

Transformación Digital de la Gestión Financiera Universitaria: Implementación del Sistema SCVRP con Inteligencia de Negocio y Dashboards Interactivos

Digital Transformation of University Financial Management: Implementation of the SCVRP System with Inteligencia Empresarial and Interactive Dashboards

Rodríguez-Campos, F.E.¹; Parada-Morado, L. ; Rodríguez-Chacón, R.L.³; Baijen-Saucedo, K.M.⁴

Datos de Adscripción:

¹ Rodríguez Campos, Francisco Eduardo.
Tecnológico Nacional de México - Instituto Tecnológico Superior de Lerdo.
Ciudad Lerdo, Durango. México.
ROR ID: <https://ror.org/04bjtbt12>
francisco.rc@itslerdo.edu.mx
ORCID: <https://orcid.org/0000-0001-5276-3681>

² Parada Morado, Lilia.
Tecnológico Nacional de México - Instituto Tecnológico Superior de Lerdo.
Ciudad Lerdo, Durango. México.
ROR ID: <https://ror.org/04bjtbt12>
lilia.pm@itslerdo.edu.mx
ORCID: <https://orcid.org/0000-0003-2169-0533>

³ Rodríguez Chacón, Rocío Lorena.
Tecnológico Nacional de México - Instituto Tecnológico Superior de Lerdo.
Ciudad Lerdo, Durango. México.
ROR ID: <https://ror.org/04bjtbt12>
rocio.rc@itslerdo.edu.mx
ORCID: <https://orcid.org/0009-0001-0842-1585>

⁴ Baijen Saucedo, Karla Mariana.
Tecnológico Nacional de México - Instituto Tecnológico Superior de Lerdo.
Ciudad Lerdo, Durango. México.
ROR ID: <https://ror.org/04bjtbt12>
212310952@itslerdo.edu.mx
ORCID: <https://orcid.org/0009-0008-1086-7851>

Resumen. La dependencia de procesos manuales para validar pagos universitarios suele provocar errores administrativos y retrasos que afectan a los estudiantes. Para resolver esta situación, se implementó el Sistema de Control de Verificación de Recibos de Pago en el Tecnológico Nacional de México, Campus Lerdo. Este trabajo no se limitó a digitalizar documentos físicos, sino que buscó convertir los datos financieros en herramientas de gestión útiles. Para el desarrollo técnico, se combinó una planificación detallada de las reglas financieras con una programación ágil, utilizando tecnologías accesibles como Laravel, MySQL y Docker para asegurar que el sistema sea seguro y escalable. El aporte principal consistió en integrar tableros interactivos y reportes automáticos que antes no existían. Ahora, los administradores pueden visualizar en tiempo real el estado de los ingresos, filtrar la información por carrera o periodo escolar y descargar reportes ejecutivos en PDF o CSV con un solo clic. Esto permitió eliminar los cálculos erróneos y centralizar toda la trazabilidad de los pagos en una base de datos confiable y segura. Las pruebas realizadas muestran que el personal redujo significativamente su carga de trabajo y aumentó la

transparencia del proceso para los usuarios. En conclusión, el proyecto demuestra que la automatización bien estructurada otorga visibilidad total para auditar y tomar decisiones estratégicas. Esta investigación confirma que el uso de software accesible moderniza la gestión educativa, beneficiando tanto a la administración como a la comunidad estudiantil, sin requerir inversiones complejas o costosas para la universidad.

Palabras Clave: Automatización de procesos, Gestión de pagos universitarios, Inteligencia de negocios, Tableros interactivos.

Abstract. Relying on manual processes to validate university payments often leads to administrative errors and delays that affect students. To address this, the Payment Receipt Verification Control System was implemented at Tecnológico Nacional de México, Campus Lerdo. This work went beyond digitizing physical documents and aimed to transform financial data into useful management tools. For technical development, detailed planning of financial rules was combined with agile programming, using accessible technologies such as Laravel, MySQL, and Docker to ensure the system is secure and scalable. The main contribution was integrating interactive dashboards and automated reports that did not previously exist. Administrators can now view income status in real time, filter information by major or school period, and download executive reports in PDF or CSV format with a single click. This eliminated erroneous calculations and centralized payment traceability in a reliable, secure database. Tests carried out show that staff significantly reduced their workload and increased process transparency for users. In conclusion, the project shows that well-structured automation provides full visibility for auditing and strategic decision-making. This research confirms that accessible software modernizes educational management, benefiting both the administration and the student community without requiring complex or costly investments for the university.

Keywords: Business intelligence; Interactive dashboards; Process automation; University payment management.

I. INTRODUCCIÓN

Administrar los recursos financieros y validar los pagos son tareas que mantienen operando a las universidades. Sin embargo, depender de métodos manuales para revisar cada recibo impreso de los estudiantes se ha convertido en un verdadero desafío administrativo. Como bien señala Huamán-Yupanqui (2024), la ausencia de automatización en este tipo de procesos no solo retrasa trámites tan importantes como las reinscripciones o las titulaciones, sino que también abre la puerta a errores humanos, a la pérdida de documentos y, por supuesto, a la frustración de los estudiantes. En las temporadas altas, esta situación se traduce en una carga de trabajo insostenible para el personal administrativo. Esta era exactamente la realidad que se vivía en el Departamento de Recursos Financieros del Tecnológico Nacional de México (TecNM) Campus Lerdo. La matrícula estudiantil seguía creciendo, pero los procedimientos se habían quedado rezagados. Para dar solución a esta problemática, se creó el Sistema de Control de Verificación de Recibos de Pago (SCVRP). Hoy en día, esta plataforma es una realidad y se encuentra en una etapa de implementación progresiva. Se comenzó por capacitar al personal y realizar pruebas piloto con el objetivo de preparar el terreno para que todos los estudiantes lo adopten. Esta transición paulatina permite asegurar que las finanzas de la institución no se detengan ante eventuales resistencias al cambio.

