

# **ESCUELA POLITÉCNICA NACIONAL**

## **ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

**DESARROLLO DE UN SISTEMA WEB INFORMATIVO DE TRÁMITES  
ACADÉMICOS Y ADMINISTRATIVOS PARA LA ESFOT**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
TECNÓLOGO EN ANÁLISIS DE SISTEMAS INFORMÁTICOS**

**DIANA JANNETH NARVÁEZ VALDEZ**

diana.narvaez@epn.edu.ec

**DIRECTOR: ING. BYRON GUSTAVO LOARTE CAJAMARCA, MSC.**

byron.loarteb@epn.edu.ec

**CODIRECTOR: ING. LUZ MARINA VINTIMILLA JARAMILLO, MSC.**

marina.vintimilla@epn.edu.ec

**Quito, diciembre 2021**

## CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por Narváez Valdez Diana Janneth como requerimiento parcial a la obtención del título de Tecnólogo en Análisis de Sistemas Informáticos, bajo nuestra supervisión:



---

**Ing. Byron Loarte, MSc.**

DIRECTOR DEL PROYECTO



---

**Ing. Luz Marina Vintimilla, MSc.**

CODIRECTORA DEL PROYECTO

## **DECLARACIÓN**

Yo Narváez Valdez Diana Janneth con CI: 1722305198 declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, soy titular de la obra en mención y otorgo una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entrego toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.

# **DEDICATORIA**

A mis padres y familiares, quienes han sido un apoyo incondicional en el transcurso de toda mi carrera universitaria, por siempre estar pendientes y alentándome a cumplir mis metas planteadas en mi vida.

**NARVÁEZ VALDEZ DIANA JANNETH**

## **AGRADECIMIENTO**

En primer lugar, agradezco a Dios, por ser el guía de mi vida, agradezco a mis padres, hermanos y amigos por su cariño, dedicación y siempre ser ese apoyo en momentos difíciles, agradezco a la Escuela Politécnica Nacional, por haber sido parte de mi formación académica y haberme brindado la oportunidad de crecer personal y profesionalmente.

Agradezco a los todos los profesores por compartir sus conocimientos cada día, quiero agradecer especialmente al Ing. Byron Loarte y a la Ing. Luz Marina Vintimilla, por haberme apoyado y guiado en la culminación de mi tesis.

**NARVÁEZ VALDEZ DIANA JANNETH**

# ÍNDICE DE CONTENIDOS

1	Introducción .....	1
1.1	Objetivo general .....	1
1.2	Objetivos específicos.....	1
1.3	Alcance.....	2
2	Metodología.....	4
2.1	Metodología de Desarrollo.....	4
2.1.1	Roles .....	4
2.1.2	Artefactos .....	6
2.2	Diseño de interfaces .....	9
2.2.1	Herramienta utilizada para el diseño.....	9
2.2.2	Sistema web.....	9
2.3	Diseño de la arquitectura.....	10
2.3.1	Patrón arquitectónico.....	10
2.3.2	Patrón arquitectónico utilizado en el sistema web.....	10
2.3.3	Herramientas utilizadas en el sistema web .....	11
3	Resultados y Discusión.....	14
3.1	<i>Sprint</i> 0. Configuración del ambiente de desarrollo.....	14
3.1.1	Definición de los requerimientos funcionales y no funcionales del sistema web .....	14
3.1.2	Diseño de la Base de Datos NoSQL en función al proyecto .....	15
3.1.3	Descripción de usuarios .....	16
3.1.4	Estructura del Proyecto .....	17
3.2	<i>Sprint</i> 1. Implementación del usuario administrador y módulos asignados a su cargo .....	17
3.2.1	Visualizar página informativa .....	18
3.2.2	Ingreso del usuario administrador.....	18
3.2.3	Registrar, visualizar, modificar y eliminar categorías .....	19
3.2.4	Registrar, visualizar, modificar y eliminar subcategorías.....	20

3.2.5	Registrar, visualizar, modificar y eliminar trámites .....	20
3.2.6	Aprobación y rechazo de formularios de quejas, novedades o sugerencias .....	21
3.2.7	Enviar notificaciones de noticias a usuarios estudiantes.....	22
3.3	<i>Sprint 2.</i> Implementación del usuario estudiante y módulos asignados a su cargo .....	22
3.3.1	Registro del usuario estudiante e inicio de sesión .....	22
3.3.2	Visualización y modificación de perfil.....	23
3.3.3	Visualización de categorías .....	24
3.3.4	Visualización de procedimientos y formularios de trámites .....	24
3.3.5	Llenar formularios de quejas, novedades o sugerencias .....	25
3.3.6	Recibir notificaciones de noticias .....	25
3.4	<i>Sprint 3.</i> Pruebas del sistema web.....	26
3.4.1	Ejecución de pruebas unitarias y resultados .....	26
3.4.2	Ejecución de pruebas de rendimiento y resultados .....	27
3.4.3	Ejecución de pruebas de compatibilidad y resultados.....	28
3.4.4	Ejecución de pruebas de aceptación y resultados .....	29
3.5	<i>Sprint 4.</i> Despliegue del sistema web .....	30
3.5.1	Despliegue a producción en el <i>hosting</i> de <i>Firebase</i> .....	30
4	Conclusiones y Recomendaciones .....	32
4.1	Conclusiones .....	32
4.2	Recomendaciones .....	33
5	Referencias .....	34
6	ANEXOS.....	38
6.1	Manual Técnico .....	38
6.2	Manual de Usuario.....	38
6.3	Manual de Instalación .....	38

## ÍNDICE DE FIGURAS

<b>Fig. 1:</b> Prototipo de inicio de sesión para el usuario administrador. ....	9
<b>Fig. 2:</b> Patrón Arquitectónico. ....	11
<b>Fig. 3:</b> Colecciones de la Base de Datos. ....	16
<b>Fig. 4:</b> Usuario administrador y usuario estudiante con sus respectivos módulos.....	16
<b>Fig. 5:</b> Componentes de archivos y directorios para el sistema web.....	17
<b>Fig. 6:</b> Página Informativa del sistema web. ....	18
<b>Fig. 7:</b> Inicio de sesión para los usuarios administrador y estudiante. ....	19
<b>Fig. 8:</b> Módulo para gestionar categorías. ....	19
<b>Fig. 9:</b> Módulo para gestionar subcategorías.....	20
<b>Fig. 10:</b> Módulo para gestionar trámites. ....	21
<b>Fig. 11:</b> Módulo para gestionar quejas, sugerencias y novedades.....	21
<b>Fig. 12:</b> Módulo de envío de noticias. ....	22
<b>Fig. 13:</b> Registro del usuario estudiante. ....	23
<b>Fig. 14:</b> Módulo de perfil para el estudiante.....	23
<b>Fig. 15:</b> Módulo para visualizar categorías. ....	24
<b>Fig. 16:</b> Módulo categorías - lista de trámites. ....	25
<b>Fig. 17:</b> Módulo de sugerencias.....	25
<b>Fig. 18:</b> Módulo de sugerencias - sección notificaciones. ....	26
<b>Fig. 19:</b> Estructura de la prueba unitaria para validar el campo email y contraseña del usuario estudiante. ....	27
<b>Fig. 20:</b> Resultado de las pruebas unitarias ejecutadas en el componente Login. ....	27
<b>Fig. 21:</b> Prueba de rendimiento de 1000 visitas en 10s. ....	28
<b>Fig. 22:</b> Creación del proyecto en Firebase. ....	30
<b>Fig. 23:</b> Sistema web en producción.....	31



## ÍNDICE DE TABLAS

<b>TABLA I:</b> Roles Asignados.....	5
<b>TABLA II:</b> Formato para la Recopilación de Requerimientos.....	6
<b>TABLA III:</b> Historia de Usuario 1 – Visualizar página informativa. ....	7
<b>TABLA IV:</b> Formato para el Product Backlog. ....	7
<b>TABLA V:</b> Formato del Sprint Backlog. ....	8
<b>TABLA VI:</b> Descripción de las herramientas que se han utilizado en el desarrollo. ....	11
<b>TABLA VII:</b> Librerías que se han utilizado en la codificación del proyecto.....	12
<b>TABLA VIII:</b> Navegadores utilizados en las pruebas de compatibilidad.....	28
<b>TABLA IX:</b> Prueba de aceptación 1 - Visualizar página informativa .....	29

## RESUMEN

Actualmente la Escuela Politécnica Nacional, dispone de un Repositorio Documental (ATENEA) el cual es de libre acceso con una serie de formularios para que los estudiantes de las diferentes carreras como: Ingeniería, Tecnología Superior y Propedéutico puedan utilizarlos en los diferentes trámites ya sean administrativos y/o académicos. Sin embargo, se ha evidenciado una serie de inconvenientes en este Repositorio Documental ya que al existir una extensa variedad de documentos y formularios esto provoca, por una parte, pérdida de tiempo en encontrar el formulario, ya que la mayoría son de difícil acceso y en otras ocasiones, no se pueden descargar, no corresponden a una categoría en particular o que algunos de ellos se encuentren incompletos, desactualizados o no se puede acceder al repositorio por fallas técnicas, entre otros. Ocasionando que los estudiantes no puedan completar el trámite solicitado.

