



Universidad Simón Bolívar  
Departamento de Computación y Tecnología de la Información  
CI3715 Ingeniería de Software I  
Septiembre - Diciembre 2018

## **Sprint I**

### **Integrantes:**

Constanza Abarca 13-10000  
Rosana García 11-10365  
Moisés Gonzalez 11-10406  
Pedro Maldonado 13-10790  
Denylson Romero 13-11270  
Fabio Suárez 12-10578

Noviembre, 2018

## Índice

<b>Introducción .....</b>	<b>3</b>
<b>Desarrollo .....</b>	<b>4</b>
Historias de Usuario.....	4
Documentación del Software .....	8
División de Actividades .....	14
Experiencia de Trabajo en Equipo .....	15
Experiencia de Trabajo con TDD .....	16
Manejo Repositorio GitHub .....	16
<b>Conclusiones.....</b>	<b>18</b>

## **Introducción**

El presente informe tiene como objetivo la presentación de los detalles de la implementación del avance especificado en el Sprint 1 siguiendo la metodología de trabajo SCRUM. El producto a entregar al final de los Sprints será la base de un software de inscripción en línea para los estudiantes de postgrado de la Universidad Simón Bolívar. Actualmente, estos miembros de la comunidad universitaria no cuentan con un sistema automatizado como lo es el sistema de inscripción de pregrado.

El equipo Delta Developers tiene como asignación para ser entregada en el Sprint 1, el diseño e implementación de la base de datos de las asignaturas pertenecientes a los programas de postgrado de la USB, cumpliendo con los requerimientos especificados en la historia de usuario número 2 del Product Backlog y haciendo uso del framework django para el desarrollo del software.

El informe contiene el análisis de las historias de usuario a implementar, los detalles relevantes del código, la experiencia del trabajo en equipo, así como las tareas asignadas, los encargados de llevarlas a cabo y el número de horas invertidas. Finalmente el trabajo cierra con la presentación de conclusiones donde se incluye los objetivos alcanzados y las recomendaciones para el Product Owner y el proyecto en general.

## Desarrollo

### Historias de Usuario

Las historias de usuario son herramientas de trabajo pertenecientes a la metodología SCRUM que permiten reconocer y analizar los requerimientos del Product Owner. Una épica es una historia de usuario grande descompuesta en varias historias de menor tamaño para ser trabajadas de forma ágil acorde a los lineamientos de SCRUM. La épica a desarrollar en el Sprint I es la siguiente:

*2. Como Jefe de Departamento, puedo gestionar el catálogo de asignaturas de postgrado de mi departamento para ofrecer un servicio de mejor calidad.*

La palabra clave de esta historia de usuario es “gestionar”, a través de este verbo es posible inferir que el usuario Jefe de Departamento puede realizar distintas acciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre la base de datos de asignaturas diseñada. De esta épica se desprenden cinco (5) historias de usuario, las cuales serán listadas a continuación. Cada historia contará con su respectivo análisis de dominio y detalles sobre el trabajo realizado para llevarla a cabo.

*2.1 Como Jefe de Departamento, puedo AGREGAR Asignaturas al catálogo de asignaturas de posgrado de mi departamento para mantenerlo actualizado.*

Para añadir asignaturas al catálogo, es necesario definir los atributos que conformarán la entidad Asignatura de la base de datos a implementar, a la que se le realizará la funcionalidad de agregar. El dominio de datos perteneciente a esta historia está comprendido por:

- Código: Campo de tipo String que debe cumplir con una formato requerido. Se utilizará la expresión regular `^[A-Z]{2}-[0-9]{4}$` para asegura que el código de la asignatura incluya el código del departamento al que pertenece y 4 dígitos que identifica a la materia dentro del departamento. Por lo tanto el tamaño del String debe ser exactamente 7 caracteres, incluyendo el carácter “-“. Este campo

identifica de forma única a la asignatura del resto, por lo tanto se utilizará como clave primaria dentro de la base de datos.

