

Analisis y Desarrollo de los Módulos de Alta de Recibos de Pagos del Sistema de Control de Verificación de Pagos (SCVRP) del TECNM Campus Lerdo.

Rodríguez-Campos, F.E.¹; Parada-Morado,L²; Rodríguez-Chacon R.L.³; García-Jiménez M.D.⁴

Datos de Adscripción:

¹ Francisco Eduardo Rodríguez Campos. Tecnológico Nacional de México/ Instituto Tecnológico Superior de Lerdo. francisco.rc@itslerdo.edu.mx
<https://orcid.org/0000-0001-5276-3681>

² Lilia Parada Morado. Tecnológico Nacional de México/ Instituto Tecnológico Superior de Lerdo. lilia.pm@itslerdo.edu.mx
<https://orcid.org/0000-0003-2169-0533>

³ Rocío Lorena Rodríguez Chacon. Tecnológico Nacional de México/ Instituto Tecnológico Superior de Lerdo. rocio.rc@itslerdo.edu.mx
<https://orcid.org/0009-0001-0842-1585>

⁴ Mayra Daniela García Jiménez. Tecnológico Nacional de México/ Instituto Tecnológico Superior de Lerdo. garcia.mayra.1bv@gmail.com
<https://orcid.org/0009-0008-8087-5435>

Resumen - Este artículo muestra el análisis y desarrollo de los módulos de alta y consulta de recibos de pagos como parte de un sistema para el control de recibos de pagos de los conceptos de inscripciones y materias de alumnos del TECNM / Instituto Tecnológico Superior de Lerdo, debido a que actualmente el proceso se realiza de manera manual por parte del departamento de servicios escolares de la institución, el presente artículo expone la innovación del desarrollo de nuevas tecnologías para la mejora de los procesos administrativos dentro de la institución. El proyecto implementa diversas metodologías de desarrollo de software, como la metodología en cascada y SCRUM, adaptando lo mejor de ambas. En cuanto a tecnologías, el proyecto implementa el lenguaje de programación PHP con el marco de trabajo Laravel para el desarrollo web y MySQL como gestor de bases de datos. Como resultado se obtienen los módulos de carga de recibos de pagos de un sistema que permite a los alumnos subir sus recibos de pagos a un sistema web de manera rápida y sencilla, además de más segura, evitando el extravío de recibos de pago al crear un expediente digital por alumno de los pagos realizados. Como conclusión se logra mejorar la gestión administrativa del control de pagos de los conceptos de inscripción y materias automatizando el proceso administrativo mejorando los tiempos de atención a los estudiantes y generando un gran impacto en la gestión de la información.

Palabras Clave - Aplicación Web, Control de pagos, Gestión administrativa, Laravel, MySQL, Procesos financieros.

Abstract - This article presents the analysis and development of registration and consultation modules for payment receipts, as part of an integrated system aimed at managing student payments for enrollment and course fees at the TECNM / Instituto Tecnológico Superior de Lerdo. Considering that the current procedure is carried out manually by the institution's Student Services Department, this research underscores the relevance of incorporating information technologies to modernize and optimize

administrative processes. The project adopts a hybrid software development methodology, combining the sequential structure of the waterfall model with the flexibility of the SCRUM framework, in order to capitalize on the strengths of both approaches. The implementation is based on PHP, using the Laravel framework for web application development, in conjunction with MySQL as the database management system. The resulting system enables students to upload payment receipts through a secure and user-friendly web platform, thereby generating a digital repository for each student. This feature mitigates the risk of document loss, facilitates record traceability, and strengthens the reliability of institutional data management.

In conclusion, the proposed system constitutes a significant advancement in the administrative management of payments associated with enrollment and academic subjects, by automating procedures, reducing service times, and enhancing the efficiency and quality of institutional information management.

Keywords - Web Application, Payment Control, Administrative Management, Laravel, MySQL, Financial Processes.

I. INTRODUCCIÓN

El área de servicios escolares del TECNM / Instituto Tecnológico Superior de Lerdo ha presentado dificultades en el proceso de la administración y verificación de pagos que realizan los estudiantes, debido al gran número de alumnos que actualmente se tienen inscritos en cada una de las carreras que la institución ofrece. En el contexto actual, el proceso se realiza manualmente, utilizando documentos físicos o herramientas básicas informáticas como hojas de cálculo, esta forma de trabajo presenta problemas de eficiencia y propicia errores en la gestión de la información, que pueden derivar en datos incompletos, duplicados o incorrectos en el momento de realizar un seguimiento de los pagos que se llevan a cabo por los alumnos. La falta de un sistema de control centralizado para la verificación de pagos dificulta el seguimiento ordenado y preciso, lo cual genera demoras en el proceso y complica la identificación de posibles inconsistencias.

Silva et al. (2013) nos hablan en su caso de estudio realizado en la Universidad Autónoma Metropolitana, acerca de la mejora significativa que puede llevar el implementar el uso de las TIC, especialmente en las áreas administrativas de las instituciones educativas donde se automatizan los procesos, lo cual permite el almacenamiento organizado de la información y hace posible realizar búsquedas eficientes y reportes especializados.