Este salto hacia lo digital está fuertemente respaldado por la literatura especializada. Investigadores como León-Velarde et al. (2024) y Ruiz-Muñoz & Narváez Vega (2024) coinciden en que transferir estos trámites a una plataforma web transforma por completo la percepción del estudiante y elimina fricciones burocráticas. Para que el SCVRP fuera rápido y seguro, se decidió construirlo bajo la arquitectura Modelo-Vista-Controlador (MVC) con el framework Laravel y PHP (2025), protegiendo al sistema contra vulnerabilidades. Para organizar la información financiera se eligió MySQL (2025), y para garantizar que el sistema soporte el tráfico de estudiantes sin colapsar, se orquestó su despliegue mediante contenedores de Docker (Docker Inc., s. f.) en servidores Linux (GeeksforGeeks, 2025). No obstante, automatizar no se trata únicamente de almacenar recibos digitales; el verdadero valor estratégico está en comprender esa información. En la actualidad, la digitalización universitaria exige que los datos brutos se transformen en conocimiento mediante reportes y dashboards (tableros visuales). Como destacan Cedillo Martínez & García Jiménez (2025) y Vargas et al. (2025), estas herramientas visuales constituyen los "ojos" de los directivos: les permiten detectar patrones de pago, auditar los ingresos y tomar decisiones en tiempo real. De hecho, en ecosistemas similares, automatizar la conciliación de reportes ha mejorado los tiempos de respuesta en más del 90% (Herrera-Pérez et al., 2022).

En este artículo se presenta el proceso de desarrollo del SCVRP y los resultados operativos de sus primeras pruebas en el TecNM Campus Lerdo. Se expone cómo la combinación de metodologías condujo al éxito: se utilizó el modelo tradicional en Cascada para analizar sin errores las reglas financieras (Delgado Olivera & Díaz Alonso, 2021), y se apoyó en Scrum y tableros Kanban para programar de forma ágil e iterativa (Bautista Villegas, 2022; Navarro Cadavid, Fernández Martínez, & Morales

Vélez, 2013; Zunzunegui, 2023). El resultado es una solución que no solo automatiza la validación, sino que, mediante su robusta generación de reportes y gráficas, ofrece una trazabilidad financiera total y moderniza el núcleo administrativo de la institución.

II. PARTE TÉCNICA DEL ARTÍCULO

El desarrollo del Sistema de Control de Verificación de Recibos de Pago (SCVRP) no fue solo un ejercicio de programación, sino una investigación tecnológica orientada a resolver un problema real y urgente del Departamento de Recursos Financieros. Para asegurar que el sistema cumpliera con las estrictas reglas de la institución, pero que al mismo tiempo pudiera adaptarse a cambios futuros, se decidió implementar un enfoque metodológico híbrido. Básicamente, se tomó la solidez y el orden de la metodología en cascada y se combinaron con la velocidad y la flexibilidad de los marcos ágiles como Scrum y Kanban, construyendo todo sobre herramientas de código abierto.

2.1 Metodología de desarrollo: Enfoque híbrido

El proyecto se dividió en dos grandes etapas. En la fase inicial, se apostó por la metodología en Cascada para levantar todos los requerimientos y diseñar la arquitectura de la base de datos (Delgado Olivera & Díaz Alonso, 2021). Esta fase era crítica: al tratarse de un sistema financiero, las reglas del juego —como los cálculos exactos de las inscripciones o el costo por materia— tenían que estar blindadas y definidas a la perfección antes de escribir una sola línea de código.

Una vez que se contó con esa base sólida, se pasó a la etapa de construcción, utilizando Scrum para trabajar de forma iterativa (Bautista Villegas, 2022). El trabajo se dividió en Sprints o bloques de tiempo, lo que permitió ir entregando módulos funcionales gradualmente. Se definieron roles claros, con el Product Owner como la voz del departamento y un equipo de desarrollo sincronizado (Zunzunegui, 2023). Para que nadie se perdiera, se utilizaron tableros Kanban visuales y reuniones diarias (Daily Scrum), lo que permitió detectar y resolver de inmediato cualquier cuello de botella técnico (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013).

2.2 Arquitectura de software y patrón MVC.

Para darle vida al sistema, se eligió el framework PHP Laravel. La razón principal fue su robustez, su código limpio y, sobre todo, sus altos estándares de seguridad (Laravel, 2025). Para mantener todo organizado, se estructuró la aplicación bajo el patrón Modelo-Vista-Controlador (MVC), el cual separa perfectamente el diseño de la lógica (Subecz, 2021):

- **Modelo (Model):** Se utilizó el ORM Eloquent de Laravel para "hablar" con la base de datos como si fuera un objeto real. Esto evitó tener que escribir consultas SQL largas y complejas, lo que hizo que la gestión de expedientes y recibos fuera mucho más ágil y segura.
- **Vista (View):** Para la interfaz de usuario, se empleó el motor de plantillas Blade combinado con Bootstrap. El objetivo era claro: crear pantallas rápidas, atractivas y responsivas para que la experiencia de uso (UX) sea excelente, ya sea desde una computadora o un celular.

- **Controlador (Controller):** Es el "cerebro" de la operación. Aquí se procesan las peticiones, se aplican las reglas de negocio definidas en la fase de Cascada y se entrega la información procesada a las vistas.

