

Desarrollo de una plataforma web para la gestión de proyectos de grado para la Universidad Santiago de Cali

Development of a web platform for the management of degree projects for the Engineering Faculty of the Universidad Santiago de Cali

Andres Albarracín Arroyave¹
andres.albarracin00@usc.edu.co

Alejandro Peña Soto¹
alejandro.pena00@usc.edu.co

Diego Fernando Loaiza Buitrago, M.S1¹
diego.loaiza02@usc.edu.co

Universidad Santiago de Cali, Facultad de Ingeniería, Programa de Ingeniería de Sistemas (1)

Resumen

El presente trabajo tiene como objetivo desarrollar una plataforma web para la gestión de proyectos de grado teniendo en cuenta la necesidad identificada de la Facultad de Ingeniería de la Universidad Santiago de Cali, al no tener un control ni gestión de los trabajos de grado desarrollados dentro del programa en tanto a agilidad del proceso, evaluación, sistematización y automatización de los mismos; por tanto, se ejecuta un análisis del proceso y estado actual de la universidad, obteniendo como resultados el diseño de un software que, recurriendo a artefactos de las metodologías ágiles de desarrollo “Scrum”, permita ir de manera directa al cumplimiento de los objetivos. Para ello, se exponen inicialmente cuadros comparativos en los que son visibles las características principales de cada uno de los modelos/herramientas y el por qué se elige el software con el que se desarrolla la investigación. Además se muestra las fases de desarrollo del programa, las ventajas, cualidades, y aportes a los cuales puede acceder la universidad con el uso e implementación de dicha plataforma.

Palabras Clave: Plataforma web; Metodologías ágiles; Scrum; Código abierto; modelo vista controlador; Universidad Santiago de Cali.

Abstract

This Project aims to develop a web platform to management of degree projects taking into account the identified need of the Faculty of Engineering of the University of Santiago de Cali, as it does not have a control or management of the degree workers developed within the program in terms of process agility, evaluation, systematization and automation; therefore, an analysis of the process and current state of the university is executed, obtaining as a result the need to develop software using artifacts from the agile methodology of development "Scrum" that allows to go directly to the fulfillment of the objectives. To do this, comparative tables are initially exposed in which the main characteristics of each of the models / tools are visible and why the software with which the research is carried out is chosen. It also shows the phases of development executed in the program, the advantages, qualities, contributions and developments that the university can access with the use and implementation of the platform.

Keywords: web platform; agile methodology; Scrum; open source; controller view model; Universidad Santiago de Cali

1. INTRODUCCIÓN

El trabajo de grado es uno de los principales requisitos para que los estudiantes puedan obtener la titulación profesional de la universidad, dicha situación se ha tornado un poco “compleja” de desarrollar en tanto a su control, estado de cada proyecto, tiempo y demás, pues actualmente, los estudiantes realizan los procesos de inscripción, radicación de anteproyectos e informe final mediante formularios de Google, los cuales generan documentos de Excel que hace que los directores de programa realicen acciones de manera manual sin tener un control preciso y detallado de los mismos generando cargas operacionales.

Por esto se desarrolló el Sistema de Gestión de Proyectos de Grado para la Facultad de Ingeniería (SGPG – USC), el cual tiene como principal objetivo el desarrollo de un software que permite ver en línea el estado en que se encuentra el proyecto,

mitigando la falta de comunicación entre los involucrados (estudiantes, director del proyecto, director del programa y jurados), llevando un registro de los tiempos de entrega y corrección, incluso generando reportes; es decir que el estudiante tendrá siempre actualizado el estado de su proyecto y sabrá si ya fue aceptado o por el contrario requiere correcciones según las anotaciones que realicen los evaluadores.

Así mismo, los directores de proyecto y jurados se beneficiarán al tener en una sola plataforma todos los proyectos asignados, los cuales podrán manipular en línea llevando así un récord histórico de cada proyecto con sus respectivas observaciones, un registro detallado por estudiante de los trabajos presentados y el estado en el que se encuentran cada uno. Además de que tendrán la posibilidad de generar técnicas de autoevaluación y gestión en tanto a crear indicadores como por ejemplo: Cantidad de trabajos de grado durante el semestre, cantidad de trabajos de grados asignados a cada docente, tiempos de respuestas de evaluación, tiempos de presentación de cada proyecto por parte del estudiante, entre otros.