Por lo antes mencionado y para dar una solución a los estudiantes de las diferentes carreras de la ESFOT, se ha desarrollado un sistema web que permita brindar información detallada sobre las distintas opciones de trámites, requisitos, formularios y el procedimiento que deben seguir en cada uno de los trámites administrativos y/o académicos. Este proyecto ha sido desarrollado utilizando la metodología ágil *Scrum* y un conjunto de herramientas actuales de desarrollo para el cumplimiento planificado de los objetivos planteados en cada avance.

El presente informe técnico se ha estructurado de la siguiente manera: en la sección I correspondiente a la Introducción se da a conocer sobre el contexto del problema, objetivo general, objetivos específicos y el alcance del proyecto. En la sección II se describe como se ha implementado de forma adecuada cada una de las fases de la metodología ágil de desarrollo *Scrum*, patrón arquitectónico y cada una de las herramientas de desarrollo para la etapa de codificación. Por otra parte, en la sección III se mencionan los resultados obtenidos en las tareas que se han planificado y ejecutado por cada *Sprint* de desarrollo. Finalmente, en la última sección se mencionan las conclusiones y recomendaciones que se han obtenido a lo largo del desarrollo de este proyecto integrador.

**PALABRAS CLAVE:** Formularios, Trámites, Angular, *Firebase*, *Scrum*, *NinjaMock*

## **ABSTRACT**

Currently, the National Polytechnic School has a Documentary Repository (ATENEA) which is freely accessible with a series of forms so that students of different careers such as: Engineering, Higher Technology and Propedeutic can use them in the different procedures, whether administrative and / or academics. However, a series of drawbacks have been evidenced in this Documentary Repository since there is a wide variety of documents and forms, this causes, on the one hand, wasted time in finding the form, since most are difficult to access and in other times, they cannot be downloaded, they do not correspond to a particular category or that some of them are incomplete, outdated or the repository cannot be accessed due to technical failures, among others. Causing students to not be able to complete the requested procedure.

Due to the aforementioned and to provide a solution to the students of the different ESFOT careers, a web system has been developed that allows providing detailed information on the different options for procedures, requirements, forms and the procedure that they must follow in each one. of administrative and / or academic procedures. This project has been developed using the agile Scrum methodology and a set of current development tools for the planned fulfillment of the objectives set in each advance.

This technical report has been structured as follows: in section I corresponding to the Introduction, information is given on the context of the problem, general objective, specific objectives and the scope of the project. Section II describes how each of the phases of the agile Scrum development methodology, the architectural pattern and each of the development tools for the coding stage has been properly implemented. On the other hand, in section III the results obtained in the tasks that have been planned and executed for each development Sprint are mentioned. Finally, in the last section the conclusions and recommendations that have been obtained throughout the development of this integrating project are mentioned.

**KEYWORDS:** Forms, Procedures, Angular, Firebase, Scrum, NinjaMock

# 1 INTRODUCCIÓN

Los estudiantes de las diferentes carreras de la ESFOT no cuentan con suficiente información sobre el proceso que deben realizar en cada uno de los trámites académicos y/o administrativos. La manera por la cual se da a conocer esta información, es por medio del uso de carteleras informativas, redes sociales, correspondencia institucional y la web de la EPN, entre otros. En los medios citados previamente, cabe recalcar que se adjuntan cuestionarios sin detallar información de forma clara, precisa y en muchas situaciones al ser información volátil, resulta molesto y tedioso encontrar dicha información para su uso respectivo [1].

Desde otra perspectiva, el desconocimiento por parte de los estudiantes sobre el procedimiento que deben realizar ya sea en un trámite académico y/o administrativo en la EPN, representa una problemática dentro de la comunidad politécnica, el cual se vuelve notorio al inicio y fin de un período académico y más aun con los estudiantes que son de otras facultades y de aquellos que ingresan por primera vez a la universidad [2].

Actualmente es esencial la utilización de la tecnología, ya que posibilita automatizar procedimientos, métodos y ofrecer información actualizada en todo momento, en una época en el cual el cliente exige velocidad, hiperconectividad y transparencia. Los sistemas web poseen todas las características de cubrir estas necesidades y las que se presenten en un futuro [3].

Por ello, se ha implementado un sistema web que permite dar información actualizada, descriptiva y en tiempo real sobre las distintas opciones de trámites, requisitos, formularios y el procedimiento que deben seguir los estudiantes en cada uno de los trámites administrativos y/o académicos. Apoyando de esta manera a la ESFOT a mejorar la forma de comunicarse con su comunidad por medio del uso de la tecnología.

## 1.1 Objetivo general

Desarrollar un sistema web de trámites académicos y administrativos para la ESFOT.

## 1.2 Objetivos específicos

- Determinar los requerimientos funcionales y no funcionales del sistema web.
- Diseñar la arquitectura, modelo de base de datos e interfaces del sistema web.
- Implementar los módulos del sistema web bajo el diseño propuesto.

- Evaluar el funcionamiento del sistema web.

### **1.3 Alcance**

En la actualidad los sistemas web son muy utilizados en diferentes campos y áreas del conocimiento, esto se debe a lo práctico y beneficioso que resultan ser. Además, facilitan de forma notable la administración de la información y procesos como un beneficio para toda la organización, convirtiéndose en herramientas indispensables en el día a día de las personas y de las organizaciones [4].

Una de las grandes ventajas que ofrece un sistema web es lograr una mayor organización de la información, almacenando y distribuyendo de forma limpia y clara todos los procesos que se administren sobre el mismo [5].

Por lo antes citado, los usuarios finales de este sistema web pueden contar con una fuente de información organizada, estructurada y en tiempo real sobre el procedimiento que tienen que realizar los estudiantes de las diferentes carreras de la ESFOT ya sea en un trámite académico y/o administrativo. Utilizando para ello una metodología ágil de desarrollo (*Scrum*) y herramientas de desarrollo modernas que sean capaces de ajustarse frente a este escenario y brindar el mejor rendimiento cuando sean puestas en producción [6].

Por otra parte, el sistema web garantiza la integridad, consistencia y seguridad de los datos, a través de varios perfiles y un módulo de autenticación. Los perfiles de usuarios que se han desarrollado son los siguientes:

#### **Perfiles en el sistema web:**

- Administrador.
- Estudiante.

#### **El perfil administrador en el sistema web permite:**

- Ingreso al sistema web.
- Actualizar su perfil.
- Registrar, listar, reemplazar y eliminar formularios.
- Registrar, listar, reemplazar y eliminar categorías.
- Registrar, listar, reemplazar y eliminar procedimiento de trámites.
- Visualizar formulario de quejas, novedades o sugerencias.

### **El perfil estudiante en el sistema web permite:**

- Registro por medio de un formulario.
- Actualización de su perfil.
- Filtrado de trámites por categorías.
- Visualización detallada del procedimiento a seguir para un trámite respectivo.
- Enviar formulario de quejas, novedades o sugerencias.

## 2 METODOLOGÍA

Las metodologías de desarrollo ágiles en la actualidad permiten que se adapten a las condiciones del proyecto consiguiendo rapidez y flexibilidad. Esto se logra ya que por medio de etapas bien definidas como la planificación, diseño, codificación y pruebas se van obteniendo resultados en cada implementación de forma flexible y modular. Logrando de esta manera tener una versión final del *software* con nuevas funcionalidades listas para ser puestas en producción [7] [8].

Es por esta razón, que el desarrollo de este sistema web se ha implementado con *Scrum*, una metodología ágil donde el equipo de desarrollo trabaja como una unidad para alcanzar un objetivo común. Por otra parte, *Scrum* permite una constante comunicación, colaboración y retroalimentación con el cliente, con el objetivo de minimizar los riesgos que afecten a la calidad del producto final y a la productividad de todo el equipo [9].

En las siguientes subsecciones se detalla el cómo se ha implementado cada una de las fases y actividades de la metodología a lo largo del desarrollo y puesta a producción del sistema web, el cual parte con una serie de actividades como: Recopilación de Requisitos, Historias de Usuario, *Product Backlog* y *Sprints* de desarrollo. Asegurando el cumplimiento del objetivo planteado y la calidad de producto final.