2.3. Herramientas, entorno de desarrollo y despliegue

Programar y lanzar un sistema de este calibre exigió herramientas de nivel profesional:

- **Desarrollo y código:** Se programó todo en Visual Studio Code (Microsoft, 2025) y se gestionaron el control de versiones y el trabajo colaborativo mediante Git (Git, 2025).
- **Entorno local y datos:** Se comenzó a probar el sistema localmente con la suite XAMPP (servidor Apache) y Composer. Para almacenar la información, se diseñó una base de datos relacional en MySQL (MySQL, 2025). Es vital mencionar la creación del Diagrama Entidad-Relación (DER), que sirvió de mapa para asegurar que las tablas de estudiantes, carreras y pagos se conectaran sin duplicar datos ni causar errores.
- **Salida a producción:** Para evitar el clásico problema de "en mi máquina sí funciona", el sistema final se empaquetó en contenedores de Docker (Docker Inc., s. f.) en un servidor Linux (GeeksforGeeks, 2025). Se cambió a Nginx para que el sistema soporte a muchos usuarios conectados simultáneamente durante los periodos de cobro, logrando así un entorno estable y a prueba de caídas.

2.4 Seguridad y validaciones

Al tratarse de procesos de recaudación, la seguridad no era opcional. El SCVRP incluye *middlewares* que protegen las sesiones y bloquean a cualquier persona que intente acceder a las rutas administrativas sin permiso. Además, cuando un estudiante sube su comprobante, el sistema inspecciona el archivo "por detrás" (*backend*) para aceptar únicamente formatos seguros (JPG, PNG, PDF) y con un peso máximo, protegiendo así a los servidores de saturarse o de recibir archivos maliciosos.

2.5 Modelado funcional y flujos del sistema

Para no programar a ciegas, primero se dibujaron los "planos" del sistema. Se elaboraron diagramas que explican, paso a paso, cómo interactúan los estudiantes y los administradores con la plataforma. En este diseño se puso especial atención en cómo el sistema procesa los datos para generar reportes y gráficas. A continuación, se detallan estos procesos.

Vista general del sistema (casos de uso)

Como punto de partida, la Figura 1 muestra quién hace qué en el SCVRP. En este diagrama se definen los dos actores principales: el estudiante (que solo puede subir recibos y ver su propio historial) y el administrador de finanzas (que tiene el poder total para validar pagos, consultar expedientes y acceder a las métricas del *dashboard*).

Diagrama de carga de comprobantes de pago

Como se puede observar en la Figura 2, este esquema modela el flujo en el que el estudiante, una vez autenticado, ingresa al sistema para subir su comprobante digital. El sistema

requiere que se elijan un concepto específico (inscripción o materias), un periodo y un año, y calcula automáticamente si el monto declarado es correcto antes de cambiar el estatus del trámite a "En revisión".

Gestión de expedientes de estudiantes.

La Figura 3 ilustra el flujo moderado para la generación de reportes de expedientes académicos y financieros mediante su número de control para visualizar de forma consolidada todos los conceptos pagados y adeudos pendientes desde su periodo de ingreso, permitiendo exportar dicha trazabilidad integral en formato PDF.

Diagrama de validación de comprobantes

En la Figura 4 se muestra este proceso crítico a cargo del administrador. El flujo ilustra detalladamente cómo el personal inspecciona los comprobantes cargados, verifica matemáticamente el costo de las materias frente al pago realizado y constata la entrega física del documento. Dependiendo de esta evaluación, se puede notar cómo el sistema bifurca la acción en la liberación del comprobante o en su rechazo definitivo.

Generación de dashbord e indicadores de pago.

Finalmente, como se puede observar en la Figura 5, se modeló la extracción de inteligencia de negocio en la plataforma. El diagrama detalla cómo el sistema recopila los datos en bruto de los comprobantes (liberados, pendientes y rechazados) y los procesa a través de filtros de año y periodo para renderizar métricas visuales mediante Chart.js, facilitando a los directivos la toma de decisiones financieras en tiempo real.

Generación de reportes de recibos de pago

Como se ilustra en la Figura 6, se modeló un flujo adicional para que el personal administrativo pueda generar listados detallados de las transacciones. El sistema permite aplicar filtros por periodo escolar, carrera o estatus del recibo, lo que facilita la consolidación de la información financiera y su exportación para el cotejo con los ingresos institucionales.

Arquitectura del sistema (patrón MVC)

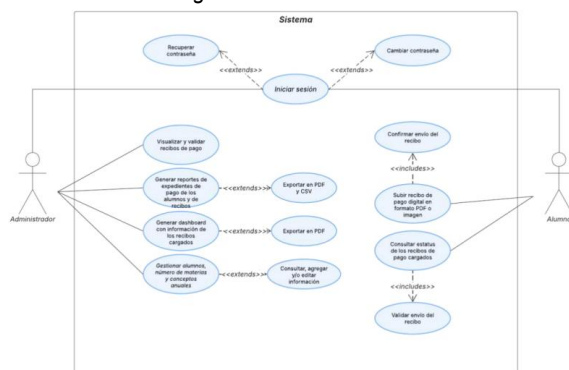
Además de los flujos operativos de los usuarios, la Figura 7 del Anexo A muestra la arquitectura interna del sistema, basada en el patrón Modelo-Vista-Controlador. Este esquema detalla la separación lógica explicada en las secciones anteriores, mostrando cómo interactúan las interfaces de usuario (Blade) con el núcleo lógico (Controladores) y el gestor de base de datos. (Eloquent).

Estructura de la base de datos

Para complementar el modelado general, la Figura 8 muestra el Diagrama de Entidades Relacionales (DER). Este esquema gráfico muestra las tablas principales (usuarios, estudiantes, carreras, conceptos y comprobantes de pago) y las llaves foráneas que garantizan la integridad referencial de toda la información financiera gestionada por la plataforma.