- Nombre: Campo de tipo String de caracteres alfabéticos para identificar la materia de forma verbosa.
- Unidades de crédito: Dato de tipo entero que representa la cantidad de créditos académicos de la asignatura. Debe ser mayor estricto que 0.
- Horas teoría: Número de horas de clase teórica que forman parte del plan de estudio de la asignatura. Se representa con un entero que debe ser mayor o igual a 0.
- Horas práctica: Número de horas de clases prácticas que forman parte del plan de estudio de la asignatura. Se representa con un entero que debe ser mayor o igual a 0.
- Horas laboratorio: Número de horas de laboratorio que forman parte del plan de estudio de la asignatura. Se representa con un entero que debe ser mayor o igual a 0.
- Requisitos: Asignatura(s) que se requieren haber aprobado para poder inscribir la materia que se está registrando en el catálogo. Se modela como un campo ManyToMany para representar la relación de dependencia entre asignaturas.
- Departamento: Clave foránea de la entidad Asignatura. Relaciona la asignatura con su respectivo departamento. Un departamento es una entidad de la base de datos que cuenta con un código de tipo String conformado por exactamente dos caracteres alfabéticos que lo identifican dentro del sistema administrativo de la Universidad (clave primaria) y un nombre, un campo de tipo String, que identifica al departamento de forma verbosa.

Para completar la historia de usuario se plantearon las siguientes tareas:

- Diseño e implementación de formulario de registro de asignaturas en el catálogo con sus respectivas validaciones.

- Diseño de vista donde se encontrará el catálogo con la lista de asignaturas de un departamento con sus respectivos datos.

Las verificaciones a tomar en cuenta en el formulario incluye que se ingresen las asignaturas correspondientes al departamento, es decir, que los dos primeros caracteres del código de la asignatura coincidan con el código del departamento y que si el código de la asignatura ya existe en la base de datos debe mostrarse un mensaje de error.

*2.2 Como Jefe de Departamento, puedo BUSCAR Asignaturas en el catálogo de asignaturas posgrado en mi departamento para consultarlas.*

La funcionalidad en el software a implementar de búsqueda de asignaturas recibirá el código de la asignatura o el nombre de asignatura como entrada. El dominio que maneja esta historia es el campo de búsqueda para filtrar la base de datos. Este campo puede ser uno de los siguientes, ambos ya definidos para la historia 2.1:

- Código
- Nombre

Las tareas relacionadas con buscar asignaturas son:

- Queries sobre la base de datos para búsqueda según nombre o código.
- Mostrar los resultados en la vista correspondiente.

Se deben manejar las excepciones en que el código o nombre ingresado por el usuario sean inválidos, es decir que no coincidan con ninguna de las asignaturas que se encuentran en la base de datos. En caso de que ocurra este evento, se debe mostrar en pantalla el mensaje de error.

*2.3 Como Jefe de Departamento, puedo MODIFICAR Asignaturas en el catálogo de asignaturas de postgrado de mi departamento para mantenerlo actualizado.*

El dominio de datos de esta historia de usuario es el mismo que el de la historia 2.1 igual que las verificaciones a realizar sobre los datos.

Las tareas correspondientes a esta historia de usuario son:

- Implementar formulario de modificación de asignatura.
- Incluir opción (botón) de editar en las entradas del catálogo de asignaturas.

*2.4 Como Jefe de Departamento, puedo ELIMINAR Asignaturas del catálogo de asignaturas de postgrado de mi departamento para mantenerlo actualizado.*

La opción de eliminar estará presente en un botón junto con el de editar en todas las entradas del catálogo mostradas en la respectiva vista del software. El dominio de esta historia está definido como la entidad Asignatura y la entidad Departamento que cuentan con todos los campos detallados en la historia de usuario 2.1.

Las tareas asociadas a esta historia de usuario son:

- Implementar una alerta que salga al tratar de eliminar una materia.
- Eliminar la traza de la asignatura de la base de datos.