Además, las TIC mejoran la gestión del conocimiento, así como el tiempo empleado en cada tarea, elementos que hacen una mejora en el cumplimiento eficiente de los procesos administrativos, como lo menciona Torres y Hernández (2024). En cuanto a este tema Valencia y Almeida (2024) hacen énfasis en la importancia del uso de nuevas tecnologías en el ámbito educativo, ampliando las posibilidades de mejora continua en diferentes áreas.

Como parte de la solución propuesta de mejora de la gestión de los pagos en el área de servicios escolares del TECNM / Instituto Tecnológico Superior de Lerdo, se propone desarrollar un Sistema de Control de Verificación de Pagos (SCVRP) que permita a los alumnos crear un expediente virtual con todos los pagos realizados durante su estancia en la universidad, con lo cual, el departamento de servicios escolares pueda realizar un seguimiento oportuno de los pagos que ha realizado cada uno de los estudiantes.

Las tecnologías elegidas para el desarrollo de este proyecto fueron el lenguaje de programación PHP empleando el marco de trabajo del framework Laravel con el uso de la base de datos MySQL, debido a que estas tecnologías cuentan con licencias de uso de código libre y que no representan un costo para la institución.

El presente documento muestra el proceso del desarrollo del Sistema de Control de Verificación de Pagos (SCVRP) empleando la metodologías ágiles en Cascada y SCRUM.

Las metodologías ágiles se enfocan en la gestión de proyectos de manera incremental, en el cual, el proyecto crece con las nuevas aportaciones que cada miembro del equipo suma al proyecto (Beck et al., 2001).

Por tanto, para el desarrollo de este proyecto se emplearon dos metodologías en conjunto, la metodología en cascada y SCRUM debido a que permiten una flexibilidad en la gestión de los proyectos.

Las metodologías ágiles tienen como punto central el gestionar los avances de manera cíclica en la que los entregables son nuevas versiones mejoradas del proyecto, tomando con base la comunicación continua, el trabajo en equipo y teniendo como prioridad el valor del entregable al cliente (Highsmith, 2010).

Según Royce (1970) la metodología en cascada permite definir etapas secuenciales como requisitos, diseño, implementación, verificación y mantenimiento en donde cada etapa debe terminar para continuar con la siguiente, esto facilita la planificación y documentación del proyecto a realizar.

La metodología ágil SCRUM es empleada en el ámbito del desarrollo de software, cuyo propósito se basa en el incremento de nuevos avances o funcionalidades de un proyecto de manera cíclica en un periodo corto a lo que se le conoce como sprints, dando como resultado un incremento del producto constante, esta metodología tiene énfasis en la inspección continua y adaptación de nuevos requerimientos (Atlassian, s.f.).

II. PARTE TÉCNICA DEL ARTÍCULO

2.1. Marco de Trabajo Laravel

Laravel es un framework que permite desarrollar sistemas web, el cual se basa en el lenguaje PHP, el marco de trabajo Laravel emplea el patrón de arquitectura Modelo-Vista-Controlador (MVC) lo que hace posible separar la lógica del sistema, la interfaz de usuario y el acceso a datos (Otwell, 2023).

Algo de lo que caracteriza este framework es el control de las rutas en un sistema de manera sencilla, un motor de plantillas BLADE, la gestión de la base de datos a partir de migraciones y el uso de Eloquent ORM que gestiona el acceso a la base de datos, además de integrar la consola de línea de comandos Artisan que permite gestionar las operaciones más comunes en el framework (Haque, 2019).

2.2. Base de datos MYSQL

MySQL es un gestor de base de datos relacional de código abierto el cual emplea el lenguaje SQL (Structured Query Language) para realizar la administración de los datos, gracias a sus características de fiabilidad y facilidad de uso, es una de las bases de datos más usadas en el desarrollo de sistemas web (DuBois, 2020).

3.1 Análisis Técnico del Proyecto

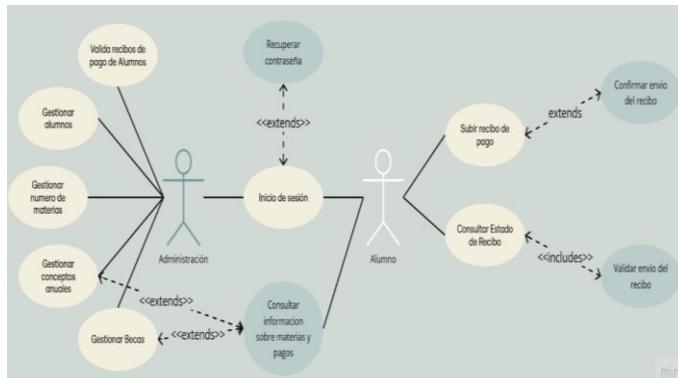
El desarrollo de un proyecto conlleva la elaboración diferentes diagramas que permitan una mejor comprensión de los diferentes procesos que se llevan a cabo y las tareas que se deberán de automatizar o sistematizar. Por tanto en el análisis del proyecto se hizo uso de diagramas de flujo y de casos de uso para el análisis profundo de los proceso que se deberán de mejorar a partir del desarrollo del sistema.

Asimismo, se elaboraron diagramas de navegación con el fin de dar una estructura y organización a los diferentes apartados que comprende el sistema. En cuanto al análisis necesario para la codificación se emplearon diagramas MVC debido a que el desarrollo del proyecto emplea un framework con este patrón de diseño de software.

