la Sección V, donde se evalúa el impacto y la efectividad del sistema desarrollado. En la Sección VI, se presentan nuestras conclusiones, resumiendo los logros y aprendizajes clave del proyecto. Finalmente, expresamos nuestros agradecimientos y los trabajos futuros en las Secciones VII y VIII

II. TRABAJOS RELACIONADOS

Mencionando trabajos previos como antecedentes, encontramos varios temas en cuales se da un enfoque parecido al nuestro respecto a la automatización de procesos facilitando el antiguo proceso manual que se solía hacer, entre estos tenemos:

En [6] nos menciona que el Business Process Management (BPM) se utiliza para gestionar los procesos

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

académicos y administrativos en Instituciones Públicas de Educación. Esto se debe a la necesidad actual de mejorar eficientemente el rendimiento de una organización, ya que el rendimiento de esta depende de la eficiencia de sus procesos. Estos procesos suelen ser actividades repetitivas, lo que genera un desperdicio de tiempo que podría utilizarse de mejor manera. Se destaca su aplicación tanto en organizaciones privadas como en instituciones públicas, incluyendo el ámbito educativo. Además, de la incorporación de las Tecnologías de la Información y Comunicación (TIC) en estos nuevos modelos de gestión.

En [8] se enfocan en mejorar la eficiencia de una organización educativa, teniendo los mismos problemas presentes que suelen tener muchas organizaciones, las cuales no se actualizaron con la tecnología. Específicamente tratan de mejorar el desempeño y eficiencia de las funciones administrativas, y para lograr este mejoramiento, se planteó un desarrollo de un sistema de gestión digital con un enfoque de procesos, beneficiando a 3960 autoridades, aclarando que la inversión que se dará será mínima, para mejorar la productividad y los tiempos de respuesta. Al terminar el proyecto se planteó puntos de cumplimiento, identificación en el diagnóstico institucional y el plan de desarrollo destacando la posibilidad de replicar el proyecto en otros departamentos, reusando la misma tecnología desarrollada.

En [9] nos mencionan que la calidad educativa en el país viene promoviendo la necesidad de evaluación y acreditación, con lo cual logran una ventaja competitiva a los programas de estudios. Sin embargo, la interpretación de los estándares para poder analizarlos, sustentarlos y valorarlos, no se puede realizar de una manera unívoca para todos los programas de estudios, debido a que el logro de cada estándar dependerá de las particularidades de cada institución y no estrictamente de lo escrito en el modelo. Por lo que proponen un modelo de información que permita recopilar y gestionar los datos relevantes para la Universidad Privada Antenor Orrego para evidenciar el nivel de cumplimiento de cada estándar, debido a que el instrumento de evaluación propuesto por el SINEACE mide sólo si se realiza actividades coherentes para cubrir cada estándar. Este modelo incluye las siguientes entidades principales: programas de estudios, estándares de acreditación, indicadores de cumplimiento y evidencias. Estas entidades se relacionan entre sí mediante evaluaciones de cumplimiento, seguimiento de acciones de mejora y un calendario de actividades. El sistema permite el registro y seguimiento de evidencias, la evaluación del cumplimiento de estándares, la generación de informes, la gestión de acciones de mejora y el seguimiento de plazos a través de un calendario y recordatorios. El modelo de información se adapta a las necesidades específicas de la UPAO y los estándares de acreditación del Sineace.

III. MATERIALES Y MÉTODOS

Para identificar la metodología de desarrollo de software más adecuada para el proyecto, se llevó a cabo un análisis detallado de las diversas metodologías disponibles, encontrándose en la bibliografía que estas se clasifican en dos grupos principales: Las metodologías tradicionales (o pesadas) y las metodologías ágiles [7]. Las metodologías tradicionales, como el modelo en cascada y el modelo en espiral, se fundamentan en las mejores prácticas de ingeniería del software y ofrecen un enfoque estructurado y secuencial para el desarrollo. Sin embargo, para un proyecto de gran envergadura como el nuestro, que demanda flexibilidad y capacidad de adaptación ante cambios, las metodologías ágiles, tales como Scrum y Kanban, se presentaron como opciones más viables [11]. Estas metodologías se caracterizan por permitir respuestas rápidas y adaptativas en entornos dinámicos, a través del uso de iteraciones cortas y colaboración estrecha con el cliente, poniendo especial énfasis en la entrega continua y la capacidad de adaptarse a cambios con agilidad [12][13].

