



**UCASAL**  

---

**UNIVERSIDAD CATÓLICA DE SALTA**

***"Desarrollo de un Sistema para  
Gestión de Becas, aplicando  
metodología RAD "***

**Fernández Sabaté, María del Milagro**

INGENIERÍA EN INFORMÁTICA

FACULTAD DE INGENIERÍA

AÑO 2023

**Título: “Desarrollo de un Sistema para Gestión de Becas, aplicando metodología RAD.”**

**Ingeniería en Informática**

**Profesor Guía:**

**Nombre: Lic. Aballay, Patricia**

**Firma:** \_\_\_\_\_

**Tribunal Evaluador:**

**Nombre:** \_\_\_\_\_

**Firma:** \_\_\_\_\_

**Nombre:** \_\_\_\_\_

**Firma:** \_\_\_\_\_

**Nombre:** \_\_\_\_\_

**Firma:** \_\_\_\_\_

**Alumna:**

**Nombre: Fernández Sabaté, María del Milagro Firma:** \_\_\_\_\_

**Fecha de Exposición:**

\_\_\_/\_\_\_/\_\_\_

## **Agradecimientos**

Quisiera expresar mi sincero agradecimiento a los docentes, por compartir sus conocimientos, por su dedicación, perseverancia y tolerancia. Agradecer a mis compañeros que me acompañaron a lo largo de la carrera. A mi profesora guía Patricia Aballay por su excelente dedicación y acompañamiento en este trabajo. Y, por último, a mi familia que estuvo siempre a mi lado apoyándome e impulsándome hasta el final.

## ÍNDICE

1.	INTRODUCCIÓN.....	9
	Justificación.....	9
2.	ESTADO DE LA CUESTIÓN.....	11
	2.2 Delimitación del objeto de estudio.....	11
3.	OBJETIVOS.....	12
	3.1 Objetivo general.....	12
	3.2 Objetivos específicos.....	12
4.	ALCANCE.....	13
5.	CRONOGRAMA.....	15
6.	ANÁLISIS DE RIESGOS.....	16
	6.1 Identificación de los Riesgos.....	16
	6.1.1 Riesgos Técnicos.....	16
	6.1.2 Riesgos Relacionados con la Planificación y el Cronograma.....	16
	6.1.3 Riesgos Relacionados con la Seguridad y Privacidad de Datos.....	16
	6.2 Evaluación de los Riesgos.....	17
	6.3 Gestión de los Riesgos.....	17
7.	ANTECEDENTES.....	19
8.	MARCO TEÓRICO.....	25
	5.1 Sistema de gestión de becas.....	25
	5.2 Desarrollo de aplicaciones.....	26
	5.3 Metodología RAD.....	27
	5.3.1 Fases RAD.....	28
	5.3.2 Beneficios de la implementación de metodología RAD.....	30
	5.3.3 Características de la metodología RAD.....	31
	5.3.4 Ventajas y desventajas en RAD.....	32
	5.3.5 Cuando utilizar los modelos Rad (Rapid Application Development).....	34

1.4.6	Prototipos .....	34
5.4	Administración de proyectos .....	35
5.4.1	Variables principales .....	36
5.5	Evaluación de proyectos .....	37
9.	DESCRIPCIÓN DEL SISTEMA ACTUAL.....	40
10.	METODOLOGÍA.....	41
7.1	Metodología de Investigación.....	41
7.1.1	Método Análisis.....	41
7.1.2	Técnicas de recolección de información.....	41
7.2	Metodología de Desarrollo RAD.....	41
7.2.1	Planificación (Modelado de la Gestión) .....	41
7.2.2	Diseño (Modelado de datos).....	43
7.2.3	Diseño (Modelado de proceso).....	43
7.2.4	Construcción (Generación de Aplicación).....	44
7.2.5	Implementación (Pruebas de entrega) .....	46
11.	DESARROLLO.....	47
8.1	Fase de Planificación de Requisitos .....	47
8.1.1	Requerimientos Funcionales:.....	47
8.1.2	Requerimientos No Funcionales:.....	48
8.2	Estudio de Factibilidad .....	48
8.2.1	Viabilidad Técnica.....	48
8.2.2	Viabilidad Económica .....	49
8.2.3	Viabilidad Operativa.....	49
8.2.4	Viabilidad Legal .....	50
8.3	Fase de Diseño (Modelado de datos y de procesos) .....	50
8.3.2	Modelado de Datos .....	50
8.2.1	Diseño de Procesos .....	55

8.3 Fase de Construcción.....	84
8.3.1 Implementación de la arquitectura lógica del sistema.....	84
8.3.2 Diseño y construcción de la base de datos.....	84
8.3.3 Desarrollo de los módulos y funcionalidades del sistema .....	86
8.3.4 Interfaces del sistema.....	91
8.4 Fase de Implementación .....	97
8.4.1 Requisitos mínimos de Hardware para la Implementación .....	97
8.4.2 Requisitos mínimos de Software para la Implementación .....	97
8.4.3 Proceso de Implementación .....	97
8.4.4 Pruebas de QA .....	98
12. Conclusión.....	101
13. Bibliografía.....	102
Anexos .....	105

## ÍNDICE DE ILUSTRACIONES, GRÁFICOS Y TABLAS

Ilustración 1: Fases RAD (Diseño propio) .....	29
Ilustración 2: Importancia de la evaluación de proyectos. Elaboración propia en base a (Baca Urbina, 2013).....	38
Ilustración 3: Diagrama Entidad-Relación General.....	54
Ilustración 4: Diagrama Entidad-Relación del Módulo de Seguridad.....	55
Ilustración 5: Flujo de pantallas Modulo de Administración. Elaboración propia. .....	67
Ilustración 6: Flujo de pantallas Portal del Alumno. Elaboración propia.....	68
Ilustración 7: Creación tabla Becas .....	85
Ilustración 8: Tabla de Becas.....	85
Ilustración 9: Creación tabla Solicitudes .....	86
Ilustración 10: Tabla de Solicitudes .....	86
Ilustración 11: Pantalla de Login.....	86
Ilustración 12: Pantalla de carga/edición de usuarios.....	87
Ilustración 13: Código PL/SQL para el proceso Agregar Usuario .....	88
Ilustración 14: Pantalla de Listado de Becas Otorgadas.....	88
Ilustración 15: Pantalla de Listado de Becas .....	89
Ilustración 16: Código PL/SQL para consumo de API GetCarreras .....	90
Ilustración 17: Pantalla de Login.....	91
Ilustración 18: Pantalla de Gestión de Becas.....	92
Ilustración 19: Pantalla de Listado de Solicitudes.....	92
Ilustración 20: Pantalla de Aprobación.....	93
Ilustración 21: Pantalla de Edición de datos de Contacto.....	93
Ilustración 22: Pantalla de listado de becas .....	94
Ilustración 23: Pantalla de Listado de Solicitudes.....	94
Ilustración 24: Pantalla de Becas Otorgadas .....	94
Ilustración 25: Pantalla de Usuarios .....	95
Ilustración 26: Pantalla de Roles .....	95
Ilustración 27: Pantalla de Listado de becas disponibles.....	96
Ilustración 28: Pantalla de Solicitudes de Becas .....	96

Gráfico 2: Triángulo de la Triple Restricción. Elaboración propia en base a (Ander Egg, 2010) ..... 36

Tabla 1: Cronograma del Proyecto. Elaboración propia ..... 15

Tabla 2: Matriz de Riesgos. Elaboración propia. .... 17

Tabla 3: Tabla de Gestión de Riesgos. Elaboración propia..... 18

Tabla 4: Modelado de Gestión. Elaboración propia ..... 42

Tabla 5: Modelado de datos. Elaboración propia ..... 43

Tabla 6: Modelado de Proceso. Elaboración propia ..... 43

Tabla 7: Construcción de Aplicaciones. Elaboración propia..... 45

Tabla 8: Implementación. Elaboración propia..... 46

## **1. INTRODUCCIÓN**

El desarrollo de un sistema de gestión de becas con la metodología RAD (Rapid Application Development) es un proceso crítico para asegurar la eficiencia y efectividad en la gestión de becas. La metodología RAD se ha convertido en una herramienta popular en el desarrollo de sistemas, especialmente en proyectos que requieren una entrega rápida y un enfoque iterativo en el desarrollo.

La gestión de becas es un proceso que requiere una gran cantidad de información y recursos, incluyendo la evaluación de solicitudes de becas, la selección de candidatos y la asignación de fondos. Con el uso de un sistema de gestión de becas, se puede automatizar y simplificar este proceso, lo que resulta en una gestión más eficiente y efectiva de las becas.

El objetivo de desarrollar un sistema de gestión de becas con la metodología RAD es garantizar una entrega rápida y un enfoque iterativo en el desarrollo, que permita a los usuarios tener una visión clara del progreso del proyecto y hacer cambios en tiempo real. La metodología RAD también permite una comunicación más efectiva entre los desarrolladores y los usuarios, que ayuda a garantizar que el sistema cumpla con los requisitos y expectativas del usuario.

El proceso de desarrollo de un sistema de gestión de becas con la metodología RAD se divide en cuatro fases principales: análisis, diseño, construcción e implementación. Durante la fase de análisis, se identifican los requisitos y expectativas del usuario para el sistema. Durante la fase de diseño, se crea un plan detallado para el desarrollo del sistema, incluyendo la estructura y la funcionalidad. Durante la fase de construcción, se desarrollan las funciones y se integran los componentes del sistema. Finalmente, durante la fase de implementación, se prueba y se entrega el sistema a los usuarios.

Un sistema de gestión de becas con metodología RAD atiende a varias necesidades clave en el proceso de gestión de becas. Algunas de estas necesidades incluyen:

### **Justificación**

La implementación de un sistema de gestión de becas con metodología RAD en una universidad es crucial para garantizar una gestión eficiente, justa y transparente de las becas. La metodología RAD es un enfoque innovador que utiliza tecnologías

avanzadas para acelerar el desarrollo de sistemas, y su aplicación en la gestión de becas puede mejorar significativamente el proceso.

Entre las mejoras de una implementación de este tipo, se puede señalar lo siguiente:

- **Automatización de procesos:** automatiza procesos críticos en la gestión de becas, como la evaluación de solicitudes, la selección de candidatos y la asignación de fondos. Esto reduce el tiempo y los esfuerzos manuales necesarios para llevar a cabo estos procesos.
- **Mejora de la eficiencia:** Al automatizar los procesos, un sistema de gestión de becas aplicando metodología RAD, mejora la eficiencia en el proceso de desarrollo del sistema.
- **Transparencia:** proporciona una mayor transparencia en el proceso de gestión de becas. Los usuarios pueden tener una visión clara de cómo se están utilizando los fondos y qué solicitudes han sido seleccionadas.
- **Reducción de errores:** Al automatizar procesos, para el proceso de análisis, diseño y desarrollo, reduce la probabilidad de errores humanos.
- **Facilidad de uso:** La metodología RAD enfoca en la iteración y la retroalimentación de los usuarios, lo que garantiza que el sistema sea fácil de usar y cumpla con las necesidades y expectativas de los usuarios.
- **Mejora de la efectividad:** puede mejorar la efectividad en el diseño y desarrollo del sistema.

Por ello, un sistema de gestión de becas con metodología RAD atiende a las necesidades críticas en el proceso de gestión de becas, incluyendo la automatización de procesos, la mejora de la eficiencia, la transparencia, la reducción de errores, la facilidad de uso y la mejora de la efectividad.

## **2. ESTADO DE LA CUESTIÓN**

La implementación de un sistema informático para gestión de becas se ha convertido en una necesidad cada vez más relevante en la actualidad, ya que la cantidad de solicitudes de becas y la complejidad de los procesos de selección y asignación de fondos han aumentado significativamente en los últimos años.

El uso de la metodología RAD para el desarrollo de sistemas informáticos se ha demostrado efectivo en términos de mejora de la eficiencia, agilidad en procesos, transparencia y reducción de errores. Además, esta metodología permite el desarrollo rápido y eficiente de sistemas, lo que es esencial en un entorno en constante evolución y cambio.

Sin embargo, el desarrollo de un sistema, utilizando metodología RAD presenta algunos desafíos, como la integración de los sistemas existentes, la necesidad de capacitación y la adaptación de los usuarios, y la protección de los datos y la privacidad de los usuarios.

A pesar de estos desafíos, el diseño e implementación de sistemas con metodologías ágiles, como la metodología RAD, se ha convertido en una tendencia creciente y se espera que continúe creciendo en el futuro. La combinación de tecnologías avanzadas y la metodología RAD puede ayudar a mejorar significativamente la eficiencia, la transparencia en el desarrollo de sistemas.

### **2.2 Delimitación del objeto de estudio**

En cuanto a la delimitación espacial, el diseño e implementación de un sistema de gestión de becas con metodología RAD se enmarca en el ámbito de una universidad privada, en este caso específico, la Universidad Católica de Salta, sede Castañares, de la ciudad de Salta Capital. En el contexto de diseñar una propuesta tecnológica que, mediante un sistema de gestión de becas, brinde posibles soluciones en cuanto a la gestión de las becas otorgadas por esta casa de altos estudios. El periodo de vigencia es el primer semestre del año 2023.

### **3. OBJETIVOS**

#### **3.1 Objetivo general**

- Análisis, diseño, desarrollo e implementación inicial de un sistema web de gestión de becas aplicando metodología RAD, que brinde soluciones a la Universidad Católica de Salta, al momento de establecer relaciones y perfiles adecuados para su asignación o beneficio, en el primer semestre del año 2023.

#### **3.2 Objetivos específicos**

- Conceptualizar a los sistemas de gestión y la metodología RAD.
- Indagar sobre los beneficios que brinda el diseño de un sistema web de gestión de becas con la utilización de la metodología RAD.
- Automatizar y mejorar procesos existentes en el sistema de gestiones de becas de la Universidad Católica de Salta.
- Desarrollar y diseñar herramientas de análisis, como especificaciones de casos de uso, diagrama de clases, flujo de pantallas y diseño de interfaz.
- Desarrollar el sistema web utilizando tecnologías Low-Code (bajo código).
- Realizar pruebas de QA para asegurar la calidad del software, permitiendo evitar errores.
- Mostrar los resultados del diseño e implementación inicial de un sistema web de gestión de becas con metodología RAD en la Universidad Católica de Salta.

#### 4. ALCANCE

Este proyecto de grado, se enfocará en las siguientes áreas claves:

- **Análisis Funcional:** Se llevará a cabo una fase de análisis funcional, que comprenderá identificación y análisis exhaustivo sobre los requisitos y necesidades del Departamento de Becas de la UCASAL.
- **Diseño del sistema:** se diseñará un sistema de gestión de becas basado en las mejores prácticas de desarrollo y adaptado a las necesidades identificadas en el punto anterior. Este sistema debe ser intuitivo y fácil de usar. Esta etapa del diseño tendrá como resultado el diagrama entidad-relación (DER), definición y especificaciones de casos de uso, flujo de pantallas y diseño de interfaces.
- **Desarrollo del sistema:**
  - La aplicación se desarrollará utilizando Oracle Application Express (APEX) como herramienta principal de desarrollo. Se empleará una base de datos Oracle Database 19C para el almacenamiento de datos, con PL/SQL como lenguaje de programación primario y SQL para consultas de datos.
  - Implementación de funcionalidades claves como alta de becas, gestión de solicitudes, módulo de seguridad, reportes de becas solicitadas y otorgadas y un Portal del Alumno para visualizar, aplicar y dar seguimiento a becas solicitadas.
- **Pruebas y Documentación:** se documentarán las pruebas realizadas sobre los casos de uso definidos, lo que asegurará la correcta funcionalidad y calidad del software en el primer prototipo.

#### **Delimitaciones:**

- El alcance del proyecto se limita a la Universidad Católica de Salta, sede Castañares, en la ciudad de Salta Capital.
- El sistema se desarrollará en su primer prototipo, con un enfoque específico en la etapa de análisis funcional, diseño y desarrollo, utilizando la metodología RAD.
- Este proyecto no incluye la implementación del sistema en la UCASAL, se limita a la fase de diseño, desarrollo y validación.

- El proyecto no incluye el código completo del sistema, solo fragmentos que hacen referencia a funciones o procedimientos relevantes para el sistema.
- Tampoco incluye la elaboración de un manual de usuario.

**Suposiciones:**

- Se asume que la UCASAL proveerá las APIs requeridas para una correcta integración del sistema de gestión de becas con el sistema SAG.
- Se asume que el sistema se utilizará en un entorno Web estándar, accesible desde cualquier navegador moderno.

## 5. CRONOGRAMA

El cronograma es una parte fundamental de cualquier proyecto, ya que es la herramienta que permite visualizar y organizar las actividades de manera secuencial a lo largo del tiempo. En esta sección del documento, se presenta de manera detallada y estructurada todas las tareas, hitos y eventos que forman parte del proyecto.

*Tabla 1: Cronograma del Proyecto. Elaboración propia*

Cronograma del proyecto		2023				
		Marzo	Abril	Mayo	Junio	Julio
1	Recolección de Información					
2	Planteamiento del Problema					
3	Planteamiento de Objetivos					
4	Metodología					
5	Marco Teórico					
6	Análisis de Datos y Resultados					
7	Desarrollo de Software					
8	Pruebas QA					
9	Conclusiones					
10	Documentación					

## 6. ANALISIS DE RIESGOS

### 6.1 Identificación de los Riesgos

Dado que este proyecto se caracteriza por su enfoque tecnológico e innovador, no podemos pasar por alto la posibilidad de que surjan ciertos riesgos que podrían comprometer de manera significativa los objetivos que pretendemos alcanzar. En esta etapa inicial de la sección, procederemos a identificar y enumerar estos riesgos. Los presentaremos en la lista a continuación, organizados de acuerdo a las áreas o ámbitos de aplicación relevantes.

#### 6.1.1 Riesgos Técnicos

**Integración con sistemas existentes:** La integración con el sistema SAG empleado por la Universidad Católica de Salta a través de APIS, puede generar problemas de integración de datos originados por diferencias en formatos y estructuras de datos. Los problemas en la integración pueden causar retrasos en el proyecto y afectar su calidad.

#### 6.1.2 Riesgos Relacionados con la Planificación y el Cronograma

**Dependencia de terceros:** Este riesgo está asociado a proveedores externos, en este caso el departamento de IT de la Universidad, quienes pueden sufrir demoras en la entrega de las APIS necesarias para la integración por diferentes factores, por ejemplo, que no se encuentren desarrolladas o falta de personal para realizarlas.

**Desviaciones en el cronograma:** Los retrasos en cualquier fase del proyecto podrían afectar el cronograma previsto y causar demoras en la entrega del sistema.

#### 6.1.3 Riesgos Relacionados con la Seguridad y Privacidad de Datos

**Violación de la seguridad de datos:** Dado que se manejarán datos sensibles de estudiantes, existe el riesgo de violaciones de seguridad o fugas de datos si no se implementan medidas de seguridad adecuadas.

**Incumplimiento Normativo:** El incumplimiento de regulaciones y normativas puede llevar a sanciones legales y problemas de reputación. Esto es especialmente crítico en proyectos relacionados con datos sensibles o regulados.

## 6.2 Evaluación de los Riesgos

Para evaluar los riesgos, se utilizará una matriz de riesgos que combina la probabilidad de ocurrencia y el impacto en el proyecto. El impacto de hacerse efectivo cada riesgo en el proyecto, se clasificarán con la siguiente escala:

1. Impacto Bajo
2. Impacto Medio
3. Impacto Alto
4. Impacto Crítico

*Tabla 2: Matriz de Riesgos. Elaboración propia.*

Denominación del Riesgo	Impacto	Probabilidad	Riesgo
Integración con sistemas existentes	3	25%	0.75
Dependencia de terceros	1	40%	0.4
Desviaciones en el cronograma	1	50%	0.5
Violación de la seguridad de datos	4	10%	0.4
Incumplimiento Normativo	3	5%	0.15

## 6.3 Gestión de los Riesgos

Luego de haber ponderado el grado de exposición al riesgo de cada uno, lo siguiente es determinar las acciones a realizar con el objetivo de mitigarlos, transferirlos o simplemente aceptarlos. A continuación, se describe brevemente estos conceptos:

- **Mitigación:** Identificar medidas preventivas y estrategias para reducir la probabilidad y el impacto de los riesgos.
- **Transferencia:** Determinar si algunos riesgos pueden ser transferidos a terceros mediante acuerdos de seguro o contratos.
- **Aceptación:** Decidir si algunos riesgos son aceptables y deben ser monitoreados sin medidas adicionales

*Tabla 3: Tabla de Gestión de Riesgos. Elaboración propia*

<b>Denominación del Riesgo</b>	<b>Mitigación</b>	<b>Contingencia</b>
Integración con sistemas existentes	Investigar si el sistema actual ofrece alguna API de integración en un lenguaje estándar (JSON o XML). En caso de que no exista, se lo solicitará al departamento de IT de la UCASAL.	Preparar un plan de contingencia que incluya alternativas de integración y un cronograma revisado en caso de problemas graves.
Dependencia de terceros	Solicitar las APIs con la debida antelación. Los riesgos de demora por falta de personal para desarrollarlas o adaptarlas se transfiere al departamento de IT.	Preparar un cronograma que incluya demoras ocasionadas por terceros y comunicar debidamente a las partes interesadas.
Desviaciones en el cronograma	Utilizar metodologías ágiles que permitan la adaptación a cambios. Por otro lado, se debe realizar estimaciones realistas y revisar el cronograma periódicamente.	Tener un plan de contingencia que incluya la posibilidad de ajustar el alcance, aumentar los recursos o reorganizar tareas para recuperar el tiempo perdido.
Violación de la seguridad de datos	Implementar medidas de seguridad adecuadas, como cifrado y control de accesos, tanto perimetrales como en los puestos de trabajo y servidores, así como también implementar una política de seguridad de la información.	Tener un plan de respuesta a incidentes de seguridad que incluya la notificación a las partes afectadas y las autoridades si es necesario.
Incumplimiento Normativo	Al poseer una probabilidad de ocurrencia muy baja y al tener un riesgo bajo sobre el proyecto, se acepta la ocurrencia del mismo.	-

## 7. ANTECEDENTES

1) Los autores Molina Granja y Bonilla Cadena (2022) elaboraron un trabajo investigativo con el título de “Desarrollo de una plataforma web para recorridos virtuales 360° mediante la metodología RAD. Caso Grupo Inmobiliario Horizonte de la ciudad de Riobamba.” En la ciudad de Riobamba, Ecuador.

**Desarrollo:** La pandemia Covid-19 declarada en diciembre de 2019 transformó el estilo de vida en todo el mundo, se paralizaron escuelas, colegios, universidades, empresas, etc. Desde entonces la tecnología se convirtió en el mejor aliado de las personas, por un lado, se crearon nuevos softwares para teletrabajo y por otro se aprovecharon las que existían para educación virtual. Un área importante que ha quedado sin progreso tecnológico es el área inmobiliaria. Por ello, se desarrolla una plataforma web para recorridos virtuales 360° mediante la metodología RAD para el Grupo Inmobiliario Horizonte de la ciudad de Riobamba. De esta forma el proyecto propuesto se centra en el beneficio de los trabajadores y usuarios para fomentar la labor, ofreciendo nuevas tecnologías como son los recorridos virtuales implementados en una página web donde se proporciona al usuario una mejor visualización de la vivienda.

**Metodología:** El tipo de investigación del presente proyecto se basó en una investigación exploratoria, donde se analizó un tema poco estudiado en el ámbito inmobiliario. Dentro de la investigación exploratoria se centró específicamente en el diseño cuasiexperimental caracterizado por estudiar a un grupo de personas no aleatorias frente a diversas situaciones.

La técnica que se utilizó se basa en la observación de campo, para ello se realizó una inspección al personal encargado del asesoramiento de ventas y secretaria, dónde fueron observados y evaluados por medio de un test de usabilidad sobre la interacción que obtuvieron con el sistema. La empresa Grupo Inmobiliario Nuevo Horizonte cuenta con un personal de ocho miembros en total, los seleccionados fueron cuatro trabajadores, tres asesores de ventas y una secretaria, los mismos que determinan la muestra de estudio y quienes evaluaron los correspondientes análisis sobre la usabilidad.

**Resultados:** Para los recorridos virtuales se obtuvieron tomas fotográficas en 360° de los inmuebles disponibles en los catálogos y para el desarrollo del software se aplicó la metodología de modelo de desarrollo rápido de aplicaciones (RAD) con sus cuatro fases. Una vez implementado el sistema web se procede a evaluar mediante

parámetros de la usabilidad, observación de campo y la aplicación de test SUS para medir la experiencia de usuario (UX) de los trabajadores encargados de evaluar el sistema.

El resultado del test de usabilidad aplicado con un resultado de 85,63 % ayuda a conocer que tan agradable e intuitiva es la experiencia del usuario al navegar por la web, es decir, lo fácil que resulta usar la página web. Para desarrollar una página se toman en cuenta los principios de usabilidad de Jakob Nielsen, estas se resumen en diez reglas heurísticas que ayudan a crear un ámbito agradable y comprensible para el usuario.

**Conclusiones:** Con el test concretado se obtuvo un resultado de 85,62 puntos sobre 100 puntos, logrando una calificación A en la satisfacción del usuario con la página web y la inserción de recorridos virtuales. El uso de la metodología RAD es caracterizada por las retroalimentaciones constantes con el usuario, se recomienda realizar al menos una en cada fase del proyecto, ayudando a reducir errores.