Por último mencionar la elaboración del diagrama de entidad relación el cual nos permite una planeación y documentación de la base de datos que utilizará el sistema para la gestión de la información por parte del sistema de manera eficiente.

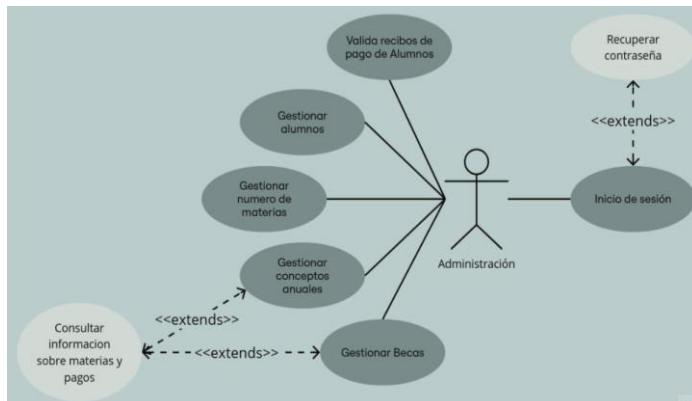
En la figura 1, se aprecian los roles de usuario administrador y alumno del sistema, en donde el administrador realiza un gran número de funciones dentro de la plataforma, como gestión de alumnos, número de materias cursadas por los alumnos, administración de conceptos anuales, alta de alumnos con becas y validación de recibos de pagos, siendo esta última actividad clave en el sistema, ya que en ella se verificarán los pagos realizados por los alumnos, por otro lado tenemos al rol del alumno el cual realiza dos actividades clave, las cuales son subir los recibos de pagos realizados y consultas del estado de los recibos cargados al sistema, brindando al alumno la posibilidad de saber si sus pagos se han acreditado en el departamento de servicios escolares o por el contrario este no cumple con los criterios de aceptación.

Figura 1
Diagrama general de casos de uso del sistema web



En la figura 2, se ilustran las diferentes acciones que realiza el administrador del sistema, las cuales son relacionadas con la gestión de alumnos, número de materias, becas y validación de los recibos de pago cargados por los estudiantes en la plataforma, cada uno de estos datos son esenciales para determinar los costos de los recibos de pago y su validación de forma correcta.

Figura 2
Diagrama de caso de uso rol de administrador



En la figura 3, se observan las acciones que realiza el rol de usuario de alumno dentro del sistema las cuales son cargar al sistema una fotografía o imagen de los recibos pagados por parte del alumno y la consulta del estado del recibo cargado el cual puede ser pendiente por cargar recibo de pago, en proceso de validación y completada su validación en el sistema.

Por último, tenemos el proceso de recuperación de contraseña en caso de que el alumno olvide sus credenciales de acceso al sistema.

En la figura 4, se muestra un diagrama de flujo del proceso de validación del recibo previamente cargado por el alumno, en cual el administrador del sistema realizará una inspección visual de la fotografía o de la imagen del recibo para validar su autenticidad y proceder a liberar el recibo de pago y pasar al estatus de completada su validación de forma satisfactoria.

Figura 3
Diagrama de caso de uso rol de alumno

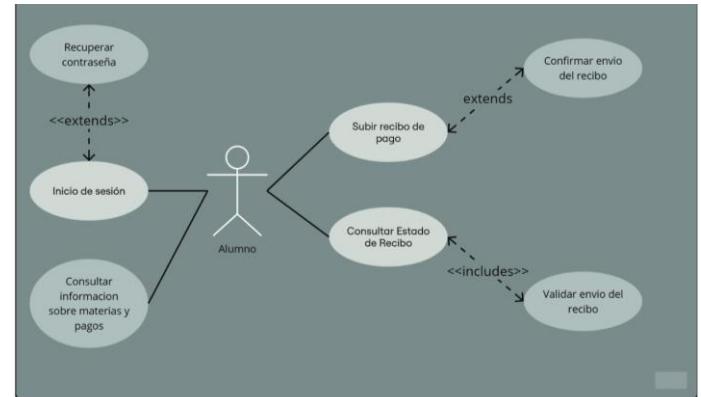
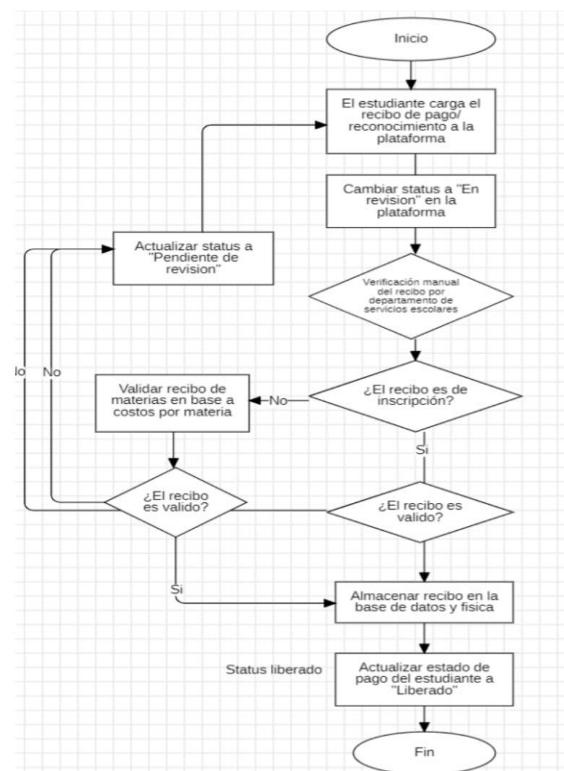