Figura 1

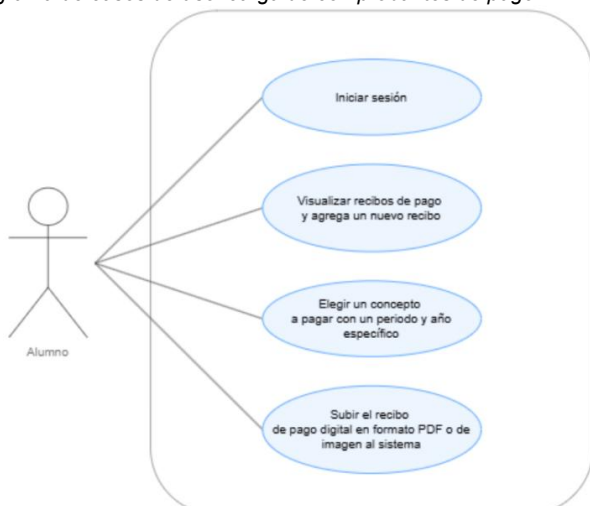
Diagrama de casos de uso general del sistema



Nota. Elaboración propia. Se muestran los dos actores del sistema y sus principales interacciones con la plataforma SCVRP.

Figura 2

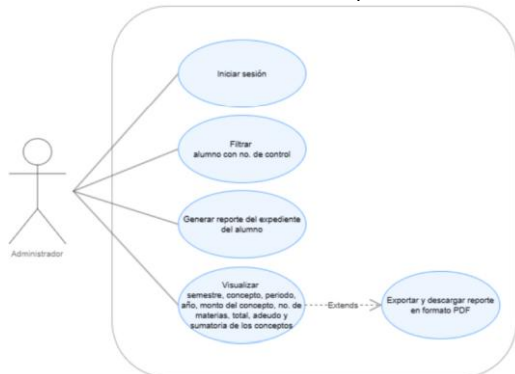
Diagrama de casos de uso: carga de comprobantes de pago



Nota. Elaboración propia. Se detalla el flujo que sigue un estudiante para registrar un nuevo comprobante de pago en el sistema.

Figura 3

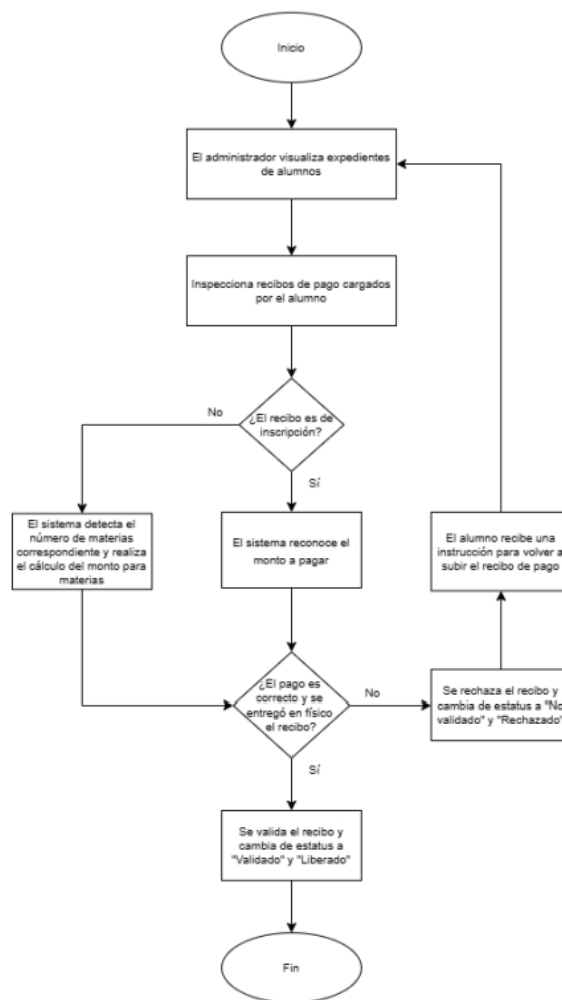
Diagrama de casos de uso: Generación de expedientes



Nota. Elaboración propia. Flujo para la consulta y exportación del historial financiero de un estudiante.

Figura 4

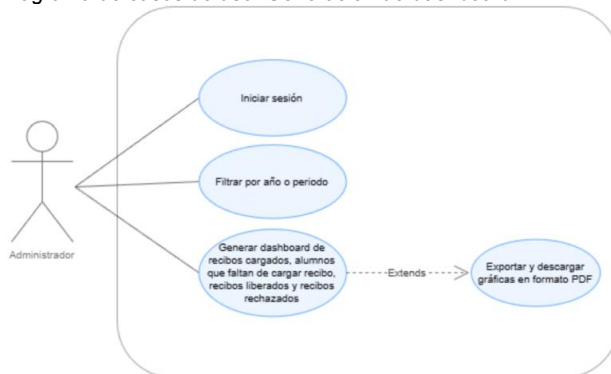
Diagrama de flujo: validación de comprobantes



Nota. Elaboración propia. Se representa el proceso de toma de decisiones del administrador financiero al revisar cada recibo.

Figura 5

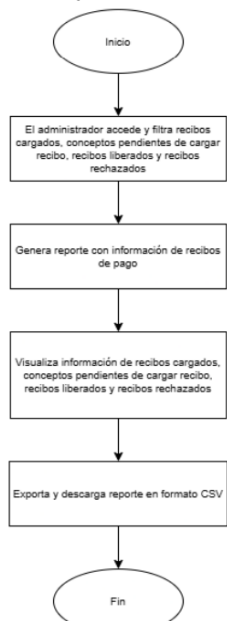
Diagrama de casos de uso: Generación de dashboard



Nota. Elaboración propia. Muestra el flujo de transformación de datos brutos en indicadores visuales.