### 2.1 Metodología de Desarrollo

La metodología ágil *Scrum* está basada en *Sprints* o comúnmente llamada iteraciones, las mismas que permiten dirigir con éxito el progreso de un proyecto de *software*. Sin embargo, una de las fases más cruciales es la fase de planificación, ya que permite al equipo de desarrollo tener todo el conocimiento de que va ocurrir en el desarrollo del proyecto. Por otra parte, los miembros del equipo frecuentemente inspeccionan el progreso del proyecto por medio de reuniones y entregas continuas con la finalidad del que el proyecto fluya y que el equipo funcione de manera auto organizada. Una vez, concluida la fase de codificación, se realiza una secuencia de pruebas las cuales sirven para la verificación del código fuente del proyecto y el nivel de aprobación [10].

#### 2.1.1 Roles

*Scrum* está basado en una serie de roles, donde la participación de cada integrante en el desarrollo del proyecto es necesaria, ya que cada uno tiene establecido un objetivo por cumplir. Además, la colaboración es indispensable para la ejecución y culminación

del objetivo planteado, debido a que tienen que estar involucrados con el mismo y son causantes del triunfo o fracaso de cada *Sprint* [11]. En este sentido, los roles que se han asignado están comprendidos de la siguiente manera:

### ***Product Owner***

Es el cliente o dueño del producto, el cual tiene como tarea comunicar la visión del producto y las características que debe poseer [12]. Este rol es desempeñado por el personal administrativo de la ESFOT, responsable de proporcionar toda la información sobre las opciones de trámites, requisitos, formularios y el procedimiento que deben realizar los estudiantes. Logrando de esta manera cumplir con la etapa de planificación y con ello definir acertadamente los prototipos, herramientas, patrón arquitectónico, perfiles, módulos y funciones para el sistema web.

### ***Scrum Master***

Es aquella persona responsable que coordina las reuniones y dirige al equipo de desarrollo en el cumplimiento de los objetivos establecidos [12]. Este rol es desempeñado por el director del proyecto de titulación. El cual tiene la capacidad y habilidades de encaminar al equipo de trabajo en la ejecución correcta del objetivo planteado y solventar aquellos impedimentos que retrasen el desarrollo del proyecto integrador.

### ***Development Team***

Son personas con habilidades técnicas para transformar los requerimientos del *Product Owner* en un producto totalmente funcional en cada *Sprint* de desarrollo [12]. Este rol lo desempeña una sola estudiante como trabajo de titulación, quien tiene los conocimientos y habilidades necesarias para llevar a cabo la codificación, pruebas y despliegue del sistema web a producción. Por último, la **TABLA I** muestra todo el equipo *Scrum* el cual está formado de la siguiente manera:

**TABLA I:** Roles Asignados

<b>ROLES</b>	<b>NOMBRES</b>
<i>Product Owner</i>	Personal administrativo de la ESFOT
<i>Scrum Master</i>	Ing. Byron Loarte, MSc.
<i>Development Team</i>	Diana Narváez



### 2.1.2 Artefactos

En *Scrum*, los artefactos son las bases necesarias para la representación de la información y mantener una estructura organizada durante el desarrollo de un producto de *software*. Además, por medio de estos artefactos se garantiza el cumplimiento de las tareas asignadas a cada uno de los miembros del equipo [13]. Por lo que, al utilizar esta metodología se han determinado los siguientes artefactos para su implementación y ejecución:

#### Recopilación de Requerimientos

Consiste en definir mediante una serie de reuniones y entrevistas todos los requisitos funcionales y no funcionales que tiene un *software*, permitiendo de esta manera trabajar sin ambigüedades en las etapas posteriores [14]. De esta manera gracias a las reuniones mantenidas con todo el equipo de trabajo se ha logrado establecer un listado de requerimientos, la cual contiene todas las funcionalidades que debe poseer el sistema web y los perfiles asignados a las mismas.

La **TABLA II** se ha utilizado como formato para elaborar la Recopilación de Requerimientos. Mientras que, la tabla completa se encuentra detallada en la sección Recopilación de Requerimientos del Manual Técnico (pág. 2-3).

**TABLA II:** Formato para la Recopilación de Requerimientos.

RECOPIACIÓN DE REQUERIMIENTOS		
TIPO DE SISTEMA	ID RR	DESCRIPCIÓN
SISTEMA WEB	RR001	Como usuario administrador y estudiante requieren, visualizar una página principal e informativa del sistema web.

#### Historias de Usuario

Una vez establecidos todos los requerimientos del sistema web y perfiles que van a interactuar sobre el mismo. Se procede con la elaboración de la Historias de Usuarios, las cuales facilita el entendimiento del requerimiento, con una serie de tarjetas que recogen de forma detallada todas las funcionalidades y restricciones que va a tener el módulo en función a las necesidades del estudiante o administrador [15].

Estas Historias de Usuario han permitido describir de manera concisa cada uno de los perfiles y funcionalidades que se deben incorporar en la etapa de codificación. Posteriormente, la **TABLA III** es un modelo y ejemplo de una de las Historias de

Usuario, mientras que las Historias de Usuario restantes se encuentran detalladas en el apartado Historias de Usuario del Manual Técnico (pág. 4-17).

**TABLA III:** Historia de Usuario 1 – Visualizar página informativa.

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU001	<b>Usuario:</b> Administrador
<b>Nombre Historia:</b> Visualizar página Informativa.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada:</b> 1	
<b>Responsable (es):</b> Diana Narváez	
<b>Descripción:</b> El usuario administrador y estudiante pueden visualizar una página principal e informativa donde se detalla las funcionalidades del sistema web, módulos disponibles y detalles del proyecto integrador.	
<b>Observación:</b> El usuario administrador y estudiante pueden acceder a las demás funcionalidades del sistema web por medio de un inicio de sesión.	

### ***Product Backlog***

Se lo denomina como una lista estructurada de todos los requerimientos, presentados a partir de la priorización de cada una de las actividades que se deben cumplir para el desarrollo del proyecto. Además, facilita notablemente el seguimiento de cada uno de los requerimientos en cada una de las iteraciones [16].

El *Product Backlog* ha permitido listar de forma ordenada todos los requerimientos en función de: complejidad de desarrollo, prioridad y necesidades del cliente, muestra de ello, se ha utilizado la **TABLA IV** como formato para listar cada uno de los requerimientos. Mientras que, la tabla completa se encuentra detallada en la sección *Product Backlog* del Manual Técnico (pág. 18 - 19).

**TABLA IV:** Formato para el *Product Backlog*.

<b>ELABORACIÓN DEL <i>PRODUCT BACKLOG</i></b>				
<b>ID-HU</b>	<b>Historia de Usuario</b>	<b>No. Iteración</b>	<b>Prioridad</b>	<b>Estado</b>
<b>HU001</b>	Visualizar página informativa	1	Media	Finalizado

## ***Sprint Backlog***

Es una nomina de actividades identificadas por el *Scrum Team* o equipo de desarrollo, se representa a través de un tablero de tareas, las cuales, hacen visibles todo el trabajo necesario para alcanzar el compromiso que se ha realizado con cliente. Además, permite ver las tareas donde el equipo puede presentar inconvenientes y en donde se puede tomar decisiones claras al respecto [17].

A partir del listado del *Product Backlog*, se establecen una serie de actividades con un periodo corto de tiempo. Sin embargo, con el objetivo de llevar un mejor control de esto, la **TABLA V** se ha utilizado como formato para listar los 5 *Sprints* de desarrollo y cada una de las actividades que se deben llevar a cabo: Configuración del ambiente de desarrollo, Implementación de módulos dirigida al usuario a cargo del sistema web, Implementación de módulos dirigida al usuario estudiante, pruebas y el sistema web puesto en producción; se encuentran detallados en el Manual Técnico – Sección *Sprint Backlog* (pág. 20 - 26).

**TABLA V:** Formato del *Sprint Backlog*.

<b>ELABORACIÓN DEL <i>SPRINT BACKLOG</i></b>						
<b>ID-SB</b>	<b>NOMBRE</b>	<b>MÓDULO</b>	<b>ID-HU</b>	<b>HISTORIA DE USUARIO</b>	<b>TAREAS</b>	<b>TIEMPO ESTIMADO</b>
SB000	Configuración del ambiente de desarrollo	-----	-----	-----	<ul style="list-style-type: none"><li>• Definir los requerimientos funcionales y no funcionales del sistema web.</li><li>• Diseño de la Base de Datos.</li><li>• Prototipos del sistema web.</li><li>• Configuración del entorno del back-end.</li><li>• Configuración del entorno del front-end.</li></ul>	10H

## 2.2 Diseño de interfaces

Tras la identificación de todos los requisitos funcionales, módulos y perfiles del sistema web, como siguiente paso se procede con el diseño de cada una de las interfaces para el sistema web, con el objetivo de que el *Product Owner* pueda tener una visión más general del producto final implementado.