Es así como el presente documento inicia con la observación y análisis tanto de proyectos nacionales como internacionales que han abordado situaciones similares en diferentes universidades. A continuación se describen en la Tabla 1. un resumen comparativo de cada uno de los proyectos consultados:

Tabla 1 Antecedentes y gestión de proyectos.

Titulo	Problema	Descripción
Sistema de información para la gestión de los trabajos de grado (Solís Poveda & Deavila Pertúz, 2013)	Manejo y control de los trabajos de grado.	El proceso de la gestión de proyectos de grado se realiza de forma manual conllevando al problema en la toma de decisiones, pérdida de información sin trazabilidad en tiempos de entrega y evaluación de trabajos.
Sistema de información y gestión de proyectos de grado (Sarmiento Forero & Quiros Traslaviña, 2013)	Dificultad en administración y control de trabajos de grado.	Existe un coordinador de trabajos de grado con un equipo específico, el cual mediante Excel llevan la información de todos los proyectos sin existencia de reportes.
Desarrollo de un sistema de información para la gestión de los proyectos de responsabilidad social del programa de ingeniería de sistemas de la Universidad Católica de Colombia (Pascagaza Gutierrez, 2018)	Inconveniente en consulta y almacenamiento de los proyectos.	Los trabajos de grado se entregan de forma física en cd y su registro se hace mediante Excel.
SIAP Sistema de información para administración de proyectos de grado (Chaparro López & Forero Sarmiento, 2005)	Conflicto en el proceso de control y seguimiento debido al volumen de proyectos.	El proceso de control y seguimiento se realiza de manera manual haciendo registro en Excel desde el computo de del coordinador de proyectos de grado, tornándose cada vez es más complejo por la cantidad de trabajo que se maneja.
Sistema de información para la gestión de proyectos para la fundación universitaria los libertadores SIGESPRO (Blanco Blanco & Hernández Zamudio, 2016)	Estancamiento en procesos de proyectos de grado y su consulta.	No se tiene la consolidación de los proyectos, ni acceso fácil a estos para la evaluación y análisis provocando pérdida de interés o desmotivación por parte de alumnos al no tener conocimiento del proceso de su proyecto.
Implementación del sistema de gestión de proyectos de investigación (G project) (Mamaui Tola, 2013)	Dificultad en la gestión de proyectos de investigación	Por gran volumen de proyectos de investigación que manejan requieren poder gestionarlos mediante un sistema.

Fuente: Elaboración propia

Se continua con el establecimiento del planteamiento del problema donde se expone explícitamente la necesidad que tiene la Institución en desarrollar un sistema que pueda gestionar los trabajos de grado de la facultad; además se plantea la pregunta de investigación: **¿Cómo desarrollar una plataforma web para la gestión de los proyectos de grado para la universidad Santiago de Cali?**

Después se definen los casos de uso, lenguaje de programación y motor de base de datos a emplear, el cual comienza a darle forma al sistema ya que se empiezan a realizar las configuraciones iniciales del proyecto donde se plasma los objetivos a desarrollar como columna vertebral de la investigación. Una vez planteado esto, se produce el desarrollo del software, donde se incluye uno a uno las vistas necesarias para implementar cada uno de los requerimientos solicitados mediante el uso de artefactos de la metodología ágil de desarrollo SCRUM aplicando la arquitectura de modelo vista controlador.

Finalmente, se realizan las pruebas necesarias con el fin de identificar y corregir problemas que puedan impedir el buen uso del software, se verifica el alcance del programa, se generan conclusiones y se reitera la necesidad de generar avances tecnológicos con la implementación de dicha aplicación dentro de los procesos actualmente desarrollados para los trabajos de grado en la facultad de ingeniería.

2. MÉTODOS/METODOLOGÍA

Se debe mencionar que existen diferentes metodologías por las cuales se pudieran desarrollar este software como por ejemplo las tradicionales: RUP (Rational Unified Procces), MSF (Microsoft Solution Framework), Win-Win Spiral Model, Iconix y las ágiles Programming [XP], Crystal, Dynamic System Development Method DSDM. (Navarro Cadavid, Fernandez Martínez, & Morales Vélez, 2013). Sin embargo, para el desarrollo de este programa se optó por emplear artefactos de SCRUM, debido a que es una metodología, qué como su mismo nombre lo indica, es ágil en los procesos y facilita su uso.