Inicialmente, se optó por implementar la metodología Scrum debido a su enfoque en reuniones frecuentes y la presentación regular de prototipos, lo que parecía alinearse bien con los requisitos de nuestro proyecto. No obstante, a medida que avanzaba el desarrollo, nos dimos cuenta de que aplicar Scrum de manera estricta no se ajustaba completamente a nuestras circunstancias particulares como estudiantes universitarios, ni a nuestro estilo de vida. Esto nos llevó a adaptar Scrum a nuestras necesidades específicas, adecuándolo en un marco de trabajo más flexible. Esta adaptación nos permitió conservar los beneficios de Scrum, como las iteraciones cortas y la colaboración estrecha con el cliente, mientras se ajustaba su estructura para que encaje mejor en nuestro contexto académico y personal, asegurando así un equilibrio entre la metodología elegida y nuestras limitaciones prácticas.

El ciclo de trabajo que se usó para la metodología se muestran en la Fig. 1 y la Fig. 2.

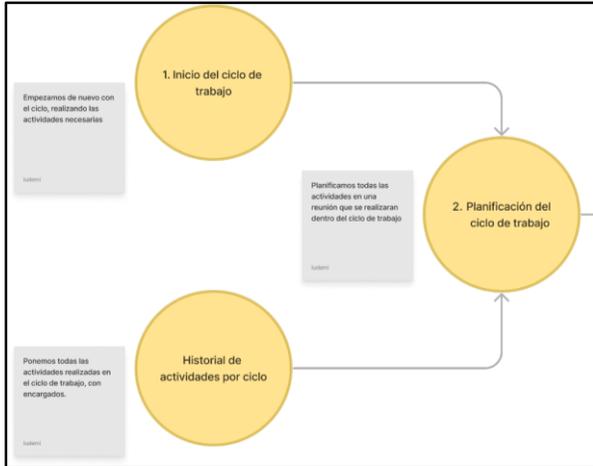


Fig. 1. Ciclo de trabajo parte 1
Fuente: Propia

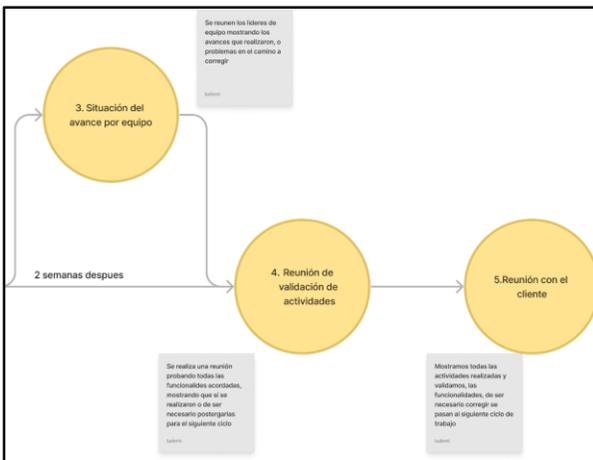


Fig. 2. Ciclo de trabajo parte 2
Fuente: Propia

Los entregables que se manejaron para cada ciclo (sprint) fueron los siguientes:

- Planificación del siguiente ciclo de trabajo
- Actividades desarrolladas durante el ciclo de trabajo
- Actas de reunión del equipo: Para poder evidenciar cómo va el avance del equipo donde solo se reúnen los líderes, y también se realizaba otra reunión antes de finalizar el ciclo de trabajo para validar las actividades realizadas.
- Matriz de trazabilidad: Luego de validar las actividades realizadas se actualizaba la matriz de trazabilidad, tomando en cuenta los requisitos ya realizados teniendo secciones como Estado, Preguntas, Comentarios.
- Informe de control de horas: Para poder medir el esfuerzo de cada uno se maneja un control de hora y se hace un reporte por cada ciclo de trabajo, usando la herramienta bixpe.

Internamente el equipo luego mantuvo la estructura de trabajo de la Fig. 3, para organizar carpetas y archivos.