### 2.2.1 Herramienta utilizada para el diseño

*NinjaMock* es un *software* amigable y gratuito, el cual permite crear prototipos, bocetos y diseños de interfaces para cualquier tipo de aplicaciones de forma intuitiva y sin tanto código [18]. Es por esta razón que a través de este *software* se ha logrado diseñar cada una de las interfaces para el sistema web tomando en cuenta la experiencia de usuario (UX) y logrando de esta manera que todo el equipo *Scrum* pueda tener una idea más concreta del diseño resultante antes de ser codificado. Más adelante, se muestra uno de los diseños que se han realizado.

### 2.2.2 Sistema web

La **Fig. 1** muestra un primer ejemplar, el cual viene a ser el prototipo donde el usuario administrador puede realizar el inicio de sesión en el sistema web. Los diseños completos de los prototipos restantes se presentan en el apartado de Diseño de Interfaces del Manual Técnico (pág. 27 - 38).



**Fig. 1:** Prototipo de inicio de sesión para el usuario administrador.

## **2.3 Diseño de la arquitectura**

Después de haber establecido los requisitos funcionales y prototipos para el sistema web, en este apartado se da a conocer el patrón arquitectónico realizado para la codificación de los módulos y perfiles; así como la compatibilidad e integración con cada una de las herramientas y librerías establecidas.

### **2.3.1 Patrón arquitectónico**

Modelo-Vista-Controlador (MVC) es el patrón arquitectónico que se ha determinado para la codificación del proyecto, el cual es una propuesta de arquitectura ampliamente utilizado en la industria del desarrollo de *software*, otorgando una serie de ventajas como: detección de errores, código más limpio y legible, componentes más modulares y la incorporación de nuevas funcionalidades conforme se vaya requiriendo a lo largo del desarrollo [19].

### **2.3.2 Patrón arquitectónico utilizado en el sistema web**

La **Fig. 2** muestra el patrón arquitectónico que se implementado en la codificación del proyecto en base a las herramientas y librerías que se han utilizado. Logrando de esta manera tener un proyecto mucho más estructurado, organizado y la integración de nuevo código de una manera más eficiente.

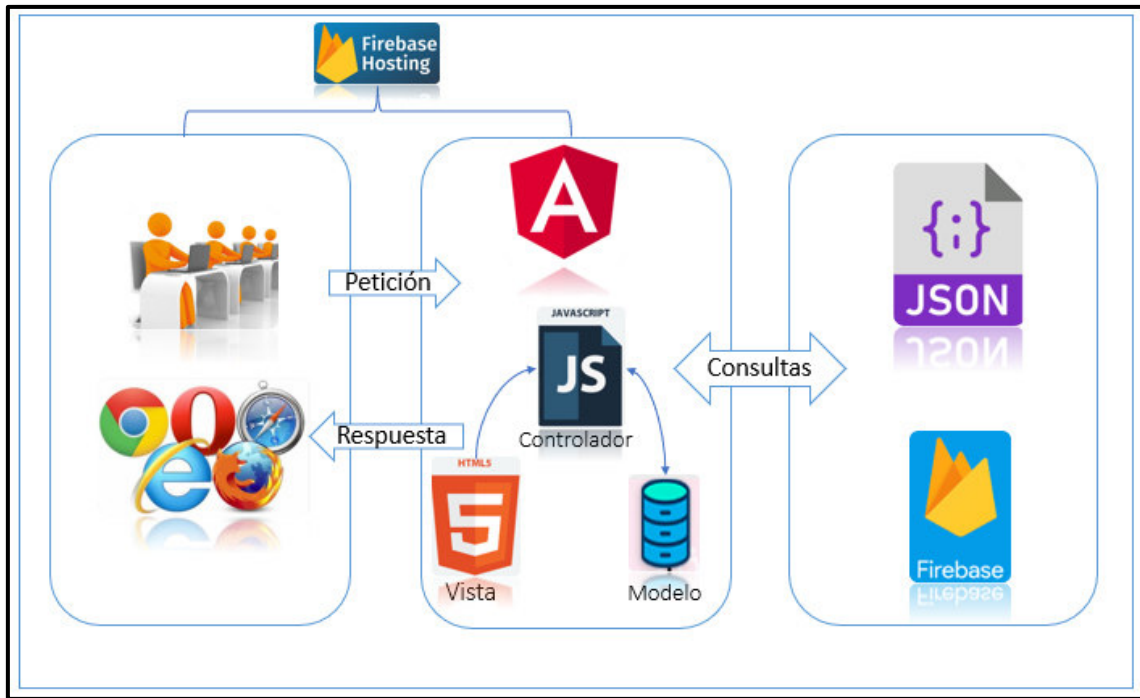


Fig. 2: Patrón Arquitectónico.

### 2.3.3 Herramientas de Desarrollo

Después de establecer requisitos funcionales, prototipos y patrón arquitectónico descritos anteriormente. A continuación, se detallan las librerías y herramientas que se han utilizado para el diseño y construcción de la Base de Datos NoSQL y la codificación de los diferentes módulos por cada perfil predeterminado.

### 2.3.4 Herramientas utilizadas en el sistema web

La siguiente **TABLA VI** especifica cada una de las herramientas, que se han utilizado en el desarrollo del proyecto y su compatibilidad con el patrón de arquitectura.

**TABLA VI:** Descripción de las herramientas que se han utilizado en el desarrollo.

HERRAMIENTAS	JUSTIFICACIÓN
<b>Angular</b>	La ejecución de este <i>Framework</i> en la codificación del sistema web permite, el manejo adecuado y organizado del código, al igual que una reutilización del mismo, teniendo en cuenta la separación clara del modelo arquitectónico MVC [20].
<b>Angular Material</b>	La ejecución de esta librería de estilos en la codificación del sistema web permite, la

	incorporación de un kit de componentes previamente desarrollados y que sea de fácil implementación en el proyecto [21].
<b>Firestore Cloud Functions</b>	La ejecución de <i>Cloud Functions</i> en la codificación del sistema web permite, guardar el código fuente en la nube de una manera particular llamada funciones. Las cuales se pueden ejecutar de forma automática en el lado del <i>backend</i> en respuesta a las solicitudes HTTP [22].
<b>Firestore</b>	La ejecución de la plataforma <i>Firestore</i> en la codificación del sistema web permite, usar un sistema de autenticación y el almacenamiento de los datos en una Base de Datos NoSQL donde se guarda la información, permitiendo una búsqueda rápida en tiempo real y otros servicios que provee esta plataforma en la nube [23].

## Librerías

La siguiente **TABLA VII** especifica cada una de las librerías, que se han utilizado en el desarrollo del proyecto y su compatibilidad con el patrón de arquitectura y herramientas de desarrollo.

**TABLA VII:** Librerías que se han utilizado en la codificación del proyecto.

LIBRERÍA	DESCRIPCIÓN
<b>"@angular/material": "^11.2.11",</b>	Esta librería permite personalizar componentes con patrones de interacción común [24].
<b>"firebase": "^8.4.3"</b>	Esta librería permite la configuración y gestión de la Base de Datos, también permite la autenticación utilizada para el sistema [25].
<b>"drawflow": "0.0.49"</b>	Esta librería permite crear flujos de datos de forma fácil y rápida [26].
<b>"tslib": "^2.0.0"</b>	Esta librería permite la ejecución para <i>TypeScript</i> que contienen funciones auxiliares [27].

<b>"@angular/compiler": "^11.2.12"</b>	Esta librería permite entender las plantillas y convertirlas a código que hace que el sistema se ejecute y renderice [28].
--	--



## 3 RESULTADOS Y DISCUSIÓN

A partir de esta sección se presentan cada uno de los resultados en base a los *Sprints* previamente definidos, conjuntamente con la configuración del ambiente de desarrollo, ejecución de pruebas y el despliegue en el *hosting* de *Firebase*.

### 3.1 *Sprint* 0. Configuración del ambiente de desarrollo

En el *Sprint* 0 se establece la configuración de las herramientas para dejar listo el ambiente de desarrollo previo a la etapa de codificación.

Actividades que se han realizado en el *Sprint* 0:

- Definición de los requerimientos funcionales y no funcionales del sistema web.
- Diseño de la Base de Datos NoSQL en función al proyecto.
- Descripción de usuarios.
- Estructura del proyecto.

#### 3.1.1 Definición de los requerimientos funcionales y no funcionales del sistema web

##### **Identificación del usuario administrador**

El usuario con perfil administrador puede entrar al sistema web llenando los campos requeridos los cuales son correo electrónico y contraseña, estas credenciales son proporcionados por el desarrollador del sistema. Además, tiene la posibilidad de editar su perfil.

##### **Administración de categorías, subcategorías y trámites**

El usuario con perfil administrador puede realizar el registro, visualización, modificación y eliminación de categorías, subcategorías y trámites según lo requiera.