(Fowler, 2005). *Los métodos ágiles son más adaptativos que predictivos*. Los métodos de ingeniería tratan de planificar gran parte del proceso del software con gran detalle durante un largo período, esto funciona bien hasta que las cosas cambien. Entonces su naturaleza es resistir el cambio. Los métodos ágiles, sin embargo, Intentan ser procesos que se adapten y prosperen en el cambio, incluso hasta el punto de cambiarse a sí mismos.

A continuación se describen algunas de sus características en comparación con las metodologías tradicionales:

Tabla 2 Metodologías Tradicionales VS Metodologías Ágiles

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Fuente: (Canós, Letelier, & Penadés, 2012)

Una de las ventajas de la metodología Scrum es que “permite dentro de este, usar varios procesos y técnicas las cuales las personas pueden afrontar problemas complejos y entregar productos de máximo valor posible” (Schwaber & Sutherland, 2017).

En cuanto a lenguajes de programación se optó por PHP ya que es un lenguaje de Open source (código abierto o software libre) apto para el desarrollo web, donde el código se ejecuta en el servidor (PHP, s.f.), a su vez se toma JavaScript debido a que permite ejecutar al usuario (Cliente) diferentes eventos, situación contraria al PHP (Condor Tinoco & Soria Solis, 2014, pág. 55).

Por otro lado referente a la Arquitectura del software se encuentran diferentes patrones a elegir entre las más significativas están cliente servidor, patrón de capas, Modelo vista controlador y demás, esta ultima el MVC (modelo vista controlador) fracciona los datos de una aplicación en el modelo, las vistas y los controladores permitiendo la actualización y mantenimiento del software de manera sencilla y en poco tiempo (Fernandez Romero & Díaz González, 2012).

El Framework de desarrollo que más se adapta a la investigación realizada es Laravel ya que es un framework para PHP

basado en la arquitectura MVC, existen otros framework con cualidades parecidas pero cada uno con sus particularidades, en la Tabla 3. se puede observar una comparación entre cada uno de ellos con sus cualidades:

Tabla 3 Cuadro Comparativo de Características y fabricantes.

Framework	PHP 4	PHP 5	MVC	Múltiple	ORM	Plantillas	Cache	Ajax	Autenticación	Módulos
KumbiaPHP	-	*	*	*	*	*	-	*	*	*
Wasp	-	*	*	-	-	*	-	*	-	*
CodeIgniter	*	*	*	*	-	*	*	-	-	-
Seagull	*	*	*	*	*	*	*	*	*	*
BlueShoes	*	*	*	*	-	*	-	-	-	*
Qcodo	-	*	*	*	-	*	-	*	-	*
Akelos	*	*	*	*	*	*	*	*	-	*
PhpOpenbiz	-	*	*	*	*	*	-	*	-	-
Zoop	*	*	*	*	-	*	*	*	-	-
Ash.MVC	-	*	*	-	-	*	-	-	-	*
Diy	-	*	*	-	*	*	*	*	-	*
Wact	*	*	*	*	-	*	-	-	-	*
Zend Framework	-	*	*	*	-	*	-	*	*	*
Yii	*	*	*	-	*	*	-	*	*	*
Laravel	-	*	*	*	*	*	*	*	*	*
CakePHP	*	*	*	*	-	*	*	*	*	*
Symfony	-	*	*	*	-	*	-	*	*	*

Fuente: (Sierra, Acosta, Ariza, & Salas, 2013)

En la Base de datos se trabajó con MYSQL ya que es un gestor con licencia GPL de distribución gratuita y posee una integración óptima con PHP y además es multiplataforma (Guzmán Reyes, 2006).

Para finalizar se contó con un controlador de versiones VCS y repositorio del proyecto Git Lab y bitbucket el cual permite llevar el control de los cambios realizados sobre los conjuntos de archivos a modo de seguro en caso de existir la necesidad de restauración de uno de estos y permite realizar una copia externa del código mediante una instancia remota del VCS.

De este último existen 3 tipos, locales, centralizados y distribuidos (Chacon, s.f.), a continuación, en la Tabla 4. se muestra un comparativo entre las más usadas en VCS.