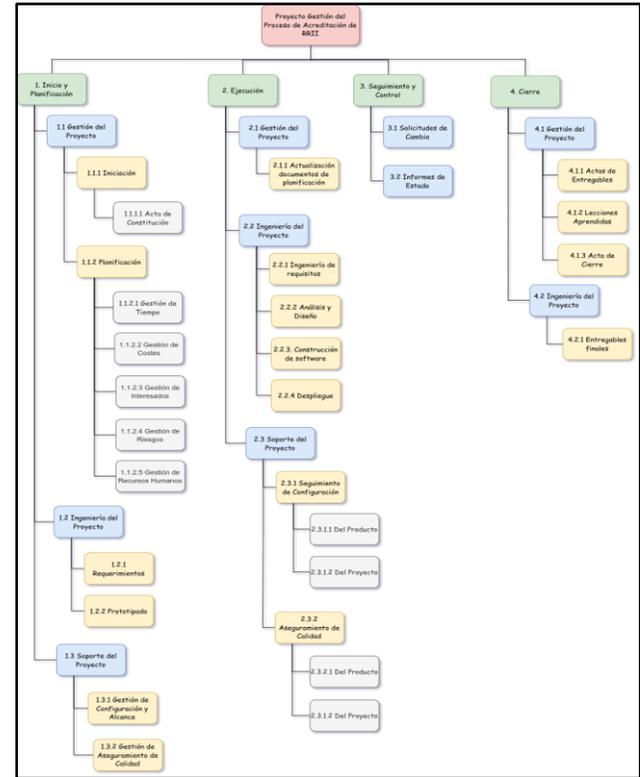


Fig. 3. EDT del proyecto
Fuente: Propia

La ingeniería de requisitos (IR) es una parte fundamental en el desarrollo de software y la gestión de proyectos. Se trata de un subconjunto de la ingeniería de sistemas que se ocupa de descubrir, desarrollar, rastrear, analizar, calificar, comunicar y gestionar los requisitos que definen el sistema en diferentes niveles de abstracción [14]. Estos requisitos son las descripciones de lo que el sistema debe realizar, incluyendo los servicios que ofrece y las restricciones bajo las cuales opera. Representan las necesidades de los clientes en cuanto a la funcionalidad que esperan del sistema [15]. El proceso de ingeniería de requisitos es crucial para garantizar que el producto final cumpla con los objetivos y funcionalidades deseadas, evitando errores costosos y retrabajos a lo largo del ciclo de vida del proyecto.

El proceso de desarrollo de requisitos software se divide en cuatro fases [16]. La primera fase, la Elicitación de requisitos, implica identificar las necesidades y restricciones de las partes interesadas mediante diversas técnicas. La segunda fase, el Análisis, busca garantizar la calidad de los requisitos y definir los límites del sistema. La tercera fase, la Especificación, consiste en documentar los requisitos acordados con un nivel de detalle adecuado. En la última

fase, la Validación, se revisan los requisitos y el documento de especificación para identificar posibles problemas antes del diseño del sistema. Durante el levantamiento de requisitos, se recopila información sobre las expectativas del sistema y se identifican relaciones lógicas entre sus elementos, lo que puede conducir a educciones e ilaciones importantes. En la Tabla 1 se muestra parte de a las educciones e ilaciones identificadas para el proyecto.

TABLA 01.
EDUCCIONES E ILACIONES

Educciones	Ilaciones
EDU-001 - Gestión de autenticación	ILA 001: Autenticación con Cuenta de Google
	ILA 002: Cierre de Sesión
	ILA 003: Tiempo de Inactividad
EDU-002 - Gestión de usuarios	ILA 004: Añadir Usuario al Sistema
	ILA 005: Cambio de Permisos de Usuario
	ILA 006: Lista de Usuarios
EDU-003 - Gestión de estándares	ILA 007: Informe de narrativas
	ILA 008: Reporte de estándar
	ILA 009: Administración del estándar
	ILA 010: Permisos de estándar
	ILA 011: Administración del estandar 08
	ILA 012: Administración del cierre de semestre

IV. DESARROLLO

El proyecto se realizó tomando en cuenta el siguiente cronograma y evidenciando lo avanzando en cada etapa:

TABLA 02
DESARROLLO EVIDENCIADO POR SPRINT

Etapa 1: Inicio y Planificación	<p>Se inició elaborando el acta de constitución del proyecto a raíz de una primera reunión con la stakeholder principal</p> <p>Planificación a través de las gestión de tiempo, gestión de costos, gestión de alcance, gestión de recursos humanos, gestión de comunicaciones, gestión de interesados y gestión de riesgos.</p> <p>Elección de la metodología y su justificación, y planes de gestión del aseguramiento de la calidad y configuración del proyecto</p>
Etapa 2: Ejecución y Desarrollo	<p>Sprint 06: Refactorización de los componentes en Figma y algunas funcionalidades y Rediseño de la base de datos</p> <p>Sprint 07: Implementación de los componentes diseñados y corrección de errores</p> <p>Sprint 08: Revisión y desarrollo de la gestión de estándares y planes de mejora (creación de</p> <p>Sprint 09: Desarrollo de la educción de gestión de usuarios y gestión de cabeceras (creación de endpoints)</p> <p>Sprint 10: Generación de reportes (endpoints), refactorización de toda la gestión de evidencias, diseño de interfaces e integración con la API y ejecución de pruebas.</p> <p>Sprint 11: Refactorización de diseño, Interfaces creadas e integración con la API.</p>
Etapa 3: Cambios, mejoras y cierre	<p>Sprint 12: Creación de casos de uso para pruebas, Diseño de los cambios solicitados, pruebas de frontend y pruebas de backend, colocar rango de periodo para los informes.</p> <p>Sprint 13: Funcionalidad de cierre de semestre, configuración de docker para el despliegue y reestructuración.</p> <p>Sprint 14: Análisis de nuevos cambios, Diseño de interfaces para los cambios, reuniones de validación.</p> <p>Sprint 15: Instalación del software en el servidor de la escuela, Implementación de formularios para reportes,</p>

	Implementación de cierre de semestre.
	Sprint 16: Presentación final del software completo y validación, Capacitación al personal docente, últimas pruebas y acta de cierre.

A. Módulos implementados

A continuación se muestran las capturas de pantalla de las educciones implementadas en el software entre los cuales el más complicado y que sería la base para el proyecto y para resaltar aún más nuestra experiencia en el proceso de desarrollo, fue la gestión de evidencias la cual resultó ser la más desafiante de todas ya que involucró la gestión de narrativas y también requirió la creación de un sistema propio como Google Drive con especificaciones detalladas para organizar adecuadamente las evidencias, códigos, formatos y enlaces. Esto representó un esfuerzo considerable por parte del equipo de frontend y otra de las funcionalidades más importantes que manejamos fue la generación de informes ya que esta usa la información de todo el sistema como planes de mejora, evidencias, estándares usuarios representando todo un desafío para el equipo al tener que adaptarlo específicamente a las necesidades del cliente, la Figura 4 muestra una evidenciad el trabajo desarrollado.

1) EDU01- Gestión de autenticación



Fig. 4. Captura de pantalla de autenticación
Fuente: Propia

Tecnologías Usadas

Durante el proyecto se empezó utilizando React como framework de trabajo para el frontend, ya que fue como lo dejó el grupo de trabajo anterior, y Laravel como backend. Sin embargo, la estructura del frontend y como estaba desarrollada, ocasionaba que al tratar de hacer un cambio, este afectara a otro bloque desarrollado, haciendo muy difícil seguir haciendo cambios al proyecto, era poco escalable; por lo que al final decidimos cambiar a Next.js y volver a realizar el proyecto desde cero quedándonos con la siguiente estructura, la Figura 5 muestra la arquitectura de tecnologías usadas

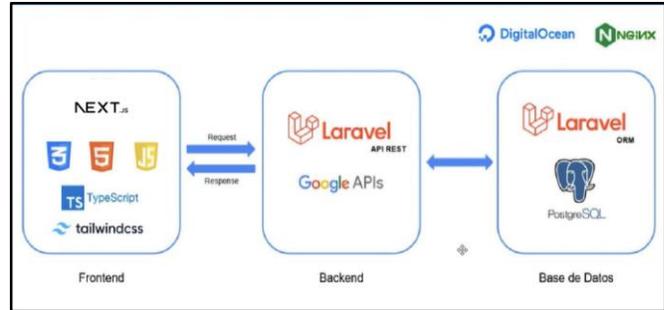


Fig. 5 Arquitectura de herramientas utilizadas
Fuente: Propia

C. Base de datos

El proyecto se sustenta en una base de datos cuidadosamente diseñada, acompañada de un diagrama de ER detallado que ilustra de manera completa nuestra propuesta. La base de datos fue aprobada por un docente especializado, garantizando la calidad y coherencia de nuestro planteamiento, el proceso de la creación de la misma fue iniciar por un diseño que fue validado por los asesores del curso, para luego implementar en el gestor correspondiente.