Figura 4
Diagrama diagrama de flujo proceso de validación de recibos de pagos



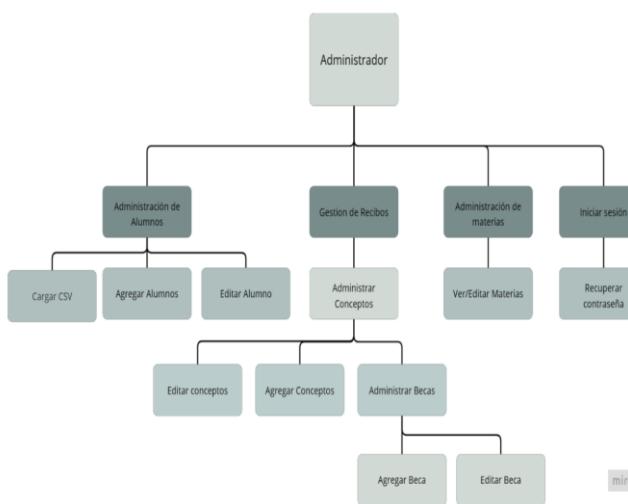
3.2 Construcción

La construcción o desarrollo de software requiere que se realicen diferentes diagramas que representen la forma en la que será construida la plataforma a desarrollar, a continuación se presenta la documentación de los diferentes diagramas diseñados en la etapa de construcción de la plataforma SCVRP.

En la figura 5, se expone el mapa de navegación del sistema del rol de administrador, en el que se observa cómo acceder a los diferentes apartados y funciones que se pueden realizar dentro

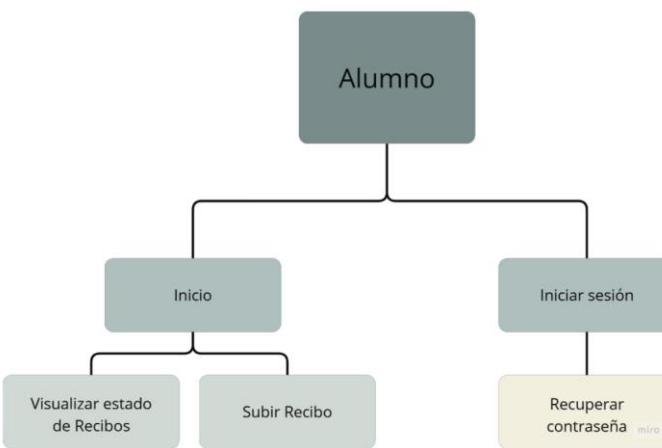
de la plataforma, como son la carga de alumnos los cuales son usuarios del sistema, la gestión de recibos de pago, administración de materias y el login o inicio de sesión.

Figura 5
Mapa de navegación del sistema rol de administrador



En la figura 6, se muestra el mapa de navegación con las diferentes rutas de acceso a los apartados que tiene acceso el rol de usuario del alumno, entre las cuales podemos observar la visualización de recibos de pago, su estado de validación y la funcionalidad del sistema que permite a los alumnos cargar recibos de pago, por último tenemos login o inicio de sesión al sistema, el cual está ligado a la opción de recuperar contraseña en caso de que el alumno no recuerde sus credenciales de acceso.

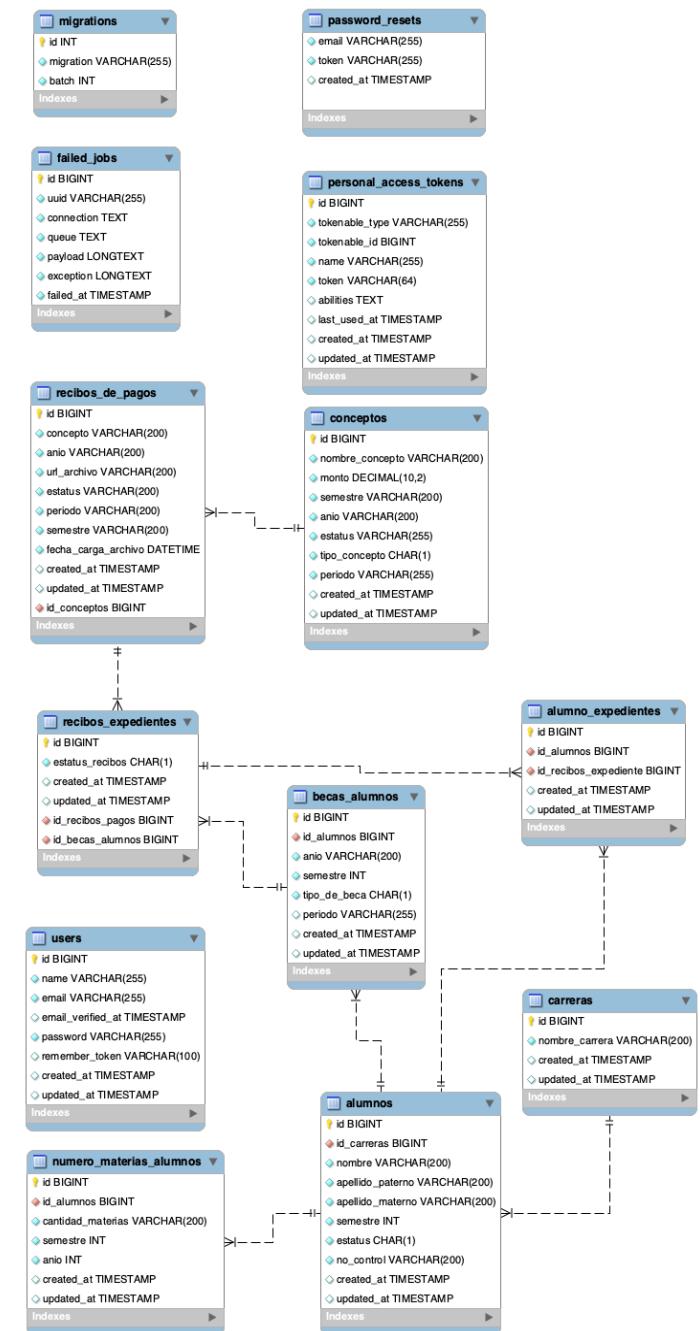
Figura 6
Mapa de navegación del sistema rol de alumno