Figura 6

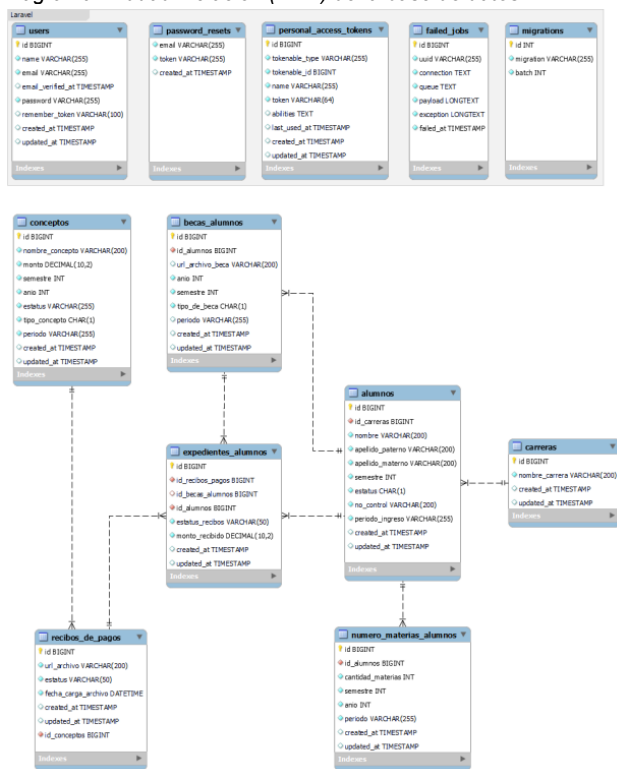
Diagrama de flujo: Generación de reportes de recibos



Nota. Elaboración propia. Proceso para obtener listados filtrados y exportables de los comprobantes registrados.

Figura 8

Diagrama Entidad-Relación (DER) de la base de datos



Nota. Elaboración propia. Representación de las entidades, atributos y relaciones de la base de datos del SCVRP.

III. RESULTADOS Y DISCUSIÓN

El desarrollo del Sistema de Control de Verificación de Recibos de Pago (SCVRP) no se limitó a la teoría; culminó en la implementación de una plataforma web real, segura y hecha a la medida para el Departamento de Recursos Financieros del TecNM Campus Lerdo. Si bien la construcción de las interfaces agilizó los flujos operativos, el mayor éxito del proyecto radicó en transformar un proceso manual "a ciegas" en un ecosistema capaz de generar reportes detallados y métricas en tiempo real.

3.1 Implementación de Módulos e Interfaces de Usuario

La interfaz gráfica fue desarrollada con principios de usabilidad, lo que asegura una navegación intuitiva y minimiza la curva de aprendizaje tanto para los estudiantes como para el personal administrativo.

Módulo de Autenticación y Carga de Comprobantes. Como se observa en la Figura 9, se desarrolló un panel en el que el estudiante puede visualizar su historial de pagos y adjuntar nuevos comprobantes. El sistema limita el tamaño del archivo a un máximo de 5 MB y requiere que el estudiante especifique el concepto y el monto pagado. Si el monto ingresado no coincide con la tarifa oficial del concepto, la carga se rechaza automáticamente y se emite una instrucción para que el estudiante acuda a los servicios escolares.

Figura 9

Interfaz del módulo de carga de comprobantes de pago para estudiantes

Historial y Seguimiento Estudiantil. Para cerrar el ciclo de transparencia, se implementó un panel de seguimiento, mostrado en la Figura 10. En esta vista, el estudiante puede consultar el listado histórico de sus comprobantes cargados y conocer en tiempo real el estatus de revisión de cada trámite (en revisión, aprobado o rechazado), evitando la incertidumbre durante los periodos de validación.

Panel de Verificación y Validación Administrativa. Para el personal de finanzas, se implementó una bandeja de entrada centralizada que permite inspeccionar los archivos cargados, como se muestra en la Figura 11. El sistema adapta el formulario de validación según el concepto: para "Inscripciones" verifica el monto recibido y, para "Materias", exige ingresar la cantidad correcta según el número de asignaturas. En ambos casos, el administrador debe marcar una casilla de confirmación de la entrega del recibo físico para liberar el trámite.

Figura 10

Panel del estudiante para el seguimiento y estatus de recibos cargados



Figura 11

Panel administrativo para la verificación y validación de comprobantes



Gestión de Expedientes de Estudiantes. El sistema permite la consulta histórica integral. En la Figura 12 se ilustra la vista mediante la cual, a través de una búsqueda por número de control, el administrador puede visualizar de forma consolidada tanto los conceptos pagados como los no pagados (adeudos) desde el periodo de ingreso del estudiante. Esta trazabilidad integral puede exportarse directamente a un reporte en formato PDF.

Figura 12

Interfaz de gestión y exportación de expedientes de estudiantes en PDF



3.2 Inteligencia de Negocio y Trazabilidad

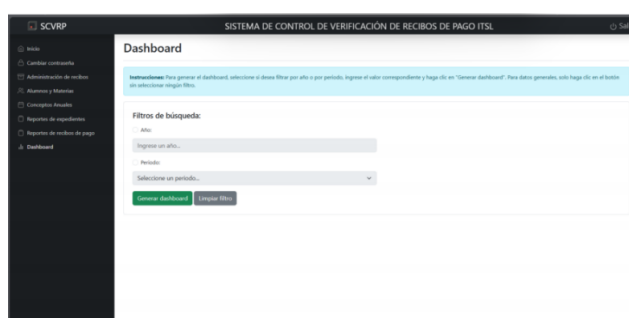
Uno de los resultados más destacables de la implementación del sistema es la integración de herramientas de inteligencia de negocio orientadas a la toma de decisiones directivas.