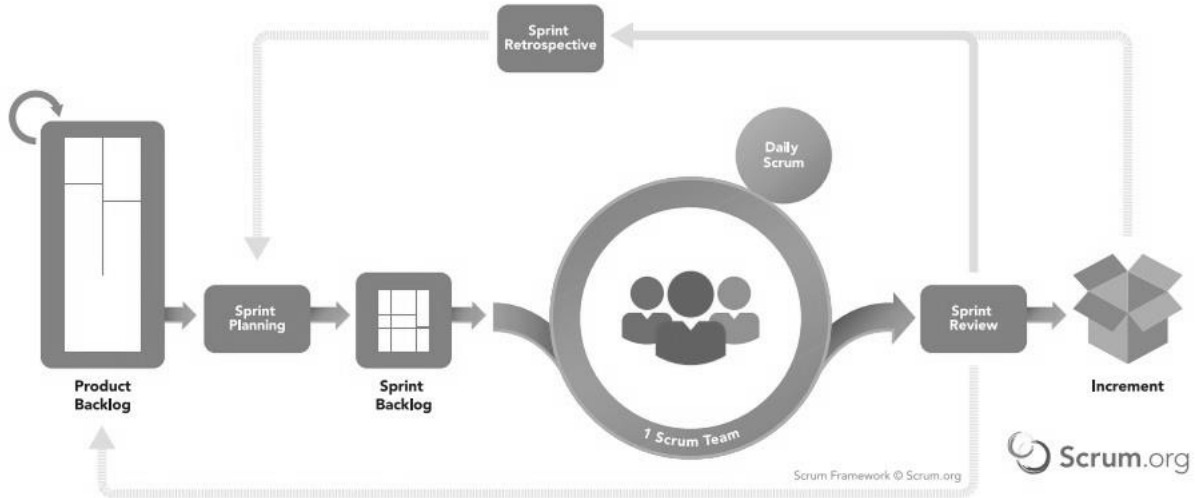
Tabla 4 Comparativo VCS populares

Software	Arquitectura de red	Resolución de conflictos	Estado de desarrollo	Compatibilidad con Bitbucket
Git	Distribuida	Fusión	Activo	Sí
Mercurial	Distribuida	Fusión	Activo	Sí
SVN	Cliente-servidor	Fusión o bloqueo	Activo	No
CVS	Cliente-servidor	Fusión	Solo mantenimiento	No

Fuente: (Organizacion, s.f.)

Ahora bien, todos los procesos para desarrollo Scrum se pueden observar en la Figura 1. Se inicia con el listado del producto (*Product Backlog*) los cuales son representados en historias de usuario suministradas por el dueño del producto (*Product Owner*) quien hizo los requerimientos, vía entrevista de que debe soportar la plataforma (Vargas Vasquez, 2018), Con el fin de que el programa se base en dichos requerimientos se realizó un listado de funcionalidades que contiene la plataforma.

Figura 1. Diagrama Proceso de Scrum



Fuente: (Schwabe & Sutherland, 2016)

Esta se estructura mediante *Sprints Backlog* con periodos cortos o largos donde el objetivo es tomar estas funcionalidades que se puede observar en la Tabla 5, e ir desarrollando el software en forma de entregable (Heys, 2011) dividiendo en fases “semanas” donde se comenzó realizando el diagrama de Casos de Uso (ver Figura 2. Diagrama Casos de Uso) y del Modelo Relacional de la BD (ver Figura 3. Diagrama Modelo Relacional de la Base de Datos), continuando con la autenticación de usuarios, definiendo los tipos que comprenden a estudiantes, tutor, jurados y directores con sus respectivos perfiles y permisos.

Seguidamente se trabajó en el desarrollo de los módulos de la aplicación (módulo de registro de anteproyecto, módulo de histórico estatus proyecto, carga del proyecto, calificación del proyecto) donde cada uno de estos se denominó como un *Sprint*.

Para la creación de esta plataforma no se tuvieron reuniones diarias, fueron reuniones esporádicas donde se presentaron los avances y así mismo se verificaron las funcionalidades (*Sprint Review*) si se estaban cumpliendo con lo solicitado, quedando pendiente el *Deployment* que consiste en pasar a producción el producto.