Para la creación de recorridos virtuales se recomienda realizar las tomas fotográficas con equipos especializados esto aporta una mayor calidad a las imágenes y una mejor visualización en el recorrido virtual, se debe tomar en cuenta las dimensiones de las imágenes para evitar una distorsión del entorno. Con respecto al alojamiento de la página web en un servidor, se debe adquirir uno con un almacenamiento superior de 5GB, puede ser gratuito o en línea. La limitación de todo servidor online es el tiempo de duración de la publicación en página web. Se recomienda, si se requiere una mayor duración, contratar un hosting.

2) La investigadora Solano Canto (2021) realizó un trabajo académico denominado “Aplicación de la metodología Bpm: Rad para la mejora de la calidad en la atención del proceso gestión de biblioteca de la Facultad de Ingeniería de Sistemas.” En la Universidad del Centro en la ciudad de Huancayo, Perú.

**Desarrollo:** surge como respuesta al análisis y caracterización de la Biblioteca Especializada como proceso de apoyo a la Formación profesional, Investigación y Proyección Social de los grupos de interés; asimismo presenta diversas connotaciones tales como: la gestión por procesos, operatividad del Sistema de Gestión de Calidad ISO 9001:2008 y el cumplimiento de estándares para la próxima acreditación de la calidad universitaria, los cuales dan origen a la necesidad de enfrentar los requerimientos de mejora y continuidad del proceso. Este trabajo de investigación se desarrolla en base a la problemática general ¿De qué manera influye la aplicación de la metodología BPM:

RAD en la mejora de la calidad en la atención del proceso gestión de Biblioteca de la Facultad de Ingeniería de Sistemas?

**Metodología:** En el desarrollo de la tesis, se aplicó la metodología BPM:RAD (Rapid Analysis & Design) como una alternativa ágil y dinámica orientada al análisis, modelización, diseño y automatización de procesos de negocio, asimismo la implementación del aplicativo web fue llevada a cabo mediante el uso del lenguaje de programación PHP y teniendo como gestor de base de datos MySQL. Los fundamentos teóricos son los relacionados a bibliotecología, ciencia social que estudia la gestión integral de bibliotecas reales o virtuales, considerando recursos humanos, materiales, financieros y tecnológicos.

**Resultados:** En los resultados se encontró, que la metodología BPM:RAD es la adecuada para un sistema de gestión de bibliotecas dentro de una universidad, brindando rapidez y procesos ágiles tanto en su diseño como en su implementación. Permitiendo que alumnos de distantes facultades logren acceder sin mayores inconvenientes al sistema, encontrando rápidamente lo que buscaban o solicitaban.

**Conclusiones:** Durante el desarrollo de la tesis surgieron problemas relacionados a la perspectiva y contexto de funcionamiento del proceso Gestión Biblioteca, las brechas relacionadas al cómo funciona y cómo debería funcionar y la orientación al cliente. Finalmente, se concluye que la aplicación de la metodología BPM: RAD influye positivamente para la mejora de la calidad en la atención del proceso Gestión de Biblioteca de la Facultad de Ingeniería de Sistemas y se recomienda otorgar la importancia y dar el énfasis necesario a este proceso durante la formulación del PEI y POI, asimismo considerar al usuario como parte esencial de la mejora de la calidad del Proceso Gestión Biblioteca con el fin de fomentar su continuidad.

3) Los autores Macías Rodríguez, Molina Chalacán y Jalón Arias (2021) llevaron a cabo una investigación titulada “Aplicativo web para la búsqueda y ubicación de libros en bibliotecas de Instituciones Escolares” en la ciudad de Quito, Ecuador. Encontrando que:

**Desarrollo:** La investigación tiene como objetivo principal implementar un aplicativo web que permita mejorar la búsqueda, ubicación, de los libros de forma sistematizada, organizada, en instituciones públicas y privadas del Ecuador. Es evidente que en los últimos años la tecnología ha incrementado de una manera extraordinaria, de modo que ha causado un gran impacto en diferentes sectores, además, ha formado parte

fundamental dentro del ámbito educativo, a tal punto que se han desarrollado plataformas virtuales que son capaces de buscar o indagar información dependiendo de las necesidades del usuario, incorporando así modelos de investigación para lograr formar individuos con espíritu investigador.

En el Ecuador la utilización de esta herramienta está avanzando continuamente, en función de que diversas instituciones tanto públicas como privadas, cuentan con bibliotecas virtuales donde se almacena información importante como artículos, papers, tesis, proyectos investigativos, revistas científicas, los cuales la mayoría se encuentran registrados digitalmente con su respectivo código, título, autor, editorial, año de publicación, idioma, y observación, de modo que el usuario o estudiante mediante la red puede acceder a documentos desde cualquier lugar en busca de información

**Metodología:** Se generó desde una concepción metodológica descriptiva documental con diseño no experimental, generando un aplicativo web, el método utilizado para el aplicativo web de búsqueda y ubicación de libros en biblioteca es la metodología RAD (Desarrollo Rápido de Aplicaciones).

**Resultados:** Se desarrolló la aplicación web para la búsqueda y ubicación de libros donde se evidenció que posee una interfaz más amigable y llamativa, un flujo de código y estructura más eficiente, y reportes con mayor comprensión. Las pruebas implementadas de implementar el aplicativo web, indican que su funcionamiento es eficiente y adecuado, además permite una óptima búsqueda de los libros que el usuario así lo requiera o necesite, de tal forma que ayuda a ahorrar el factor tiempo.

El diseño y elaboración de la biblioteca virtual, concuerda con la estructura elaborada, posee una interfaz amigable y entendible, el cual permite realizar búsqueda de forma eficiente y segura. se ofrece una herramienta tecnológica como un método de ayuda, por medio de la cual los usuarios y/o estudiantes podrán investigar y navegar sin algún tipo de limitaciones o barreras y puedan obtener toda la información que así lo requieran. La biblioteca virtual se asume como un producto y una herramienta tecnológica que tendrá el propósito de mostrar y divulgar sus materiales bibliográficos.

**Conclusiones:** Se desarrolló la aplicación web para la búsqueda y ubicación de libros donde se evidenció que posee una interfaz más amigable y llamativa, un flujo de código y estructura más eficiente, y reportes con mayor comprensión.

Las pruebas implementadas de implementar el aplicativo web, indican que su funcionamiento es eficiente y adecuado, además permite una óptima búsqueda de los

libros que el usuario así lo requiera o necesite, de tal forma que ayuda a ahorrar el factor tiempo. Otro punto importante es que el aplicativo web tiene múltiples beneficios donde el usuario puede registrarse de forma gratuita (sin costo alguno), además cuenta con una gran variedad de libros donde pueden solicitarlo a través del admin o el bibliotecario encargado.

4) El investigador Gomez Zea (2019) realizó su trabajo titulado " Implementando scrum + rad para la gestión y desarrollo de proyectos de software en equipos de trabajo con personal limitado y eventual". En el departamento de desarrollo del Instituto Tecnológico de Villahermosa, en Tabasco, México

**Desarrollo:** Las metodologías de desarrollo ágil han transformado la manera de construir software; de métodos rígidos y lentos a procesos flexibles y en dientes. Rad y Scrum son marcos de trabajo ágil que proponen un conjunto de prácticas y roles para: elaborar software de calidad, obtener resultados anticipados, dar flexibilidad y adaptación, permitir el retorno de inversión, promover la productividad; entregar con frecuencia software que funcione, entre otros. Este artículo propone un conjunto de buenas prácticas (combinación de la metodología scrum y el desarrollo rápido de aplicaciones rad) para el análisis, diseño y desarrollo de proyectos de software de calidad en períodos menores a 60 días, aplicado en entornos con poco personal o personal eventual. Además, se plantean variables y elementos para su óptimo funcionamiento.

**Metodología:** La metodología empleada en esta investigación fue un estudio de caso complementada marginalmente con la investigación experimental, analizando variables de: Técnicas y estándares, Escenarios de trabajo, Horarios y cantidad de integrantes, Herramientas, Tamaño y tiempo de proyectos, Artefactos, Pruebas y Documentación.

**Resultados:** La implementación se realizó en el departamento de centro de cómputo del Instituto Tecnológico de Villahermosa, en el estado de tabasco, México. En este período se contaba con un jefe de departamento y un apoyo de soporte técnico. Dentro de sus funciones están establecidas actividades de: Administración de servicios de red, diseño y administración de sistemas de información, soporte técnico y de mantenimiento a los equipos tecnológicos, administración del centro de datos, entre otros. Las solicitudes de nuevos sistemas de información fueron en aumento de 2 a 3 sistemas por semestre, determinado por los eventos académicos, sociales y culturales de

la institución. Los sistemas solicitados requieren al menos procesamiento de información a una base de datos, envío de emails, administración de usuarios, difusión de registros del evento, reportes estadísticos, impresión y funcionalidad para su acceso desde el internet.

**Conclusiones:** Las metodologías ágiles de gestión y desarrollo de software proponen una filosofía adaptable, flexible y escalable para integrarlas en variados escenarios de trabajo. Además, ofrecen beneficios de productividad y eficiencia sin perder calidad en los productos. Este artículo propone mecanismos, herramientas y técnicas que permiten a las instituciones y empresas con escaso personal o personal eventual, establecer un área de desarrollo de software, apoyado de practicantes y/o residentes, integrando elementos de la metodología ágil scrum combinada con la metodología de desarrollo rad, que le permitan la construcción de sistemas de información de calidad, en periodos cortos de tiempo. En este contexto, se requiere del esfuerzo inicial del personal de base el cual debe considerar el aprendizaje de un marco de trabajo robusto con una curva de aprendizaje corta, cubriendo al principio los elementos de estandarización de los módulos o códigos, adaptación de un escenario adecuado para la ejecución de la metodología, reclutamiento de practicantes y residentes, dominar un conjunto de herramientas case de desarrollo y conocer la filosofía y técnicas de las metodologías.

## **8. MARCO TEÓRICO**

### **5.1 Sistema de gestión de becas**

Terrones Melgarejo (2021) define que un sistema de gestión de becas es un software que se utiliza para administrar y automatizar el proceso de solicitud, selección y gestión de becas. Este sistema puede incluir la captura de información de los solicitantes, la evaluación de solicitudes y la selección de beneficiarios, así como la gestión de la distribución y seguimiento de fondos de becas.

Continuando con Terrones Melgarejo (2021) el sistema puede ser utilizado por una variedad de organizaciones, incluyendo universidades, fundaciones y grupos sin fines de lucro. Puede ayudar a agilizar y simplificar el proceso de becas, asegurando una gestión más eficiente y una toma de decisiones más justa y objetiva.

Asimismo, desde el análisis de Cisneros Gutiérrez (2015) un buen sistema de gestión de becas debe ser fácil de usar, intuitivo y ofrecer una experiencia de usuario amigable para los solicitantes y los administradores. Debe ser seguro, confiable y proteger la información personal de los solicitantes. Además, debe ser escalable para manejar una gran cantidad de solicitudes y tener la capacidad de integrarse con otros sistemas relevantes.

La finalidad principal de un sistema de gestión de becas es automatizar y simplificar el proceso de solicitud, selección y administración de becas, lo que permite una gestión más eficiente y una toma de decisiones más justa y objetiva. Algunos de los objetivos específicos de un sistema de gestión de becas incluyen:

- Agilizar el proceso de solicitud: Un sistema de gestión de becas permite a los solicitantes ingresar y enviar sus solicitudes de manera rápida y eficiente, lo que agiliza el proceso de solicitud.
- Evaluación objetiva: Un sistema de gestión de becas permite la evaluación objetiva de las solicitudes de becas, lo que reduce la posibilidad de errores o sesgos en la toma de decisiones.
- Distribución justa de fondos: Un sistema de gestión de becas permite la distribución equitativa de los fondos de las becas, asegurando que los solicitantes más calificados reciban las becas que necesitan.
- Seguimiento efectivo: Un sistema de gestión de becas permite un seguimiento efectivo de los beneficiarios de las becas, lo que garantiza que se estén utilizando adecuadamente.

- Mejora de la transparencia: Un sistema de gestión de becas aumenta la transparencia en el proceso de solicitud y administración de becas, lo que aumenta la confianza de los solicitantes y los administradores (Cisneros Gutiérrez, 2015, págs. 58-59).

Concluyendo, un sistema de gestión de becas tiene como objetivo simplificar y automatizar el proceso de solicitud, selección y administración de becas, garantizando una gestión más eficiente, una toma de decisiones más justa y objetiva, y un seguimiento efectivo de los fondos de las becas (Terrones Melgarejo, 2021).

## **5.2 Desarrollo de aplicaciones**

Pressman (2010) señala que el desarrollo de aplicaciones es el proceso de creación de software con un propósito específico. Las aplicaciones pueden ser de uso personal o empresarial y pueden ejecutarse en una variedad de plataformas, incluidas computadoras de escritorio, dispositivos móviles y servidores web.

Mientras tanto, Mesa Casado (2022) sostiene que el desarrollo de aplicaciones implica varios pasos, incluyendo la definición de los requisitos del usuario, el diseño de la aplicación, la implementación del código, la prueba y el despliegue. También puede incluir la mantención y mejora de la aplicación a lo largo del tiempo.

A su vez, Castillo (2021) refiere que el desarrollo de aplicaciones se realiza por un equipo de desarrolladores de software, que pueden incluir programadores, diseñadores de interfaces de usuario, ingenieros de pruebas y especialistas en seguridad de la información.

Existen diferentes metodologías de desarrollo de aplicaciones, como Agile, Waterfall, RAD y DevOps, que dictan cómo se deben abordar y ejecutar los diferentes pasos en el proceso de desarrollo. Pressman (2010) refiere que la metodología elegida depende de varios factores, como el tamaño y la complejidad del proyecto, el tiempo de entrega y los recursos disponibles.

El modelo de desarrollo rápido de aplicaciones (RAD, por sus siglas en inglés) es una técnica ágil de desarrollo de software que da prioridad a las entregas e iteraciones rápidas de prototipos. A diferencia de la metodología de cascada, RAD tiene más en cuenta el uso del software y la opinión del usuario que la planificación rigurosa y el registro de los requisitos (Mena, 2015).

### **5.3 Metodología RAD**

El desarrollo rápido de aplicaciones o RAD (acrónimo en inglés de rapid application development), concebido en la década de 1970 pero presentado oficialmente por James Martin en 1991, es una metodología que se centra en desarrollar aplicaciones rápidamente por medio de iteraciones frecuentes y aprobaciones con comentarios continuos de los clientes. Comprende el desarrollo interactivo, la construcción de prototipos y el uso de utilidades CASE (Computer Aided Software Engineering). Al priorizar los lanzamientos de prototipos ágiles y rápidos, RAD incide en la usabilidad del software, los comentarios de los usuarios y la entrega rápida a través de una planificación a largo plazo y un único conjunto de requisitos iniciales para la creación de elementos, como las aplicaciones personalizadas. Gracias a su rapidez y agilidad, la popularidad de RAD va en aumento (Martin, 1991).

Mena (2015) refiere que el modelo de desarrollo rápido de aplicaciones (RAD) es un modelo de desarrollo que promueve la creación rápida de prototipos y la retroalimentación inmediata en lugar de ciclos de desarrollo y pruebas largos y prolongados. Con la ayuda de los modelos de desarrollo rápido de aplicaciones, los programadores pueden realizar varias iteraciones y modificaciones en el software en poco tiempo sin tener que empezar desde el principio cada vez. Esto contribuye a garantizar que el resultado final esté más centrado en la calidad y esté en sintonía con las necesidades de los usuarios finales.

La plataforma APEX de Oracle, es una herramienta RAD, la cual está disponible gratuitamente para compañías que tienen bases de datos Oracle y permite la creación de soluciones customizadas en menos tiempo y en un menor costo que cualquier otra herramienta, convirtiéndose en una de las plataformas low-code líderes del mercado.

RAD se utiliza a menudo en proyectos de software con un tiempo de entrega ajustado, o en proyectos en los que el alcance o las especificaciones pueden cambiar con frecuencia. La metodología también es útil en proyectos en los que el cliente desea tener una participación activa en el desarrollo y la toma de decisiones (Reingart, 2011)

Es importante destacar que RAD no es adecuado para todos los proyectos de software, ya que puede resultar en un producto final de baja calidad si no se lleva a cabo adecuadamente. Por lo tanto, es importante evaluar cuidadosamente si RAD es la metodología adecuada para un proyecto en particular antes de adoptarla.

Los beneficios clave de la metodología RAD son:

- Reducción del tiempo de desarrollo y aceleración de la entrega.
- Mejora de la flexibilidad y la adaptabilidad.
- Mejor gestión de riesgos.
- Menos programación manual y tiempos de prueba más cortos.
- Comentarios de los usuarios constantes, relevantes y en tiempo real (Mena, 2015, pág. 11).

El desarrollo rápido de aplicaciones (Rad) evolucionó con el paso del tiempo. Reingart (2011) menciona que se adaptó su estructura para cumplir con los requisitos previos de la época sin dejar de adherirse a los principios fundamentales de su crecimiento.

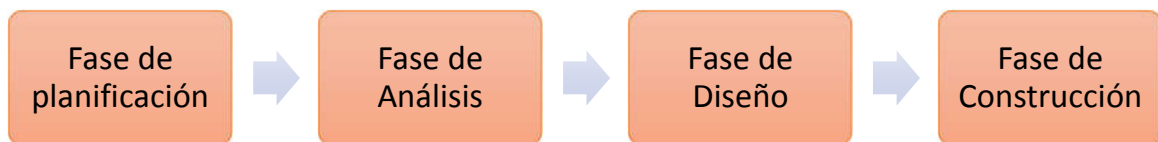
### **5.3.1 Fases RAD**

Según James Martin, en su libro "Rapid Application Development" (1991), la metodología RAD consta de cuatro fases principales:

- **Fase de Planificación:** En esta fase se establecen los objetivos del proyecto, se identifican los requisitos del sistema y se definen los objetivos del prototipo. Se lleva a cabo una evaluación del riesgo y se establecen los recursos necesarios para el desarrollo del proyecto. También se define el equipo de desarrollo y se establece un cronograma de trabajo.
- **Fase de Análisis:** En esta fase se lleva a cabo un análisis detallado de los requisitos del sistema. Se identifican las funcionalidades y características que se incluirán en el prototipo, y se elabora un diseño detallado del sistema. Se establece una lista de prioridades y se define el alcance del prototipo.
- **Fase de Diseño:** En esta fase se crea un diseño detallado del sistema, incluyendo la arquitectura, la interfaz de usuario, las bases de datos, la lógica de negocio y la integración de los componentes. Se desarrolla un diseño funcional del prototipo, que servirá como base para la construcción del mismo.

- **Fase de Construcción:** En esta fase se lleva a cabo la implementación del prototipo. Se desarrollan los componentes del sistema, se integran y se prueban en un entorno controlado. Se realizan pruebas de funcionamiento, pruebas de integración y pruebas de usuario para asegurar que el prototipo cumple con los requisitos establecidos.

Cabe destacar que en la metodología RAD, las fases son iterativas y se llevan a cabo de forma rápida y secuencial, con un enfoque en la creación de prototipos funcionales que puedan ser evaluados y refinados en cada iteración. Esto permite una entrega temprana de funcionalidades y una retroalimentación constante del cliente, lo que facilita la adaptación a los cambios y requisitos cambiantes del proyecto.



*Ilustración 1: Fases RAD (Diseño propio)*

Existen dos paradigmas principales de desarrollo de software: ágil y prescriptivo. El método tradicional de desarrollo de software es en cascada. Este se centra en un proceso lineal estricto que depende en gran medida de la aprobación del cliente. Las creaciones de este tipo pueden durar meses sin que los clientes vean un producto final, lo que provoca muchos problemas para actualizar los requisitos o recibir comentarios adicionales que afectan el proyecto. Puede resultar difícil cambiar las funciones y características principales del software (Martin, 1991).

Mena (2015) refiere que ágil es una de las metodologías más utilizadas. Se creó como respuesta a las limitaciones de las técnicas de gestión estructuradas tradicionalmente. RAD, un tipo de metodología ágil, ofrece resultados en tiempo real y funciona bien cuando debe entregar rápidamente un producto y actualizar las características según sea necesario. Aunque se hace hincapié en la velocidad, no se basa en un período de tiempo específico. Lo que hace que el proceso RAD sea único es que se basa en el proceso y se centra en los prototipos de prueba y en los cambios rápidos para entregar un producto completo en menos tiempo.

Siguiendo con Mena (2015) las tecnologías RAD y ágil comparten pasos similares. Sin embargo, RAD se centra en prototipos mientras que la tecnología ágil

divide los proyectos en características para poder realizar las entregas en varios sprints durante el ciclo de desarrollo. RAD cuenta con un conjunto definido de cuatro pasos necesarios para completar un proyecto. El objetivo de RAD es reducir el tiempo de planificación y centrarse en la construcción y creación de un producto. Por tanto, aunque se repitan algunos pasos, se obtiene un producto del que tanto su equipo como las partes interesadas pueden estar orgullosos.

El software y las aplicaciones se prueban minuciosamente y las partes interesadas pueden proponer cambios o aportar nuevas ideas a medida que se detectan problemas. No debería haber muchos errores, ya que la ventaja de RAD es que la mayoría de los errores pueden verse en tiempo real durante de la fase de creación de prototipos y ajustarse después de inmediato. Una vez que las partes interesadas estén satisfechas con su producto, puede completarlo (Reingart, 2011).

De acuerdo con Martin (1991) la etapa final consiste en hacer una versión optimizada del producto final que sea estable y fácil de mantener para que dure más. Las características, las funciones y la estética se rematan junto con la parte interesada. Una vez que se pasa a producción, los usuarios pueden realizar pruebas o formación a gran escala. En ese momento, el producto está listo para presentarlo a la parte interesada.

### **5.3.2 Beneficios de la implementación de metodología RAD**

La metodología RAD tiene varios beneficios:

- **Desarrollo rápido:** La metodología RAD permite la entrega rápida de prototipos funcionales, lo que significa que el proyecto puede avanzar de manera eficiente y el cliente puede ver el progreso temprano.
- **Mayor involucración del cliente:** RAD permite una mayor participación del cliente en el desarrollo del software, lo que significa que sus necesidades y deseos se pueden incorporar de manera más efectiva en el producto final.
- **Flexibilidad:** La metodología RAD permite una mayor flexibilidad en el desarrollo del software, lo que significa que puede adaptarse fácilmente a los cambios en el alcance o las especificaciones.
- **Retroalimentación temprana:** La interacción continua y la entrega rápida de prototipos permiten una retroalimentación temprana del cliente, lo que

significa que se pueden hacer ajustes temprano en el proceso de desarrollo.

- Mejora de la calidad del producto: La retroalimentación temprana y la participación activa del cliente pueden conducir a una mejor comprensión de sus necesidades y a un producto final más satisfactorio.

Es dable destacar que, para obtener los mejores resultados, la metodología RAD debe llevarse a cabo adecuadamente y por un equipo experimentado. Reingart, (2011) señala que RAD no es adecuado para todos los proyectos de software, por lo que es importante evaluar cuidadosamente si es la metodología adecuada para un proyecto en particular antes de adoptarla.

### **5.3.3 Características de la metodología RAD**

#### **A. Utiliza Equipos Híbridos**

Equipos compuestos por alrededor de seis personas, incluyendo desarrolladores y usuarios de tiempo completo del sistema, así como aquellas personas involucradas con los requisitos.

#### **B. Utiliza herramientas especializadas que soportan:**

- Desarrollo "visual"
- Creación de prototipos falsos (simulación pura)
- Creación de prototipos funcionales
- Múltiples lenguajes
- Calendario grupal
- Herramientas colaborativas y de trabajo en equipo
- Componentes reusables
- Interfaces estándares (API)

#### **C. Utiliza "Timeboxing"**

- Las funciones secundarias son eliminadas como sea necesario para cumplir con el calendario.

#### **D. Utiliza Prototipos Iterativos y Evolucionarios.**

- Reunion JAD (Joint Application Development):
  - Se reúnen los usuarios finales y los desarrolladores.
  - Lluvia de ideas para obtener un borrador inicial de los requisitos.
- Iterar hasta acabar:

- Los desarrolladores construyen y depuran el prototipo basado en los requisitos actuales.
- Los diseñadores revisan el prototipo.
- Los clientes prueban el prototipo, depuran los requisitos.
- Los clientes y desarrolladores se reúnen para revisar juntos el producto, refinar los requisitos y generar solicitudes de cambios.
- Los cambios para los que no hay tiempo no se realizan. Los requisitos secundarios se eliminan si es necesario para cumplir el calendario.
- Notas
  - Las iteraciones requieren entre 1 día a 3 semanas.
  - En algún momento, los prototipos exploratorios pueden convertirse en prototipos operativos.
  - Sesiones de grupos focales
    - Duran alrededor de 2 horas
    - Son dirigidos por un facilitador experimentado que mantiene el grupo "en foco":
      - Al tener objetivos claros con respecto al tipo de información que debe obtenerse,
      - Preparando una agenda orientada a los temas antes de la reunión,
      - Asegurando que la discusión adecuada se dirija hacia cada tema,
      - Asegurando que todos tengan la oportunidad adecuada de participar.
    - Son seguidos por un informe del facilitador

(Mena, 2015, págs. 41-42-43).

#### **5.3.4 Ventajas y desventajas en RAD**

Como toda metodología RAD tiene sus ventajas y desventajas, entre las ventajas podemos encontrar la rápida entrega de prototipos funcionales, una mayor participación del cliente, una mayor flexibilidad en el desarrollo del software, una retroalimentación temprana. Entre las desventajas, la metodología RAD puede ser más costosa que otras

metodologías de desarrollo de software debido a la necesidad de un equipo más grande y la entrega rápida de prototipos, puede ser más compleja que otras metodologías debido a la iteración continua y la retroalimentación temprana, si no se tiene la experiencia suficiente para estos eventos, debido a la naturaleza iterativa de la metodología RAD, puede haber una pérdida de control sobre el proceso de desarrollo y la dirección general del proyecto, como también, al inicio del proyecto puede llevar a una implementación incorrecta o a una pérdida de tiempo en ajustes posteriores.