*2.5 Como Jefe de Departamento, puedo LISTAR (ascendente, descendente) por orden alfabético o por código las asignaturas de posgrado de mi departamento.*

El dominio asociado a esta historia es el de las entidades Asignatura y Departamento que contienen todos los datos explicados en la historia 2.1. Además incluye el tipo de listado a realizar, el cual podrá tomar los valores: ascendente, descendente, orden alfabético, orden por código de asignaturas. Dependiendo del tipo de listado realizado se mostrará en la vista el resultado deseado. La tarea relacionada será la de la implementación de las funciones que se encargarán de devolver las entradas de la tabla de forma ordenada según el criterio indicado por el usuario.

## Documentación del Software

Una vez analizadas las historias de usuario se procede a detallar la documentación del software, en primer lugar se presenta el modelo Entidad Relación de la base de datos de las asignaturas de postgrado que fue diseñado en base a los requerimientos del cliente indicados en el Product Backlog. De color azul se encuentra la entidad correspondiente a la épica presentada anteriormente y sus respectivos atributos que conforman el catálogo de asignaturas a implementar por el equipo desarrollador.

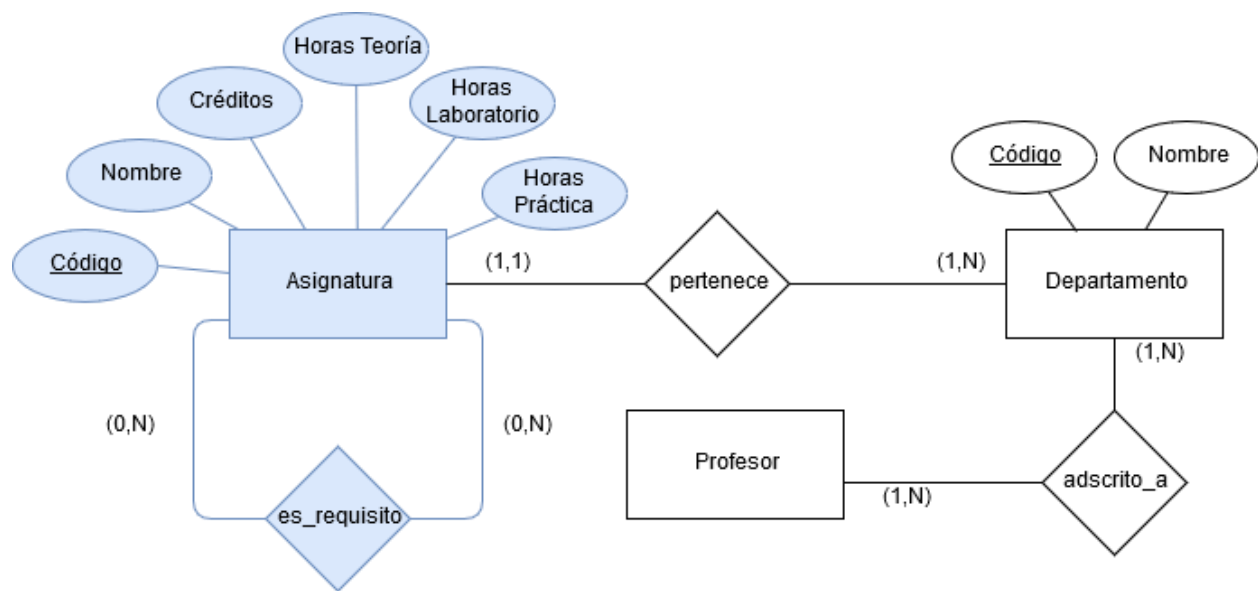


Figura 1. Modelo ER

Una vez diseñado el modelo de la base de datos se inició la implementación en django. Al trabajar con este framework se utilizó el Modelo Vista Template (MTV) el cual facilitó la modularidad del software a desarrollar. Una visión general de este esquema de trabajo incluye el archivo models.py donde se encuentra el modelo de la Figura 1 implementado. En el directorio templates/Asignaturas se encuentran los templates que se utilizaron para el front-end del software, en el archivo views.py se encuentran los controladores del sistema que renderizan los templates con los URL de la aplicación y realizan las operaciones con las entradas ingresadas por el usuario en la interfaz del software.