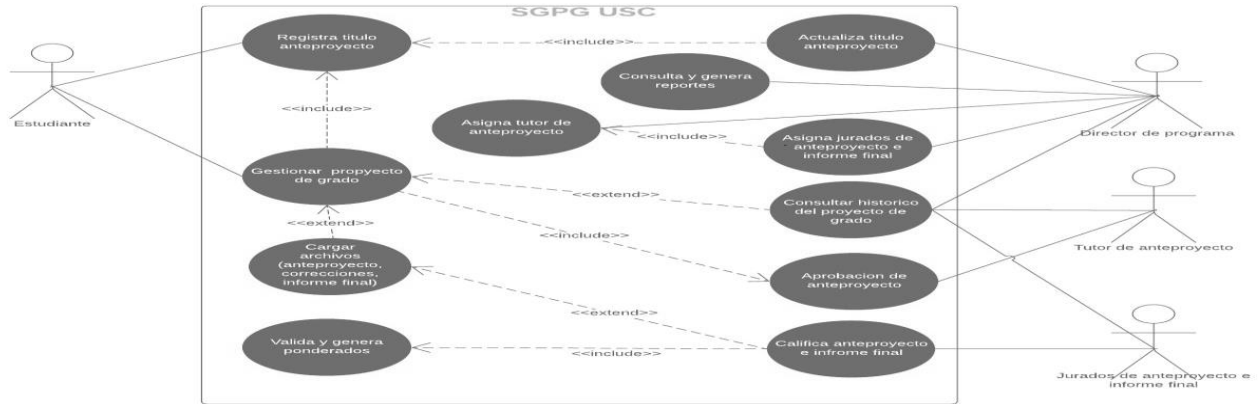
Tabla 5 Funcionalidades.

Funcionalidades	
1.	El sistema debe permitir que en la pantalla de inicio de sesión, realice la validación del usuario registrado con sus respectivas credenciales.
2.	El sistema debe restringir a usuarios el acceso a módulos que no tiene permisos.
3.	El sistema debe cargar las opciones según el perfil del usuario que ha iniciado sesión.
4.	El sistema debe ser parametrizable para los periodos académicos que defina el director.
5.	El sistema debe permitir que el estudiante inscriba idea, anteproyecto, trabajo final.
6.	El sistema debe controlar que el director de proyecto avale la información del trabajo a entregar.
7.	El sistema debe permitir que en la vista del jurado puedan aprobar, no aprobar o correcciones.
8.	El sistema debe permitir que en la vista del director de programa, asignar tutor y jurados de anteproyecto o de informe final.
9.	El sistema debe controlar que el estudiante solo pueda corregir el proyecto una sola vez en cada etapa.
10.	El sistema debe contener un módulo de reportes.
11.	El módulo de reportes debe tener los procesos que superen el tiempo de entrega o revisión.
12.	El módulo de reporte debe mostrar qué jurados están atrasados o con procesos pendientes.
13.	El módulo de reporte debe tener qué estudiantes no continuaron el proceso y determinar los desertores para actualizar los estatus en el sistema.
14.	El sistema permitir que el estudiante siempre pueda consultar en que procesos se encuentra su anteproyecto.
15.	El sistema debe permitir observar siempre los estatus de los proyectos.
16.	El sistema debe permitir que todas las fechas del sistema puedan ser parametrizables.
17.	El sistema debe realizar la calificación o ponderación según el formato entregado por el producto owner.

18. El sistema debe mostrar a un docente sus proyectos asignados.
19. El sistema debe permitir en la vista jurado agregar las observaciones pertinentes, según la fase actual en la que se encuentre el proyecto, llevando un histórico de las mismas.
20. El sistema debe permitir el cargue de información por medio de archivos externo.
21. El sistema debe permitir la consulta de los proyectos ya finalizados a modo informativo.