A continuación, se describen las ventajas y desventajas de mayor relevancia:

Ventajas:

- Los entregables pueden ser fácilmente trasladados a otra plataforma.
- El desarrollo se realiza a un nivel de abstracción mayor.
- Visibilidad temprana.
- Mayor flexibilidad.
- Menor codificación manual.
- Mayor involucramiento de los usuarios.
- Posiblemente menos fallas.
- Posiblemente menor costo.
- Ciclos de desarrollo más pequeños.
- Interfaz gráfica estándar.

Desventajas:

- Costo de herramientas integradas y equipo necesario.
- Progreso más difícil de medir.
- Menos eficiente (para proyectos de gran escala).
- Menor precisión científica.
- Riesgo de revertirse a las prácticas sin control de antaño.
- Prototipos pueden no escalar, un problema mayúsculo.
- Funciones reducidas (por “timeboxing”).
- Dependencia en componentes de terceros: funcionalidad de más o de menos, problemas legales (Reingart, 2011, pág. 50).

**Algunos inconvenientes:**

Para proyectos grandes, aunque por escalas, el RAD requiere recursos humanos suficientes como para crear el número correcto de equipos RAD. Se requiere clientes y desarrolladores comprometidos en las rápidas actividades necesarias para completar un

sistema en un marco de tiempo abreviado. Si no hay compromiso, por ninguna de las partes constituyentes, los proyectos RAD fracasaran.

No todos los tipos de aplicaciones son apropiados para RAD. Si un sistema no se puede modelizar adecuadamente, la construcción de los componentes necesarios para RAD será problemática. Si está en juego el alto rendimiento, y se va a conseguir el rendimiento convirtiendo interfaces en componentes de sistema, el enfoque RAD puede que no funcione. RAD no es adecuado cuando los riesgos técnicos son altos. Esto ocurre cuando una nueva aplicación hace uso de tecnologías nuevas, o cuando el nuevo software requiere un alto grado de interoperabilidad con programas de computadora ya existentes (Mena, 2015).

### **5.3.5 Cuando utilizar los modelos Rad (Rapid Application Development)**

Cuando se dispone de menos tiempo para la creación del producto, como por ejemplo en un lapso de pocos días, se utiliza el modelo Rapid Application Development (Rad). Se utiliza cuando ya se ha tomado una decisión sobre los entregables y los requisitos. Los modelos de desarrollo rápido de aplicaciones (Rad) pueden utilizarse cuando se da al usuario final o al cliente la opción de participar en todas las fases del ciclo de vida del producto; esto se conoce como "participación del cliente o del usuario".

Se puede utilizar en caso de que el presupuesto sea lo suficientemente amplio; será posible contratar diseñadores. Para desarrollar códigos con herramientas automatizadas, que exigen un mayor presupuesto, es necesario disponer de un presupuesto mayor (Reingart, 2011).

#### **1.4.6 Prototipos**

Según Martin, J. (1991) la metodología de desarrollo RAD es un enfoque ágil para el desarrollo de software que se centra en la rápida creación de prototipos y la iteración continua basada en la retroalimentación del usuario. El objetivo es desarrollar y entregar el software en un plazo muy corto, utilizando técnicas de desarrollo rápido y herramientas automatizadas para acelerar el proceso.

La metodología RAD se basa en la creación de prototipos funcionales rápidos del software, lo que permite a los desarrolladores obtener rápidamente una comprensión

clara de los requisitos del usuario y su retroalimentación. El proceso de desarrollo se divide en ciclos cortos de iteración, donde se crea un prototipo, se revisa y se mejora basado en la retroalimentación del usuario, para luego seguir con la siguiente iteración.

La metodología RAD se enfoca en la colaboración continua entre los desarrolladores y los usuarios, a través de la utilización de prototipos. La colaboración continua ayuda a garantizar que el software desarrollado cumpla con las necesidades y expectativas del usuario final.

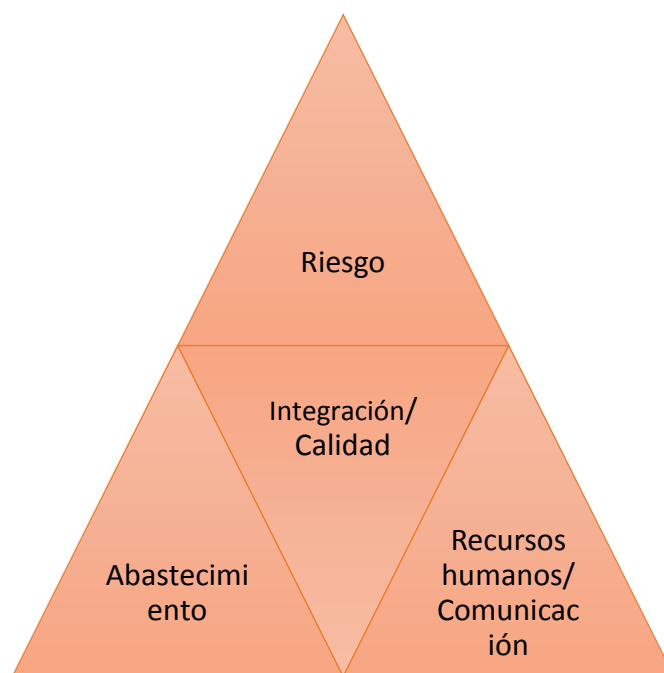
Los prototipos en la metodología RAD pueden ser desarrollados utilizando herramientas de desarrollo rápido y automatizado, como las herramientas de diseño de interfaz de usuario, generadores de código y editores de código. Esto ayuda a reducir el tiempo de desarrollo y mejora la productividad de los desarrolladores.

En resumen, la metodología RAD se basa en la creación rápida de prototipos y la colaboración continua con el usuario final, para asegurar que el software entregado cumpla con las expectativas y necesidades del usuario. El enfoque en ciclos cortos de iteración, la utilización de herramientas de desarrollo automatizado y la retroalimentación continua del usuario final son algunos de los elementos clave de la metodología RAD.

#### **5.4 Administración de proyectos**

La administración de proyectos es la aplicación de conocimientos, experiencias, herramientas y técnicas para proyectar actividades con el fin de alcanzar o exceder las expectativas y necesidades del proyecto, alcanzar o exceder las necesidades o expectativas involucra invariablemente hacer un balance entre demandas. El director de proyecto es la persona responsable de alcanzar los objetivos del proyecto (Baca Urbina, 2013).

- Alcance, Tiempo, Costos y Calidad.
- Inversionistas con diferentes necesidades y expectativas.
- Requerimientos Identificados, así como los no Identificados.
- Adaptar las especificaciones, los planes, y el enfoque a las diversas inquietudes y expectativas de los diferentes interesados.



*Gráfico 1: Triángulo de la Triple Restricción. Elaboración propia en base a (Ander Egg, 2010)*

#### **5.4.1 Variables principales**

Baca Urbina (2013) menciona que la administración de proyectos se refiere a la aplicación de conocimiento, habilidades, herramientas y técnicas para lograr objetivos específicos dentro del presupuesto especificado y las restricciones de tiempo establecidas. Entre las actividades de administración de proyectos están el planear el trabajo, evaluar el riesgo, estimar los recursos requeridos para realizar el trabajo, organizarlo, adquirir los recursos humanos y materiales, asignar tareas, dirigir actividades, controlar la ejecución del proyecto, informar sobre el progreso y analizar los resultados. Como en otras áreas de la empresa, la administración de proyectos para los sistemas de información debe lidiar con cinco variables principales: alcance, tiempo, costo, calidad y riesgo.

El alcance define qué trabajo se incluye o no en un proyecto. Por ejemplo, el alcance del proyecto para un nuevo sistema de procesamiento podría ser contener nuevos módulos para introducir pedidos y transmitirlos a producción y contabilidad, pero ningún cambio en los sistemas relacionados de cuentas por cobrar, fabricación, distribución o control de inventario. La administración de proyectos define todo el

trabajo requerido para completar un proyecto con éxito y debe asegurar que el alcance del mismo no se expanda más allá de lo que estaba planeado en un principio (Baca Urbina, 2013).

El tiempo es el lapso requerido para completar el proyecto. Por lo general, la administración de proyectos establece la cantidad de tiempo necesaria para completar los componentes principales de un proyecto. Cada uno de estos componentes se divide a su vez en actividades y tareas. La administración de proyectos trata de determinar el tiempo requerido para completar cada tarea y establecer un itinerario para terminar el trabajo (Baca Urbina, 2013).

El costo se basa en el tiempo para completar un proyecto, multiplicado por el costo de los recursos humanos requeridos para finalizar ese proyecto. Los costos de un proyecto de sistemas de información también consideran el costo del hardware, software y el espacio de trabajo. La administración de proyectos desarrolla un presupuesto para el proyecto y monitorea los gastos continuos del mismo (Baca Urbina, 2013).

La calidad es un indicador de qué tan bien cumple el resultado final de un proyecto con los objetivos especificados por la gerencia. La calidad de los proyectos de sistemas de información se reduce por lo general a las mejoras en el desempeño organizacional mejorado y en la toma de decisiones. La calidad también considera la precisión y actualidad de la información producida por el nuevo sistema, además de su facilidad de uso (Baca Urbina, 2013).

El riesgo se refiere a los problemas potenciales que amenazan el éxito de un proyecto. Estos problemas potenciales podrían evitar que un proyecto logre sus objetivos al incrementar el tiempo y el costo, reducir la calidad de los resultados del proyecto o evitar que éste se complete (Baca Urbina, 2013).

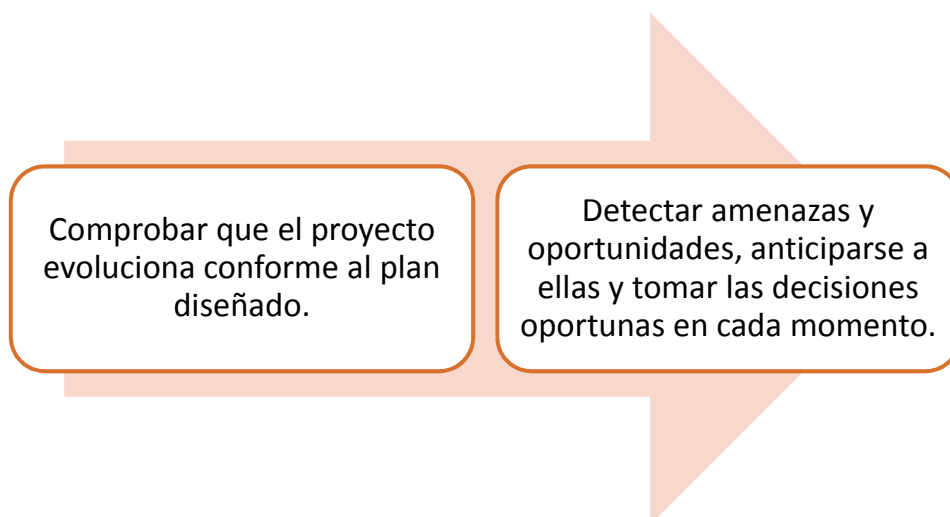
## **5.5 Evaluación de proyectos**

Sapag Chaín, Sapag Chaín y Sapag Puelma (2014) refieren que la evaluación de los proyectos es un proceso de valoración en el cual se analizan todos los elementos que intervienen en el proyecto con el fin de determinar su viabilidad y eficacia, calcular los posibles riesgos y determinar las respuestas. Se trata de una fase fundamental, con independencia de sus características y tamaño del proyecto en cuestión.

La evaluación desde el punto de vista semántico significa estimar, apreciar, calcular el valor de una cosa. Como mecanismo institucional es el proceso integral y

continuo de investigación y análisis de los cambios, más o menos permanentes, que se materializan en el mediano y el largo plazos, como consecuencia directa o indirecta del quehacer institucional o de un proyecto de desarrollo socio-económico, en un contexto, una población y con el apoyo de las organizaciones participantes (Ruíz Domínguez & Becerra Lois, 2015).

Baca Urbina (2013) afirma que la importancia de la evaluación de proyectos radica en que supone implementar un seguimiento y control que permita establecer una comparación para poder determinar y medir la evolución del proyecto, detectar desviaciones y necesidades y establecer las medidas de mejora necesarias a lo largo del proceso. Además, todo proceso de evaluación precisa de un monitoreo, un seguimiento y control continuo que permita:



*Ilustración 2: Importancia de la evaluación de proyectos. Elaboración propia en base a (Baca Urbina, 2013)*



## **9. DESCRIPCIÓN DEL SISTEMA ACTUAL**

La Universidad Católica de Salta actualmente dispone del sistema SAG (Sistema de Autogestión), el cual, tiene un módulo dentro de Gestiones Administrativas para que el alumno pueda solicitar becas. Seguidamente, se le presenta al alumno la opción para descargar el reglamento de becas de la UCASAL, el instructivo del uso del sistema y el cronograma de presentación de solicitudes de becas. Luego de haber leído el reglamento, el alumno debe aceptar las condiciones para poder ingresar al formulario.

En el formulario se deberán cargar los siguientes datos (esta información se encuentra detallada en el **Anexo II** y **Anexo III**):

- Datos laborales del alumno.
- Domicilio actual.
- Información de otras becas en la cual es beneficiario.
- Información relacionada a su situación económica.
- Información del grupo familiar.

Finalmente, se graba la solicitud y se genera un archivo PDF.

La solicitud queda almacenada en el sistema para ser posteriormente evaluada por el comité e informar al alumno sobre el resultado de la misma.

## **10.METODOLOGÍA**

### **7.1 Metodología de Investigación**

Los métodos de desarrollo que se utilizó para el desarrollo del Sistema de Gestión de Becas de la Universidad Católica de Salta, son los siguientes:

#### **7.1.1 Método Análisis**

En este trabajo se realizó una detallada investigación bibliográfica en la cual se obtuvo datos necesarios y relevantes que deben ser tenidos en cuenta y comprendidos para el desarrollo de la aplicación.

#### **7.1.2 Técnicas de recolección de información**

##### **7.1.2.1 Entrevista**

Se realizó una entrevista a un empleado administrativo del Departamento de Becas de la UCASAL para conocer el sistema actual, entender los procesos que se llevan a cabo y relevar los requerimientos necesarios para el desarrollo del nuevo Sistema de Gestión de Becas. Para observar el detalle de la entrevista revise el Anexo I.

### **7.2 Metodología de Desarrollo RAD**

Para realizar este trabajo se utilizó la metodología RAD (Rapid Application Development) que consta de las siguientes fases principales: Planificación (Modelado de la Gestión), Diseño (Modelado de datos y procesos), Construcción (Generación de Aplicación) y la Implementación (Pruebas de entrega).

#### **7.2.1 Planificación (Modelado de la Gestión)**

El flujo de información se modela teniendo en cuenta las siguientes preguntas (Universidad Tecnológica de Pereira, 2011).

##### **7.2.1.1 ¿Qué información conduce el proceso de gestión?**

La UCASAL deberá proveer Web Service que sean necesarios para obtener la información académica del alumno, así también como la información personal, para determinar si el alumno cumple con los requisitos necesarios para aplicar a la beca.

##### **7.2.1.2 ¿Qué información se genera?**

La información que se genera a través del método get pertenece a los alumnos de la UCASAL.

##### **7.2.1.3 ¿Quién la genera?**

La genera el equipo técnico informático de la Universidad.

#### 7.2.1.4 ¿A dónde va la información?

La información debe llegar al equipo de desarrollo para diseñar el sistema, definir los procesos y desarrollar la aplicación.

#### 7.2.1.5 ¿Quién las procesa?

El equipo de desarrollo es el encargado de identificar los procesos necesarios en la aplicación para obtener los resultados requeridos.

*Tabla 4: Modelado de Gestión. Elaboración propia*

<b>Fase 1: Modelado de Gestión</b>		
<b>Descripción del Documento técnico (artefacto)</b>	<b>Descripción detallada</b>	<b>Observación</b>
<b>Descripción del sistema a desarrollarse</b>	Desarrollo de una aplicación web para la gestión de Becas de la Universidad Católica de Salta.	Se puede emplear herramientas de Ofimática.
<b>Áreas de Influencia del software a desarrollarse</b>	Universidad Católica de Salta (UCASAL).	-
<b>Atributos del sistema a desarrollarse</b>	Los procesos definidos deben ser automatizados en detalle a través de la aplicación web.	Se puede emplear herramientas de Ofimática.
<b>Funciones del sistema a desarrollarse</b>	<ul style="list-style-type: none"> <li>• Obtener datos básicos del alumno,</li> <li>• obtener información académica del alumno,</li> <li>• obtener información de las becas,</li> <li>• gestionar solicitud,</li> <li>• generar reportes.</li> </ul>	Se puede emplear herramientas de Ofimática.
<b>Especificación de requerimientos</b>	Listado de requerimientos	Se puede emplear herramientas de Ofimática.
<b>Alcance del sistema a desarrollarse</b>	Área de Becas de la UCASAL.	Se puede emplear herramientas de Ofimática.

### 7.2.2 Diseño (Modelado de datos)

En esta fase se deben definir los atributos de cada uno de los objetos y la relación entre los mismos ((Universidad Tecnológica de Pereira, 2011).

*Tabla 5: Modelado de datos. Elaboración propia*

<b>Fase 2: Modelado de datos</b>		
<b>Descripción del documento tecnico de Ingeniería</b>	<b>Descripción detallada</b>	<b>Observación</b>
<b>Modelamiento de datos</b>	<b>Web Service:</b> <ul style="list-style-type: none"> <li>➤ GetAlumno</li> <li>➤ GetCarreras</li> </ul> <b>Modelo de Base de datos</b> <ul style="list-style-type: none"> <li>➤ DER (Diagrama Entidad-Relación)</li> </ul>	Herramientas utilizadas: Oracle APEX, Oracle SQL y Microsoft Visio.

### 7.2.3 Diseño (Modelado de proceso)

Los objetos de datos que son definidos en la fase de modelado de datos, se transforman con el objetivo de lograr el flujo de información necesario para el sistema.

Se describen procesos que se crean para intercomunicar los objetos, ya sea para añadir, editar, eliminar o recuperar datos (Universidad Tecnológica de Pereira, 2011).

*Tabla 6: Modelado de Proceso. Elaboración propia*

<b>Fase 2: Modelado de Proceso</b>		
<b>Descripción del documento tecnico de Ingeniería</b>	<b>Descripción detallada</b>	<b>Observación</b>
<b>Modelamiento de procesos</b>	<b>Casos de Uso:</b> <b>Módulo de Administración:</b> <ol style="list-style-type: none"> <li>1. Registrar Usuario del Sistema</li> <li>2. Editar Usuario del Sistema</li> </ol>	Modelamiento del comportamiento del sistema desde la perspectiva del

	<ol style="list-style-type: none"> <li>3. Registrar Administrador del Sistema</li> <li>4. Registrar Rol de Usuario</li> <li>5. Editar Rol de Usuario</li> <li>6. Configurar Rol de Usuario</li> <li>7. Registrar Ingreso al Sistema (Login)</li> <li>8. Recuperar Contraseña del Usuario</li> <li>9. Cambiar Contraseña del Usuario</li> <li>10. Registrar Beca</li> <li>11. Editar Beca registrada</li> <li>12. Inhabilitar Beca</li> <li>13. Aprobar Solicitud</li> <li>14. Rechazar Solicitud</li> <li>15. Editar datos de contacto del Alumno</li> <li>16. Consultar Listado de Becas</li> <li>17. Consultar Listado de Solicitudes</li> <li>18. Consultar Historial de Becas otorgadas</li> </ol> <p><b>Portal del Alumno:</b></p> <ol style="list-style-type: none"> <li>19. Registrar Solicitud</li> <li>20. Consultar Estado de Solicitud</li> </ol>	usuario
--	--	---------

#### **7.2.4 Construcción (Generación de Aplicación)**

En la metodología RAD asume la utilización de técnicas de cuarta generación, en lugar de crear software con lenguajes de tercera generación.

Las técnicas de cuarta generación de software, también conocidas como 4GL (Fourth Generation Language) en inglés, son enfoques y herramientas de desarrollo de software que buscan aumentar la productividad y la eficiencia en la creación de aplicaciones informáticas. Estas técnicas suelen utilizar lenguajes de programación de alto nivel y abstracciones que permiten a los desarrolladores crear aplicaciones de

manera más rápida y sencilla, con menor cantidad de código y con un enfoque orientado a la declaración de lo que se quiere lograr, en lugar de cómo lograrlo.

En el presente proyecto se ha trabajado con Oracle APEX 22.2 y la base de datos con Oracle Database 19c (SQL).

*Tabla 7: Construcción de Aplicaciones. Elaboración propia*

<b>Fase 3: Construcción de aplicaciones</b>		
<b>Descripción del documento técnico de ingeniería</b>	<b>Descripción detallada</b>	<b>Observación</b>
<b>Construcción prototipo</b>	<p><b><u>Modulo Administración:</u></b></p> <p><b>Submodulo de Gestión:</b></p> <ul style="list-style-type: none"> <li>-Pantalla de ABM de becas.</li> <li>-Pantalla de Gestión de Solicitudes.</li> </ul> <p><b>Submódulo de Reporteria:</b></p> <ul style="list-style-type: none"> <li>-Listado de becas</li> <li>-Listado de solicitudes</li> <li>-Historial de becas otorgadas</li> </ul> <p><b>Submódulo de Seguridad:</b></p> <ul style="list-style-type: none"> <li>-Pantalla de Usuarios</li> <li>-Pantalla de Roles</li> </ul> <p><b><u>Portal del Alumno:</u></b></p> <ul style="list-style-type: none"> <li>- Pantalla de listado de becas ofrecidas</li> <li>- Pantalla para completar datos de solicitud de beca</li> <li>-Pantalla de consulta de estado de solicitudes aplicadas por el alumno.</li> </ul>	<p>Oracle APEX 22.2 y la base de datos con Oracle Database 19c (SQL).</p>

### 7.2.5 Implementación (Pruebas de entrega)

La metodología RAD se centra en la reutilización de componentes ya testeados para reducir tanto el tiempo de desarrollo como de validación de los mismos. Sin embargo se deben probar los componentes nuevos y validar a fondo las interfaces.

*Tabla 8: Implementación. Elaboración propia*

<b>Fase 4: Implementación</b>		
<b>Descripción del documento técnico de ingeniería</b>	<b>Descripción detallada</b>	<b>Observación</b>
<b>Plan de Pruebas</b>	Pruebas de Caja negra	Se evalúa el funcionamiento del sistema sin conocer su estructura interna. El análisis se enfoca en la entrada y salida del sistema.

## **11.DESARROLLO**

Luego de llevada a cabo la primera iteración, se obtendrá un primer prototipo completamente funcional para el cliente.

Las fases que se siguieron para obtener este prototipo, son:

### **8.1 Fase de Planificación de Requisitos**

Para definir los requerimientos y el alcance del sistema, se realizó una entrevista al personal encargado del departamento de Gestión de Becas de la Universidad Católica de Salta. En la misma se buscó indagar sobre las características del sistema actual, sobre los procesos que se realizan a lo largo del sistema de gestión (Carga de becas, solicitud, gestión y evaluación) y los procedimientos que podrían tener una mejora, así como las funcionalidades faltantes. Se adjunta entrevista en **Anexo I**.

Como resultado de esta acción, se lograron definir los siguientes requerimientos:

#### **8.1.1 Requerimientos Funcionales:**

- El sistema debe permitirle al usuario, tanto cargar como dar de baja, diferentes tipos de becas, asignando a cada una diferentes requisitos, fechas de vigencia y beneficios ofrecidos.
- El sistema debe ofrecer a los alumnos que cumplen con los requisitos establecidos, un listado de las becas generadas.
- Se debe permitir editar o adicionar datos a las solicitudes generadas, en caso en el que el postulante haya cometido un error o cargado información incompleta.
- El sistema debe ofrecer al usuario del departamento de becas, la posibilidad de ver el listado de todas las becas solicitadas, discriminando por tipo de beca y por fecha.
- Se debe permitir evaluar las solicitudes cargadas, determinando si las mismas están aprobadas o no. En caso de un rechazo, indicar los motivos del mismo.
- El sistema debe ofrecer la posibilidad de generar un reporte histórico con el listado de becas otorgadas, y de becas rechazadas.
- Debe contener una interfaz en donde los alumnos de la facultad puedan aplicar a las becas que tienen disponibles, ingresando los datos requeridos en el formulario y la documentación adicional.

- El solicitante debe poder realizar el seguimiento de sus solicitudes mediante una interfaz que muestre un listado de las mismas junto a su estado correspondiente.

### **8.1.2 Requerimientos No Funcionales:**

- Seguridad: el sistema deberá disponer de diferentes roles para controlar el acceso a diferentes menús.
- Disponibilidad: El sistema deberá estar disponible en todo momento para que los alumnos puedan realizar las solicitudes.
- Mantenimiento: El sistema debe ser de fácil mantenimiento.
- Portabilidad: el sistema deberá ser adaptable a diferentes dispositivos (computadoras o dispositivos móviles)
- Usabilidad: El sistema deberá ser fácil de usar y comprender para todos sus usuarios.

## **8.2 Estudio de Factibilidad**

Se llevará a cabo un estudio de factibilidad del proyecto de desarrollo, el cual evalúa la viabilidad técnica, económica y operativa del proyecto, y proporciona una base para tomar decisiones informadas sobre su implementación.