D. Aseguramiento de calidad

Para asegurar la calidad del proyecto se optó por utilizar la ISO/IEC 25010 la cual proporciona un conjunto de normas y guías que abordan diferentes aspectos de la calidad del software, incluyendo la definición de requisitos de calidad, la evaluación y medición de la calidad, y la gestión de la calidad a lo largo del ciclo de vida.

Durante el proyecto nos dimos cuenta que encontrar un formato específico el cual seguir y poder decir que se cumple la calidad del proyecto es complicado, para lo cual vimos otros proyectos que plantearon aseguramiento de calidad como en [10] “Educación Superior en el Perú: Retos para el Aseguramiento de la Calidad”, dándonos como idea que es posible usar cualquier formato siempre y cuando se muestre la información necesaria estructurada. La calidad de un software puede llegar a ser complicado, sin embargo la mayoría de actividades se basan en documentación que en fases más adelante del proyecto resulta ser muy útil, sin embargo a nuestro alcance solo pudimos desarrollar un bloque de la ISO 25010 la cual es adecuación funcional que corresponde:

1. Completitud Funcional:

a) Documentación de requisitos:

Presentando una evidencia de todos los requisitos del proyecto en la tabla 02 mostradas:

TABLA 03. EDUCACIONES DEL PROYECTO

Código	Descripción
EDU-001	Gestión de autenticación
EDU-002	Gestión de usuarios
EDU-003	Gestión de estándares
EDU-004	Gestión de Planes de mejora
EDU-005	Gestión de evidencias
EDU-006	Gestión de cabeceras
EDU-007	Gestión de narrativas

b) Matriz de Trazabilidad:

Desarrollamos una matriz la cual muestra todos los requisitos funcionales implementados, no implementados en el software con comentarios, preguntas de ser necesario.

c) Registro de Tareas y Objetivos: (Tablero, Calendario)

El calendario de desarrollo se realizó tomando en cuenta todo lo que conlleva el semestre A y B.

2. Corrección Funcional:

a) Resultados de Pruebas:

Las pruebas de software, que incluyeron pruebas unitarias, de integración y de sistema, así como la revisión del código fuente, indican un nivel de corrección funcional del 75%. Se observó que la mayoría de las funciones operan correctamente, pero existen áreas donde se pueden realizar mejoras para aumentar la precisión de los resultados.

Se recomienda la implementación de pruebas adicionales, especialmente en escenarios de borde y condiciones excepcionales, para abordar posibles situaciones no contempladas. Los resultados de nuestras pruebas realizadas son los siguientes, manteniendo todos la estructura de la Figura 20.

- Pruebas de errores de código 59/59 aprobadas
- Prueba de verbos en HTTP 160 /160 aprobadas

- Pruebas funcionales manuales 86/90 aprobadas

3. Pertinencia Funcional:

a) Encuestas del Cliente:

Manejamos todas las encuestas con el cliente por sprint, recaudando información como los pasos de reunión y los acuerdos nuevos siendo firmado por el líder de proyecto y entrevistado para validar todo.

b) Registro de Retroalimentación:

Al mostrar las funcionalidades desarrolladas siempre hay modificaciones o mejoras recomendadas esto las registramos y en caso de haber un cambio drástico se maneja una solicitud de cambio luego otro documento para analizar y aprobar este cambio.

V. RESULTADOS

Se logró en su totalidad implementar el software respecto a funcionalidades validadas al 100% considerando funcional el proyecto a un 91%, quedando detalles por desarrollar. El sistema diseñado e implementado permite la creación, almacenamiento y seguimiento eficiente de los documentos relacionados con los estándares de calidad para el proceso de acreditación.

Las pruebas realizadas sobre el software nos revelan que la confiabilidad del sistema es del 97.2% , siendo el restante 2.8% defectos los cuales se van a corregir en una fase de mantenimiento posterior.

Se establece un flujo de trabajo claro y estandarizado para la generación, revisión y seguimiento de las evidencias para cada estándar por parte de los docentes y gestión de los parámetros de evaluación.

Todos los documentos y procesos implementados cumplen con los estándares y requisitos establecidos por SINEACE para la acreditación.