Un diagrama entidad relación nos permite representar de manera gráfica la estructura de la base de datos a partir de entidades o tablas. En la figura 7, se ilustra el diagrama entidad relación el

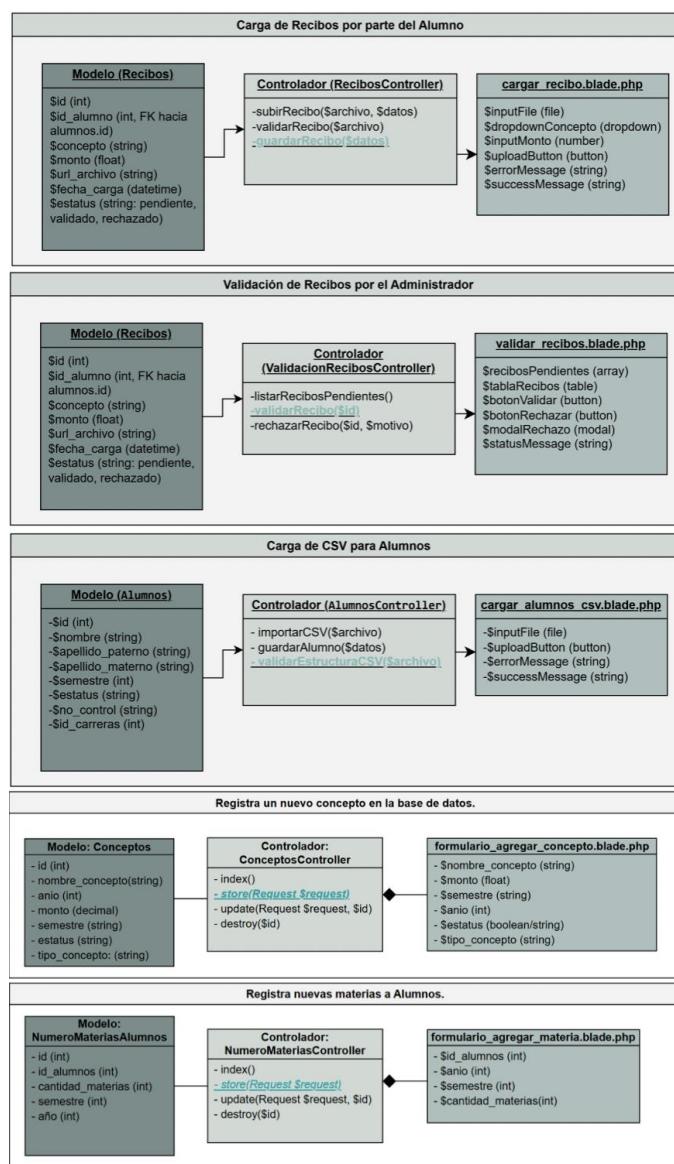
cual nos muestra diferentes tablas de la base de datos, algunas de las más cruciales para el sistema son las tablas de alumnos, materias, becas, alumnos y recibos de pagos, en estas tablas almacenaremos los datos de los alumnos que ingresarán al sistema para cargar sus recibos de pago y que a partir de estas entidades de datos se podrá calcular los costos de inscripción en base a las materias cursadas por cada alumno y la omisión de pago del concepto de materias por parte de los alumnos que cuenten con una beca.

Figura 7
Diagrama entidad relación de la base de datos.



En la figura 8, se presenta el diagrama con los principales modelos, controladores y vistas del sistema en base al modelo MVC el cual emplea el framework Laravel para la codificación del sistema. Los modelos en el marco de trabajo de Laravel representan una tabla de la base de datos y permiten el acceso de los datos, mientras que los controladores contiene la lógica o las operaciones que se realizarán por el sistema a desarrollar y como elemento final tenemos a las vistas o interfaces del sistema, las cuales muestran al usuario la información o datos empleando variables o estructuras de datos diversas, así también, permiten la interacción con el sistema a través de elementos gráficos, cabe mencionar que cada uno de los componentes del modelo MVC son elementos que se conectan entre sí para realizar las diferentes funciones y acciones en el sistema.