Reportes de recibos en CSV. Se desarrolló un submódulo que permite aplicar filtros por carrera o periodo escolar para contabilizar recibos pendientes, liberados, rechazados y totales cargados. Toda esta información puede ser exportada a formato CSV. El resultado de esta exportación genera un archivo estructurado, el cual permite al personal de finanzas organizar la situación del alumnado de forma masiva e interoperar con otras herramientas contables.

Generación de Dashboard Financiero. Los directivos ya no tienen que adivinar cómo va la recaudación. Para acceder a esta inteligencia de negocio, el administrador cuenta primero con un panel de filtros. En la figura 13 se observa cómo es posible segmentar la información según el año o periodo escolar que desea auditar, lo que le brinda total flexibilidad a sus consultas.

Figura 13

Panel de configuración y filtros para la generación del dashboard



Una vez aplicados los filtros, toda la información procesada alimenta un *dashboard* interactivo (Figura 14) que traduce y transforma miles de registros aburridos en gráficas de barras y de pastel claras y dinámicas. Esta vista permite analizar al instante el índice de pagos liberados, rechazados y pendientes, segmentado por carrera, e incluye una distribución general y tablas estadísticas detalladas.

Figura 14

Dashboard interactivo con gráficas de métricas financieras por carrera



Finalmente, sabemos que las juntas directivas y las auditorías requieren documentos formales. Por ello, con un solo clic, el sistema consolida todas estas gráficas en un reporte ejecutivo en formato PDF. En la Figura 15 se ilustra este documento final, que se genera automáticamente, está organizado y listo para descargar, asegurando que los altos mandos tengan siempre a la mano una fotografía exacta de la salud financiera de la institución.

Figura 15
Reporte ejecutivo del dashboard financiero exportado en formato PDF



3.3 Discusión y Pruebas de Implementación

Poner a prueba el SCVRP en un entorno real nos demostró que la ingeniería de software puede transformar por completo la administración escolar. Al comparar los resultados con el modelo de gestión manual documentado en las fases iniciales del proyecto, destacamos lo siguiente:

Optimización de tiempos. Automatizar las sumas matemáticas y digitalizar la validación redujeron drásticamente el estrés y la carga operativa del personal, eliminando los clásicos cuellos de botella característicos durante los periodos críticos de cierre de semestre y reinscripciones.

Mitigación de errores. Al aplicar validaciones estrictas desde el código (backend) y bloquear montos incorrectos, erradicamos incidencias históricas como registros incompletos, la pérdida de comprobantes físicos y las alteraciones en los pagos declarados por los estudiantes.

Trazabilidad y seguridad. Logramos centralizar las finanzas en una bitácora digital imposible de alterar. Hoy, cada movimiento (aprobación o rechazo) deja una huella transparente, lo que mejora sustancialmente la percepción de calidad del servicio por parte de la comunidad estudiantil.

Visibilidad y Auditoría mediante Reportes. El cambio más profundo fue darle "ojos" a la administración. La implementación central de reportes exportables (PDF/CSV) y dashboards interactivos resolvió la falta de visibilidad analítica del departamento. La capacidad de auditar cientos de pagos en segundos marca el salto tecnológico más importante del proyecto y le otorga a la institución herramientas proactivas para la toma de decisiones.

IV. CONCLUSIONES

El desarrollo y puesta en marcha del SCVRP dejó una lección clara: modernizar la administración financiera universitaria ya no es un lujo, es una necesidad urgente. Depender del papel, de las calculadoras y de las validaciones manuales puede superarse con éxito mediante la ingeniería de software aplicada, tal como señalan Huamán-Yupanqui (2024) y Ruiz-Muñoz & Narváez Vega (2024).

A nivel metodológico, el enfoque híbrido fue un acierto. La metodología en Cascada (Delgado Olivera & Díaz Alonso, 2021) nos proporcionó la rigurosidad necesaria para definir las reglas financieras sin errores, mientras que Scrum (Bautista Villegas, 2022) nos permitió programar de forma iterativa y adaptarnos a los cambios. Las herramientas elegidas —Laravel, MySQL y Docker— demostraron ser suficientes para construir un sistema seguro, escalable y capaz de soportar la carga de miles de estudiantes en periodos críticos.

Más allá del código, el mayor logro del proyecto fue transformar un ecosistema "ciego" en uno analítico. Los reportes automatizados, las exportaciones en CSV y los dashboards interactivos permitieron al Departamento de Recursos Financieros auditar los ingresos en tiempo real y tomar decisiones basadas en datos, no en estimaciones. La implementación progresiva aún no alcanza al 100% de los estudiantes, por lo que se requiere continuar con la capacitación y el acompañamiento.

Limitaciones y trabajo futuro. La implementación progresiva aún no alcanza al 100% de los estudiantes, por lo que se requiere continuar con la capacitación y el acompañamiento.

V. AGRADECIMIENTOS

Los autores desean agradecer a la Dirección General y al Departamento de Subdirección de Posgrado e Investigación del TECN – Instituto Tecnológico Superior de Lerdo, Dgo., por el apoyo brindado para la realización de este proyecto, así como a los departamentos de Control Escolar y Recursos Financieros del ITSL por facilitar el acceso a los procesos y datos necesarios para el desarrollo del sistema.