### **8.2.1 Viabilidad Técnica**

La viabilidad técnica del proyecto se ve respaldada por el hardware disponible en la Universidad, que consiste en máquinas con conexión a internet. Aunque el rendimiento de las máquinas no resulta crítico debido a que el sistema se alojará en la nube, es importante destacar que estas máquinas proporcionarán un entorno de trabajo adecuado para el acceso y uso del sistema.

En términos de infraestructura de red, la empresa cuenta actualmente con un proveedor de internet por cable. Sin embargo, para garantizar la disponibilidad continua del sistema, se recomienda la contratación de un segundo proveedor de internet como respaldo en caso de posibles cortes de conexión. Esto asegurará que los usuarios puedan acceder al sistema de manera ininterrumpida y evitará interrupciones en la operación.

Además, al optar por alojar el sistema en la nube, se aprovecharán las ventajas que ofrece este modelo, como la escalabilidad y la flexibilidad. El sistema se beneficiará de la capacidad de almacenamiento y procesamiento de la nube, sin requerir recursos significativos en las máquinas locales.

### 8.2.2 Viabilidad Económica

La viabilidad económica del sistema web de gestión de becas se evaluará teniendo en cuenta los costos involucrados en su desarrollo, implementación y operación, así como los beneficios esperados. A continuación, se presentará una estimación de los costos y beneficios.

#### Costos:

- **Desarrollo del sistema:** a continuación se detalla los costos asociados al desarrollo:

Aplicaciones	Cantidad de Pantallas	Horas
Sistema de Gestión	13	32 hrs
Portal del Alumno	10	18 hrs

Desarrolladores	Horas de Análisis	Horas de Desarrollo	Horas de Implementación	Costo por Hora	Costo Total
1	18 hrs	50 hrs	8 hrs	50 USD	3800 USD

- **Licencias:** los costos asociados a licencias de base de datos se detalla a continuación, cabe destacar que la licencia de Oracle APEX está incluida en la licencia de la Base de Datos Cloud.

Licencias	Costo Mensual	Costo Anual
Oracle Cloud (BD)	202,32 USD	2427,84 USD

En base a los costos detallados, la inversión total estimada es de **6.227,84 USD**.

### 8.2.3 Viabilidad Operativa

La viabilidad operativa del proyecto se ve fortalecida por el perfil del equipo de trabajo de la Universidad, compuesto por personas jóvenes y familiarizadas con el uso de tecnologías. Acostumbrados al manejo de sistemas computacionales, lo que indica que están acostumbrados a trabajar con interfaces de usuario.

En este sentido, se espera que el equipo de trabajo no encuentre dificultades en adaptarse al nuevo sistema de gestión de becas, ya que la interfaz gráfica diseñada será

mucho más amigable y fácil de usar en comparación con el sistema actual. La transición a un sistema más intuitivo y moderno permitirá que los usuarios se sientan cómodos y puedan aprovechar al máximo las funcionalidades ofrecidas.

Además, se tiene previsto brindar capacitación segmentada por áreas a los usuarios que utilizarán el sistema. Esta capacitación personalizada facilitará la comprensión y el dominio de las diferentes funcionalidades según las necesidades y responsabilidades de cada área. De esta manera, se asegurará que los usuarios adquieran las habilidades necesarias para utilizar el sistema de manera eficiente y aprovechar todas sus ventajas.

#### **8.2.4 Viabilidad Legal**

Se realizó una revisión de las regulaciones internas de la Universidad Católica de Salta para garantizar que el Sistema de Gestión de Becas cumpla con todos los requisitos legales y normativos establecidos por la institución.

Durante el análisis, se verificó que el sistema cumple con las políticas de acceso a la información y divulgación de datos establecidas por la Universidad, evitando cualquier conflicto con las regulaciones legales vigentes.

### **8.3 Fase de Diseño (Modelado de datos y de procesos)**

La fase de diseño de la metodología RAD es la etapa en donde se definen y modelan los datos y procesos de la aplicación a desarrollar, incluye diagramas que permiten visualizar la construcción de un sistema. Cada iteración aborda de manera cada vez más detallada el diseño del sistema.

#### **8.3.2 Modelado de Datos**

En esta subfase se definen y diseñan las estructuras de datos que serán utilizadas en la aplicación a desarrollar. Esto incluiría la creación de modelos de datos, que son representaciones visuales de las entidades y sus relaciones en la aplicación.

##### **8.3.2.1 Web Service solicitado**

El Web Service que se solicita a la UCASAL es el siguiente:

## GetAlumno

Este Web Service será el encargado de proporcionar toda la información relacionada al alumno, discriminada de la siguiente manera:

- Datos Generales
- Domicilio
- Información Académica

Respuesta del Web Service:

```
{
  "RESPUESTA": "APROBADO",
  "DATOS_ALUMNO": {
    "DATOS_GENERALES": [
      {
        "NOMBRE": "MILAGRO",
        "APELLIDO": "FERNANDEZ SABATE",
        "DNI": "22222222",
        "SEXO": "FEMENINO",
        "FECHA_NACIMIENTO": "15/05/1992"
      }
    ],
    "DOMICILIO": [
      {
        "CALLE": "URQUIZA",
        "NUMERO": "325",
        "CP": "4400",
        "BARRIO": "CENTRO",
        "LOCALIDAD": "SALTA",
        "PROVINCIA": "SALTA"
      }
    ],
    "INFORMACION_ACADEMICA": [
      {
        "FACULTAD": 4,
        "CARRERA": 84,
        "MODO": 1,
        "ALUMNO_REGULAR": "SI",
        "MATERIAS_APROBADAS": "42",
        "PLAN": "2015",
        "PROMEDIO": "7,50"
      }
    ]
  }
}
```

*Modelo de JSON requerido como respuesta a la API GetAlumno*

## GetCarreras

Este Web Service será el encargado de proporcionar toda la información relacionada a las facultades y carreras asociadas de la Universidad Católica de Salta. Esta información se presentará de la siguiente manera:

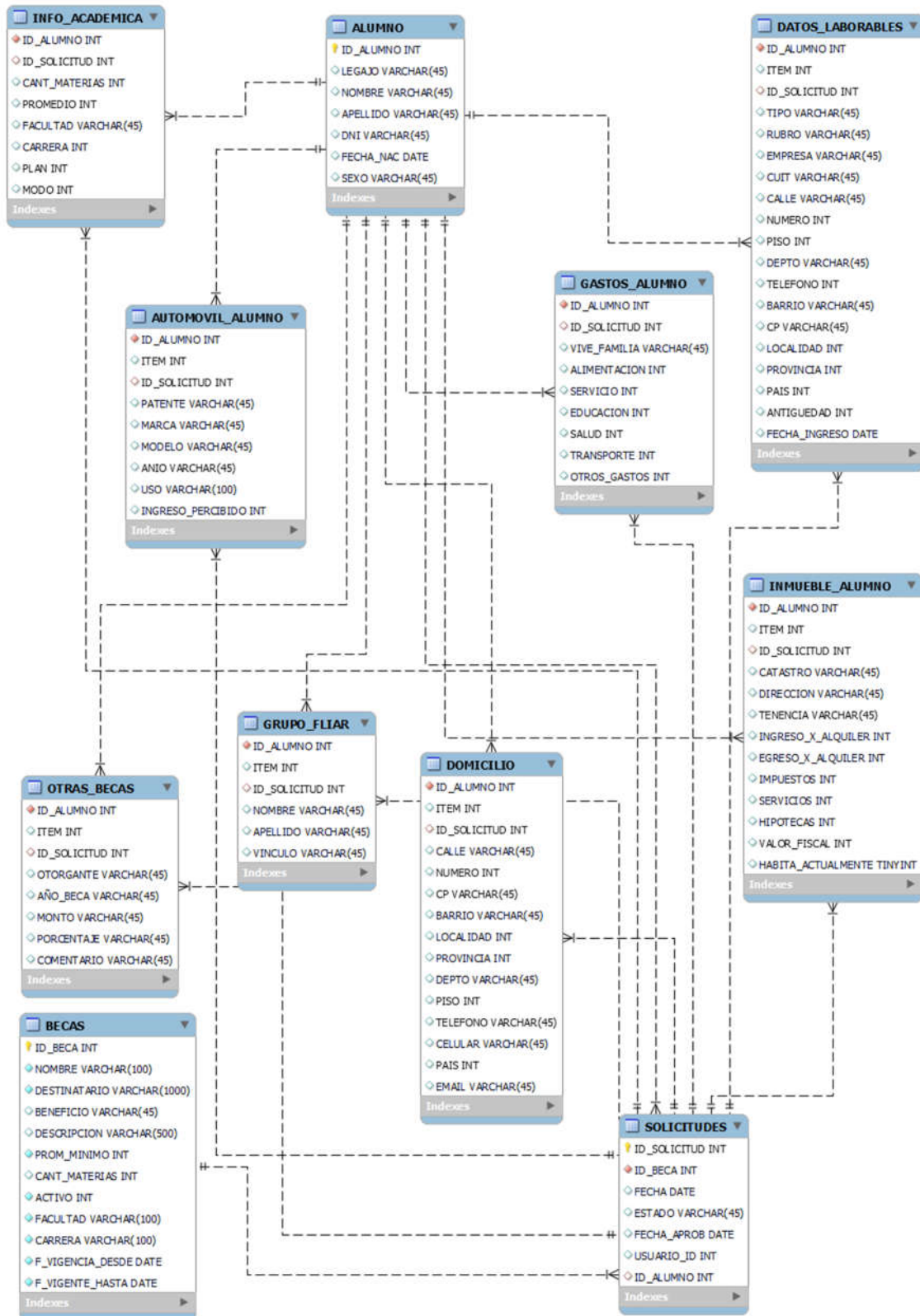
```
{
  "RESPUESTA": "APROBADO",
  "LISTADO_FACULTADES": [
    {
      "IDENTIFICADOR": "4",
      "NOMBRE": "INGENIERÍA",
      "CARRERAS": [
        {
          "IDENTIFICADOR": "84",
          "NOMBRE": "INGENIERÍA EN INFORMÁTICA"
        },
        {
          "IDENTIFICADOR": "85",
          "NOMBRE": "INGENIERÍA INDUSTRIAL"
        },
        {
          "IDENTIFICADOR": "86",
          "NOMBRE": "INGENIERÍA CIVIL"
        }
      ]
    },
    {
      "IDENTIFICADOR": "5",
      "NOMBRE": "ECONOMÍA Y ADMINISTRACIÓN",
      "CARRERAS": [
        {
          "IDENTIFICADOR": "70",
          "NOMBRE": "LIC. EN ECONOMÍA"
        },
        {
          "IDENTIFICADOR": "71",
          "NOMBRE": "CONTADOR PÚBLICO"
        },
        {
          "IDENTIFICADOR": "72",
          "NOMBRE": "LIC. EN RECURSOS HUMANOS"
        }
      ]
    },
    {
      "IDENTIFICADOR": "6",
      "NOMBRE": "EDUCACIÓN",
      "CARRERAS": [
        {
          "IDENTIFICADOR": "60",
          "NOMBRE": "LIC. EN HISTORIA"
        }
      ]
    }
  ]
}
```

```
    },  
    {  
      "IDENTIFICADOR": "61",  
      "NOMBRE": "LIC. EN GESTIÓN EDUCATIVA"  
    },  
    {  
      "IDENTIFICADOR": "62",  
      "NOMBRE": "PROF. EN EDUCACIÓN PRIMARIA"  
    }  
  ]  
}  
]
```

#### **8.2.2.2 Diseño de Base de Datos**

A continuación, se presentan las ilustraciones que muestran en detalle el modelo de base de datos que se utilizará para respaldar el Sistema de Gestión de Becas y asegurar que cumpla con todas las funciones necesarias.

Es importante destacar que se ha desarrollado tanto un Diagrama Entidad-Relación (DER) General como uno específico para el Módulo de Seguridad (Usuarios y Roles).



*Ilustración 3: Diagrama Entidad-Relación General*



**14. Rechazar Solicitud**

**15. Editar datos de contacto del Alumno**

16. Consultar Listado de Becas

17. Consultar Listado de Solicitudes

18. Consultar Historial de Becas otorgadas

**Portal del Alumno:**

**19. Registrar Solicitud**

20. Consultar Estado de Solicitud

### **8.2.1.1 Especificaciones de Casos de Uso**

Los casos de uso son una técnica utilizada para describir cómo los usuarios interactúan con la aplicación y cómo ésta responde a esas interacciones. Los casos de uso son escenarios o situaciones que describen cómo se utilizan esas funcionalidades en situaciones reales.

Los casos de uso suelen representarse en forma de diagramas de casos de uso, que muestran las interacciones entre los actores (los usuarios o sistemas externos que interactúan con la aplicación) y los casos de uso (las funcionalidades de la aplicación). Cada caso de uso describe una acción o tarea específica que realiza un actor en la aplicación y cómo la aplicación responde a esa acción (Kendall & Kendall, 2011).

A continuación, se desarrollarán las Especificaciones de Casos de Uso que fueron marcadas en negrita en el Listado de Casos de Uso.

<b>ID_CU</b>	Registrar Usuario del Sistema
<b>Descripción</b>	El usuario administrador podrá registrar a otros usuarios desde el módulo de seguridad.
<b>Actor</b>	Administrador
<b>Precondiciones</b>	El usuario administrador debe haber iniciado sesión al sistema de gestión.
<b>Postcondiciones</b>	Debe quedar registrado el usuario en el sistema para poder loguearse.

**Escenario principal del éxito**

<b>Actor</b>	<b>Sistema</b>
1. Iniciar Caso de Uso	
2. Seleccionar opción para agregar Usuario	
	3. Mostrar vista para registración de usuario
4. Completar Nombre y Apellido del usuario	
5. Completar mail	
6. Completar contraseña	
7. Completar nombre de usuario.	
8. Seleccionar opción asignar rol al usuario	
9. Seleccionar opción de guardar.	
	10. Validar mail
	11. Validar contraseña
	12. Validar nombre de usuario
	13. Guardar datos del usuario
	14. Mostrar mensaje de éxito.
	15. Fin de Caso de Uso

**Escenario de alternativas**

10. Validar mail (el mail tiene un formato incorrecto)

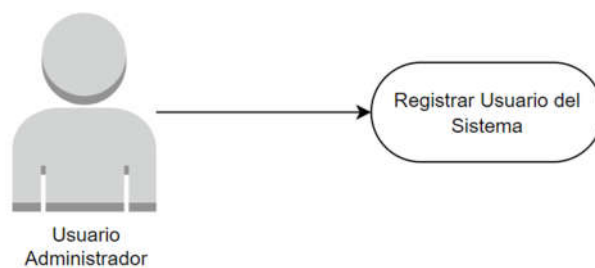
<b>Actor</b>	<b>Sistema</b>
	10.1 Mostrar error “formato incorrecto”
10.2 Ingresar nuevamente el mail	
	10.3 Volver al paso 9

11. Validar contraseña (la contraseña ingresada no cumple con la cantidad de caracteres)

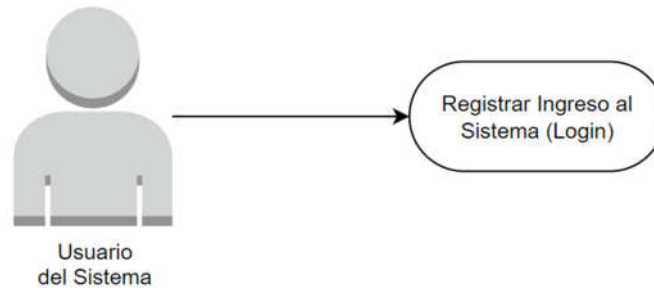
<b>Actor</b>	<b>Sistema</b>
	11.1 Mostrar error “Debe contener como mínimo 6 caracteres”
11.2 Ingresar nuevamente el mail	
	11.3 Volver al paso 9

12. Validar nombre de usuario (el nombre de usuario ya existe)

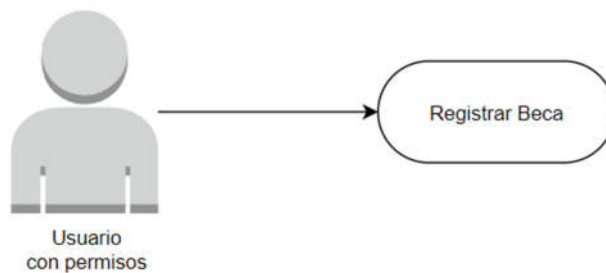
Actor	Sistema
	12.1 Mostrar error “el nombre de usuario ya existe”
12.2 Ingresar nuevamente nombre de usuario	
	12.3 Volver al paso 9



<b>ID_CU</b>	Registrar Ingreso al Sistema (Login)	
<b>Descripción</b>	El usuario accede al sistema empleando sus credenciales de acceso.	
<b>Actor</b>	Cualquier usuario del sistema	
<b>Precondiciones</b>	➤ El usuario debe tener un usuario creado	
<b>Postcondiciones</b>	➤ Debe quedar registrada la beca y disponible para el alumno.	
<b>Escenario principal del éxito</b>		
	<b>Actor</b>	<b>Sistema</b>
	1. Iniciar Caso de Uso	
		2. Solicitar usuario y clave
	3. Completar usuario	
	4. Completar clave	
	5. Seleccionar opción de inicio sesión	
		6. Validar usuario y clave
		7. Mostrar Menú principal
		8. Fin de Caso de Uso
<b>Escenario de alternativas</b>		
6. Validar Usuario y Clave (usuario o clave ingresada erróneamente)		
	<b>Actor</b>	<b>Sistema</b>
		6.1 Mostrar mensaje “Usuario y/o Clave incorrecta”
	6.2 Ingresar Usuario y clave nuevamente	
		6.3 Volver al paso 5



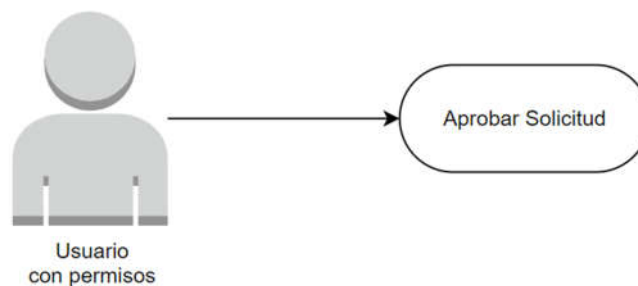
<b>ID_CU</b>	Registrar Beca	
<b>Descripción</b>	El usuario del departamento de becas, en el módulo de gestión, podrá cargar en el sistema las becas con sus especificaciones correspondientes.	
<b>Actor</b>	Cualquier usuario del sistema con los permisos correspondientes.	
<b>Precondiciones</b>	➤ El usuario del departamento de becas debe haber iniciado sesión.	
<b>Postcondiciones</b>	➤ Debe quedar registrada la beca y disponible para el alumno.	
<b>Escenario principal del éxito</b>		
<b>Actor</b>	<b>Sistema</b>	
1. Iniciar Caso de Uso		
2. Seleccionar la opción de registración de beca		
	3. Mostrar vista de becas ya cargadas	
4. Seleccionar opción para agregar Beca		
	5. Mostrar vista de Carga/Edición de Becas	
6. Completar Nombre de Beca		
7. Completar descripción de destinatarios (a quienes va dirigido el beneficio)		
8. Completar descripción del beneficio		
9. Seleccionar opción para guardar		
	10. Guardar datos de beca	
	11. Mostrar mensaje de éxito	
	12. Fin de Caso de Uso	
<b>Escenario de alternativas</b>		



<b>ID_CU</b>	Aprobar Solicitud
<b>Descripción</b>	El usuario del departamento de becas, en el módulo de gestión, podrá realizar la aprobación de las solicitudes, y así el alumno verá reflejado en el Portal del Alumno el estado de la misma y recibirá una notificación cuando la solicitud sea aprobada.
<b>Actor</b>	Usuario del sistema con los permisos correspondientes.
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>➤ El usuario debe haber iniciado sesión en el sistema.</li> <li>➤ Debe existir una solicitud de beca registrada y pendiente de revisión.</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>➤ El estado de la solicitud de beca se actualiza como "Aprobada" en el sistema.</li> <li>➤ El alumno recibe una notificación a través del Portal del Alumno, indicando la aprobación de su solicitud.</li> </ul>

**Escenario principal del éxito**

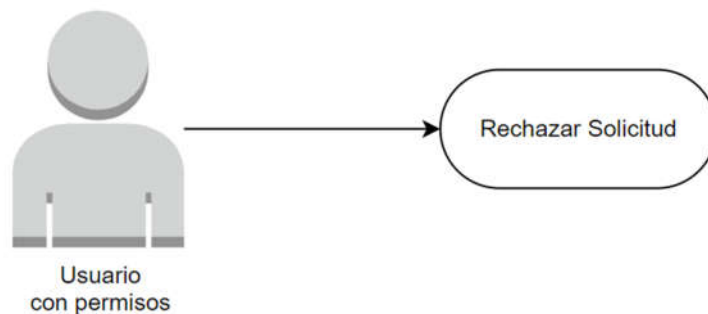
<b>Actor</b>	<b>Sistema</b>
1. Iniciar Caso de Uso	
2. Acceder a la funcionalidad de gestión de solicitudes en el sistema	
	3. Mostrar vista de solicitudes pendientes
4. Seleccionar solicitud de beca que desea aprobar	
	5. Mostrar los detalles de la solicitud, incluyendo la información del alumno y los documentos adjuntos.
6. Revisa la solicitud y decide aceptarla.	
	7. Actualizar estado de solicitud a aprobada.
	8. Emitir notificación por correo
	9. Actualiza estado de la solicitud en el portal del alumno.
	10. Mostrar mensaje de éxito
	11. Fin de Caso de Uso



<b>ID_CU</b>	Rechazar Solicitud
<b>Descripción</b>	Este caso de uso permite al usuario del departamento de becas, en el módulo de gestión, rechazar una solicitud de beca. Como resultado, el estado de la solicitud se actualizará y el alumno correspondiente recibirá una notificación y la verá reflejada en el Portal del Alumno.
<b>Actor</b>	Usuario del sistema con los permisos correspondientes.
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>➤ El usuario debe haber iniciado sesión en el sistema.</li> <li>➤ Debe existir una solicitud de beca registrada y pendiente de revisión.</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>➤ El estado de la solicitud de beca se actualiza como "Rechazada" en el sistema.</li> <li>➤ El alumno recibe una notificación y verá el estado en el Portal del Alumno, indicando el rechazo de su solicitud y mostrando el motivo proporcionado.</li> </ul>

**Escenario principal del éxito**

Actor	Sistema
1. Iniciar Caso de Uso	
2. Acceder a la funcionalidad de gestión de solicitudes en el sistema	
	3. Mostrar vista de solicitudes pendientes
4. Seleccionar solicitud de beca que desea rechazar	
	5. Mostrar los detalles de la solicitud, incluyendo la información del alumno y los documentos adjuntos.
6. Revisa la solicitud y decide rechazarla.	
7. Ingresar un motivo para el rechazo de la solicitud.	
	8. Actualizar estado de solicitud a rechazada.
	9. Emitir notificación por correo
	10. Actualiza estado de la solicitud en el portal del alumno.
	11. Mostrar mensaje de éxito
	12. Fin de Caso de Uso



<b>ID_CU</b>	Editar datos de contacto del alumno
<b>Descripción</b>	Este caso de uso permite al usuario con privilegios de administrador o personal autorizado editar los datos de contacto del alumno en el sistema de gestión de becas.
<b>Actor</b>	Administrador / personal autorizado
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>➤ El usuario debe haber iniciado sesión en el sistema.</li> <li>➤ El usuario debe tener asignado los permisos correspondientes</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>➤ Los datos de contacto del alumno se actualizan en el sistema.</li> <li>➤ Se registra la fecha y usuario de la actualización de los datos de contacto.</li> </ul>

**Escenario principal del éxito**

<b>Actor</b>	<b>Sistema</b>
1. Iniciar Caso de Uso	
2. Seleccionar opción de reporte de solicitudes	
3. Seleccionar opción para ver datos del alumno.	
	4. Mostrar vista de datos del alumno.
5. Accede a la funcionalidad de edición de datos de contacto del alumno	
	6. Habilitar campos (teléfono/ correo/ domicilio) para su edición
7. Realiza modificaciones necesarias en los campos de datos de contacto.	
8. Seleccionar opción para guardar cambios	
	9. Validar teléfono
	10. Validar correo
	11. Actualiza datos de contacto del alumno
	12. Mostrar mensaje de éxito
	13. Fin de Caso de Uso

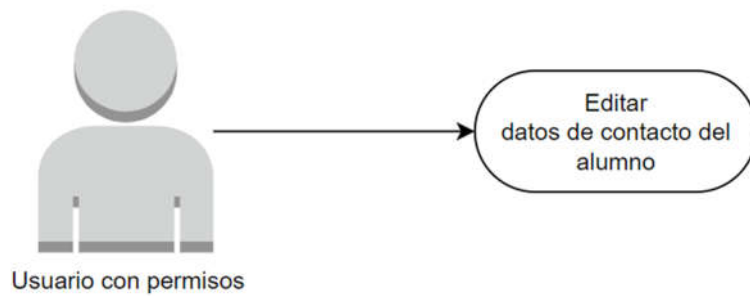
**Escenario de alternativas**

9. Validar teléfono (el teléfono tiene un formato incorrecto)

<b>Actor</b>	<b>Sistema</b>
	9.1 Mostrar mensaje debajo del campo erróneo “formato incorrecto”
9.2 Completar nuevamente	
	9.3 Volver al paso 8