Figura 8
Diagrama MVC Laravel (Modelos, Controladores, Vistas).

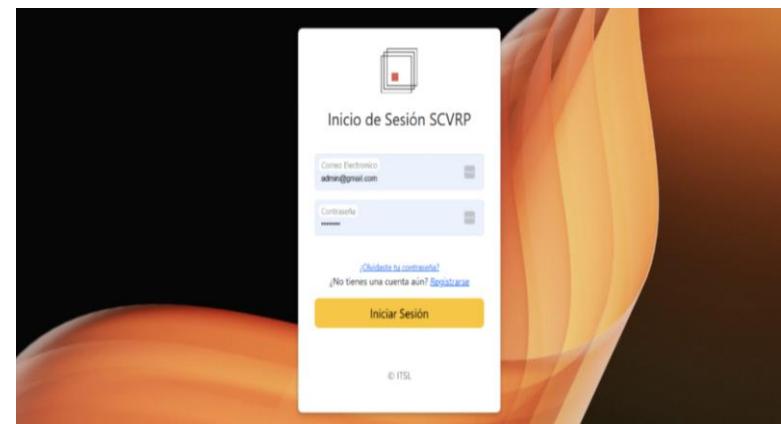


3.3 Producción

A continuación se muestra en este apartado los resultados que se obtuvieron como parte de la etapa de codificación de la plataforma SCVRP, mostrando algunas de las pantallas principales del sistema en ejecución.

En la figura 9, se muestra la interfaz de acceso al sistema, en la cual se ingresan las credenciales necesarias para el inicio de sesión tanto del rol de administrador así como del rol de alumno dentro del sistema SCVRP.

Figura 9
Interfaz acceso al sistema SCVRP.



En la figura 10, se puede apreciar la vista de inicio del sistema, la cual da la bienvenida al usuario que ha iniciado sesión y muestra el menú del sistema con las diferentes opciones que se pueden realizar dentro del sistema.

Figura 10
Interfaz inicio del sistema.



En la figura 11, es posible visualizar la interfaz de los expedientes digitales de recibos de pagos, en esta vista se podrán visualizar los recibos de pagos que los alumnos han cargado al sistema en forma de un expediente digital individual por alumno, así como también será posible realizar búsqueda a partir de filtros como el nombre del alumno o el número de control, el cual es un identificador único por alumno.

Figura 11
Interfaz de expedientes digitales alumnos.

The screenshot shows the 'SISTEMA DE CONTROL DE VERIFICACIÓN DE RECIBOS DE PAGO ITSL' interface. On the left, a sidebar menu includes 'Inicio', 'Administración de Recibos', 'Alumnos y Materias', 'Conceptos Anuales', and 'Becas'. The main area is titled 'Inicio Administración' and displays a table of student records. The columns include 'Número de Control', 'Carrera', 'Semestre', 'Nombre del Alumno', 'Estatus', and 'Opciones'. Each row shows a student's name, their degree program (e.g., Ingeniería en Sistemas, Ingeniería Industrial), the semester they are in (e.g., 5, 6, 7, 8), their name (e.g., Juan Pérez, María López), their status (e.g., Activo, Baja Temporal, Baja Definitiva), and a 'Ver expediente' button. At the bottom, there are navigation buttons for 'Previo', 'Siguiente', and '1 2 3'.

En la figura 12, se muestra la vista del sistema que permite ver los recibos de pagos cargados por los alumnos para su posterior validación por el administrador del sistema, cada recibo de pago debe de contener los datos del periodo que corresponde, así como datos del concepto al cual pertenece el pago realizado, datos necesarios para su validación satisfactoria en caso de que los datos proporcionados del pago correspondan con la fotografía o imagen del pago cargado por el alumno.

Figura 12
Interfaz de expediente digital de recibos de pagos.

The screenshot shows the 'SISTEMA DE CONTROL DE VERIFICACIÓN DE RECIBOS DE PAGO ITSL' interface. The sidebar menu is identical to Figure 11. The main area is titled 'Expediente @alumno' and shows a profile for 'MIRYÁ GARCÍA'. It includes fields for 'Número de Control', 'Carrera', and 'Semestre'. Below this is a table of payment receipts. The columns include 'Semestre', 'Periodo', 'Año', 'Concepto', 'Nº. Concepto', 'Costo por Concepto', 'Pago Recibo(s) / Beca / Reconocimiento', 'Monto Recibo', 'Estatus Recibo', 'Fecha de Carga', 'Entrega de Recibo Físico', 'Costo Total a Pagar', 'Abierto', and 'Estatus'. The table lists several entries, such as 'INSCRIPCIÓN' and 'MATERIAS', with details like the amount paid (\$2,500.00, \$160.00), the date of payment (2024-01-10, 2024-08-15), and the status (VALIDADO, NO VALIDADO).