##### **Administración de formularios de quejas, novedades o sugerencias**

El usuario con perfil administrador puede llevar a cabo la visualización de las diferentes notificaciones de los formularios de quejas, novedades o sugerencias, al igual que, la remisión respecto a los avisos de respuesta tales como aprobar o negar dichos formularios.

### **Administración de notificaciones de noticias**

El usuario con perfil administrador puede enviar a los usuarios ya registrados en el sistema web notificaciones de noticias respecto a los tramites académicos y administrativos para la ESFOT.

### **Registro de usuarios**

Para un adecuado registro de los usuarios estudiantes en el sistema web es realizar el ingreso de los campos requeridos en el formulario de registro.

### **Inicio de sesión**

Para un adecuado inicio de sesión de los usuarios con perfil estudiante en el sistema web, el ingreso de las credenciales es a través del correo electrónico y contraseña, previamente registrados en el módulo de registro.

### **Restablecimiento de contraseña**

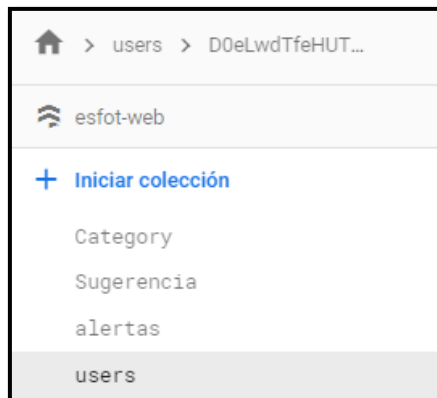
Los usuarios con perfil estudiante poseen la opción de restablecer su contraseña por medio del envío de un enlace al correo electrónico previamente otorgado en el proceso de registro.

### **Visualización de categorías**

El usuario con perfil estudiante tiene la opción de visualizar las categorías y subcategorías creadas por el administrador y cada uno de los trámites por cada subcategoría.

#### **3.1.2 Diseño de la Base de Datos NoSQL en función al proyecto**

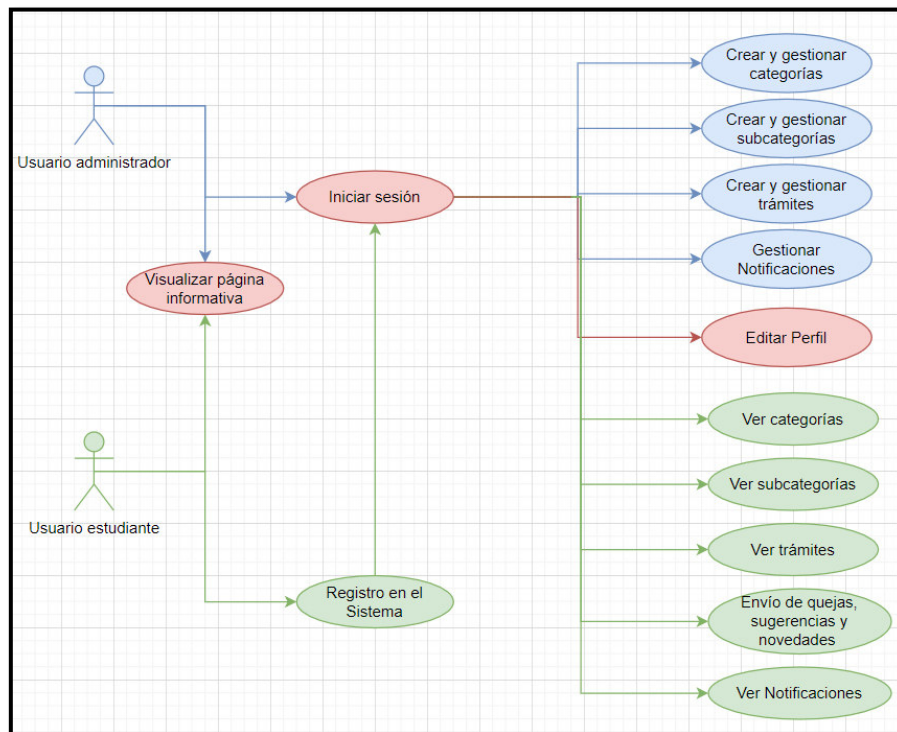
En base a los requisitos del proyecto, se ha diseñado una Base de Datos NoSQL, la cual en vez de tablas utiliza colecciones y documentos para el registro de la información como se ilustra en la **Fig. 3**, esta Base de Datos está gestionada por *Cloud Firebase*, el cual es parte de los servicios de la plataforma *Firebase*, en donde se gestiona la totalidad de la información referente al sistema web. Por último, el diseño completo de la Base de Datos NoSQL se puede observar en el Manual Técnico – Apartado Diseño de la Base de Datos (pág. 35).



**Fig. 3:** Colecciones de la Base de Datos.

### 3.1.3 Descripción de usuarios

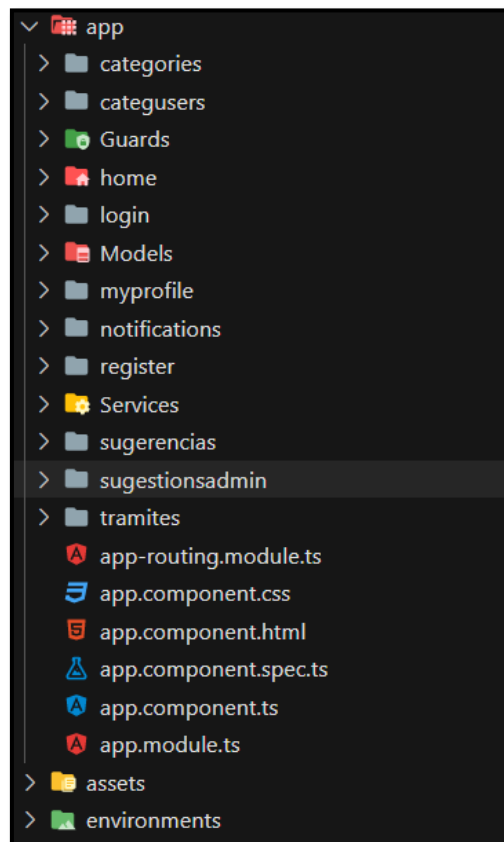
La **Fig. 4** muestra a todos los usuarios que intervienen en el desarrollo del sistema web, al igual que los módulos, en donde cada uno de los usuarios tienen accesos a los diferentes módulos dependiendo el rol que desempeñen, el color rojo muestra los módulos en común entre los usuarios, el color azul muestra los módulos a cargo del usuario administrador y el color verde muestra los módulos a cargo del usuario estudiante.



**Fig. 4:** Usuario administrador y usuario estudiante con sus respectivos módulos.

### 3.1.4 Estructura del Proyecto

Para tener un proyecto mejor organizado y escalable a medida que se requiera se ha implementado el patrón arquitectónico Modelo - Vista - Controlador (MVC), el cual permite mantener un orden claro y una excelente compatibilidad con el *Framework* Angular lo que permite una codificación mucho más rápida y que el mantenimiento y escalabilidad sea mucho más robusta y ordenada. Muestra de ello, la **Fig. 5** ilustra una serie de directorios al igual que archivos, los cuales permiten una separación de cada uno de los componentes dependiendo de su funcionalidad.



**Fig. 5:** Componentes de archivos y directorios para el sistema web.

## 3.2 *Sprint* 1. Implementación del usuario administrador y módulos asignados a su cargo

El *Sprint* 1 se establece cada uno de los módulos que tiene el usuario con perfil administrador.

Actividades que forman parte del *Sprint* 1:

- Visualizar página informativa.

- Ingreso del usuario administrador.
- Registrar, visualizar, modificar y eliminar categorías.
- Registrar, visualizar, modificar y eliminar subcategorías.
- Registrar, visualizar, modificar y eliminar trámites.
- Aprobación y rechazo de formularios de quejas, novedades o sugerencias.
- Enviar notificaciones de noticias a usuarios estudiantes.

### 3.2.1 Visualizar página informativa

Una vez subido al *hosting* de *firebase* el sistema web, todos los usuarios pueden visualizar la página principal la cual brinda opciones para el inicio de sesión y las principales categorías, subcategorías y trámites creados por el usuario administrador, como se muestra en la **Fig. 6**.



**Fig. 6:** Página Informativa del sistema web.

### 3.2.2 Ingreso del usuario administrador

La **Fig. 7** muestra los campos requeridos y las validaciones respectivas de los mismo para el ingreso al sistema web, cabe recalcar que las credenciales para el usuario administrador son otorgadas por la desarrolladora del sistema web. Posterior al ingreso de las credenciales el sistema verifica a que perfil pertenece el usuario para luego redireccionarlo a cada uno de los módulos asignado a su cargo. Por último, en el Manual Técnico se da a conocer el proceso de restablecimiento de contraseña.



Fig. 7: Inicio de sesión para los usuarios administrador y estudiante.