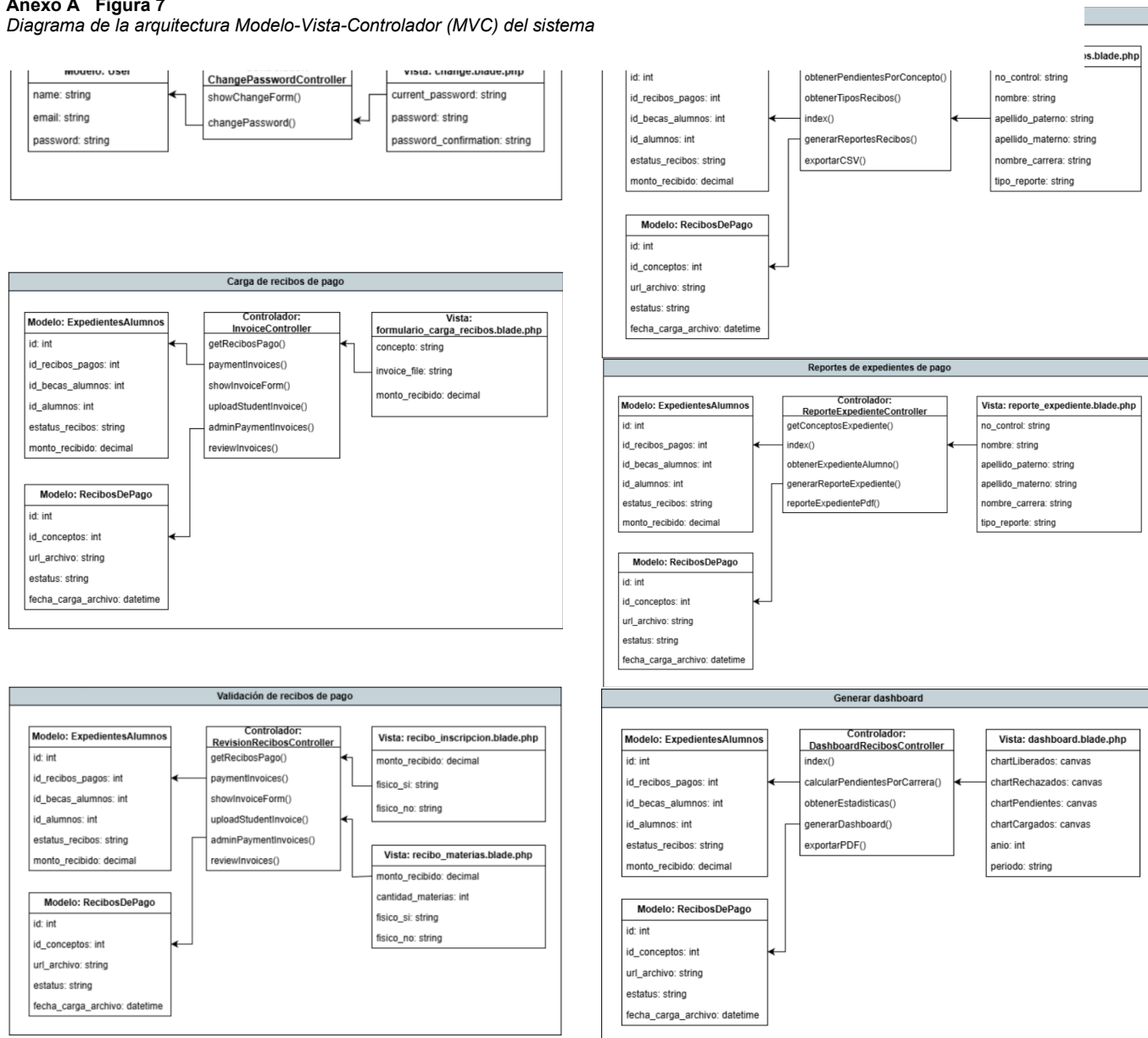
VI. REFERENCIAS

Álvarez Campos, D. A., & Seijas Velásquez, S. (2017). Implementación de un registro computarizado de matrículas y pagos en la Institución Superior Tecnológico Público Jorge Desmaison