En la figura 13, se muestra la interfaz para cargar la evidencia del pago realizado por parte del alumno por un determinado concepto, esta evidencia puede ser una fotografía del recibo de pago o imagen del mismo, donde se deberá de especificar datos como periodo, año y semestre al cual corresponde el pago.

Figura 13
Interfaz carga de recibos de pagos.

The screenshot shows the 'SISTEMA DE CONTROL DE VERIFICACIÓN DE RECIBOS DE PAGO ITSL' interface. The sidebar menu is identical to Figure 11. The main area is titled 'Carga de Recibos de Pago' and contains a form titled 'Formulario de Carga de Recibos'. It includes dropdown menus for 'Selección Periodo', 'Selección Año', 'Selección Semestre', and 'Subir Archivo'. There is also a note: 'Instrucciones: Para agregar un nuevo concepto a la lista, solo de clic en el botón "Agregar nuevo concepto" y completar el formulario. Si desea editar y/o eliminar, de clic en la opción que necesita en la sección de "Opciones" en la tabla.' A 'Cargar Recibo' button is at the bottom.

Como parte del control de recibos de pago el sistema permite dar de alta nuevos conceptos, los cuales los alumnos deberán de realizar el pago correspondiente según sea el semestre que cursa el alumno. En la figura 14, es posible visualizar los conceptos cargados al sistema así como las opciones disponibles para editar o eliminar algún concepto.

Figura 14
Interfaz alta de concepto de pago.

The screenshot shows the 'SISTEMA DE CONTROL DE VERIFICACIÓN DE RECIBOS DE PAGO ITSL' interface. The sidebar menu is identical to Figure 11. The main area is titled 'Carga de Conceptos Anuales' and shows a table of annual concepts. The columns include 'Concepto', 'Periodo', 'Año', 'Importe', and 'Opciones'. The table lists entries like 'INSCRIPCIÓN' and 'MATERIAS' for various years (2024, 2025) and periods (ENE-JUN, AGO-DIC). Each entry has 'Editar' and 'Eliminar' buttons in the 'Opciones' column.

En la figura 15, se presenta la interfaz a la cual accede el administrador del sistema para cargar la información de los alumnos de nuevo ingreso, así como las materias que cursa cada alumno, esto con la finalidad de poder calcular el pago que debe realizar cada alumno por el concepto de pago de materias, el cual se calcula en base a las materias cursadas en el anterior semestre al que se encuentra cursando.

Figura 15
Interfaz caga de alumno y materias.

The screenshot shows the 'SISTEMA DE CONTROL DE VERIFICACIÓN DE RECIBOS DE PAGO ITSL' interface. The sidebar menu is identical to Figure 11. The main area is titled 'Carga de alumnos y materias' and contains a note: 'Instrucciones: Seleccione un módulo para continuar.' Below this are two buttons: 'Cargar alumnos' and 'Cargar materias'.

III. RESULTADOS Y DISCUSIÓN

El resultado del uso de la metodología Cascada nos dio como resultado la división del proyecto en diferentes fases, entre las cuales se destaca el análisis, diseño y codificación e implementación las cuales fueron secuenciados.

En cuanto al uso de la metodología de SCRUM cada una de estas fases fue gestionada en base a diferentes roles de la metodología (Scrum Master, Product Owner y Equipo de Desarrollo) en la cual las tareas que se realizaron para el desarrollo del proyecto se pudo lograr desarrollar un proyecto que fuera en constante crecimiento dando como resultado una proyecto incremental en base a cada una de las versiones que se estaban desarrollando.

Las primeras fases del proyecto se determinaron los roles y funciones que desempeñarán los diferentes usuarios del sistema a partir de los múltiples diagramas expuestos en este artículo y así como fue posible determinar cuál sería la mejor manera de gestionar los recursos con los que contaría el sistema, cabe mencionar que gracias al estudio o análisis previo a la codificación del sistema el desarrollo del sistema fue fluido y de forma secuencial con lo que la aplicación de las metodologías Cascada y SCRUM es fundamental para la optimización y agilidad en los proyectos. En la fase de diseño del proyecto se realizó la planeación de las diferentes interfaces que comprenden el sistema, gran parte de las interfaces son el resultado de este análisis que se exponen en el apartado técnico que corresponde a esta publicación, los resultados obtenidos fueron interfaces amigables con el usuario y que cubren las necesidades de los procesos que realizarán en el sistema, cabe mencionar que esto se hizo en base a entrevistas con el Product Owner y los requisitos que se solicitaron. En la figura 16 se muestra cómo es que se documenta un requerimiento o historia de usuario para posteriormente el desarrollador pueda crear una nueva funcionalidad incremental al sistema, todo esto empleando la plataforma de control de versiones GitLap y sus funcionalidades de creación de tareas (Issues).

Es importante mencionar que el uso de las metodologías Cascada y SCRUM en conjunto no constituye una nueva metodología si no que fueron empleadas en conjunto debido a la naturaleza del proyecto el cual cumple con múltiples fases y que el desarrollo del proyecto destaca por ser un producto de software el cual se construye de forma incremental a partir de los requerimientos que se plantearon por los futuros usuarios del sistema.