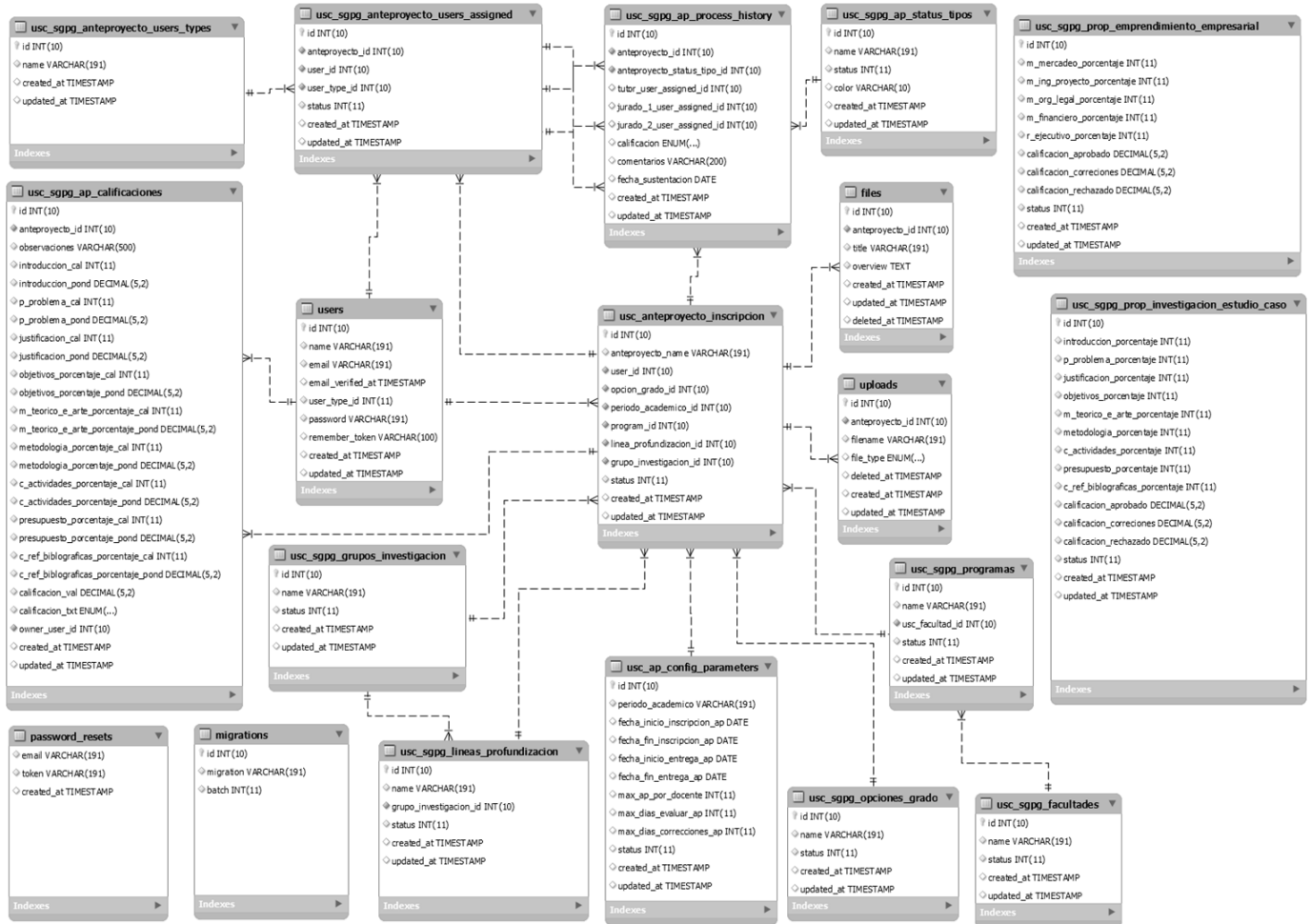
Fuente: Elaboración propia

Figura 2. Diagrama Casos de Uso.



Fuente: Elaboración propia

Figura 3 Modelo Relacional de la Base de Datos.



Fuente: Elaboración propia

3. RESULTADOS Y DISCUSIÓN

Uno de los principales resultados obtenidos con esta investigación es el desarrollo mismo de la plataforma web para la gestión de proyectos de grado de la facultad de ingeniería de la Universidad Santiago de Cali, consiguiendo desarrollar cada uno de los objetivos propuestos mediante las actividades realizadas en la metodología establecida; a continuación se grafica algunas de las pruebas realizadas en el desarrollo del programa tratando de que fuera un ejercicio las más cercano a la realidad.

- Validación de usuario al inicio de sesión, el cual se verifica con la base de datos creada permitiendo ingresar a la vista que corresponda, sea estudiante, tutor, jurado o director de programa.

Figura 4 Login Usuarios.



Fuente: Elaboración propia

- Permitir inscribir Anteproyecto

Figura 5 Inscripción de Anteproyecto.