### 3.2.3 Registrar, visualizar, modificar y eliminar categorías

La Fig. 8 muestra el módulo para las categorías en el cual el usuario con perfil administrador tiene opción de registrar, visualizar, modificar y eliminar una categoría en base a sus necesidades, si el administrador desea crear una categoría debe ingresar la información en todos los campos necesarios y el sistema web verifica la información ingresada a través de una serie de validaciones para su posterior inserción en la Base de Datos. Por otra parte, en el Manual de Usuario se da a conocer el proceso, diseño de prototipos y sus validaciones para la visualización, modificación y eliminación de categorías.

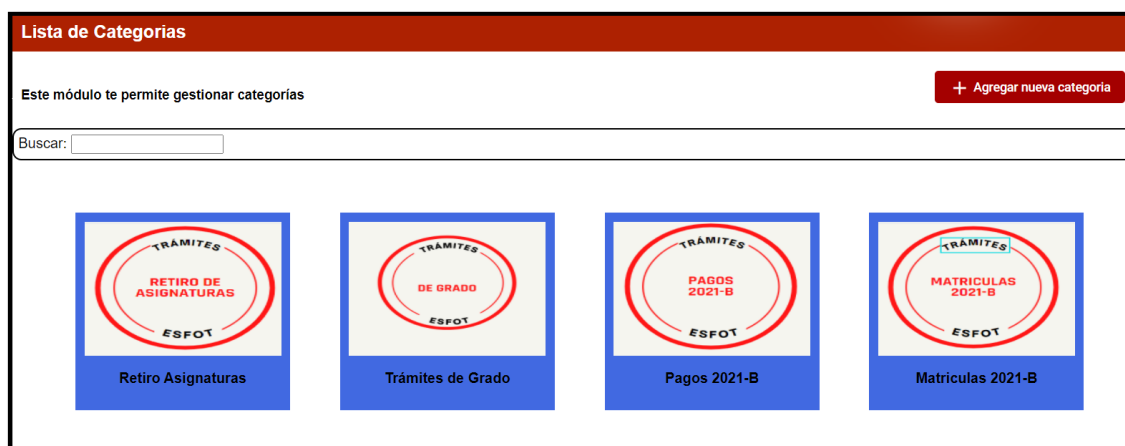
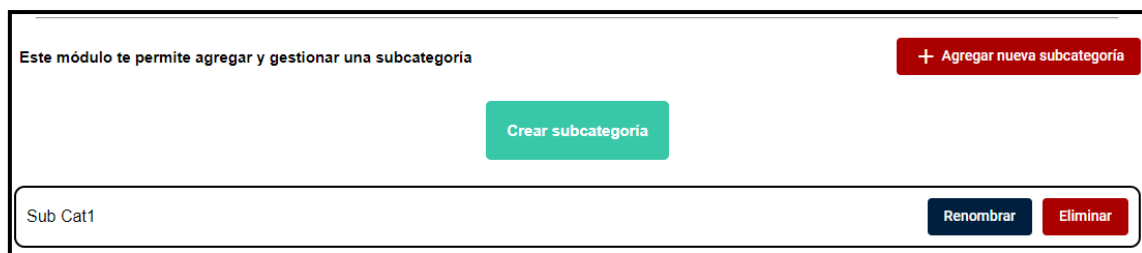


Fig. 8: Módulo para gestionar categorías.

### 3.2.4 Registrar, visualizar, modificar y eliminar subcategorías

La **Fig. 9** muestra el módulo para las subcategorías en el cual el usuario con perfil administrador tiene opción de registrar, visualizar, modificar y eliminar una subcategoría en base a sus necesidades, si el administrador desea crear una subcategoría debe ingresar la información en todos los campos necesarios al igual que una categoría previamente registrada y el sistema web verifica la información ingresada a través de una serie de validaciones para su posterior inserción en la Base de Datos. Por otra parte, en el Manual de Usuario se da a conocer el proceso, diseño de prototipos y sus validaciones para la visualización, modificación y eliminación de subcategorías.



**Fig. 9:** Módulo para gestionar subcategorías.

### 3.2.5 Registrar, visualizar, modificar y eliminar trámites

La **Fig. 10** muestra el módulo para los trámites en el cual el usuario con perfil administrador tiene opción de registrar, visualizar, modificar y eliminar un trámite en base a sus necesidades. Si el administrador desea crear un trámite debe ingresar la información en todos los campos necesarios al igual que una categoría y subcategoría previamente registrada. Posterior a ello, el sistema web verifica la información ingresada a través de una serie de validaciones para su posterior inserción en la Base de Datos. Por otra parte, en el Manual de Usuario se da a conocer el proceso, diseño de prototipos y validaciones para la visualización, modificación y eliminación de trámites.

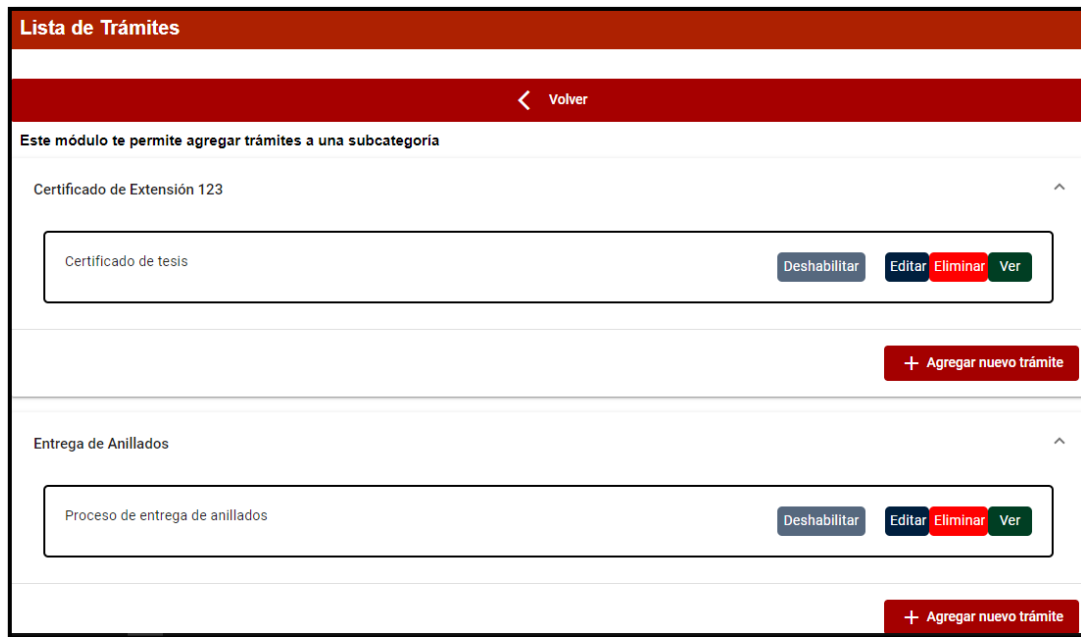


Fig. 10: Módulo para gestionar trámites.

### 3.2.6 Aprobación y rechazo de formularios de quejas, novedades o sugerencias

La Fig. 11 muestra el módulo para los formularios de quejas, novedades o sugerencias donde el usuario con perfil administrador tiene opción de aprobar o rechazar un formulario en base a sus necesidades. Si un formulario es aceptado automáticamente le llega una notificación al usuario con perfil estudiante de la aprobación del mismo, caso contrario si el formulario es rechazado en la notificación de respuesta se detalla del porque su formulario ha sido rechazado. Por otra parte, en el Manual de Usuario se da a conocer el proceso para el rechazo de un formulario.

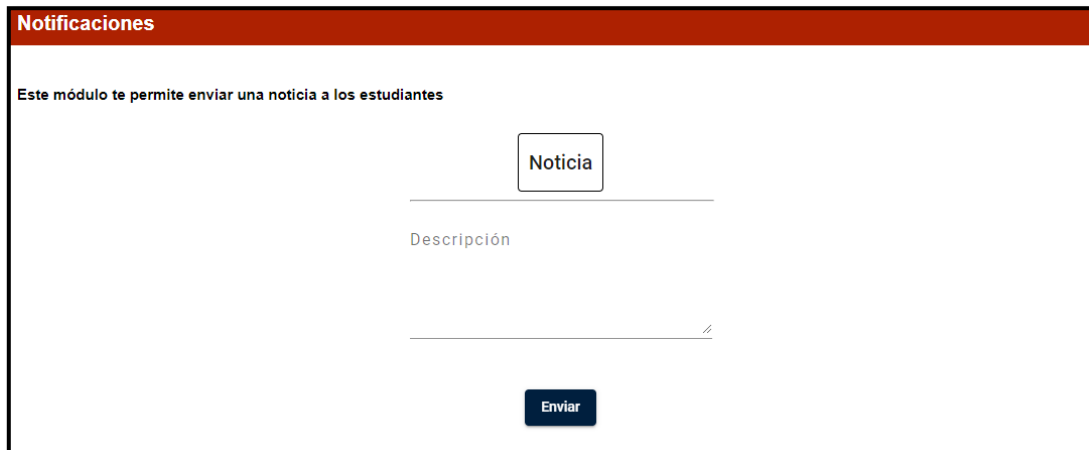


Fig. 11: Módulo para gestionar quejas, sugerencias y novedades.