Como punto final es importante mencionar que los resultados que se obtuvieron en la elaboración del proyecto Sistema de Control de Verificación de Pagos (SCVRP) son gracias a que las metodologías que facilitaron y permitieron tener un mejor control de avances y nuevos cambios al sistema desarrollado así como también hizo posible llevar un seguimiento de las actividades de una manera efectiva.

Figura 16

Documentación de historia de usuario plataforma GitLap. empleando la metodología SCRUM.

The screenshot shows a GitHub issue titled "Carga de Recibo de Pago". The issue is labeled "open" and was created 1 month ago by Francisco Eduardo Rodríguez Campos. The ID is HU-002. The description states: "Como alumno, quiero cargar un recibo de pago de inscripción (Imagen o PDF) para el semestre seleccionado, y que el sistema valide que el monto pagado corresponde al número de materias inscritas con base en un catálogo de costos, para asegurar que mi inscripción esté correctamente verificada." Below the description is a section titled "Criterios de Aceptación" with a numbered list of requirements. At the bottom, there is a note about the catalog of costs and file types, followed by standard GitHub interaction buttons and a dropdown menu.

IV. CONCLUSIONES

El proyecto de desarrollo del Sistema de Control de Verificación de Pagos (SCVRP) ha creado una aportación significativa en la modernización del proceso administrativo de gestión de pagos realizados por los alumnos dentro del TECNM / Instituto Tecnológico Superior de Lerdo. Al automatizar el proceso que anteriormente se realizaba de forma manual, con lo cual es posible reducir los errores humanos derivados de la sobrecarga de trabajo que los empleados del departamento de servicios escolares pueden llegar a tener en los períodos de inscripción de alumnos.

Un punto importante que se debe de tener en cuenta en este proyecto es que el uso de esta plataforma mejorará los tiempos en la gestión de recibos de pagos realizados por los alumnos, permitiendo hacer más eficiente el proceso de verificación de recibos de pago, brindando al área de servicios escolares una herramienta que mejore los tiempos de respuesta ante la demanda creciente de nuevos alumnos.

El uso de las tecnologías como lo es el framework Laravel y la base de datos MySQL empleadas en el desarrollo de esta plataforma web son claves en futuras actualizaciones, ya que permiten desarrollar nuevos módulos del sistema de una manera más ágil en comparación con otras tecnologías existentes, además de no representar un costo extra al no existir pago de licencias por su uso, creando un entorno de desarrollo escalable para futuras versiones del sistema.

Como mejoras futuras del proyecto se puede considerar la creación de nuevos módulos enfocados a la generación de reportes de datos que permitan realizar estrategias de mejora continua en base a los resultados que se obtengan de los reportes de la plataforma.

Por último, mencionar que este proyecto puede ampliar sus alcances y expandirse a nuevas instituciones educativas que cuenten con problemáticas semejantes a las que aborda el desarrollo de esta plataforma.

Este proyecto no solo plantea la automatización y mejora de un proceso clave para el departamento de escolares del TECNM / Instituto Tecnológico Superior de Lerdo, si no que abre la puerta a crear una plataforma que integre futuros procesos que tomen de base los datos de alumnos inscritos en la institución.

V. AGRADECIMIENTOS

Agradecemos al departamento de servicios escolares y de investigación del TECNM / Instituto Tecnológico Superior de Lerdo por todos los recursos que proporcionaron para el desarrollo de este proyecto, especialmente al titular del área Subdirección de Investigación y Desarrollo Tecnológico del ITSL, Dr. Jesús Alejandro Valdés Nieblas por su apoyo constante a cada uno de los miembros que participaron en este proyecto.

VI. REFERENCIAS

- Atlassian. (s. f.). What is Scrum? Atlassian.
<https://www.atlassian.com/agile/scrum>
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... Thomas, D. (2001). Manifesto for Agile Software Development. <https://agilemanifesto.org>
- DuBois, P. (2020). MySQL (5th ed.). Addison-Wesley Professional.
- Haque, A. (2019). Beginning Laravel: Build websites with Laravel 5.8. Apress.
- Highsmith, J. (2010). Agile project management: Creating innovative products (2nd ed.). Addison-Wesley Professional.
- Otwell, T. (2023). Laravel documentation. Laravel.
<https://laravel.com/docs>
- Royce, W. W. (1970). Managing the development of large software systems. Proceedings of IEEE WESCON, 1–9.
- Silva, R., Cruz, E., Méndez, I., & Rodríguez, J. (2013). Sistema de gestión digital para mejorar los procesos administrativos de instituciones de educación superior: Caso de estudio en la Universidad Autónoma Metropolitana. Perspectiva Educacional, 52(2), 104–134. <https://doi.org/10.4151/07189729-Vol.52-Iss.2-Art.142>
- Torres-Espriú, F. A., & Hernández López, M. S. (2024). Influencia de la gestión del conocimiento, la administración del tiempo y el manejo de TIC y TAC en el cumplimiento de procesos administrativos en instituciones educativas. IE Revista de Investigación Educativa de la REDIECH, 15, e2165.
https://doi.org/10.33010/ie_rie_rediech.v15i0.2165

Valencia Camacho, C., & Almeida Delgado, V. (2024). La tecnología en la gestión educativa. Revista de Investigación Latinoamericana en Competitividad Organizacional, 6(23), 73–79.
<https://doi.org/10.51896/rilco.v6i23.648>