A continuación, se presentarán las interfaces desarrolladas y se explicará su correspondencia con las historias de usuario analizadas. En primer lugar, se presenta la vista de inicio de sesión y registro, donde el jefe y demás miembros del departamento puede ingresar al sistema.



## Sistema de Oferta de Asginaturas - Postgrado

Inicio de Sesión
Registrarse

Nombre de usuario:

Contraseña:

Entrar

Figura 2. Inicio de Sesión

## Sistema de Oferta de Asginaturas - Postgrado

Inicio de Sesión
Registrarse

Nombre de usuario: 
Requerido. 150 caracteres como máximo. Únicamente letras, dígitos y @/./+/\_

Departamento:

Contraseña:

- Su contraseña no puede asemejarse tanto a su otra información personal.
- Su contraseña debe contener al menos 8 caracteres.
- Su contraseña no puede ser una clave utilizada comunmente.
- Su contraseña no puede ser completamente numérica.

Contraseña (confirmación): 
Para verificar, introduzca la misma contraseña anterior.

Registrar

Figura 3. Registro

Ahora se presenta el formulario realizado para el registro de asignaturas en la base de datos, dando respuesta a la historia de usuario 2.1.

[Registrar asignatura](#) [Ver asignaturas](#) [Salir](#)

### Registro de Asignatura

---

Código:

Nombre:

Unidades de Crédito:

Horas de Teoría:

Horas de Práctica:

Horas de Laboratorio:

Requisitos:

Departamento:

Figura 2. Formulario de Registro de Asignaturas

El funcionamiento del formulario es correcto pues los datos ingresados pueden ser visualizados en el catálogo de asignaturas presentado en la Figura 3. Además esta visualización de asignaturas representa las historias de usuario 2.2, 2.4 y 2.5, pues es posible modificar, eliminar y ordenar entradas de la tabla.









### Sistema de Oferta de Asignaturas - Postgrado

[Registrar asignatura](#) [Ver asignaturas](#) [Salir](#)

Ver  asignaturas por página
Buscar:

Código	Nombre	Créditos	Horas Teoría	Horas Práctica	Horas Laboratorio	Requisitos	Departamento		
CI-6432	Inteligencia Artificial	4	6	0	0		CI	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
CI-7431	Bases de Datos Lógicos	4	4	0	0		CI	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Mostrando 1 de 1

Figura 3. Catálogo de Asignaturas

Adicionalmente, al dar click en el botón de modificar, aparece otro formulario para editar los campos de la respectiva asignatura como se muestra en la Figura 4. Con esto se cumple con el requerimiento de la historia de usuario 2.3.

[Registrar asignatura](#) [Ver asignaturas](#) [Salir](#)

### Modificar Asignatura

Código:

Nombre:

Unidades de Crédito:

Horas de Teoría:

Horas de Práctica:

Horas de Laboratorio:

Requisitos:

Departamento:

[Modificar](#)

Figura 4. Modificar Asignatura

Si se desea eliminar un elemento del catálogo, se selecciona el botón “Eliminar” y aparece un mensaje de alerta tal como se muestra en la Figura 5. Si se da click en “Eliminar” nuevamente, la entrada del catálogo será eliminada. Esta operación cumple con la historia de usuario 2.4.