Se implementó medidas de seguridad sólidas para proteger la confidencialidad, integridad y disponibilidad de los documentos relacionados con el proceso de acreditación. Se establecen controles de acceso adecuados y autenticación segura. Esto garantiza la protección de la información sensible y reduce los riesgos de filtración, pérdida o manipulación de los documentos.

Se desarrollaron una variedad de manuales para poder mantener un correcto funcionamiento del proyecto como:

- Manuales del sistema

- Manual de despliegue del sistema
- Manual de Gitlab
- Manual de horas individuales
- Manual de Jira
- Manual de Slack
- Manual de usuario
- Manual de estándar de commits.

Se realizaron capacitaciones a los usuarios para el correcto uso del software, así mismo se entregaron los manuales correspondientes, se cerró el proyecto con la firma del acta de cierre.

VII. DISCUSIÓN

Luego de analizar y comparar respectivamente los resultados en otros estudios relacionados, se llegó a las siguientes conclusiones, empezando con el sistema BPM desarrollado para que funcione en la Universidad Técnica de Ambato (UTA) mostrado en [8], indican en sus resultados una mejora significativa en la gestión de procesos académicos y administrativos, aunque no menciona cifras específicas, aclara que existe una mejora eficiente y mayor productividad, que es lo que esperamos al notar resultados en otras organizaciones respecto a la gestión de procesos, archivos, informes, entre otros.

De igual manera en [7], al realizar un análisis de los procesos que conforman la gestión académica y administrativa en las escuelas, se llegó a dar como resultado, que es necesario viabilizar los procesos de la gestión académica teniendo un desarrollo idóneo en los procesos y un logro de calidad, que recalca como beneficio a nuestro proyecto, la necesidad de implementar un sistema de gestión administrativo.

Para finalizar, la discusión con el último artículo en [9], el cual es el más cercano respecto a desarrollo con nuestro proyecto, propone un modelo de información que permite recopilar y gestionar los datos relevantes para la Universidad Privada Antenor Orrego, y así evidenciar el nivel de cumplimiento de cada estándar propuesto por el SINEACE; nos evidencia el valor y que su modelo de información se adapta a las necesidades específicas de la UPAO y los estándares de acreditación del Sineace, siendo este un caso de éxito evidenciado para esperar grandes resultados respecto a la acreditación con nuestro sistema implementado.

VII. CONCLUSIONES

Este proyecto presenta una solución integral y sólida para la gestión de procesos de calidad en la Facultad de RRII. La implementación de un sistema de gestión documental, el establecimiento de un flujo de trabajo eficiente y el cumplimiento de estándares de acreditación demuestran ser pilares fundamentales para agilizar el proceso de acreditación por parte de SINEACE.

El proyecto ha logrado casi en su totalidad implementar el software, validando el 87% de las funcionalidades con un 91% de consideración funcional del proyecto, dejando solo detalles por desarrollar. El sistema diseñado e implementado permite una gestión eficiente de los documentos relacionados con los estándares de calidad, estableciendo un marco de trabajo claro y estandarizado que facilita el proceso de acreditación y mejora la gestión educativa en la facultad.

La elección de la metodología Scrum se justifica por su capacidad para adaptarse a cambios frecuentes y la necesidad de presentar prototipos con regularidad, lo que refuerza la viabilidad y efectividad del proyecto en entornos dinámicos.

Los cursos CAPSTONE permiten desafiar a los estudiantes a situaciones reales donde los propios protagonistas son ellos mismos y la asesoría y acompañamiento juega un papel muy importante para el éxito de los proyectos.

VIII. RECOMENDACIONES

Consideramos que en el principio del proyecto es muy importante estructurar y jerarquizar todo lo que respecta a documentación y burocracia. Documentos como los de solicitudes de cambio, de reunión, manuales, estructura, definir la metodología más adecuada para el proyecto ya que cada proyecto es distinto, con distintas necesidades.

Y lo más importante poder organizar y lograr que el equipo de trabajo lo haga conjuntamente involucrando a cada participante con el proyecto, ya que al no haber muchos integrantes y tener un proyecto casi en su totalidad real con un cliente, cada rol es de suma importancia, ya que en caso de ausentarse uno este puede afectar el proyecto.

Se recomienda continuar desarrollando manuales y documentación que ayuden a mantener un correcto funcionamiento del proyecto, así como a facilitar la integración de nuevos miembros al equipo y la transferencia de conocimientos. Estos manuales podrían abarcar áreas como el sistema, el despliegue, herramientas de colaboración y estándares de desarrollo.