10. Validar correo (el correo tiene un formato incorrecto)

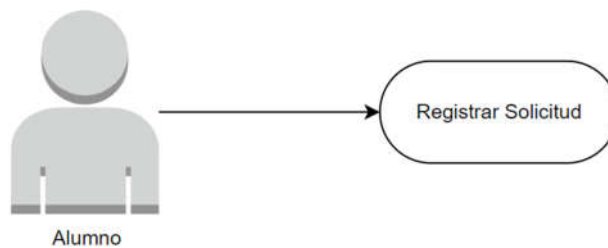
<b>Actor</b>	<b>Sistema</b>
	10.1 Mostrar mensaje debajo del campo erróneo “formato incorrecto”
10.2 Completar nuevamente	
	10.3 Volver al paso 8



<b>ID_CU</b>	Registrar Solicitud
<b>Descripción</b>	Este caso de uso permite al alumno aplicar a una beca a través del portal del alumno, siempre y cuando cumpla con los requerimientos mínimos establecidos, como la cantidad de materias y el promedio necesario para aplicar a la beca.
<b>Actor</b>	Alumno
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>➤ El alumno debe haber iniciado sesión en el SAG.</li> <li>➤ El alumno debe cumplir con los requerimientos mínimos para aplicar a la beca.</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>➤ La solicitud del alumno queda en estado pendiente y disponible para gestionarla desde el sistema de gestión de becas.</li> </ul>
<b>Escenario principal del éxito</b>	
<b>Actor</b>	<b>Sistema</b>
1. Iniciar Caso de Uso	
2. Ingresar al SAG (Sistema de Autogestión) de la UCASAL	
3. Seleccionar opción "Solicitud de beca"	
	4. Redirecciona al Portal del Alumno en una nueva pestaña.
	5. Validar requerimientos mínimos del alumno
	6. Mostrar listado de becas disponibles para el alumno
7. Seleccionar beca que desea aplicar	
	8. Validar que el alumno no haya aplicado a la beca previamente.
	9. Mostrar formulario de solicitud de beca.
	10. Mostrar datos generales del alumno a modo informativo (nombre, DNI, legajo, fecha de nacimiento)
	11. Solicitar completar datos de domicilio
12. Completar calle, numero, depto., piso, CP, país, localidad, provincia, barrio, teléfono, celular, email	
	13. Validar presencia de datos
	14. Validar formato de datos (teléfono, celular y email)
	15. Seleccionar opción de agregar

	domicilio.
Repetir pasos 10-14 hasta cargar los domicilios	
16. Seleccionar opción siguiente	
	17. Mostrar datos de información académica del alumno
18. Seleccionar carrera (en caso de tener más de una)	
	19. Mostrar datos relacionados a la carrera seleccionada.
20. Seleccionar opción siguiente	
	21. Solicitar completar datos laborables
22. Completar, en caso de tener un trabajo, datos laborables (tipo, rubro, empresa, CUIT, calle, numero, piso, depto., teléfono, barrio, CP, localidad, provincia, país, antigüedad, fecha de ingreso)	
23. Seleccionar opción siguiente	
	24. Validar formato numérico de datos
	25. Solicitar completar datos de grupo familiar
26. Completar nombre y apellido del familiar	
27. Seleccionar opción del vínculo del familiar (madre, padre, hijo/a, hermano/a, pareja, etc.)	
28. Seleccionar opción de agregar.	
Repetir pasos 25-27 hasta completar todo el grupo familiar	
29. Seleccionar opción siguiente	
	30. Solicitar completar datos de gastos
31. Completar montos de gastos	
32. Seleccionar opción siguiente	
	33. Validar presencia de datos
	34. Validar formato numérico de datos
	35. Solicitar completar datos de otras becas
36. Completar datos de otras becas	
37. Seleccionar opción de finalizar	
	38. Guardar registración de solicitud
	39. Generar un N° de transacción
	40. Mostrar mensaje de solicitud finalizada con éxito + N° de transacción
	41. Fin de Caso de Uso
<b>Escenario de alternativas</b>	
5. Validar requerimientos mínimos del alumno (cantidad de materias aprobadas, promedio)	
<b>Actor</b>	<b>Sistema</b>
	5.1 Mostrar mensaje “Usted no cumple con las condiciones necesarias para aplicar a una beca”
5.2 Seleccionar opción “Aceptar”	
	5.3 Volver al paso 2.

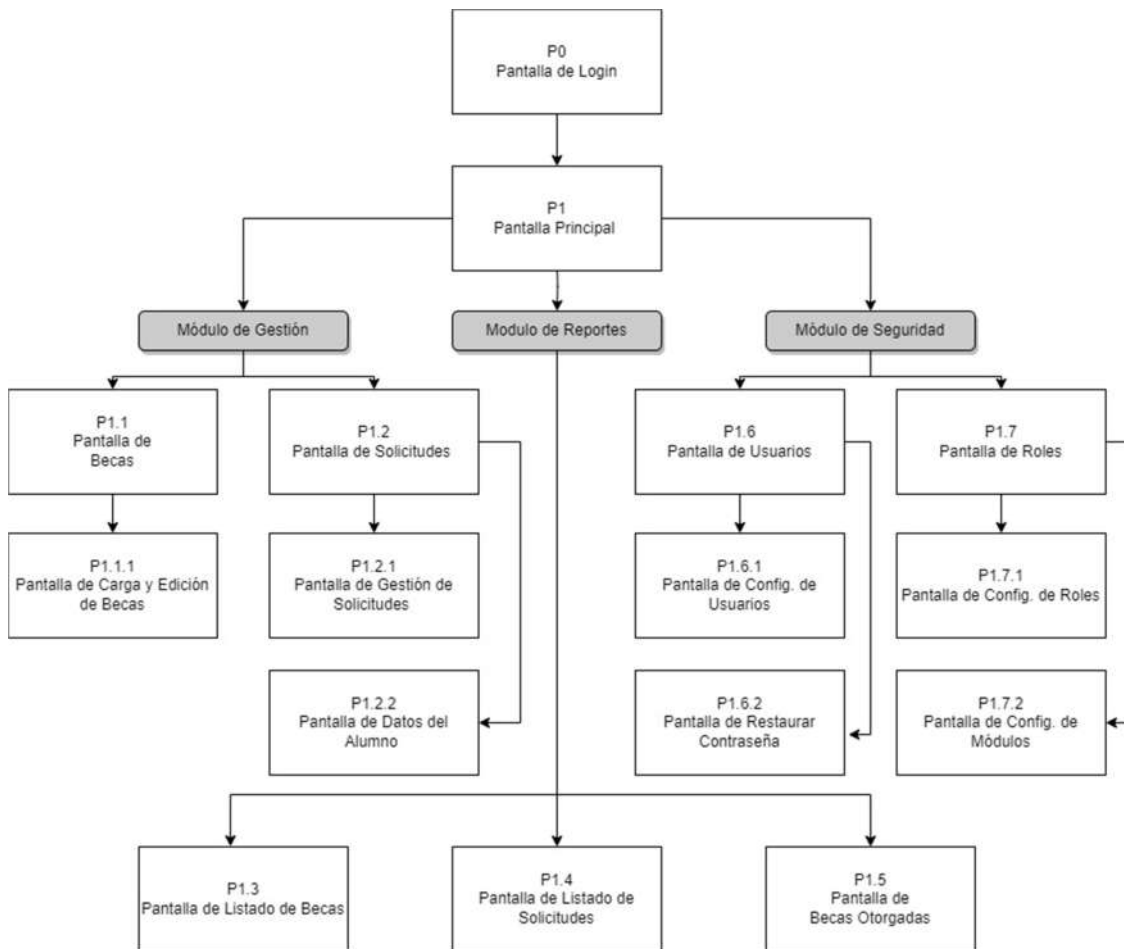
8. Validar que el alumno no tenga una solicitud activa en la beca.	
Actor	Sistema
	8.1 Mostrar mensaje “Ya posees una solicitud activa para esta beca.”
	8.2 Volver al paso 6
13. Validar presencia de datos (que se hayan ingresado todos los datos obligatorios)	
Actor	Sistema
	13.1 Mostrar error “Debe ingresar...”
	13.2 Volver al paso 11.
14. Validar formato de datos (teléfono y mail)	
Actor	Sistema
	14.1 Mostrar error “Formato incorrecto”
	14.2 Volver al paso 11.
24. Validar formato numérico de datos (CUIT, numero, piso, teléfono, antigüedad)	
Actor	Sistema
	24.1 Mostrar error “Debe ingresar solo números”
	24.2 Volver al paso 21.
33. Validar presencia de datos (ingresar monto de gasto en alimentos, servicios, educación, salud, transporte y otros gastos)	
Actor	Sistema
	33.1 Mostrar error “Debe ingresar un monto”
	33.2 Volver al paso 30.
34. Validar formato numérico de datos	
Actor	Sistema
	34.1 Mostrar error “Debe ingresar solo números”.
	34.2 Volver al paso 30.



### 8.2.1.2 Flujo de Pantallas

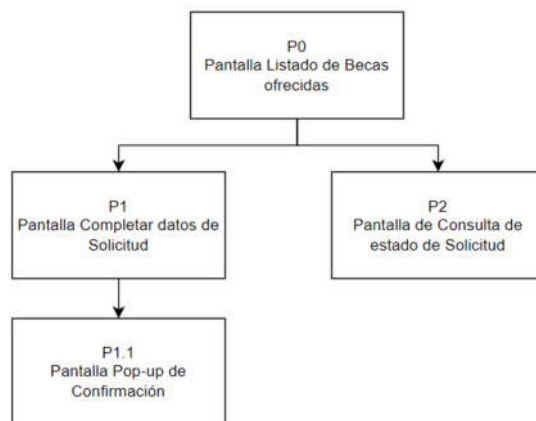
A continuación, se detallará la navegación entre las distintas pantallas que confirman la interfaz del sistema, tanto para el Módulo de Administración como para el Portal del Alumno.

#### Módulo de Administración



*Ilustración 5: Flujo de pantallas Modulo de Administración. Elaboración propia.*

## Portal del Alumno



*Ilustración 6: Flujo de pantallas Portal del Alumno. Elaboración propia*

### 8.2.1.3 Diseño de Interfaces

A continuación, se presentan los diseños de interfaces del Módulo de Administración, también conocidos como "Mockups", junto con sus correspondientes referencias y observaciones.

Pantalla de Becas	P1.1																																								
<div style="border: 1px solid black; padding: 10px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <h2 style="margin: 0;">Carga de Becas</h2> </div> <div style="text-align: right; margin-top: 10px;"> <span style="border: 1px solid blue; padding: 2px 10px; color: blue; font-weight: bold;">Agregar Beca</span> </div> <div style="margin-top: 10px;"> <div style="text-align: center; border-bottom: 1px solid black; padding-bottom: 5px;"> <span style="font-weight: bold;">Listado de Becas</span> </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 15%;">Tipo de Beca</th> <th style="width: 30%;">Destinatario</th> <th style="width: 10%;">Beneficio</th> <th style="width: 15%;">Descripción</th> <th style="width: 5%;">Prom Mínimo</th> <th style="width: 5%;">Cant Materias</th> <th style="width: 5%;">Facultad</th> <th style="width: 5%;">Carrera</th> <th style="width: 10%;">Fecha Vigencia</th> </tr> </thead> <tbody> <tr> <td style="text-align: left; vertical-align: middle;">1 ← </td> <td style="text-align: left;">Beca Abanderado</td> <td style="text-align: left; font-size: 0.8em;">Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.</td> <td>50%</td> <td>De la cuota mensual</td> <td>7</td> <td>40</td> <td>Todas</td> <td>-</td> <td>30-06-2023</td> </tr> <tr> <td style="text-align: left; vertical-align: middle;"></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td style="text-align: left; vertical-align: middle;"></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> </div> </div>		Tipo de Beca	Destinatario	Beneficio	Descripción	Prom Mínimo	Cant Materias	Facultad	Carrera	Fecha Vigencia	1 ←	Beca Abanderado	Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.	50%	De la cuota mensual	7	40	Todas	-	30-06-2023																					
	Tipo de Beca	Destinatario	Beneficio	Descripción	Prom Mínimo	Cant Materias	Facultad	Carrera	Fecha Vigencia																																
1 ←	Beca Abanderado	Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.	50%	De la cuota mensual	7	40	Todas	-	30-06-2023																																
<p>Observaciones:</p> <ol style="list-style-type: none"> <li>El ícono editar que se encuentra dentro del reporte de listado de becas, por cada registro, abre la pantalla pop-up de edición de Becas (P 1.1.1), al igual que el botón “Agregar Beca”, con la diferencia que al editar ya se mostrará en pantalla todos los datos previamente cargados para su edición.</li> </ol>																																									

Editar/Cargar Beca

Tipo de Beca

Beneficio

Prom. Mínimo

Cant. Materias

Destinatario

Activo

Fecha Desde

Fecha Hasta

Facultad

Carrera

Descripción

Cancelar → 2

Guardar → 1

Observaciones:

1. El botón “**Guardar**”, guarda cambios y cierra la ventana modal (pop-up). Rellenando así el reporte de listado de becas con lo agregado o editado, mostrando un mensaje “Beca agregada exitosamente” en el caso que se cargue una nueva, si no el mensaje será “Beca editada exitosamente”.
2. El botón “**Cancelar**”, no guarda cambios y cierra la ventana modal (pop-up).
3. Lista de valores: mostrará el listado de Facultades, al igual que el de Carreras, cuyos datos se obtendrán desde la API solicitada.
4. Campo tipo fecha: despliega un calendario para poder seleccionar una fecha específica.

Pantalla de Solicitudes						P
						1.2
<b>Observaciones:</b>						
1. El botón “ <b>Filtrar</b> ” carga el reporte de listado de solicitudes, los datos que concuerdan con los ítems rellenados.						


Pantalla Pop-up de Gestión de Solicitudes	P 1.2.1
<div data-bbox="336 465 1211 775" style="border: 1px solid black; border-radius: 15px; padding: 20px; text-align: center;"><p>¿Desea Aprobar la solicitud de beca realizada por el alumno XXXXX?</p><div style="display: flex; justify-content: space-around; margin-top: 10px;"><span data-bbox="501 689 647 734" style="border: 1px solid red; border-radius: 5px; padding: 5px 15px;">Rechazar</span> → 2      <span data-bbox="896 689 1043 734" style="border: 1px solid blue; border-radius: 5px; padding: 5px 15px;">Aprobar</span> → 1</div></div>	
<p>Observaciones:</p> <ol style="list-style-type: none"><li>1. El botón “<b>Aprobar</b>” aprueba la solicitud del alumno, enviándole una notificación y cambiando el estado de la solicitud a “Aprobada”. Luego se cierra la pantalla pop-up.</li><li>2. El botón “<b>Rechazar</b>” rechaza la solicitud del alumno, enviándole una notificación y cambiando el estado de la solicitud a “Rechazada”. Luego se cierra.</li></ol>	

Observaciones:

1. Presionando el ícono del lápiz habilita el campo para su edición. Solo están para teléfono, correo y domicilio.
2. El botón “**Guardar cambios**” será para guardar los campos editables únicamente. Mostrará un mensaje en pantalla “Cambios guardados exitosamente”.
3. La pantalla constará con diferentes pestañas para visualizar toda la información cargada por el alumno.

<b>Pantalla de Reporte de Listado de Becas</b>	<b>P 1.3</b>																																													
<div style="border: 1px solid black; padding: 10px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="margin-left: 20px;"> <h3 style="margin: 0;">Reporte de Listado de Becas</h3> </div> </div> <div style="margin-top: 10px; border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="display: flex; gap: 10px;"> <input type="text" value="Tipo de Beca"/> <input type="text" value="Facultad"/> <input type="text" value="Carrera"/> <input type="text" value="Fecha Vigencia"/> <input type="text" value="Promedio"/> </div> <div style="text-align: right;"> <span>1 ←</span> <span style="background-color: #007bff; color: white; padding: 2px 5px; border-radius: 3px;">Filtrar</span> </div> </div> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr style="background-color: #f2f2f2;"> <th colspan="9">Listado de Becas</th> </tr> <tr> <th style="width: 10%;">Tipo de Beca</th> <th style="width: 20%;">Destinatario</th> <th style="width: 10%;">Beneficio</th> <th style="width: 15%;">Descripción</th> <th style="width: 5%;">Prom Mínimo</th> <th style="width: 5%;">Cant Materias</th> <th style="width: 5%;">Facultad</th> <th style="width: 5%;">Carrera</th> <th style="width: 10%;">Fecha Vigencia</th> </tr> </thead> <tbody> <tr> <td>Beca Abanderado</td> <td style="font-size: 0.8em;">Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.</td> <td style="text-align: center;">50%</td> <td style="text-align: center;">De la cuota mensual</td> <td style="text-align: center;">7</td> <td style="text-align: center;">40</td> <td style="text-align: center;">Todas</td> <td style="text-align: center;">-</td> <td style="text-align: center;">30-06-2023</td> </tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> </div>		Listado de Becas									Tipo de Beca	Destinatario	Beneficio	Descripción	Prom Mínimo	Cant Materias	Facultad	Carrera	Fecha Vigencia	Beca Abanderado	Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.	50%	De la cuota mensual	7	40	Todas	-	30-06-2023																		
Listado de Becas																																														
Tipo de Beca	Destinatario	Beneficio	Descripción	Prom Mínimo	Cant Materias	Facultad	Carrera	Fecha Vigencia																																						
Beca Abanderado	Este beneficio está destinado a alumnos ingresantes que egresan el año anterior a su ingreso a la universidad como abanderados de la enseñanza nacional. Se concede a 10 estudiantes con los mejores promedios generales entre los postulados.	50%	De la cuota mensual	7	40	Todas	-	30-06-2023																																						
<p><b>Observaciones:</b></p> <ol style="list-style-type: none"> <li>La interfaz dispone de una sección de filtros para visualizar la información según el campo deseado.</li> </ol>																																														

<b>Pantalla de Reporte de Solicitudes</b>	<b>P 1.4</b>																																													
<div style="border: 1px solid black; padding: 10px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="margin-left: 20px;"> <h3 style="margin: 0;">Reporte de Listado de Solicitudes</h3> </div> </div> <div style="margin-top: 10px; border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="display: flex; gap: 10px;"> <input type="text" value="Tipo de Beca"/> <input type="text" value="Facultad"/> <input type="text" value="Carrera"/> <input type="text" value="Legajo"/> <input type="text" value="DNI Alumno"/> </div> <div style="text-align: right;"> <span>→ 1</span> <span style="background-color: #007bff; color: white; padding: 2px 5px; border-radius: 3px;">Filtrar</span> </div> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> <input type="text" value="Fecha Solicitud"/> <input type="text" value="Estado"/> </div> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr style="background-color: #f2f2f2;"> <th colspan="9">Listado de Solicitudes del Alumno</th> </tr> <tr> <th style="width: 10%;">Tipo de Beca</th> <th style="width: 10%;">Nombre</th> <th style="width: 10%;">Apellido</th> <th style="width: 10%;">Legajo</th> <th style="width: 10%;">DNI</th> <th style="width: 10%;">Fecha Solicitud</th> <th style="width: 10%;">Facultad</th> <th style="width: 10%;">Carrera</th> <th style="width: 10%;">Estado</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> </div>		Listado de Solicitudes del Alumno									Tipo de Beca	Nombre	Apellido	Legajo	DNI	Fecha Solicitud	Facultad	Carrera	Estado																											
Listado de Solicitudes del Alumno																																														
Tipo de Beca	Nombre	Apellido	Legajo	DNI	Fecha Solicitud	Facultad	Carrera	Estado																																						
<p><b>Observaciones:</b></p> <ol style="list-style-type: none"> <li>La interfaz dispone de una sección de filtros para visualizar la información según el campo deseado.</li> </ol>																																														

**Reporte de Becas Otorgadas** 





Tipo de Beca Facultad Carrera Legajo  
Aprobador Fecha Desde Fecha Hasta  1

Tipo de Beca	Facultad	Carrera	Legajo	Alumno	Fecha	Aprobador

Observaciones:

1. La interfaz dispone de una sección de filtros para visualizar la información según el campo deseado.

A continuación, se detallan las pantallas del Módulo de Seguridad.

Pantalla de Usuarios						P 1.6
<h3>Listado de Usuarios</h3> 						
<a href="#">Crear Usuario</a>						
(P 1.6.1) ←						
	Nombre	Apellido	Username	Correo	Password	Activo
					<a href="#">Restaurar Contraseña</a>	
					<a href="#">Restaurar Contraseña</a>	1
					<a href="#">Restaurar Contraseña</a>	
↓ (P 1.6.2)						

Observaciones:

1. “Restaurar contraseña”: te brinda la posibilidad de cambiar tu clave de acceso.

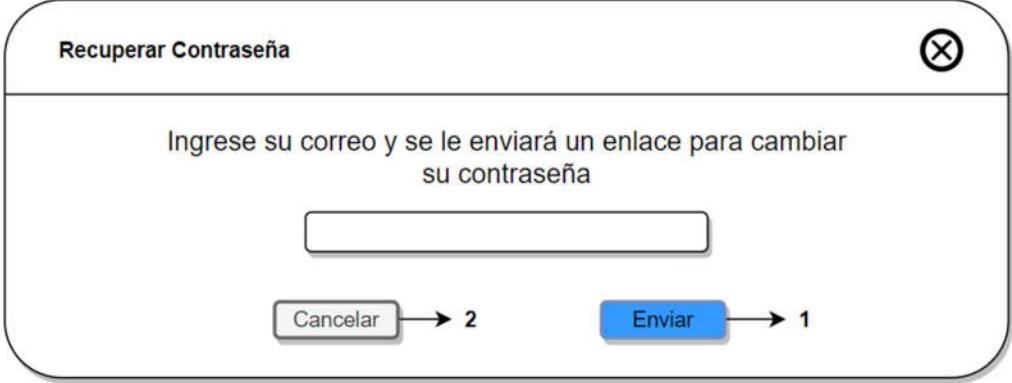
Pantalla Pop up de Configuración De Usuarios P 1.6.1

The screenshot shows a pop-up window titled "Agregar/Editar Usuario" with a close button in the top right corner. The window contains the following elements:

- Input fields for "Nombre", "Apellido", "Username" (two separate fields), "Correo", and "Activo" (a dropdown menu).
- Two "Roles" selection boxes with arrows between them, labeled "4".
- An "Activo" toggle switch.
- Buttons for "Cancelar" (labeled "2") and "Guardar" (labeled "1").
- A "3" is located near the "Activo" dropdown menu.


Observaciones:

1. Guarda cambios y cierra ventana pop up.
2. Cierra ventana pop-up.
3. Despliega lista de valores “Si, No”.
4. Selección de roles para el usuario.

Pantalla Pop up de Restaurar Contraseña	P 1.6.2
	
<p>Observaciones:</p> <ol style="list-style-type: none"><li>1. Se le envía un enlace al correo para recuperar contraseña. Cierra ventana pop-up</li><li>2. Cierra ventana pop-up.</li></ol>	

Pantalla de Roles	P 1.7
-------------------	-------

### Listado de Roles

  
Crear Rol → (P 1.7.1)


Nombre	Módulos
ADMINISTRADOR	<span style="border: 1px solid gray; padding: 2px 10px;">Módulos</span>
ADMINISTRATIVO	<span style="border: 1px solid gray; padding: 2px 10px;">Módulos</span> → 1 ↓ (P 1.7.2)

Observaciones:

1. Abre una ventana pop up de listado de módulos para permitir seleccionar los módulos disponibles para cada rol.

Pantalla Pop up de Configuración de Roles	P 1.7.1
---	---------

### Agregar/Editar Roles



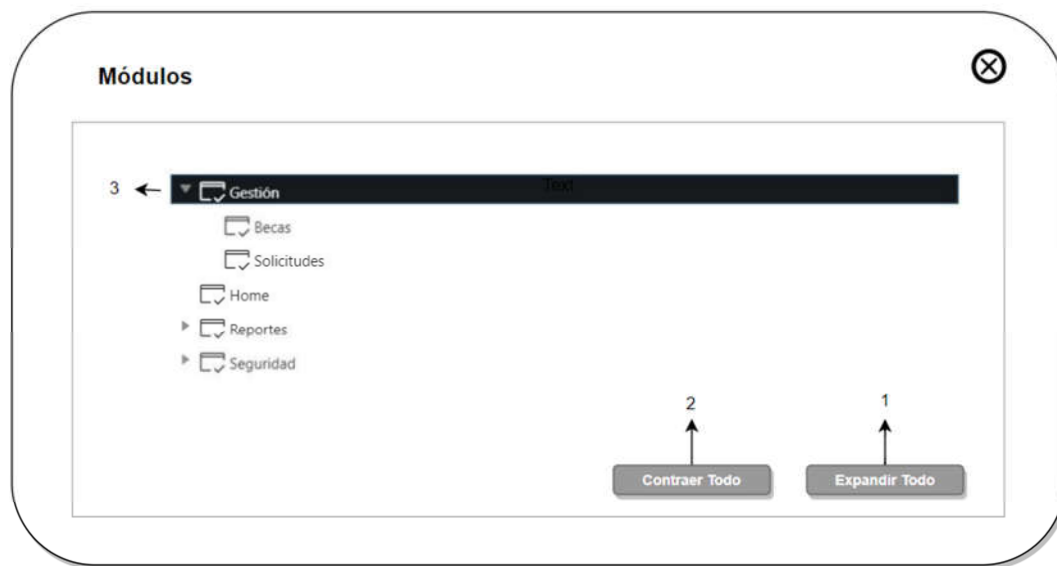
Nombre del Rol:

Cancelar → 2      Guardar → 1

Observaciones:

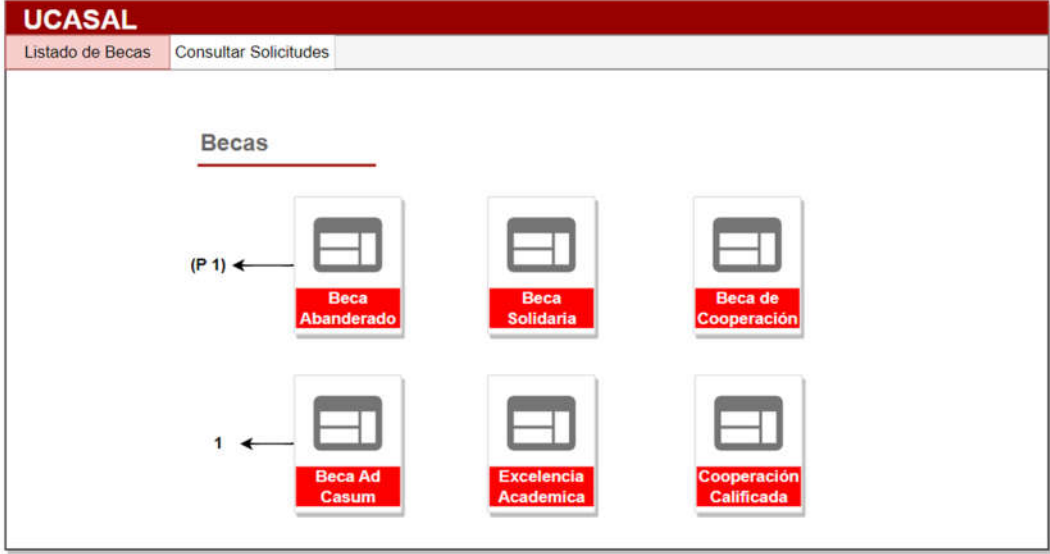
1. Guarda un rol nuevo o editado. Cierra ventana pop up.
2. Cierra ventana pop-up.