- Seminario, Pacasmayo. *Sciéndo Ingenium*, 12(4), 19-32.
<https://revistas.unitrú.edu.pe/index.php/PGM/article/view/1513>
- Bautista Villegas, J. (2022). *Metodologías ágiles para el desarrollo de software*. Editorial Universitaria.
- Carballo Mendivil, B., Pérez Peralta, P. N., & Arellano Pineda, B. A. (2019). Sistema de información de apoyo a la medición de indicadores de trayectoria escolar en educación superior. *Revista de Investigación en Tecnologías de la Información (RITI)*, 7(14), 64–74. <https://dialnet.unirioja.es/descarga/articulo/7242769.pdf>
- Cedillo Martínez, I. A., & García Jiménez, L. F. (2025). Modelado de un sistema de reinscripciones escalable sobre una nube híbrida usando Learning Analytics. *TIES, Revista de Tecnología e Innovación en Educación Superior*, 14, 35–47.
<https://doi.org/10.22201/dgtic.26832968e.2025.14.99>
- Charcopa, J. L. S. (2022). Sistema de registro de estudiantes para el proceso de matriculación online (Instituto Superior Técnico y Tecnológico de Esmeraldas). *Dominio de las Ciencias*, 8(2), 528–542.
<https://dominiodelasciencias.com/ojs/index.php/es/article/view/2701>
- Chiriboga Peña, P. M. (2024). Implementación de dashboards para la gestión académica y administrativa en instituciones de educación superior. *Revista Internacional de Investigación y Desarrollo Global (RIIDG)*, 3(4), 33–48.
<https://dialnet.unirioja.es/servlet/articulo?codigo=10461822>
- Delgado Olivera, C., & Díaz Alonso, L. (2021). Metodologías ágiles en el desarrollo de software. *Revista Cubana de Ciencias Informáticas*, 15(1), 1–15.
- Docker Inc. (s. f.). Docker documentation. <https://docs.docker.com/>
- GeeksforGeeks. (2025). LEMP Stack: Linux, Nginx, MySQL, PHP. GeeksforGeeks. <https://www.geeksforgeeks.org/>
- Git Project. (2025). Git documentation. <https://git-scm.com/doc>
- Herrera-Pérez, Y., Moya-Carrera, Y., Sánchez-Cali, F., & Tamayo, M. (2022). Sistema de gestión académica Ignug para la mejora del proceso de matrícula del Instituto Superior Tecnológico Yavirac. *Investigación, Tecnología e Innovación*, 14(15), 22–31.
<https://doi.org/10.53591/iti.v14i15.1605>
- Huamán-Yupanqui, N. C. (2024). Implementación de un sistema web para mejorar la gestión académica en universidades: un estudio de caso. *Revista Amazonía Digital*, 3(2), e289.
<https://revistas.unamad.edu.pe/index.php/rad/article/view/289>
- Laravel Holdings Inc. (2025). *Laravel documentation*.
<https://laravel.com/docs>
- León-Velarde, C. G., Solano, R., Lureña, C. L., Rivera, E. M., & Muro, R. G. (2024). Diseño de una aplicación web ASP.NET para la gestión de matrícula y registro de notas de estudiantes en una universidad nacional. *Aula Virtual*, 5(12), e307.
https://ve.scielo.org/scielo.php?script=sci_arttext&pid=S2665-03982024000202021
- López-Navarro, S. M., Guamán-Lumbisí, G. E., Analuiza-Analuiza, E. F., & Romero-Gaviláñez, F. R. (2023). Automatización de la gestión de matrícula y calificaciones del Instituto Superior Tecnológico “San Gabriel”. *RCTU*, 10(1), 10–21.
http://scielo.senescyt.gob.ec/scielo.php?script=sci_arttext&pid=S1390-76972023000100010
- Microsoft. (2025). Visual Studio Code documentation.
<https://code.visualstudio.com/docs>
- MySQL. (2025). MySQL documentation. <https://dev.mysql.com/doc>
- Navarro Cadavid, A., Fernández Martínez, L., & Morales Vélez, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 11(2), 30–39.
- PHP Documentation Group. (2025). *PHP manual*.
<https://www.php.net/manual/>
- Rizo Rabelo, N., Mendoza Otero, J. N., Concepción Morales, E. R., Beltrán Alonso, H., & Concepción Rizo, C. (2022). Sistema informático para la gestión de la admisión, matrícula y registro de créditos a los programas de formación doctoral de la Universidad de Cienfuegos. *Revista Conrado*, 18(S4), 593–600.
<https://conrado.ucf.edu.cu/index.php/conrado/article/view/2854>
- Ruiz-Muñoz, G. F., & Narváez Vega, E. A. (2024). Innovación y mejora continua en los procesos de matriculación en la educación superior. *Reincisol*, 3(5), 531–552.
[https://doi.org/10.59282/reincisol.V3\(5\)531-552](https://doi.org/10.59282/reincisol.V3(5)531-552)
- Subecz, Z. (2021). Web-development with Laravel framework. *Gradus*, 8(1), 211–218. https://gradus.kefo.hu/archive/2021-1/2021_1_CSC_006_Subecz.pdf
- Vargas, R., Aizprúa-A, S., Esquivel, C., & Delgado, J. L. (2025). Digitalización en el manejo de información estudiantil del CAIPI de la Universidad de Panamá. *Revista Científica Orbis Cognitionis*, 9(2), 142–157. <https://doi.org/10.48204/j.orbis.v9n2>
- Vázquez-Ingelmo, A., & Therón, R. (2020). Beneficios de la aplicación del paradigma de líneas de productos software para generar dashboards en contextos educativos. *RIED. Revista Iberoamericana de Educación a Distancia*, 23(2), 1–20.
<https://www.redalyc.org/articulo.oa?id=331463171009>
- Zunzunegui, S. (2023). Roles en Scrum: Product Owner, Scrum Master y Equipo de Desarrollo. *Scrum Manager*.

Anexo A Figura 7

Diagrama de la arquitectura Modelo-Vista-Controlador (MVC) del sistema



Nota. Se desglosa visualmente la interacción entre vistas, controladores y modelos en Laravel.

No.	Rol	Descripción Nota 1: favor de eliminar los corchetes una vez colocado su nombre Nota 2: elimine el todo texto de apoyo que se proporcionó en los asteriscos de cada fila y debajo de cada nombre.
1	Conceptualización [Conceptualization]	Francisco Eduardo Rodríguez Campos.
2	Curación de datos [Data curation]	Karla Mariana Baijen Saucedo.
3	Análisis formal [Formal analysis]	Lilia Parada Morado.
4	Obtención de fondos [Funding acquisition]	Rocío Lorena Rodríguez Chacón.
5	Investigación [Investigation]	Francisco Eduardo Rodríguez Campos. Lilia Parada Morado. Rocío Lorena Rodríguez Chacón.
6	Methodología [Methodology]	Francisco Eduardo Rodríguez Campos; Karla Mariana Baijen Saucedo
7	Administración del proyecto [Project administration]	Francisco Eduardo Rodríguez Campos.
8	Recursos [Resources]	Karla Mariana Baijen Saucedo
9	Software [Software]	Francisco Eduardo Rodríguez Campos; Karla Mariana Baijen Saucedo
10	Supervision [Supervision]	Rocío Lorena Rodríguez Chacón.
11	Validación [Validation]	Karla Mariana Baijen Saucedo
12	Visualización [Visualization]	Lilia Parada Morado.
13	Redacción - borrador original [Writing – original draft]	Francisco Eduardo Rodríguez Campos. Karla Mariana Baijen Saucedo
14	Redacción - revisión y edición [Writing – review & editing]	Lilia Parada Morado.