Además, se debe investigar formas de hacer que el sistema sea más escalable y adaptable a medida que crezcan

las necesidades y los requisitos de la institución. Esto puede implicar la incorporación de nuevas funcionalidades, la optimización del rendimiento y la implementación de procesos de mejora continua que permitan al sistema evolucionar con el tiempo.

Se logró realizar el despliegue del proyecto en el servidor utilizando Docker como contenedor, para poder subir el proyecto sin afectar el entorno.

IX. TRABAJOS FUTUROS

Por el momento se desarrolló el software para asegurar la calidad de una escuela al validar la eficacia y utilidad para la escuela y que logre la acreditación lo que esperamos es que otras escuelas también lo realicen y en que se de apoyo para toda nuestra universidad. Para esto es necesario explorar formas de hacer que el sistema de gestión documental sea aún más adaptable a entornos cambiantes y dinámicos, permitiendo una flexibilidad óptima para integrar nuevos requisitos y estándares de acreditación.

También se plantea la posibilidad de integrar el sistema de gestión de calidad con otras áreas de la institución, como la gestión académica o administrativa, para lograr una sinergia efectiva y una visión completa de la gestión institucional

AGRADECIMIENTOS

Se realiza un reconocimiento especial al equipo de desarrollo integrado por estudiantes y liderado por docentes.

Agradecimiento especial a los directores de las escuelas de Ingeniería de Sistemas y de relaciones industriales por permitir y brindar todas las facilidades para el desarrollo de esta experiencia.

REFERENCIAS

- [1] Escuela profesional de Ingeniería de Sistemas de la Universidad Nacional de San Agustín de Arequipa url: <https://fips.unsa.edu.pe/ingenieriadesistemas/>
- [2] Accreditation Board for Engineering and Technology ABET url: <https://www.abet.org/>
- [3] Gonzales-Saji, Freddy, et al. "Experience in the development of the CAPSTONE course under ABET guidelines at the Professional School of Systems Engineering." Proceedings of the LACCEI International Multi-Conference for Engineering, Education and Technology. doi. Vol. 10. 2020.
- [4] Cadavid, A. N., Martínez, J. D. F., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 11(2), 30-39.
- [5] Volz, R., Oberle, D., Staab, S. y Motik, B. (mayo de 2003). KAON SERVER-Un sistema de gestión de web semántica. En WWW (Pistas de papel alternativas) .
- [6] Maida, E. G., & Pacienza, J. (2015). "Metodologías de desarrollo de software."
- [7] Flores, A., Lavín, J. M., Calle, X., y Álvarez, E. (2014). "Buscando la excelencia educativa: Gestión de procesos académicos y administrativos en Instituciones Públicas de Educación mediante BPM." *Maskana*, 5.

- [8] Silva, R., Cruz, E., Méndez, I., & Hernández, J. Á. (2013). "Sistema de Gestión Digital para mejorar los procesos administrativos de Instituciones de Educación Superior: Caso de estudio en la Universidad Autónoma Metropolitana." *Perspectiva Educativa, Formación de Profesores*, 52(2), 104-134.
- [9] Castillo Robles, E. F. (2021). "Modelo de información para el monitoreo del cumplimiento de estándares de acreditación del Sineace en los programas de estudios de la UPAO."
- [10] Yamada, G., Castro, J. F., & Rivera, M. (2012). Educación superior en el Perú: Retos para el aseguramiento de la calidad.
- [11] Cadavid, A. N., Martínez, J. D. F., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 11(2), 30-39.
- [12] Trigás Gallego, M. (2012). Metodología scrum.
- [13] Lendínez, L. C. (2019). Kanban. Metodología para aumentar la eficiencia de los procesos. 3c *Tecnología: glosas de innovación aplicadas a la pyme*, 8(1), 30-41.
- [14] J. Dick, E. Hull y K. Jackson, "Requirements Engineering", 2017. doi:10.1007/978-3-319-61073-3
- [15] I. Sommerville, "Software engineering," 9th ed., Boston, Addison-Wesley, 2011.
- [16] R. A. Vera, M. M. Mata, J. D. Mendoza y J. U. Pech, "Explorando la influencia de los roles de Belbin en la especificación de requisitos de software," *Revista Ibérica de Sistemas e Tecnologías de Informação*, no. 36, pp. 34-49, 2020.