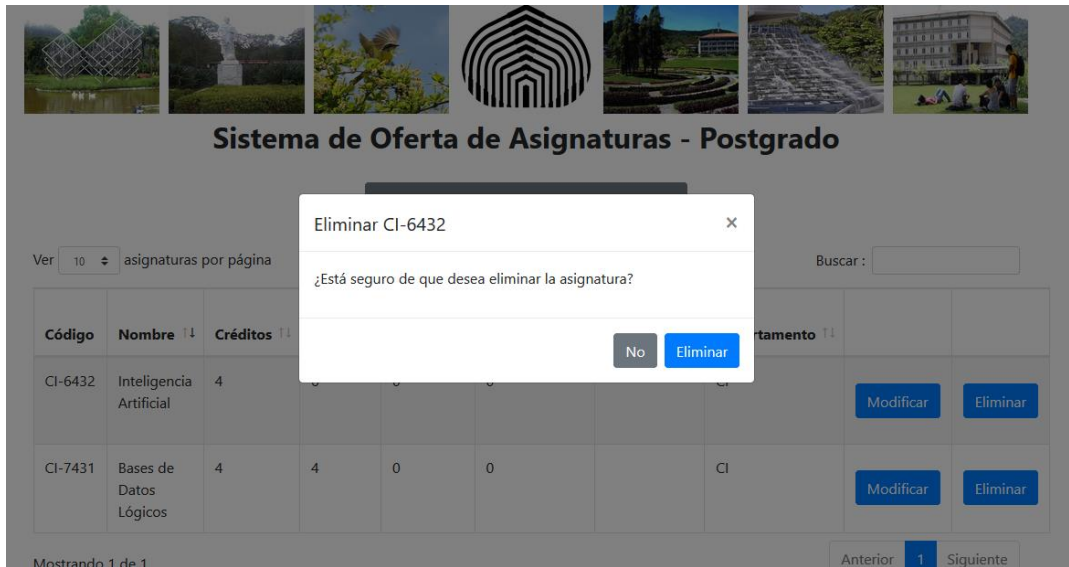


Figura 5. Alerta de Eliminación

Por lo tanto, las funciones CRUD de la base de datos fueron implementadas de forma correcta en el software desarrollado.

Adicionalmente, la búsqueda se realiza en la barra que se encuentra en la parte superior derecha de la tabla. Al ingresar el nombre o código de la asignatura el sistema arroja los resultados correspondientes, o indica que no se encontraron asignaturas en caso contrario tal como se muestra en la Figura 6. Esta funcionalidad da respuesta a la historia 2.2.

Ver  asignaturas por página

Buscar :

Código	Nombre	Créditos	Horas Teoría	Horas Práctica	Horas Laboratorio	Requisitos	Departamento		
CI-6432	Inteligencia Artificial	4	6	0	0		CI	Modificar	Eliminar

Mostrando 1 de 1 (Filtrado de 2 total asignaturas registradas)

Anterior  Siguiente

Ver  asignaturas por página

Buscar:

Código	Nombre <span>↑↓</span>	Créditos <span>↑↓</span>	Horas Teoría <span>↑↓</span>	Horas Práctica <span>↑↓</span>	Horas Laboratorio <span>↑↓</span>	Requisitos <span>↑↓</span>	Departamento <span>↑↓</span>		
No se encontraron asignaturas									

No hay asignaturas disponibles (Filtrado de 2 total asignaturas registradas)

Anterior Siguiente

Figura 6. Búsqueda de Asignaturas

Por otro lado, en el encabezado de la tabla se puede ordenar las entradas de forma alfabética ascendente o descendente tal como se muestra en la Figura 7, cumpliendo con lo requerido por la historia de usuario 2.5.