Observaciones:

1. Expande toda la lista de módulos del sistema
2. Contrae toda la lista de módulos del sistema
3. Haciendo doble click se visualiza o se oculta el módulo para el rol del usuario

A continuación, se presentan los diseños de interfaces (Mockups) del Portal del Alumno, junto con sus correspondientes referencias y observaciones.

Pantalla del Portal del Alumno	P 0
	
<p>Observaciones:</p> <ol style="list-style-type: none"><li>1. Las becas disponibles para el alumno serán presentadas en forma de tarjetas (cards) en la interfaz de la aplicación.</li></ol>	

**UCASAL**  
Completar Solicitud de Beca

Datos Grales Domicilio Info Académica Datos Laborables Grupo Familiar Gastos Otras Becas Resumen

Siguiete → 1

2

**UCASAL**  
Completar Solicitud de Beca

Datos Grales Domicilio Info Académica Datos Laborables Grupo Familiar Gastos Otras Becas Resumen

Finalizar → (P 1.1)

3

Observaciones:

1. Consiste en una secuencia de interfaces con ítems agrupados por categoría, presionando el botón “Siguiete” se accede a la siguiente categoría para completar los datos requeridos.
2. Se mostrarán los grupos de datos que se deberán cargar por categoría.
3. Una vez completado todo el formulario, se previsualiza toda la información cargada a modo de Resumen.

Pantalla Pop-up de Confirmación de Solicitud	P 1.1
<div data-bbox="391 286 1173 571" style="border: 1px solid black; border-radius: 15px; padding: 20px; text-align: center;"><p>¿Está seguro que desea enviar la solicitud?</p><div style="display: flex; justify-content: space-around; margin-top: 10px;"><span data-bbox="539 488 671 533" style="border: 1px solid gray; border-radius: 5px; padding: 2px 10px;">Cancelar</span> → 2      <span data-bbox="890 488 1023 533" style="background-color: #4a86e8; color: white; border-radius: 5px; padding: 2px 10px;">Confirmar</span> → 1</div></div>	
<p>Observaciones:</p> <ol style="list-style-type: none"><li>1. Envía la solicitud en estado “Pendiente” y genera un N° de Solicitud que se mostrará en pantalla (P 1) una vez confirmada la operación. Cierra ventana Pop-up.</li><li>2. Cierra la ventana pop-up.</li></ol>	

Pantalla de Solicitudes	P 2																
<div data-bbox="236 1064 1310 1630" style="border: 1px solid gray; padding: 10px;"><div style="background-color: #800000; color: white; padding: 5px;"><b>UCASAL</b></div><div style="background-color: #f0f0f0; padding: 5px; margin-top: 5px;"><span>Listado de Becas</span>   <span style="background-color: #f08080; padding: 2px 5px;">Consultar Solicitudes</span></div><div style="margin-top: 20px;"><p><u>Mis solicitudes</u></p><table border="1" data-bbox="363 1272 1203 1496" style="width: 100%; border-collapse: collapse;"><thead><tr style="background-color: #f08080;"><th>Nro de Solicitud</th><th>Nombre de Beca</th><th>Fecha de Solicitud</th><th>Estado</th></tr></thead><tbody><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr></tbody></table></div></div>		Nro de Solicitud	Nombre de Beca	Fecha de Solicitud	Estado												
Nro de Solicitud	Nombre de Beca	Fecha de Solicitud	Estado														

### 8.3 Fase de Construcción

Para la construcción de la aplicación web se emplea la interfaz de desarrollo Low-Code, Oracle Application Express (APEX) en su versión 23.1.0-16, mientras que para la base de datos se utiliza Oracle SQL 19C. Además, para el consumo de Web Services se utiliza los servicios RESTFUL nativos de APEX.

Es importante destacar que la cuenta vinculada para realizar este prototipo es una cuenta gratuita. Si bien la misma no posee limitaciones en cuanto al desarrollo, su base de datos no es configurable en modo administrador. Esta limitación no impide la creación de secuencias, tablas, paquetes, etc.

#### 8.3.1 Implementación de la arquitectura lógica del sistema

La arquitectura lógica del sistema se compone de los siguientes elementos y sus relaciones correspondientes:

- **Interfaz de usuario:** es la capa que permite la interacción del usuario con la aplicación.
- **Desarrollo de la aplicación:** se encarga del desarrollo y programación de la aplicación web en Oracle APEX.
- **Gestión y almacenamiento de datos:** abarca la gestión, almacenamiento y acceso a los datos en la base de datos Oracle SQL 19C.
- **Gestión y administración:** es el componente responsable de la gestión y administración de las solicitudes y procesos de otorgamiento de becas en la aplicación web.

#### 8.3.2 Diseño y construcción de la base de datos

Se lleva a cabo la construcción física de la base de datos en SQL Oracle 19C. Esta tarea implica la creación de tablas, vistas, secuencias y restricciones necesarios para la implementación del modelo de datos diseñado en la fase de análisis.

Durante esta etapa, se establecen las relaciones entre las tablas y se definen las claves primarias y foráneas que permiten mantener la integridad de la base de datos. Además, se emplean las reglas de normalización con el fin de asegurar que la información alojada en la base de datos se encuentre libre de redundancias y anomalías, contribuyendo así a la integridad y calidad de los datos almacenados.

➤ **Creación de tablas y campos**

A continuación, se puede ver la sentencia SQL “CREATE TABLE” para la creación de tablas, en este caso es para la tabla “BECAS” la cual contiene definido los nombres de los campos, tipo de dato de cada uno, tamaño del campo, si permite nulo o no, restricciones, etc. Además, se define una restricción llamada "BECAS\_PK" que establece que el campo "ID\_BECA" es la clave primaria de la tabla. Se utiliza un índice para mejorar el rendimiento de las consultas en esta clave primaria.

```

1 CREATE TABLE "BECAS"
2 (
3   "ID_BECA" NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY MINVALUE 1 MAXVALUE 99999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,
4   "NOMBRE" VARCHAR2(100 CHAR) NOT NULL ENABLE,
5   "DESTINATARIO" VARCHAR2(1000 CHAR) NOT NULL ENABLE,
6   "DESCRIPCION" VARCHAR2(500 CHAR),
7   "PROM_MINIMO" NUMBER,
8   "CANT_MATERIAS" NUMBER,
9   "ACTIVO" NUMBER NOT NULL ENABLE,
10  "FACULTAD" VARCHAR2(100 CHAR),
11  "CARRERA" VARCHAR2(100 CHAR),
12  "F_VIGENCIA_DESDE" DATE NOT NULL ENABLE,
13  "F_VIGENCIA_HASTA" DATE NOT NULL ENABLE,
14  "BENEFICIO" VARCHAR2(45),
15  CONSTRAINT "BECAS_PK" PRIMARY KEY ("ID_BECA")
16  USING INDEX ENABLE
17 );
    
```

*Ilustración 7: Creación tabla Becas*

Oracle APEX también permite visualizar la tabla de una manera más intuitiva como podemos ver en la siguiente ilustración:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default
BECAS	ID_BECA	NUMBER	22	-	-	1	-	"WKSP_MILUCASAL"."ISEQ\$\$_186726529".nextval
	NOMBRE	VARCHAR2	100	-	-	-	-	-
	DESTINATARIO	VARCHAR2	1000	-	-	-	-	-
	DESCRIPCION	VARCHAR2	500	-	-	-	✓	-
	PROM_MINIMO	NUMBER	22	-	-	-	✓	-
	CANT_MATERIAS	NUMBER	22	-	-	-	✓	-
	ACTIVO	NUMBER	22	-	-	-	-	-
	FACULTAD	VARCHAR2	100	-	-	-	✓	-
	CARRERA	VARCHAR2	100	-	-	-	✓	-
	F_VIGENCIA_DESDE	DATE	7	-	-	-	-	-
	F_VIGENCIA_HASTA	DATE	7	-	-	-	-	-
	BENEFICIO	VARCHAR2	45	-	-	-	✓	-

*Ilustración 8: Tabla de Becas*

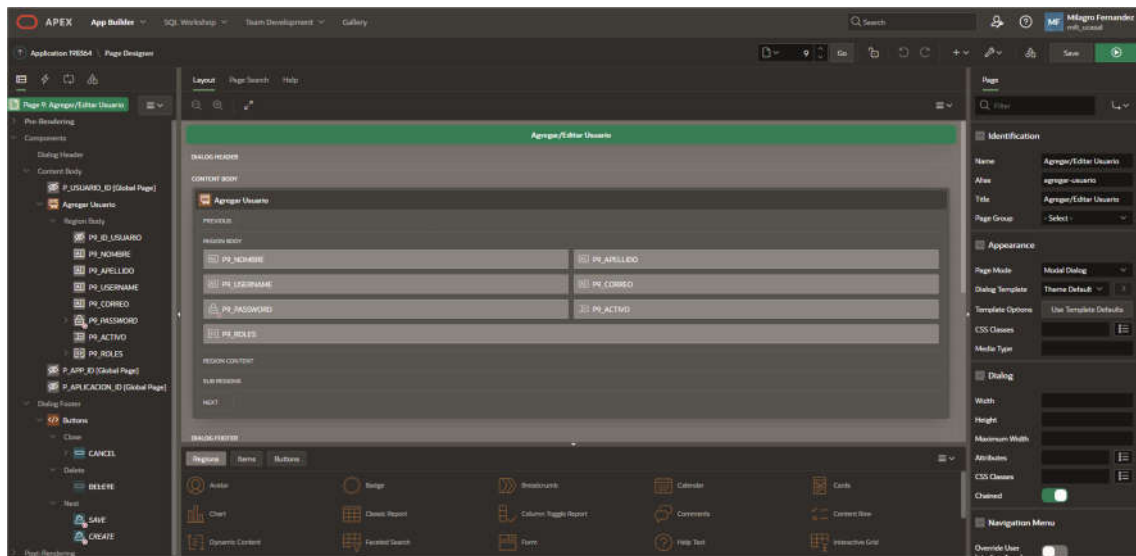
En la ilustración 17 se muestra el proceso de creación de la tabla "SOLICITUDES", con sus respectivos campos, definiciones y restricciones. La sentencia ALTER TABLE se utiliza para realizar modificaciones en la estructura de una tabla tente.

En este caso, se utilizó la sentencia ALTER TABLE para agregar claves foráneas a los campos ID\_ALUMNO, ID\_BECA e ID\_USUARIO en la tabla.



En la ilustración 19, se presenta la interfaz de desarrollo de Oracle Application Express (APEX). En la misma se puede apreciar el desarrollo de la pantalla de inicio de sesión (Login) de la aplicación de gestión de becas. A la izquierda visualizamos los ítems que compone la pantalla, en este caso, el nombre de usuario, la contraseña y el botón del Login. En la región central se presenta la ubicación de estos ítems en pantalla. Y a la derecha los atributos de cada ítem.

## Pantalla de creación/edición de usuarios



*Ilustración 12: Pantalla de carga/edición de usuarios*

En la ilustración 20, se puede visualizar el desarrollo de la pantalla de creación/edición de usuarios. En este caso, se puede observar que la pantalla está compuesta por los siguientes ítems: ID del usuario (oculto), nombre, apellido, nombre de usuario, correo, contraseña (encriptada), activo. En cuanto a los botones están el de Cancelar (“CANCEL”), Eliminar (ítem oculto “DELETE”), luego, los botones Guardar (“SAVE”) y Crear (“CREATE”) se mostrarán dependiendo si el usuario existe o no.

A continuación, en la ilustración 20 se visualiza a la izquierda el listado de procesos y a la derecha vemos el código del proceso seleccionado “*Process from Agregar Usuario*” en lenguaje PL/SQL con sus comentarios correspondientes.

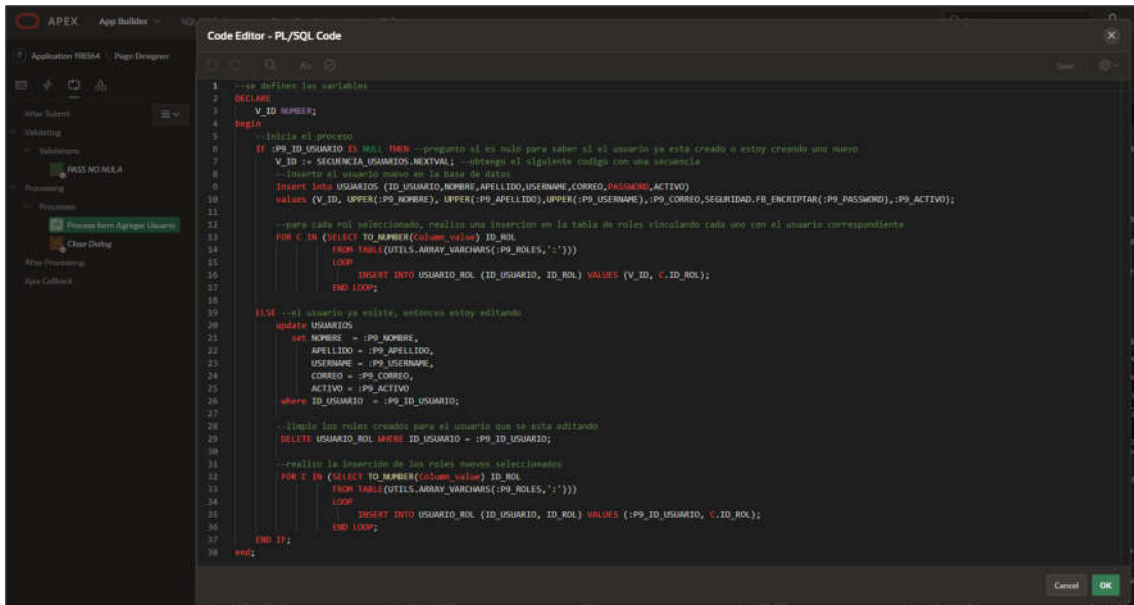


Ilustración 13: Código PL/SQL para el proceso Agregar Usuario

Este procedimiento comienza definiendo las variables que se utilizarán a lo largo del proceso. A continuación, se inicia el proceso identificando si la variable ":P9\_ID\_USUARIO" es nula. Si es nula, implica que se está creando un nuevo usuario, por lo que se realiza la inserción en la tabla "USUARIOS" y se ejecuta un ciclo "FOR" para asignar los roles correspondientes. En cambio, si la variable no es nula, se trata de una edición de usuario, por lo que se actualiza la tabla "USUARIOS" y se eliminan los roles asignados previamente para seleccionar nuevos.

## Pantalla de Listado de Becas Otorgadas

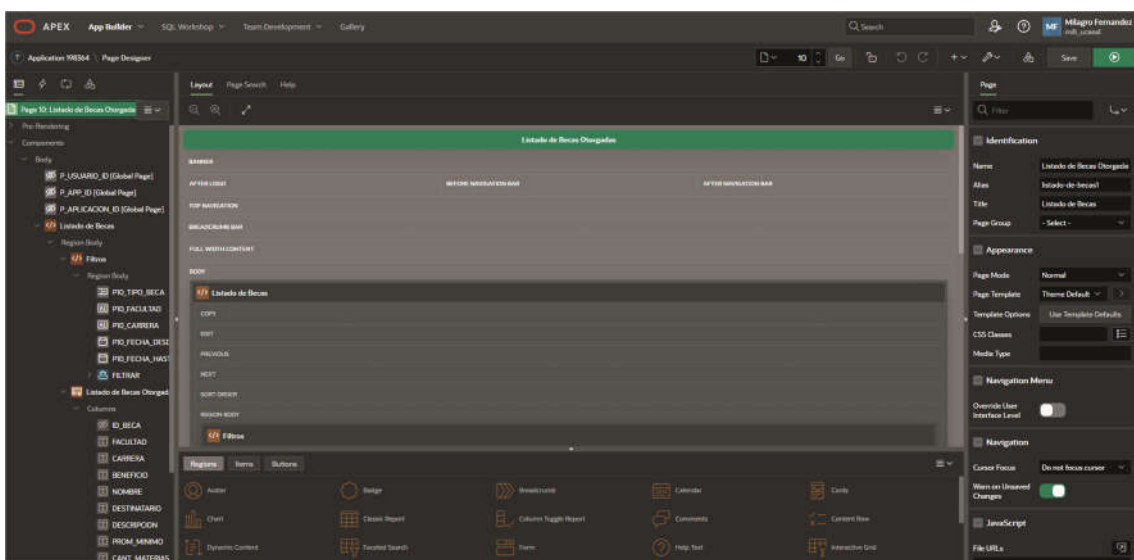
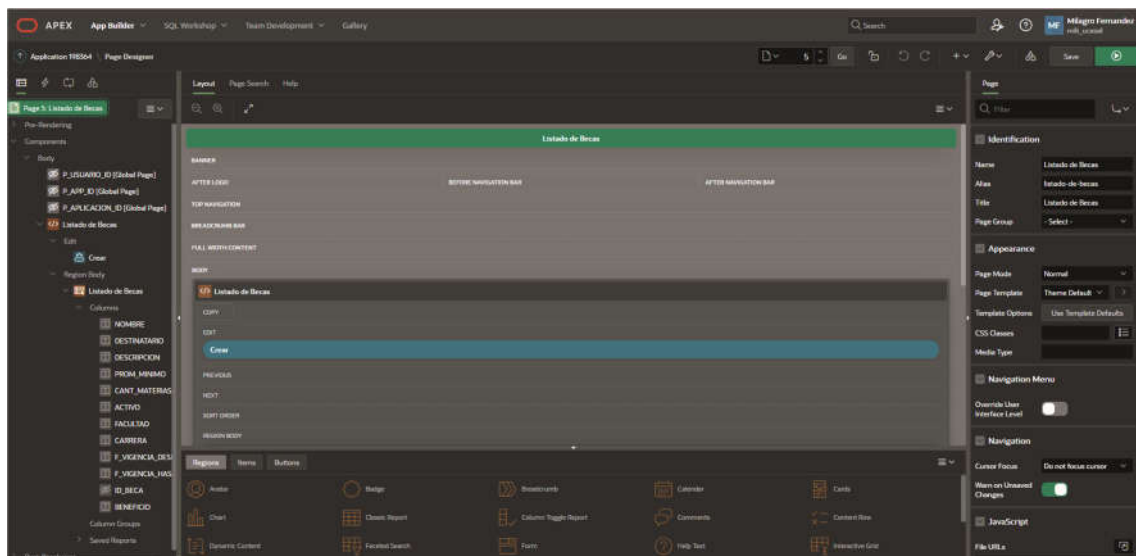


Ilustración 14: Pantalla de Listado de Becas Otorgadas

En la ilustración 21 se observa la pantalla de Becas Otorgadas. La misma posee una región denominada “Listado de Becas Otorgadas” de tipo reporte interactivo, que contiene una consulta a la base de datos para mostrar esta información. Los datos se muestran en forma de tabla y se pueden realizar diversas operaciones, como ordenar por columna, filtrar registros basados en criterios específicos, agregar totales y subtotales, exportar datos en diferentes formatos (Excel o PDF). En la parte superior de esta región, se encuentran los elementos utilizados como filtros, junto con el botón "FILTRAR" que permite actualizar el reporte en base a los criterios seleccionados.

### **Pantalla de Listado de Becas**



*Ilustración 15: Pantalla de Listado de Becas*

En la ilustración 22 representa la pantalla de Listado de Becas de tipo reporte interactivo. Esta pantalla proporciona la visualización todas las becas cargadas en el sistema por parte del personal del departamento de becas. Al acceder a esta pantalla, los usuarios podrán ver una lista completa de todas las becas disponibles y visibles para el alumno, según condiciones necesarias. También se observa el botón “Crear” el cual abre la pantalla de carga de becas.

➤ **Consumo de API**

```
declare
  pJson CLOB;
  vJson CLOB;
begin
  pJson := '{
    "legajo": '||v_legajo||'
  }';

  --obtengo datos del api de credit para validar
  v_token_seg := seguridad.get_token_apis;
  apex_web_service.g_request_headers(1).name := 'Content-Type';
  apex_web_service.g_request_headers(1).value := 'application/json';
  apex_web_service.g_request_headers(2).name := 'Authorization';
  apex_web_service.g_request_headers(2).value := 'Bearer ' ||v_token_seg;
  vjson := apex_web_service.make_rest_request(
    p_url => 'https://www.ucasal.edu.ar/apis/GetCarreras',
    p_http_method => 'POST',
    p_transfer_timeout => 10,
    p_body => pJson);

  apex_json.parse(vjson);
```

*Ilustración 16: Código PL/SQL para consumo de API GetCarreras*

En la imagen previa se muestra el código utilizado para consumir las API proporcionadas por la facultad. En este caso específico, se realiza una consulta a la API **GetCarreras** para obtener datos sobre las facultades y las carreras asociadas a ellas. Para lograr esto, se utiliza la librería `apex_web_service`, que contiene procedimientos y funciones para invocar servicios web en estilo SOAP y RESTful. En el ejemplo, `g_request_headers` se utiliza para determinar el nombre y el valor de una cabecera de información, mientras que la función `make_rest_request` construye la solicitud con los parámetros establecidos (URL, método de consulta, cuerpo de la solicitud, tiempo de espera, entre otros).

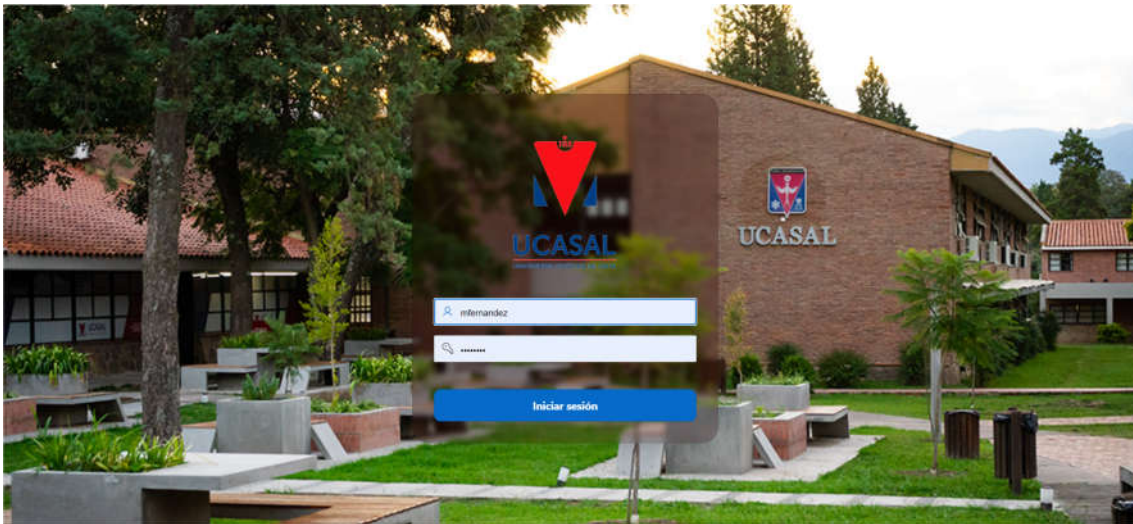
### 8.3.4 Interfaces del sistema

Se presentan las imágenes de las interfaces del sistema de gestión. Estas interfaces fueron diseñadas y desarrolladas para facilitar la interacción de los usuarios con el sistema, brindando una experiencia intuitiva y eficiente.