Fuente: Elaboración propia

- Permitir observar estados de inscripción (vista estudiante)

Figura 6 Vista General Estudiante.

Anteproyecto	Estudiante	Opcion grado	Periodo	Programa	Estado	Acciones
Sistema integral de seguridad en la Universidad, basado en tarjetas identificador de identidad.	Alejandro Peña Soto	Proyecto de investigacion aplicada	2019B	Sistemas	Título inscrito	

Fuente: Elaboración propia

- Permitir observar vista histórica

Figura 7 Vista Histórica de un Proyecto.

Fecha	Estado	Tutor	Jurado # 1	Jurado # 2	Comentarios	Calificación	Sustentación
2019-09-02 06:12:58	Título inscrito						

Fuente: Elaboración propia

- Permitir observar todos los trabajos inscritos, asignación tutor y jurados, además del histórico de cada proyecto (vista director de programa).

Figura 8 Vista General Director de Programa.



Fuente: Elaboración propia

- Permitir asignación de tutores (Vista director de programa)

Figura 9 Asignar Tutor.



Fuente: Elaboración propia

- Evidenciar cambios sobre los proyectos inscritos en cuanto a su estado (Vista director de programa).

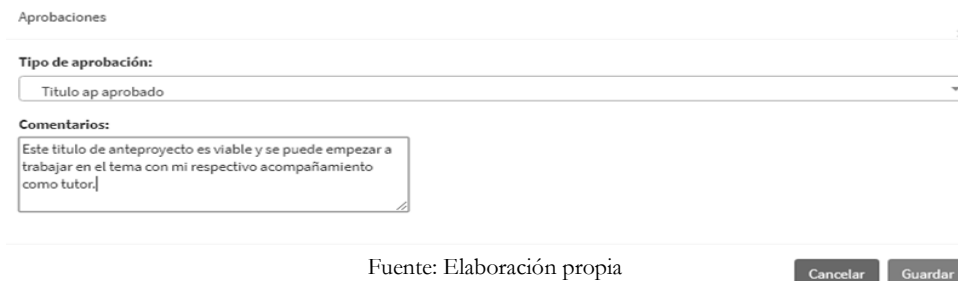
Figura 10 Estado de Anteproyectos.



Fuente: Elaboración propia

- Permitir realizar comentarios al proyecto inscrito (Vista tutor) en otras.

Figura 21 Aprobación de Proyecto.



Fuente: Elaboración propia

- Permite ver histórico con comentarios realizados

Figura 32 Históricos con Comentarios.

Historico anteproyecto

Show 10 entries

Search:

Fecha	Estado	Tutor	Jurado # 1	Jurado # 2	Comentarios	Calificación	Sustentación
2019-09-02 08:53:31	Titulo aprobado	Tutor			Este titulo de anteproyecto es viable y se puede empezar a trabajar en el tema con mi respectivo acompañamiento como tutor.		
2019-09-02 08:02:40	Tutor asignado	Tutor					
2019-09-02 06:12:58	Titulo inscrito						

Showing 1 to 3 of 3 entries

Previous 1 Next

Cerrar

Fuente: Elaboración propia

Además, se logró identificar que con la implementación de esta plataforma en la facultad se espera:

1. Agilizar tiempos para docentes involucrados en los proyectos.
2. Poder medir cantidad de proyectos que fueron asignados, evaluados, finalizados y pendientes durante cada semestre.
3. Se puede observar la carga académica asignada a cada docente.
4. Disminuye carga operacional ya que permite que director del programa tenga un control de los proyectos asignados a su planta de docente.
5. La posibilidad de que los directores del programa generen indicadores de gestión y calidad que mida la eficiencia y eficacia de los procesos administrativos de trabajos de grados desarrollados dentro de la facultad.

4. CONCLUSIONES

Se considera que la implementación de la plataforma web para la gestión de los proyectos de grado mejoraría el proceso que se lleva a cabo en la actualidad, agilizándolo y evitando el desconocimiento por parte de los estudiantes en tanto al estado de entrega de su proyecto, evitando también incurrir en el tiempo de entrega por parte de la universidad y el mismo estudiante. Además de generar un seguimiento de control constante sobre cada uno de los proyectos impulsando un proceso de autoevaluación, eficiencia, rendimiento, calidad y desarrollo por las partes involucradas, de tal modo que permite el crecimiento de la facultad de ingeniería pasando de procesos manuales a procesos actualizados, los cuales permiten medir el servicio prestado no solamente a nivel educativo si no en administrativo.