Ver  asignaturas por página

Buscar:

Código	Nombre <span>↑↓</span>	Créditos <span>↑↓</span>	Horas Teoría <span>↑↓</span>	Horas Práctica <span>↑↓</span>	Horas Laboratorio <span>↑↓</span>	Requisitos <span>↑↓</span>	Departamento <span>↑↓</span>		
CI-7431	Bases de Datos Lógicos	4	4	0	0		CI	Modificar	Eliminar
CI-6432	Inteligencia Artificial	4	6	0	0		CI	Modificar	Eliminar

Mostrando 1 de 1

Anterior **1** Siguiente

Ver  asignaturas por página

Buscar:

Código	Nombre <span>↑↓</span>	Créditos <span>↑↓</span>	Horas Teoría <span>↑↓</span>	Horas Práctica <span>↑↓</span>	Horas Laboratorio <span>↑↓</span>	Requisitos <span>↑↓</span>	Departamento <span>↑↓</span>		
CI-6432	Inteligencia Artificial	4	6	0	0		CI	Modificar	Eliminar
CI-7431	Bases de Datos Lógicos	4	4	0	0		CI	Modificar	Eliminar

Mostrando 1 de 1

Anterior **1** Siguiente

Figura 7. Tablas en Orden Alfabético Ascendente y Descendente

Para complementar la documentación visual expuesta en este informe, en el repositorio de Github se encuentra el código comentado y más detalles de las funcionalidades desarrolladas en el archivo README.

## División de Actividades

A continuación se presentan las tareas asignadas internamente en el equipo para llevar a cabo la entrega del Sprint 1 con sus respectivos responsables y número de horas invertidas.

**Tabla N°1 “Tareas Asignadas para el Desarrollo del Sprint 1”**

Tarea	Responsable	Tiempo
Reunión Pre-Sprint 1	Delta Developers	1 hora
Análisis de Requerimientos	Delta Developers	Continuamente durante toda la etapa de desarrollo
Diseño de Diagrama ER	Rosana García, Pedro Maldonado	2 horas
Creación de repositorio en Github	Constanza Abarca	10 minutos
Crear ambiente de programación en django	Delta Developers	10 minutos
Implementar BD en django	Fabio Suárez	2 horas
Página de Inicio con Login / Registro.	Constanza Abarca, Pedro Maldonado	2 horas
Fixture de django para cargar datos en la BD.	Fabio Suárez	1 hora
Formulario de Asignaturas	Moises González	3 horas
Validación de campos en la BD.	Fabio Suárez, Pedro Maldonado	1 hora
Validación de campos en el formulario de Asignaturas.	Moises González	1 hora
Vista de tabla de Asignatura	Constanza Abarca	3 horas
Modificar datos en la tabla de asignaturas	Constanza Abarca	2 horas
Eliminar entrada en la tabla de asignaturas	Constanza Abarca	2 horas

Búsqueda de Asignaturas en la tabla por código o por nombre.	Constanza Abarca, Moises González	3 horas
Listar las asignaturas según criterio de orden.	Constanza Abarca, Moises Gonzalez	1 hora
Diseño de Pruebas	Denylson Romero	2 hora
Implementación de Pruebas	Denylson Romero	2 horas
Redactar informe Sprint I	Pedro Maldonado	4 horas
Revisiones finales	Delta Developers	1 hora

## Experiencia de Trabajo en Equipo

La experiencia de trabajo en equipo en el desarrollo del software fue complicada debido a la cantidad de eventualidades ocurridas desde el inicio de la reunión de Pre-Sprint 1, desde falta de agua en la Universidad hasta fallas eléctricas dificultaron el desempeño de los roles de cada integrante del equipo. Además de que se juntaron muchas actividades que habían quedado aplazadas por problemas del agua en la Universidad para la misma semana de la entrega del Sprint, lo que significó un grado de dificultad mayor, sin embargo, es deber del equipo organizarse de la forma más efectiva para evitar este tipo de inconvenientes que pueden seguir sucediendo a lo largo del proceso.

En general el equipo trabajó de forma correcta, sin embargo, la comunicación no se realizó de la forma más adecuada, ya que, existieron temas tratados que no fueron discutidos con todos los miembros, lo que desencadenó un fallo en el análisis de requerimientos y fue necesaria una reestructuración del trabajo realizado así como una reorganización interna del equipo y del plan de comunicación establecido en la fase de formación del equipo.