### 3.2.7 Enviar notificaciones de noticias a usuarios estudiantes

La **Fig. 12** muestra el módulo de notificaciones en el cual el usuario con perfil administrador tiene opción de enviar notificaciones a los usuarios con perfil estudiante. Si el administrador desea crear y enviar una notificación debe ingresar la información en todos los campos necesarios y el sistema web verifica la información ingresada a través de una serie de validaciones para el envío respectivo.



**Fig. 12:** Módulo de envío de noticias.

## 3.3 *Sprint 2*. Implementación del usuario estudiante y módulos asignados a su cargo

El *Sprint 2* se establece cada uno de los módulos que tiene el usuario con perfil estudiante.

Actividades que forma parte el *Sprint 2*:

- Registro del usuario estudiante e inicio de sesión.
- Visualización y modificación de perfil.
- Visualización de categorías.
- Visualización de procedimientos y formularios de trámites.
- Llenar formularios de quejas, novedades o sugerencias.
- Recibir notificaciones de noticias.

### 3.3.1 Registro del usuario estudiante e inicio de sesión


La **Fig. 13** muestra los campos y las validaciones pertinentes para llevar a cabo el registro exitoso del usuario estudiante en el sistema web, una vez realizado el registro, puede realizar el inicio de sesión con el correo electrónico y contraseña.


**REGISTRO**

Nombres \_\_\_\_\_

Apellidos \_\_\_\_\_

Correo \_\_\_\_\_

Contraseña \* \_\_\_\_\_ 

Repetir Contraseña \* \_\_\_\_\_ 


**Registrar**

**Fig. 13:** Registro del usuario estudiante.

### 3.3.2 Visualización y modificación de perfil

Después de haber ya iniciado sesión en el sistema web la **Fig. 14** muestra el perfil del estudiante, donde se puede visualizar los datos personales y modificarlos si así lo desea. Por otra parte, en el Manual de Usuario se da a conocer el proceso para la modificación de su perfil.

**Actualizar Cuenta**



Nombres  
**Diana**

Apellidos  
**Narváez**

Correo  
**narvaezdiana4@gmail.com**

**Cambiar Foto**

**Actualizar**

**Fig. 14:** Módulo de perfil para el estudiante.

### 3.3.3 Visualización de categorías

La Fig. 15 muestra el módulo de categorías en donde el estudiante debe seleccionar una de ellas para obtener la lista de subcategorías y trámites existentes. Por otra parte, en el Manual de Usuario se da a conocer el proceso para la visualización a detalle de cada una de las categorías ya registradas.

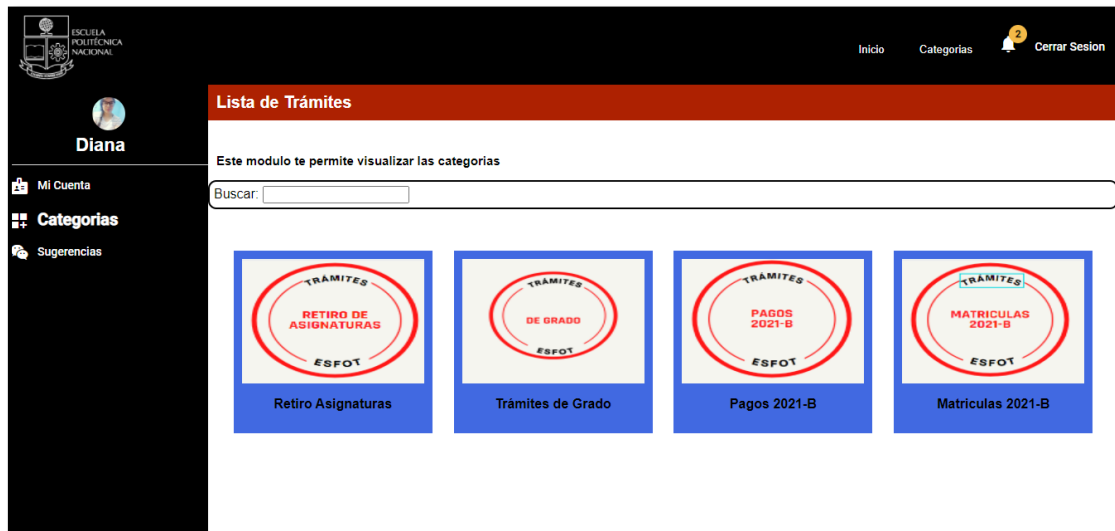


Fig. 15: Módulo para visualizar categorías.

### 3.3.4 Visualización de procedimientos y formularios de trámites

La Fig. 16 muestra el módulo de categorías y una lista de trámites donde el usuario estudiante selecciona la subcategoría para acceder a la visualización completa del trámite deseado, por cada trámite se despliega la información, el cual, permite visualizar el detalle completo de los pasos a seguir y sus respectivos formularios.

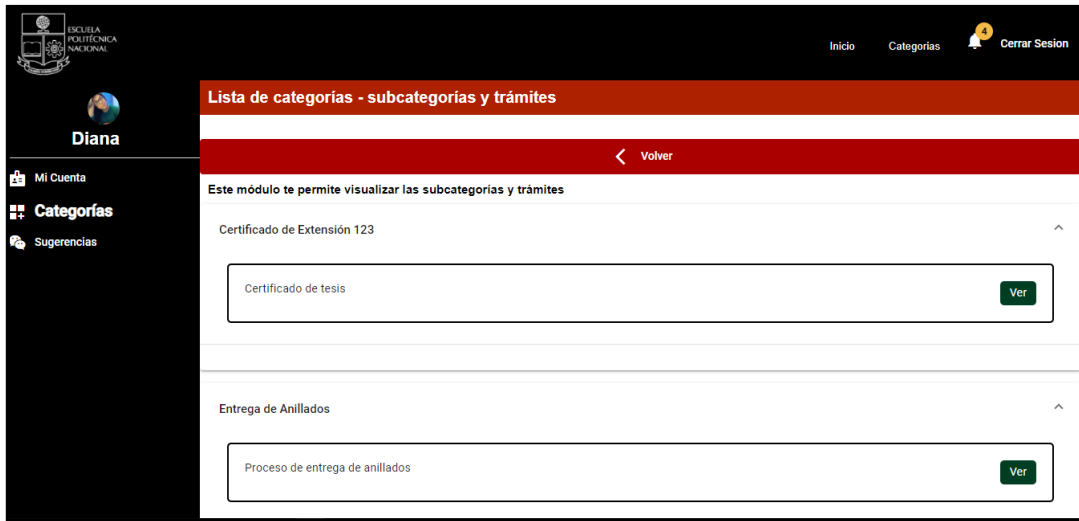


Fig. 16: Módulo categorías - lista de trámites.

### 3.3.5 Llenar formularios de quejas, novedades o sugerencias

La Fig. 17 muestra el módulo de sugerencias para el envío de quejas, novedades y/o sugerencias por parte del usuario estudiante, el cual debe llenar los campos solicitados y enviar el mismo para que sea procesado por el usuario administrador.

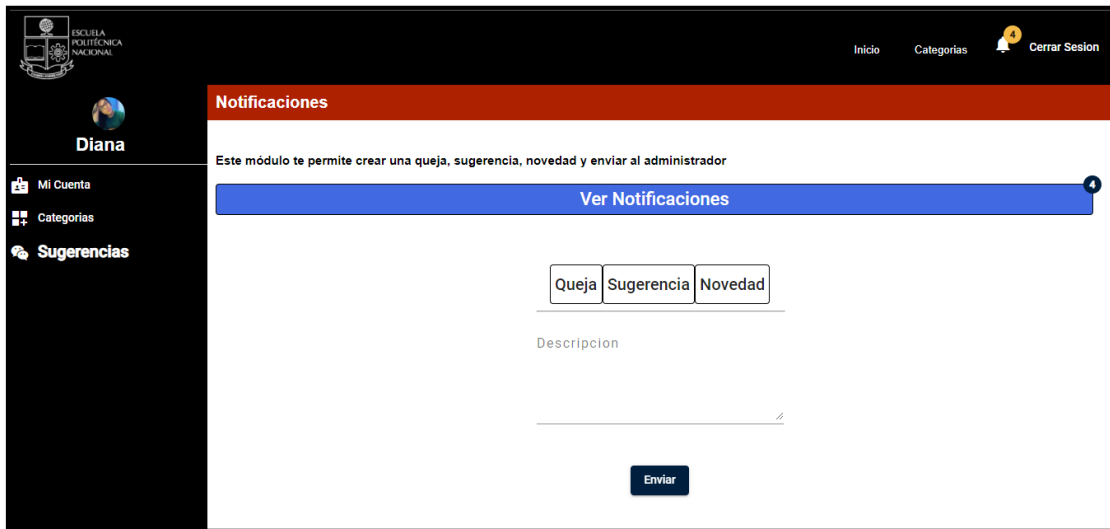


Fig. 17: Módulo de sugerencias.