#### Módulo de Gestión

➤ Pantalla de Login

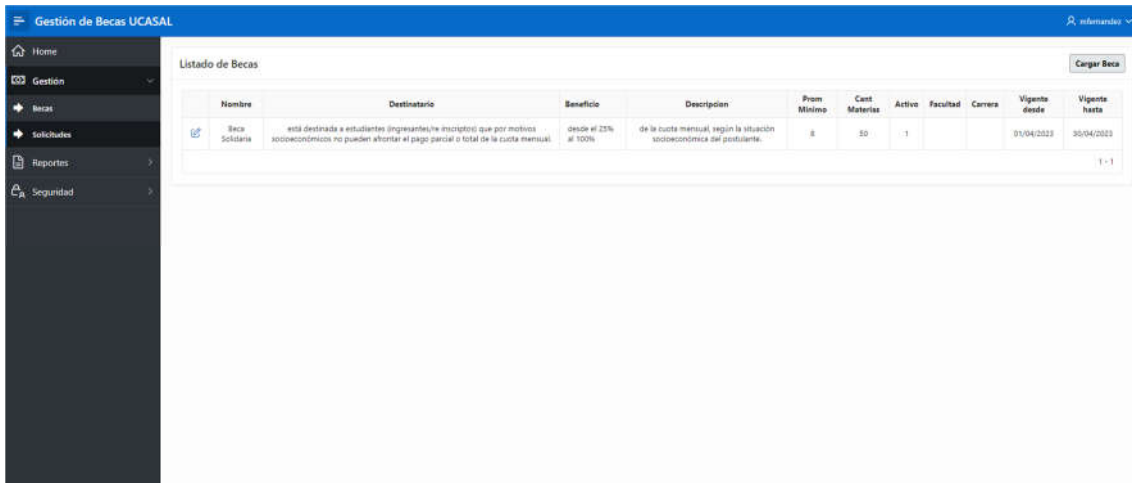
En la siguiente ilustración se visualiza la pantalla de Login donde el usuario, previamente registrado, debe ingresar su usuario y contraseña para acceder.



*Ilustración 17: Pantalla de Login*

➤ Pantalla de Listado de Becas

En la ilustración se muestra la pantalla donde el usuario (personal del departamento de becas) puede observar el listado de becas existentes, editarlas y cargar una nueva.

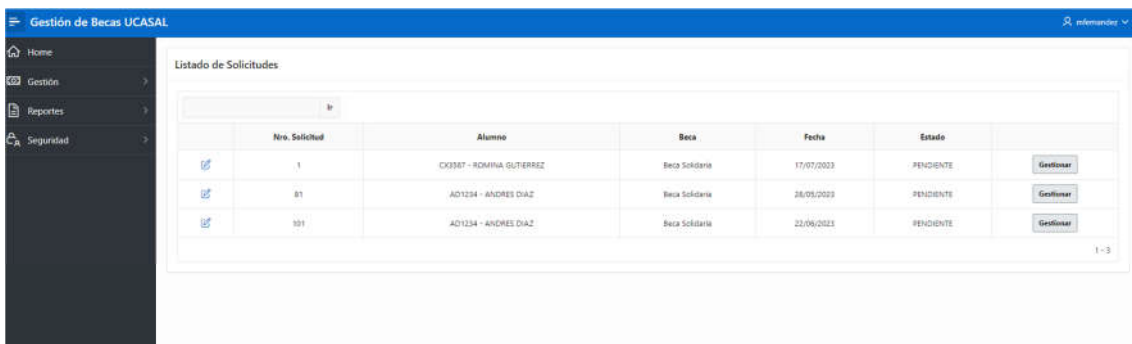


Nombre	Destinatario	Beneficio	Descripción	Prom. Mínimo	Cant. Materias	Activo	Facultad	Carrera	Vigente desde	Vigente hasta
Beca Solidaria	está destinada a estudiantes (ingresantes/ya inscritos) que por motivos socioeconómicos no puedan afrontar el pago parcial o total de la cuota mensual.	desde el 25% al 100%	de la cuota mensual, según la situación socioeconómica del postulante.	8	50	1			01/04/2023	30/04/2023

*Ilustración 18: Pantalla de Gestión de Becas*

➤ Pantalla de Listado de Solicitudes

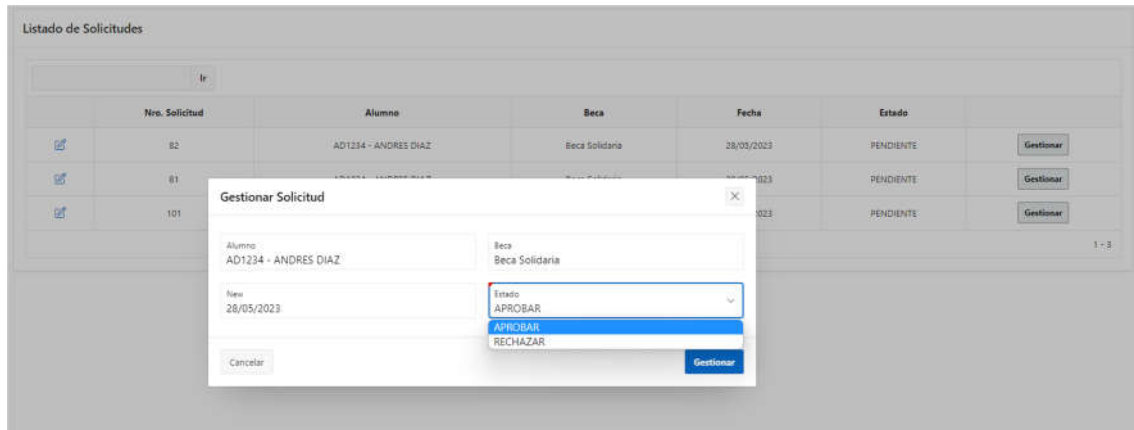
En las siguientes ilustraciones se observa la pantalla de listado de solicitudes en estado pendiente, con la posibilidad de gestionar cada una de ellas y editar los datos de contacto, en caso de ser necesario.



Nro. Solicitud	Alumno	Beca	Fecha	Estado	
1	CK387 - ROMINA GUTIERREZ	Beca Solidaria	17/07/2023	PENDIENTE	Gestionar
81	AD1234 - ANDRES DIAZ	Beca Solidaria	26/05/2023	PENDIENTE	Gestionar
101	AD1234 - ANDRES DIAZ	Beca Solidaria	22/06/2023	PENDIENTE	Gestionar

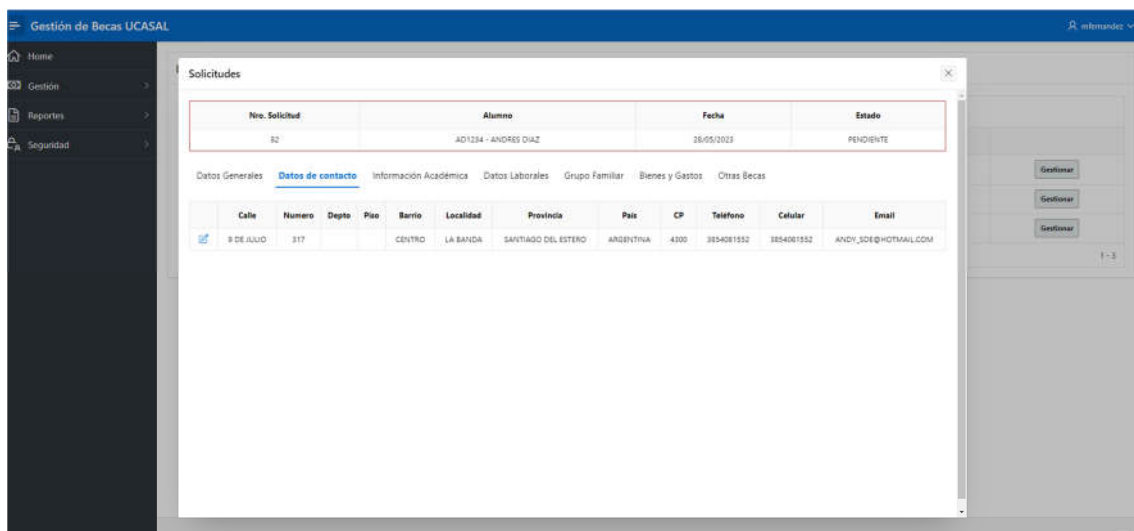
*Ilustración 19: Pantalla de Listado de Solicitudes*

➤ Pantalla de Aprobación/Rechazo de Solicitudes



*Ilustración 20: Pantalla de Aprobación*

➤ **Pantalla de Edición de datos de contacto**



*Ilustración 21: Pantalla de Edición de datos de Contacto*

**Pantallas de Reportaría:**

A continuación, se presentan las pantallas de reportaría, diseñadas específicamente para brindar una visualización clara y concisa. Cada pantalla cuenta con su sección de filtros, lo cual facilita la búsqueda y el análisis de la información deseada.

➤ **Listado de Becas**

UCASAL – Ing. Informática - *Desarrollo de un sistema para gestión de becas, aplicando metodología RAD*

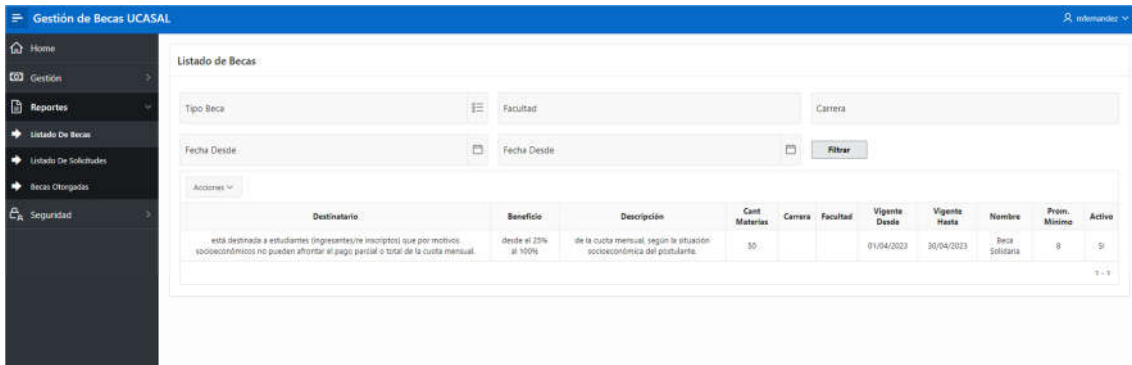


Ilustración 22: Pantalla de listado de becas

➤ Listado de Solicitudes

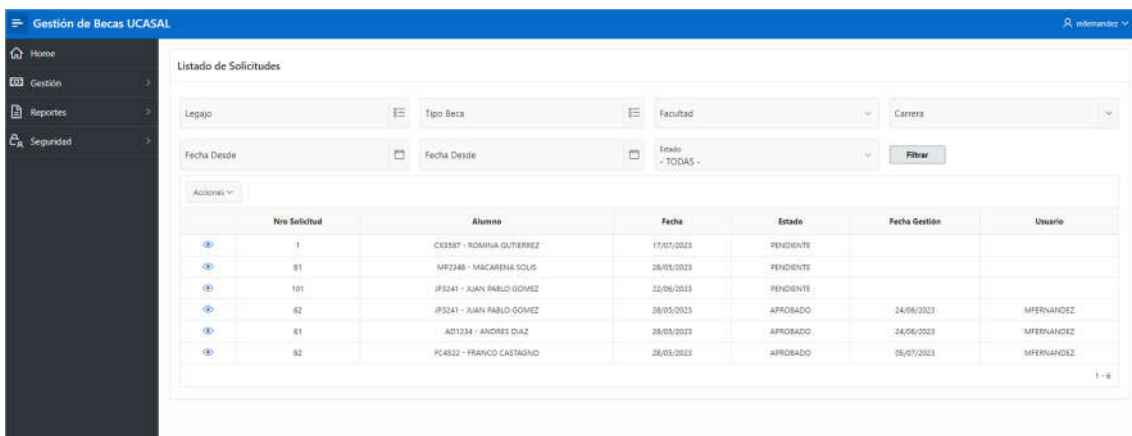


Ilustración 23: Pantalla de Listado de Solicitudes

➤ Listado de becas otorgadas

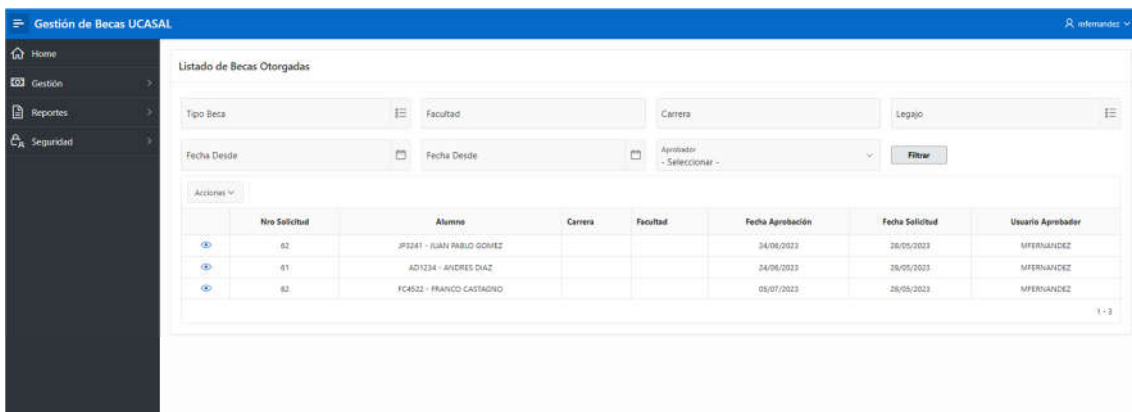


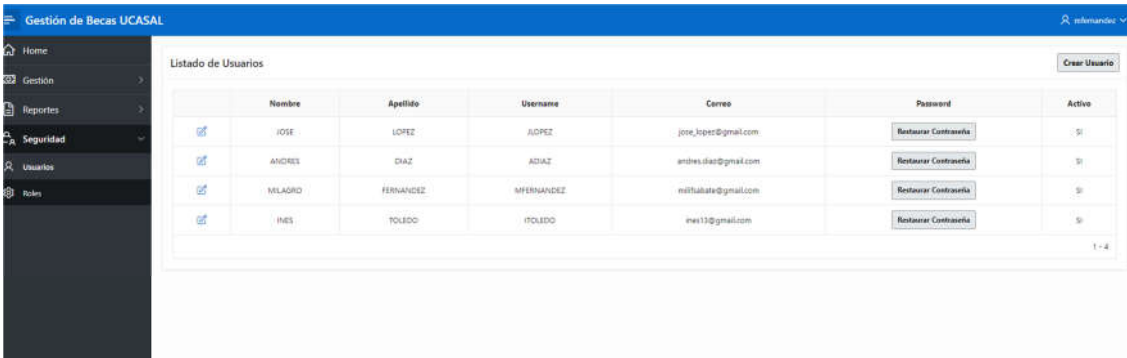
Ilustración 24: Pantalla de Becas Otorgadas

## Pantallas de Seguridad

A continuación, se presentan las pantallas del módulo de seguridad del sistema. En la primera pantalla, los administradores del sistema tienen la posibilidad de agregar nuevos usuarios, además ofrece la posibilidad de restaurar las contraseñas en caso de que los usuarios olviden sus credenciales de acceso.

La segunda pantalla corresponde al módulo de roles. En esta interfaz, los administradores pueden crear nuevos roles y asignar permisos específicos a cada uno de ellos. Esto permite establecer niveles de autorización personalizados para diferentes tipos de usuarios, garantizando que solo tengan acceso a las funcionalidades y datos pertinentes a sus responsabilidades.

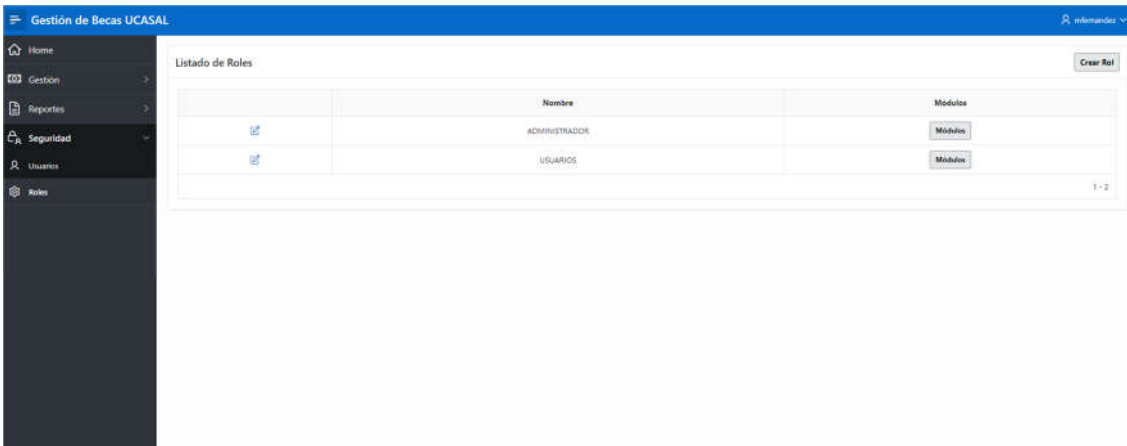
### ➤ Pantalla de Creación/Edición de Usuarios



	Nombre	Apellido	Username	Correo	Password	Activo
	JOSE	LOPEZ	JLOPEZ	jose_lopez@gmail.com	<a href="#">Restaurar Contraseña</a>	SI
	ANDRES	CHAZ	ACHAZ	andres.chaz@gmail.com	<a href="#">Restaurar Contraseña</a>	SI
	MILAGRO	FERNANDEZ	MFFERNANDEZ	milagrote@gmail.com	<a href="#">Restaurar Contraseña</a>	SI
	IMES	TOLEDO	ITOLEDO	imes13@gmail.com	<a href="#">Restaurar Contraseña</a>	SI

*Ilustración 25: Pantalla de Usuarios*

### ➤ Pantalla de Roles



	Nombre	Módulos
	ADMINISTRADOR	<a href="#">Módulos</a>
	USUARIOS	<a href="#">Módulos</a>

*Ilustración 26: Pantalla de Roles*

## Portal del Alumno

Por último, se presentan las pantallas correspondientes al portal del alumno en el sistema web de gestión de becas. En estas pantallas, los alumnos pueden aplicar a las becas, completar las solicitudes y realizar un seguimiento del estado de las mismas en la sección de "Consultar Solicitudes". Aquí podrán visualizar si su solicitud ha sido aprobada o rechazada. Estas funcionalidades proporcionan una experiencia interactiva y transparente para los alumnos, permitiéndoles participar activamente en el proceso de solicitud y tener conocimiento actualizado sobre el estado de su solicitud de beca.

### ➤ Pantalla de Listado de Becas disponibles



*Ilustración 27: Pantalla de Listado de becas disponibles*

### ➤ Pantalla de Solicitudes de Becas



*Ilustración 28: Pantalla de Solicitudes de Becas*

## **8.4 Fase de Implementación**

La fase de Implementación incluye la puesta a prueba y la capacitación del usuario final, la conversión de datos y la implementación del sistema de aplicación.

### **8.4.1 Requisitos mínimos de Hardware para la Implementación**

Estos requisitos son las especificaciones técnicas mínimas necesarias para garantizar el correcto funcionamiento del Sistema de Gestión de Becas. Son necesarios para asegurar que la aplicación pueda funcionar de forma óptima.

#### **Servidor Web**

Memoria RAM: 2 GB

Procesador: 2 GHz.

Disco Duro: 1.5 Gb (Libre)

#### **Servidor Cliente**

Memoria RAM: 512 Mb

Procesador: 1 GHz.

Tarjeta de red

### **8.4.2 Requisitos mínimos de Software para la Implementación**

#### **Servidor Web**

##### **Sistema Operativo**

Windows 7

Arquitectura x86\_64

##### **Sistema de Gestión de Base de Datos**

SQL 19C.

#### **Cliente**

##### **Navegador Web**

Google Chrome, Mozilla Firefox, Microsoft Edge, Zafari.

### **8.4.3 Proceso de Implementación**

Una vez completada la fase de desarrollo, se procede a la implementación del mismo. La primera tarea es la integración con el sistema SAG, que es el sistema principal de la universidad. Para lograr esto, el Departamento de Informática debe

agregar una URL en el sistema SAG que dirija a la página de inicio del Sistema de Gestión de Becas (Portal del Alumno). Junto con esta URL, se debe enviar un parámetro que contenga el número de legajo del alumno logueado en el SAG. El legajo permitirá realizar las consultas a las APIs correspondientes que devolverán la información asociada al alumno.

Una vez que se ha logrado la integración con el sistema SAG, el Departamento de Becas deberá crear los usuarios en el sistema y asignarles roles apropiados. Los roles pueden incluir administrador, operador, entre otros. Los usuarios serán asignados a uno o varios roles según su función dentro del sistema.

Una vez que los usuarios han sido creados y los roles asignados, se procede a la carga de las becas. El Departamento debe ingresar la información en el sistema, incluyendo el nombre de la beca, destinatarios, beneficios de la misma.

Finalmente, se realizan pruebas exhaustivas para garantizar que el sistema esté funcionando correctamente y cumpla con los requisitos establecidos en la fase de análisis. Esto incluye pruebas de calidad y de funcionalidad.

#### **8.4.4 Pruebas de QA**

El proceso de pruebas de QA es una parte esencial del desarrollo del sistema de gestión de becas. Las pruebas se realizan para garantizar que el sistema funcione correctamente y cumpla con los requisitos establecidos en la fase de análisis y diseño. En este apartado, se desarrolla las pruebas de Caja Negra para los Casos de Uso desarrollados en la Fase de Análisis.

#### **Módulo de Administración:**

➤ Registrar Usuario del Sistema

<b>Datos de ingreso</b>	<b>Resultado Obtenido</b>	<b>Estado</b>
Creación de un usuario nuevo y que sus datos sean almacenados correctamente en la base de datos.	Usuario registrado correctamente.	Validado
Creación de usuarios con campos vacíos o con información inválida.	Mensaje de error: "Nombre debe	Validado

	tener algún valor”	
Verificar que no se puedan crear usuarios con un nombre de usuario que ya existe en la base de datos.	Mensaje de error: “Usuario existente”	Validado

➤ Registrar Ingreso al Sistema (Login):

Datos de ingreso	Resultado Obtenido	Estado
Iniciar sesión con sus credenciales correctas	Ingreso al sistema	Validado
Iniciar sesión con sus credenciales correctas	Mensaje de error: “Credenciales de conexión no válidas”	Validado

➤ Registrar Beca:

Datos de ingreso	Resultado Obtenido	Estado
Registrar becas nuevas y que sus datos sean almacenados correctamente en la base de datos.	Beca registrada	Validado
Registrar becas con campos vacíos o con información inválida.	Mensaje de error: “El campo debe tener algún valor”	Validado

➤ Aprobar Solicitud:

Datos de ingreso	Resultado Obtenido	Estado
Aprobar una solicitud de beca correctamente y que se actualice su estado en la base de datos.	Beca aprobada	Validado

➤ Rechazar Solicitud

Datos de ingreso	Resultado Obtenido	Estado
Se pueda rechazar una solicitud de beca correctamente y que se actualice su estado en la base de datos.	Beca rechazada	Validado

➤ Editar datos de contacto del Alumno:

Datos de ingreso	Resultado Obtenido	Estado
Se puedan editar los datos de contacto del alumno y que se actualicen correctamente en la base de datos.	Datos de contacto editados.	Validado
No se puedan editar los datos de contacto del alumno con formato inválido.	Mensaje de error	Validado
No se pueda guardar cambios si no se cargaron los datos de contacto requeridos.	Mensaje de error	Validado

**Portal del Alumno:**

➤ Registrar Solicitud:

Datos de ingreso	Resultado Obtenido	Estado
El alumno pueda registrar una solicitud de beca correctamente y que se almacene en la base de datos.	Beca registrada	Validado
No se puedan registrar solicitudes de beca con campos vacíos o con información inválida.	Mensaje de error	Validado
El alumno no se pueda registrar dos veces en la misma beca.	Mensaje: “Usted ya posee una solicitud activa para esta beca”	Validado

## **12.CONCLUSIÓN**

En conclusión, el Sistema web de Gestión de Becas desarrollado con la metodología RAD y tecnologías Low-Code tiene el potencial de mejorar significativamente los procesos de gestión de becas en la Universidad Católica de Salta.

La automatización de procesos y la utilización de herramientas de análisis permiten una gestión más eficiente y efectiva de las becas, reduciendo el tiempo y los costos involucrados en el proceso de asignación y seguimiento de las mismas.

En particular, la integración del sistema con el sistema SAG permite un acceso rápido y seguro a la información de los estudiantes, lo que facilita la asignación y seguimiento de las becas. Además, la posibilidad de registrar las solicitudes de becas en línea por parte de los estudiantes a través del portal del alumno agiliza y simplifica el proceso, mejorando la experiencia del usuario.

En el transcurso del proyecto, me encontré con algunos desafíos. Uno de ellos fue la implementación de la nueva plataforma de desarrollo de software, que si bien, una de sus ventajas es la corta curva de aprendizaje, requirió destinarle tiempo y adaptación. Además, otro obstáculo importante fue la necesidad de familiarizarme y dominar una nueva metodología de desarrollo, ya que mi experiencia laboral se basó en el uso de metodología Scrum.

En lo personal, fue muy importante para mi formación académica y profesional el haber realizado este proyecto. El seminario me brindó la oportunidad de profundizar una metodología moderna, entrar en contacto con clientes reales y aplicar muchos de los procesos aprendidos a lo largo de mi carrera. Ha sido una experiencia enriquecedora que me ha permitido desarrollar habilidades prácticas y adquirir un mayor entendimiento de los desafíos que implica implementar soluciones tecnológicas en un entorno real. Estoy sumamente agradecida por la oportunidad y aprendizajes obtenidos durante este proyecto.