La reorganización mencionada incluyó la reasignación de tareas en parejas para aplicar la programación por pares y así evitar posibles conflictos en el repositorio. Por

otro lado, quedó definido que todas las decisiones deben discutirse con todos los miembros del equipo y que cualquier duda que se presente expresarla directamente con el SCRUM Master o el Product Owner.

## **Experiencia de Trabajo con TDD**

En el caso del proceso de TDD, todos los miembros del equipo ya cuentan con destrezas en esta área, pues ya se han realizado dos tareas en la asignatura relacionadas con el desarrollo orientado a pruebas. Se utilizaron las herramientas que proporciona django para la realización de pruebas sobre la base de datos implementada. Django admin se utilizó para pruebas iniciales en la base de datos y crear el usuario Jefe de Departamento. Por otro lado, en el módulo de pruebas test.py proporcionado por django, se realizaron las pruebas unitarias para los formularios evaluando que se realizaran las validaciones de los campos de forma correcta. Una vez aprobadas las pruebas, se realizaron los procedimientos de refactorización para mejorar el estilo del código sin dañar las funcionalidades ya desarrolladas y al mismo tiempo acomodar detalles estéticos en las vistas y los formularios para la presentación del producto.

## **Manejo Repositorio GitHub**

En el repositorio de git se crearon varias ramas para trabajar simultáneamente en diferentes tareas y así agilizar el trabajo en el equipo.

- Master
  - Base de Datos (BD)
  - Comprobaciones
  - Arreglar BD
  - Página de Inicio / Login / Registro (Pedro)
  - Documentación (Rosana)
  - Alerta Eliminar



- Pruebas (Denylson)
- Solo Jefe (Moises)

Como consecuencia de la reestructuración en la estrategia de trabajo del equipo, se realizó un nuevo repositorio para trabajar en base al catálogo especificado en el Product Backlog. Por lo tanto, en la rama master ya se encuentra varias de las funcionalidades del software, como lo son los formularios y las operaciones CRUD. En las ramas BD, Comprobaciones y Arreglar BD se trabajaron las modificaciones necesarias para satisfacer los requerimientos del cliente, así como adaptar las respectivas verificaciones indicadas en las historias de usuario trabajadas. En la rama de Página de Inicio se desarrolló una vista donde se integra en login o un registro, en caso de ser un usuario nuevo, para poder acceder al sistema donde se encuentra el catálogo de las asignaturas. Se creó una rama de pruebas para realizar las pruebas unitarias de tipo frontera, esquina y malicia con el fin de verificar la funcionalidad de los formularios, como se indicó en la experiencia de trabajo en TDD.

## **Conclusiones**

Una vez finalizada la etapa de desarrollo es posible concluir que se cumplió con el objetivo planteado del Sprint 1 de diseñar e implementar la base de datos para las asignaturas de postgrado dictadas en la Universidad Simón Bolívar a pesar de todas las dificultades presentadas a lo largo de las dos semanas de trabajo.

La metodología de trabajo SCRUM ayudó a la solución de algunos problemas ya mencionados y agilizó la asignación de tareas a cada miembro del equipo para que cada uno desempeñe sus respectivos roles de trabajo en función de las historias de usuario proporcionadas en el Product Backlog y así lograr el objetivo establecido.

En cuanto a las recomendaciones se tiene la de contar con un tiempo de holgura para la entrega de los siguientes Sprint, ya que, será necesario para prevenir las eventualidades que pueden continuar ocurriendo en la Universidad.

Finalmente, se espera que para la entrega del Sprint 2 se pueda tener una mejor organización del equipo y establecer una comunicación ágil y directa entre todos los miembros y cualquier tipo de duda expresarla al SCRUM Master o al cliente para obtener un excelente desempeño y reducir el impacto de las eventualidades que pueden darse sea en la Universidad o en el entorno de trabajo de cada integrante.