Se recomienda saber y escoger los tipos de herramientas a usar antes de iniciar con el desarrollo del proyecto ya que son piezas fundamentales a la hora de obtener resultados, las cuales podrán facilitar un poco la realización de este obteniendo beneficios reflejados en la duración del proyecto. También se sugiere a futuro desarrollar nuevos reportes que permitan sacar el mayor provecho posible de esta plataforma ya que el proyecto tiene implementado un módulo de reportes flexible que permite construir cualquier reporte a partir de código SQL.

REFERENCIAS

Blanco Blanco, P. A., & Hernández Zamudio, M. (2016). *Los libertadores Fundación Universitaria*. Retrieved from <https://repository.libertadores.edu.co/handle/11371/742>

- Canós, J. H., Letelier, P., & Penadés, M. C. (2012, 03 13). *roa.ult.edu.cu*. Retrieved from <http://roa.ult.edu.cu/handle/123456789/476>
- Chacon, S. (n.d.). *Git-scm*. Retrieved from <https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones>
- Chaparro López, G. A., & Forero Sarmiento, L. A. (2005). *Universidad Javeriana*. Retrieved from <https://repository.javeriana.edu.co/handle/10554/7487>
- Condor Tinoco, E. E., & Soria Solis, I. (2014). *Programacion Web con CSS, JavaScript, PHP y AJAX*. Universidad Nacional José María Arguedas.
- Fernandez Romero, Y., & Díaz González, Y. (2012). Patrón Modelo-Vista-Controlador. *Revista Digital de las Tecnologías de la Información y las Comunicaciones Telemática*, 47-57.
- Fowler, M. (2005, 12 13). *martinfowler.com*. Retrieved from <https://www.martinfowler.com/articles/newMethodology.html>
- Guzmán Reyes, D. (2006, 11). *Bases de datos distribuidas con una solución LAMP (Linux, Apache, MySQL y PHP)*. Retrieved from <http://dgsa.uaeh.edu.mx:8080/bibliotecadigital/handle/231104/359>
- Heys, W. B. (2011, 01 18). *Developer Microsoft*. Retrieved from <https://blogs.msdn.microsoft.com/billheys/2011/01/18/branching-for-scrum/>
- Mamaui Tola, A. R. (2013, 11). *revistasbolivianas.org.bo*. Retrieved from http://www.revistasbolivianas.org.bo/scielo.php?script=sci_arttext&pid=S2305-60102013000200002&lng=es&nrm=iso
- Navarro Cadavid, A., Fernandez Martínez, J. D., & Morales Vélez, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 31.
- Organizacion, b. (n.d.). *bitbucket.org*. Retrieved from <https://bitbucket.org/product/es/version-control-software>
- Pascagaza Gutierrez, J. M. (2018). *Universidad Católica de Colombia*. Retrieved from <https://repository.ucatolica.edu.co/handle/10983/16047>
- PHP, G. (n.d.). *php*. Retrieved from <https://www.php.net/manual/es/intro-what-is.php>
- Sarmiento Forero, J. A., & Quiros Traslaviña, F. (2013, 02 14). *Universidad Libre*. Retrieved from <https://repository.unilibre.edu.co/handle/10901/8875>
- Schwabe, K., & Sutherland, J. (2016, 06). *Scrum Org*. Retrieved from <https://www.scrum.org/resources/scrum-framework-poster>
- Schwaber, K., & Sutherland, J. (2017). *La guía de Scrum: Las reglas del Juego*. Retrieved from <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>
- Sierra, F., Acosta, J., Ariza, J., & Salas, M. (2013, 10 15). Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web. *Investigación y desarrollo en TIC*, 11-12. Retrieved from <http://revistas.unisimon.edu.co/index.php/identific/article/view/2480>
- Solís Poveda, M., & Deavila Pertúz, J. C. (2013, 12 31). <http://www.unicartagena.edu.co>. Retrieved from <http://190.242.62.234:8080/jspui/handle/11227/295>
- Vargas Vasquez, J. E. (2018). Proceso Actual USC - requerimientos. (A. F. Albarracin, & A. Peña, Interviewers)