### 13. BIBLIOGRAFÍA

- Abrego Almazán, D., Sánchez Tovar, Y., & Medina Quintero, J. (2017). Influencia de los sistemas de información en los resultados organizacionales. *Contaduría y administración*, 62(2), 14-20.
- Ander Egg, E. (2010). *Cómo elaborar un proyecto*. Buenos Aires: Gránica.
- Arellano Rodríguez, M. (2010). Sistemas de información: ¿adecuación a los cambios tecnológicos o herramienta de gestión? *Revista de Ciencias Sociales*, 14(3), 31-36.
- Baca Urbina, G. (2013). *Evaluación de proyectos*. Ciudad de México: Mc Graw Hill.
- Canossa Montes de Oca, H. (2022). Gestión de proyectos como estrategia para la evaluación de desempeño del talento humano en las empresas. *Ciencias administrativas*, 9(36), 9-16.
- Casanova, C. (2010). *Factores clave para el éxito o el fracaso en proyectos de implementación de sistemas ERP*. Obtenido de [http://www.pragmaticaconsultores.com/elementos/articulosPDF/Factores\\_clave\\_exito\\_ERP\\_PETROTECNIAABRIL2010.pdf](http://www.pragmaticaconsultores.com/elementos/articulosPDF/Factores_clave_exito_ERP_PETROTECNIAABRIL2010.pdf)
- Castillo, C. (2021). *Desarrollo de aplicaciones web*. Buenos Aires: Alfaomega - Marcombo.
- Cisneros Gutiérrez, A. (2015). *Propuesta de mejora de la calidad del servicio en el proceso de administración de becas y ayudas económicas del Instituto de Fomento al Talento Humano*. Quito: IFTH.
- Gomez Zea, J. M. (2019). *Implementando scrum + rad para la gestión y desarrollo de proyectos de software en equipos de trabajo con personal limitado y eventual*. Obtenido de [https://www.researchgate.net/publication/346060429\\_Implementando\\_scrum\\_rad\\_para\\_la\\_gestion\\_y\\_desarrollo\\_de\\_proyectos\\_de\\_software\\_en Equipos\\_de\\_trabajo\\_con\\_personal\\_limitado\\_y\\_eventual](https://www.researchgate.net/publication/346060429_Implementando_scrum_rad_para_la_gestion_y_desarrollo_de_proyectos_de_software_en Equipos_de_trabajo_con_personal_limitado_y_eventual)
- González Pérez, A. (2014). *Programación de Bases de Datos con C#*. Madrid: RA-MA.
- Kaplan, R., & Norton, D. (1992). *Cuadro de mando integral*. Buenos Aires: Planeta.
- Kendall, K., & Kendall, J. (2011). *Análisis y diseño de sistemas, Octava Edición*. México: Pearson Educación.
- Kosciuk, N. (2015). *Sistemas de información gerencial*. Buenos Aires: AlfaEpsilon.
- Laudon, K., & Laudon, J. (2012). *Sistemas de información gerencial*. México: Pearson.

- Macías Rodríguez, A., Molina Chalacán, L., & Jalón Arias, E. (2021). *Aplicativo web para la búsqueda y ubicación de libros en bibliotecas de Instituciones Escolares*. Caracas: FUNDACIÓN KOINONIA.
- Martin, J. (1991). *Metodología RAD*. New Jersey: UP.
- Mena, S. (2015). *RAD*. Querétaro: Universidad de Querétaro.
- Mesa Casado, M. (2022). *Desarrollo de aplicaciones con Java*. Buenos Aires: IC Editorial.
- Molina Granja, F., & Bonilla Cadena, J. (Diciembre de 2022). *Desarrollo de una plataforma web para recorridos virtuales 360° mediante la metodología RAD. Caso Grupo Inmobiliario Horizonte de la ciudad de Riobamba*. Obtenido de <http://dspace.unach.edu.ec/handle/51000/10023>
- Piattini, M., García Rubio, F., García Rodríguez de Guzmán, I., & Pino, F. (2018). *Calidad de sistemas de información*. Buenos Aires: Alfaomega.
- Pino, F., García, F., Ruiz, F., & Piattini, M. (2005). *Adaptación de las normas ISO/IEC 12207: 2002 e ISO/IEC 15504: 2003 para la evaluación de la madurez de procesos software en países en desarrollo*. Granada: Actas de las X Jornadas de Ingeniería del Software y Bases de Datos.
- Pressman, R. (2010). *Ingeniería del Software*. México: Mc Graw Hill.
- Reingart, M. (2011). *Plataforma de Desarrollo Rápido de Aplicaciones bajo el Proceso de Software Personal: en búsqueda de agilidad, solidez y disciplina para la Ingeniería de Software*. Morón: Facultad de Informática, Ciencias de la Comunicación y Técnicas Especiales.
- Ruíz Domínguez, R., & Becerra Lois, F. (2015). Una propuesta para la evaluación integral de los proyectos de desarrollo local. El caso de estudio TROPISUR. *Economía y Desarrollo*, 154(1), 25-40.
- Sapag Chaín, N., Sapag Chaín, R., & Sapag Puelma, J. (2014). *Preparación y evaluación de proyectos*. México: Mc Graw Hill.
- Serna, E., Martínez, R., & Tamayo, P. (2021). Una revisión a la realidad de la automatización de las pruebas del software. *Computación y Sistemas*, 23(1), 9-22.
- Solano Canto, D. (2021). *Aplicación de la metodología Bpm: Rad para la mejora de la calidad en la atención del proceso gestión de biblioteca de la Facultad de Ingeniería de Sistemas*. Huancayo: Universidad Nacional del Centro del Perú.

UCASAL – Ing. Informática - *Desarrollo de un sistema para gestión de becas, aplicando metodología RAD*

Suárez Concepción, F., Piñero Aguilar, R., Prieto Moreno, A., Alfonso Cordoví, A., Carbó Castro, J., & Llanes Santiago, O. (2022). Metodología para la automatización de procesos tecnológicos en la industria farmacéutica cubana. *Ingeniería Industrial*, 30(2), 59-68.

Sutherland, J. (2021). *Scrum: El Revolucionario Método Para Trabajar el Doble en la Mitad de Tiempo*. México: Booket.

Tellez Castro, F. (Junio de 2019). *SISTEMA DE INFORMACIÓN ADMINISTRATIVO PARA UNA MICROEMPRESA COMERCIAL EN VALLE DE CHALCO SOLIDARIDAD*. Obtenido de <http://ri.uaemex.mx/bitstream/handle/20.500.11799/105635/Fatima.pdf?sequence=1&isAllowed=y>

Terrones Melgarejo, C. (2021). *Análisis del sistema de gestión de la calidad basado en procesos de la norma ISO 9001 en un programa de becas*. Lima: PUCP.

## **ANEXOS**

### **Anexo I: Entrevista al personal del Departamento de Becas**

#### **¿Qué datos se le solicitan al alumno al momento de la solicitud? Listarlos**

- La solicitud de becas se realiza de manera virtual a través del S.A.G.

#### **¿Cómo y quienes definen a qué candidato se le otorgan las becas?**

- Para definir a los beneficiarios de las becas de la Universidad Católica de Salta, se reúne un Comité de Becas el cual revisa que se cumplan los requerimientos necesarios y el Reglamento de Becas según el tipo de becas.

#### **¿Quiénes manejan o administran el sistema?**

- Para administrar el sistema de Becas, la universidad dispone de un departamento de Dirección de Becas.

#### **¿Tienen accesos a diferentes pantallas dependiendo del tipo de usuario? (Roles, ej.: administrador, operario, etc.)**

- No hay roles definidos para utilizar el sistema, todo el equipo de la Dirección de Becas tiene acceso a todas las pantallas y funcionalidades del sistema. Quizás sería interesante disponer de diferentes permisos para asignarles a los diferentes usuarios.

#### **¿Quién carga los tipos de Becas en el sistema? (Ej.: equipo informático, departamento de Becas, etc.)**

- El departamento de Dirección de Becas no configura las becas disponibles, ya están precargadas en el sistema (lo hace el área de Desarrollo de Software). Suele ocurrir que en determinados casos necesitamos modificar algunos parámetros de esas becas. Esto es algo tedioso ya que debemos coordinar con el equipo de desarrollo.

#### **¿El sistema emite reportes o datos estadísticos? Si la respuesta es "Si", ¿Cuáles?**

- No emite ningún tipo de reportes, ni tampoco estadísticas.

**¿Están conformes con el proceso de carga/solicitud/gestión/aprobación de becas?**

**¿Por qué?**

- Estoy conforme con la posibilidad de que el sistema se encuentre de manera online, ya que es más accesible para los estudiantes. Sin embargo, se podría disponer de alguna funcionalidad más, como, por ejemplo, reportes.

**¿Qué crees que podría ayudarte a hacer el trabajo más rápido o eficiente?**

- Contar con más recursos humanos en el momento del análisis.

## Anexo II: Solicitud de Beca en el SAG de la UCASAL

El SAG le permite descargar el formulario Solicitud de Beca, para presentar en la Dirección de Becas, Unidad de Gestión o Delegación.

1. Ingrese a la página web de la Universidad Católica de Salta <http://www.ucasal.edu.ar/>
2. Seleccione sobre el margen derecho la pestaña **Sistema de Autogestión (SAG)**.
3. En el cuadro **Alumno registrados** ingrese sus datos Usuario y Contraseña. Luego, haga clic en el botón **Iniciar Sesión**.



Alumnos registrados

\* Usuario

[¿Olvidó su cuenta de usuario?](#)

\* Contraseña

[¿Olvidó su contraseña?](#)

**Iniciar sesión**

4. A continuación elija la **Carrera** en la que desea solicitar Beca



Gestiones Académicas

Mis Carreras / Cursos

**INSCRIPCIÓN**

Notificaciones Generales

Bienvenidos a la nueva versión del Sistema de Autogestión de Alumnos - SAG

Aranceles y Créditos por Examen

Acreditación de Pagos

Seleccione la Carrera / Curso en la que desea realizar sus transacciones

Al elegir la carrera o curso accederá a una lista de transacciones disponibles asociadas a su elección. La disponibilidad de transacciones está sujeta al Calendario Académico.

Carrera	Lugar	Modalidad	Facultad	Tipo de Estudio	Plan	Estado
<a href="#">ABOGACÍA</a>	SALTA - DISTANCIA	CAMPUS VIRTU...	Ciencias Jurídicas	Grado	8707	Normal

Carreras / Cursos pendientes de acreditación de pago de matrícula / arancel

**A tener en cuenta:**

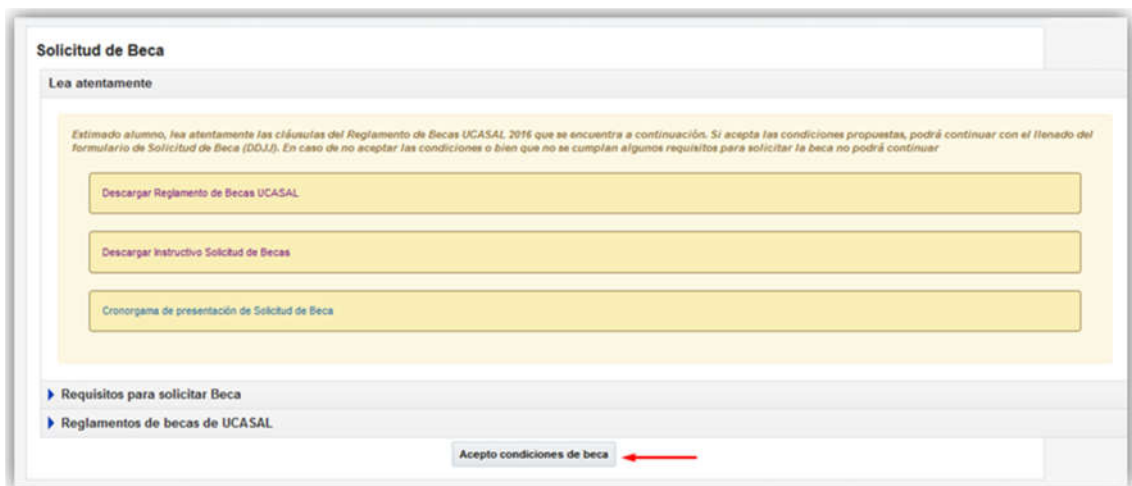
La carrera o curso que visualiza a continuación se debe a que se encuentra pendiente de acreditación el pago de la matrícula/arancel dependiendo si es una carrera o un curso. Una vez acreditado el pago, podrá acceder a todas las opciones de la carrera o curso. Si cuando inició el proceso de inscripción no completó sus datos personales, deberá realizarlo desde la transacción Completar solicitud de ingreso del menú Gestiones Académicas.

Carrera	Lugar	Modalidad	Facultad	Tipo Estudio	Fecha Inscripción
No data to display.					

5. Luego del menú **Gestiones Administrativas** seleccione la opción **Solicitud de Beca**, en el panel izquierdo de la pantalla.



6. Posteriormente, descargue y lea atentamente tanto el Reglamento como el Instructivo y Cronograma que se presentan como hipervínculos en la sección amarilla. Seguidamente, haga clic en el botón **Acepto condiciones de Beca** para poder ingresar y completar el formulario de solicitud de Beca.



7. Las siguientes pantallas las deberá completar de acuerdo a lo expresado en el **Instructivo** disponible para su descarga en el paso anterior, seguidamente aparecerán las siguientes pantallas para completar sus datos correspondientes.

Tenga presente que los campos marcados con un (\*) son obligatorios y que para navegar entre pantallas deberá utilizar los botones **Anterior** y **Siguiente** de cada pantalla; puede salir de la solicitud haciendo clic en el botón **Cancelar** y también puede completar cada pantalla e ir grabando parcialmente estos datos para luego continuar con los demás datos de las siguientes pantallas.

**Datos laborales del alumno** Anterior Siguiente Cancelar Grabar parcialmente

Tipo

\* Rubro SE

Empresa

Cuit

Calle

Número País Departamento

Teléfono

Barrio

Código Postal

\* País SE

\* Provincia SE

\* Localidad SE

Antigüedad

Ingreso

**Domicilio Actual del Alumno** Anterior Siguiente Cancelar Grabar parcialmente

Domicilio de residencia del alumno mientras cursa sus estudios.

\* Calle Lote

\* N°

\* País SE

\* Departamento

\* Categoría Ocupacional

\* Teléfono

Celular

Barrio

Código Postal

\* País ARGENTINA

\* Provincia SAULT

\* Localidad SAULT CAPITAL

\* E-mail

Facebook

Twitter

\* Medio preferido para el contacto

**Otras becas** Anterior Siguiente Cancelar Grabar parcialmente

Información de otras becas o beneficios que ha recibido.  
Especifique si ya ha recibido becas de la UCASAL anteriormente.  
Especifique si está solicitando beca también a otra empresa u organismo para el "Año en curso". En este último caso, en el cuadro Comentarios coloque la leyenda "En trámite".

Agregar beca

Otorgante	Año de la Beca	Monto	Porcentaje	Comentarios
No se ha encontrado ningún dato				

**Inmuebles** Anterior Siguiente Cancelar Grabar parcialmente

Inmueble del grupo familiar incluyendo vivienda en la que el alumno resida mientras cursa sus estudios

*Importante: En todos los casos que corresponda, deberá acreditar constancia de pagos en conceptos de hipotecas, alquiler, gastos compartidos, etc.*

- Si posee más de una vivienda deberá presentar constancia de inmuebles (N° de Catastro / Cedula Parcelaria).
- Si existe quiebra o embargo que afecte la vivienda, deberá presentar documentación que lo acredite

Agregar

Catastro	Dirección	Tenencia	Ingreso \$ x Alquiler	Egreso \$ x Alquiler	Impuestos	Servicios	Hipotecas	Valor fiscal	Habita actualmente
No data to display.									

**Automóviles** Anterior Siguiente Cancelar Grabar parcialmente

Agregar

Patente	Marca	Modelo	Año	Uso	Ingreso Percibido
No data to display.					

### Gastos

¿Convive con su grupo familiar? Si

#### Gastos del Grupo Familiar

Incluyendo los gastos del alumno, si el alumno convive con la familia mientras cursa sus estudios o si se mantiene solo con sus ingresos.

\* Alimentación \$

\* Servicios \$

\* Educación \$

\* Salud \$

\* Transporte \$

\* Otros gastos \$

Total gastos \$

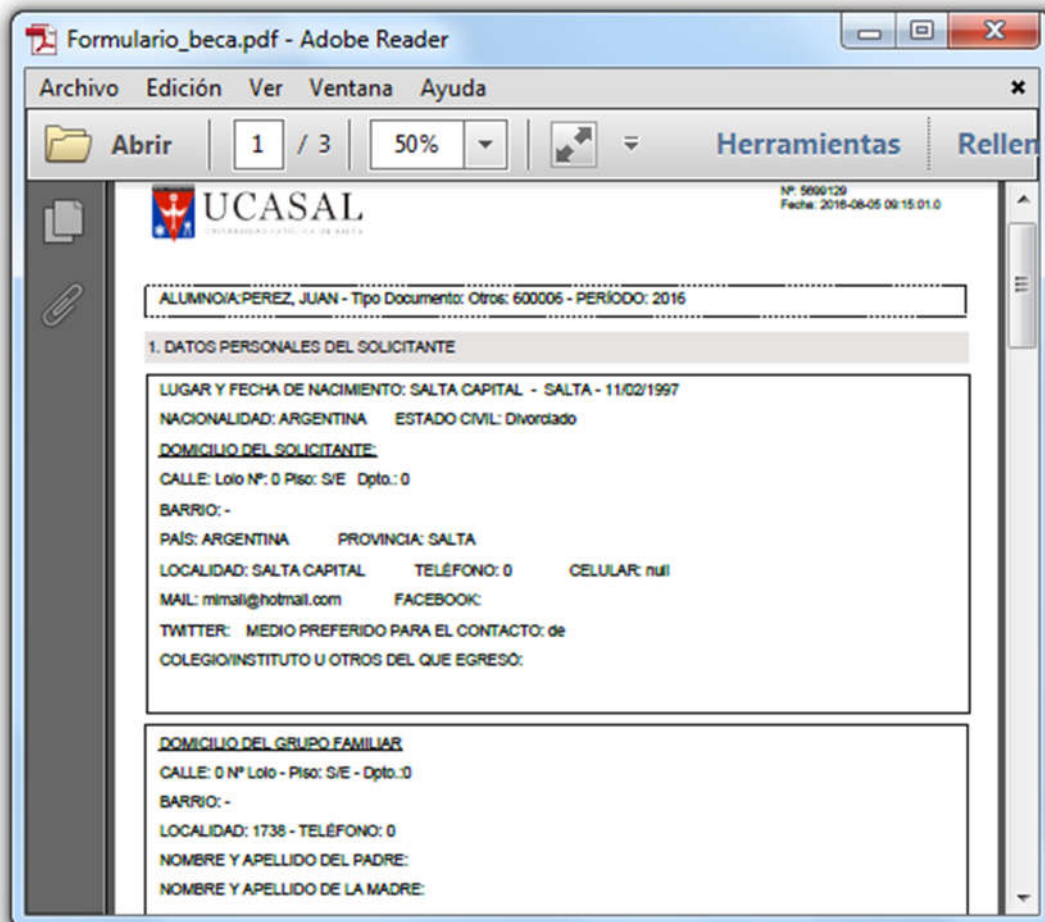
### Conformación del grupo familiar

Declara las personas que conforman su grupo familiar y las personas con las que comparte residencia (por ejemplo: declarar compañero con los que comparte alquiler de una residencia).

Apellido y Nombre	Vínculo	Grupo	
No data to display.			

### Paso final

Paso Final: Grabar e imprimir formulario de Solicitud de Beca, Usted completó correctamente su Solicitud de Beca.



### **Anexo III: Instructivo para completar formulario de Becas**

Estimado/a Alumno/a:

Antes de comenzar a llenar el formulario, por favor lee atentamente este instructivo. El formulario debe presentarse en la Dirección de Becas, por duplicado, acompañado de toda la documentación que avale los datos que se solicitan en la DDJJ (Declaración Jurada), firmado por vos y por el jefe/a del grupo familiar.

DEBERÁS TENER PRESENTE QUE:

- El formulario DEBE ser llenado conjuntamente con el jefe del grupo familiar, quien también se hará responsable de lo que se declare,
- EL formulario será recibido por la Dirección de Becas solo si está completo en cuanto a los datos solicitados y la documentación respaldatoria; no se aceptarán solicitudes incompletas,
- Si se comprobara que se ha faltado a la verdad en los antecedentes proporcionados en la DDJJ para acreditar la situación socioeconómica, se perderá el derecho a obtener cualquier beneficio otorgado por la Universidad, y se aplicará el art. 131 del Reglamento de Alumnos publicado en el Manual del Alumno,
- Para el caso de alumno ingresante o con pase de una Institución educativa, al momento de recibirla Solicitud se fijará la fecha y hora de una entrevista personal, entrevista que tiene carácter de obligatoria,
- Para cualquier gestión posterior debes conservar tu Formulario de Solicitud de Becas (DDJJ)

MOMENTO 1. COMPLETAR FORMULARIO DE SOLICITUD DE BECA (DDJJ)

I. Lee con atención los requisitos que se despliegan en la página; si cumples con esos requisitos continua con el siguiente paso,

II. La página te mostrará el Reglamento de Becas de la Universidad Católica de Salta; por favor léelo atentamente, y si estás de acuerdo en todo con él, clickea en **“Acepto condiciones de Beca”**.

III. **PASO 1: Tus datos laborales.** El cuadro te da la posibilidad de elegir distintas opciones; si no trabajas elegí esta opción y pasarás al cuadro siguiente; si trabajas elegí las opciones que se te presentan y completa los datos solicitados.

**Domicilio Actual del Alumno**, en él deberás cambiar los datos que no estén correctos y agregar aquellos que no hayas completado. En este cuadro te da la opción de agregar tu face y twitter y además, elegir la red social que preferís se te contacte desde la Dirección de Becas para invitaciones, entrevistas, noticias, etc.

**Datos relativos a becas.** En caso de haber recibido beca deberás consignar los datos que se soliciten: Becas de la UCASAL recibidas de años anteriores y si has solicitado beca para el año en curso a otra/s Institución/es, deberás agregar los datos en el mismo cuadro como NUEVA BECA. En el caso de otras Instituciones, al momento de conocer el otorgamiento o no, deberás informarlo a la Dirección de Becas con una nota que presentarás en la oficina de Becas, caso contrario perderás el beneficio de acuerdo a Reglamento. **IMPORTANTE**, es necesario que llenes la tabla con todas las becas UCASAL que tuviste Y CON TODAS LAS Becas que solicitaste, para el año en curso.

IV. **PASO 2: Gastos del Grupo Familiar, Gastos del Alumno Inmuebles y Automóviles.** En este paso y para que la información sea completa y real, deberás hacerlo acompañado de padres o tutores. Es necesario que se coloquen todos los gastos del grupo familiar. En el cuadro Informe Económico del Alumno, solo lo completarás si tu grupo familiar reside en un lugar diferente al de la universidad y tenés que desplazarte para estudiar, o si te mantienes solo con tus ingresos.

**En Inmueble y automóviles** deberás entrar a EDITAR y allí colocar TODOS los datos que se solicitan. En inmueble podrás editarlos datos de inmueble/s propio/s o el inmueble que alquila tu grupo familiar o vos. En la opción TENENCIA se despliegan

distintos conceptos según la condición del inmueble en donde vivís vos y/o tu grupo familiar.

**V. PASO 3: Conformación del Grupo Familiar:** en este paso tenés que tener en cuenta que se consideran como integrantes del **grupo familiar**, a saber: tus parientes por consanguinidad o adopción, siempre que estas personas residan en una misma casa y compartan ingresos y gastos. Se considerará **grupo familiar y de convivencia** a los parientes y todas aquellas personas que vivan en la misma casa. **Tus padres** se considerarán siempre integrantes del grupo familiar, aun cuando no residan en la misma casa, si comparten ingresos y gastos con vos. También se considerará integrante del grupo familiar la persona que aporte económicamente para tu manutención, aunque no sea pariente.

Deberás, entonces, agregar cada integrante del grupo familiar y/o de convivencia sin olvidar llenar todos los datos que allí se indiquen. No te olvides al llegar al final de controlar que todos los datos que pusiste son correctos y posteriormente clickea GRABAR, de lo contrario se perderán todos los datos cargados; completarás esta planilla tantas veces como cantidad de integrantes tenga el grupo familiar y /o de convivencia.

**VI. PASO 4: Grabar e Imprimir el Formulario de Solicitud de Beca.**

Una vez que hayas revisado el formulario completo y esté correcto, podrás grabarlo e imprimirlo. Recordá que debés imprimir dos copias.

**VII. PASO 5: Adjuntar toda la documentación al Formulario de Solicitud.**

Una vez impresa la DDJJ, firmada por vos y el Jefe/a del grupo familiar, deberás adjuntar toda la documentación que se enumera en el artículo n° 8 del Reglamento de Becas 2015 y toda aquella que acredite los datos que consignaste en el formulario.

MOMENTO 2. PRESENTACIÓN DE LA SOLICITUD DE BECAS

**VIII. PASO 6:** Una vez que tengas la solicitud completa, con toda la documentación (paso 5), tendrás que llevarla a la Dirección de Becas, cuyas oficinas están ubicadas en el Campus Universitario (Campo Castañares - Salta) o bien a tu Unidad de Gestión o Delegación

**RECORDA QUE SE RECIBIRÁN LAS SOLICITUDES COMPLETAS DESDE EL 1º DE DICIEMBRE AL 26 DE DICIEMBRE PARA TODAS LAS CARRERAS; Y A PARTIR DE 26 DE ENERO, CADA SEMANA SE RECIBIRÁN LAS SOLICITUDES POR FACULTAD DE ACUERDO AL CRONOGRAMA VIGENTE.**

**MUCHAS GRACIAS  
DIRECCION DE BECAS**