### 3.3.6 Recibir notificaciones de noticias

La Fig. 18 muestra el módulo sugerencias, sección notificaciones donde el usuario estudiante puede visualizar las repuestas a sus quejas, novedades y/o sugerencias en el listado de notificaciones, al igual puede visualizar las noticias que son enviadas por

parte del administrador referente a modificaciones y nuevos procesos de trámites que hayan sido creados.

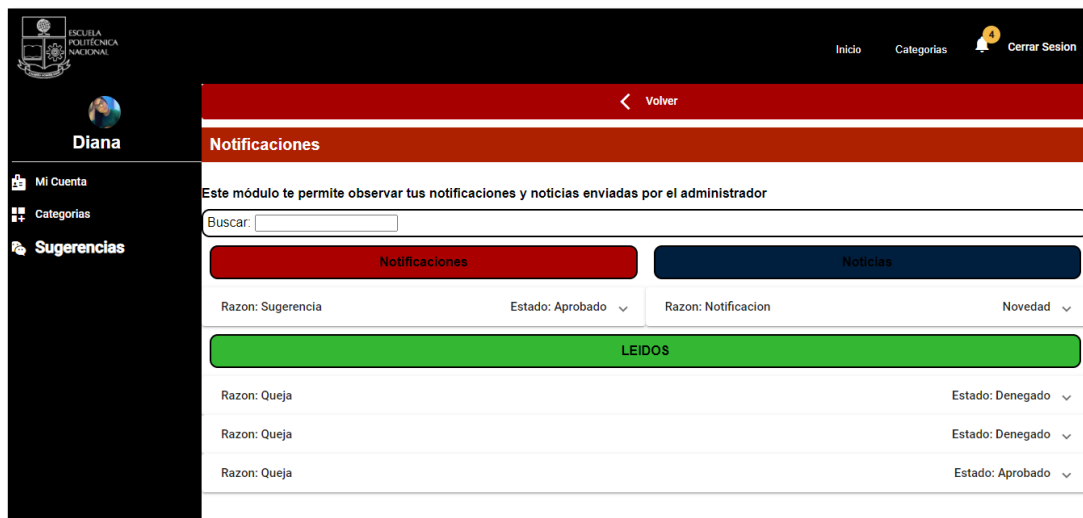


Fig. 18: Módulo de sugerencias - sección notificaciones.

### 3.4 *Sprint 3*. Pruebas del sistema web

El *Sprint 3* establece cada una de las pruebas que se han realizado al sistema web una vez que se ha finalizado la etapa de codificación.

Actividades que forma parte el *Sprint 3*:

- Ejecución de pruebas unitarias y resultados.
- Ejecución de pruebas de rendimiento y resultados.
- Ejecución de pruebas de compatibilidad y resultados.
- Ejecución de pruebas de aceptación y resultados.

#### 3.4.1 Ejecución de pruebas unitarias y resultados

Las pruebas unitarias, permiten descomponer funcionalidades del programa en pequeñas unidades individuales que sean fáciles de probar y detectar errores. Estas pruebas son un proceso más de lo que se realiza en una metodología ágil de trabajo [29].

Por otra parte, estas pruebas unitarias han sido realizadas en todos los componentes principales del sistema web, con el objetivo de verificar la funcionalidad de cada módulo

por separado, facilitar la pronta detección de errores, verificar que el código sea de calidad e impedir posibles fallos en posteriores implementaciones.

Para la ejecución de las pruebas unitarias se utilizan las herramientas *Jasmine* y *Karma* las cuales están incluidas en el *Framework* Angular, de esta forma permite programar y ejecutar a la vez cada una de las pruebas. La ventaja principal de estas herramientas es que brindan un reporte visual, en el cual se observa las ejecuciones exitosas, las que no se han ejecutado y las que se deben corregir. Finalmente, el comando para la ejecución de estas pruebas es *ng test* el cual permite la ejecución total de las pruebas existentes en cada uno de los componentes principales del sistema web.

La **Fig. 19** muestra la correcta estructura para realizar una prueba unitaria con la validación respectiva. El total de las pruebas unitarias se encuentra detallado en el Manual de Técnico – Sección Pruebas unitarias (pág. 40 - 43).

```
it('Debería verificar que los campos del correo electrónico y contraseña no esten vacios', () => {  
  
  const fixture= TestBed.createComponent(AuthServiceService);  
  const component2= fixture.componentInstance;  
  const cp= {mail: "narvaezdiana4@gmail.com", pass: '123456789'} || {};  
  const verificacion = component2.loginMailUser(cp.mail, cp.pass);  
  
  expect(verificacion['required']).toBeFalsy();  
  expect(verificacion ['mail']).toBeFalsy();  
  expect(verificacion ['pass']).toBeFalsy();  
  
});
```

**Fig. 19:** Estructura de la prueba unitaria para validar el campo email y contraseña del usuario estudiante.

Tras la ejecución de estas pruebas y en base a los resultados alcanzados, se determina que el sistema web actúa de manera eficiente a todas las validaciones que realiza, como se muestra en la **Fig. 20** .

```
LoginComponent  
• Debería crear el componente  
• Debería verificar que los campos del correo electrónico y contraseña no esten vacios
```

**Fig. 20:** Resultado de las pruebas unitarias ejecutadas en el componente *Login*.

### 3.4.2 Ejecución de pruebas de rendimiento y resultados

Las siguientes pruebas son las de rendimiento que vienen a ser una técnica de pruebas de *software* no funcional, las mismas que determinan la estabilidad, velocidad,

escalabilidad y la capacidad de repuesta de un sistema *software* bajo una determinada carga de trabajo [30].

La **Fig. 21** muestra la ejecución de una prueba de rendimiento que se ha realizado en la herramienta *JMeter*. El total de las pruebas de rendimiento se encuentra detallado en el Manual de Técnico – Sección Pruebas de rendimiento (pág. 44 - 45).

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	13:00:29.207	Primera Prueba 1-2	HTTP Request	186	✓	1365	122	186	14
2	13:00:29.217	Primera Prueba 1-3	HTTP Request	201	✓	1365	122	201	15
3	13:00:29.236	Primera Prueba 1-5	HTTP Request	182	✓	1365	122	182	13
4	13:00:29.226	Primera Prueba 1-4	HTTP Request	212	✓	1365	122	212	16
5	13:00:29.247	Primera Prueba 1-6	HTTP Request	191	✓	1365	122	191	14
6	13:00:29.257	Primera Prueba 1-7	HTTP Request	184	✓	1365	122	184	14
7	13:00:29.197	Primera Prueba 1-1	HTTP Request	244	✓	1365	122	244	20
8	13:00:29.266	Primera Prueba 1-8	HTTP Request	186	✓	1365	122	186	14
9	13:00:29.288	Primera Prueba 1-10	HTTP Request	189	✓	1365	122	189	14
10	13:00:29.278	Primera Prueba 1-9	HTTP Request	201	✓	1365	122	201	16
11	13:00:29.298	Primera Prueba 1-11	HTTP Request	191	✓	1365	122	191	14
12	13:00:29.307	Primera Prueba 1-12	HTTP Request	192	✓	1365	122	192	15
13	13:00:29.317	Primera Prueba 1-13	HTTP Request	201	✓	1365	122	201	14
14	13:00:29.327	Primera Prueba 1-14	HTTP Request	208	✓	1365	122	208	13
15	13:00:29.338	Primera Prueba 1-15	HTTP Request	207	✓	1365	122	207	14
16	13:00:29.346	Primera Prueba 1-16	HTTP Request	202	✓	1365	122	202	15
17	13:00:29.367	Primera Prueba 1-18	HTTP Request	194	✓	1365	122	194	15
18	13:00:29.357	Primera Prueba 1-17	HTTP Request	219	✓	1365	122	219	17

**Fig. 21:** Prueba de rendimiento de 1000 visitas en 10s.

Tras la ejecución de estas pruebas y en base a los resultados que se han obtenido, se determina que el sistema web actúa de manera eficiente en base a todas las peticiones de los usuarios con respuesta inmediatas.

### 3.4.3 Ejecución de pruebas de compatibilidad y resultados

Para que un sistema web sea compatible con todos los navegadores actuales, se lleva a cabo el uso de pruebas de compatibilidad, las cuales permiten verificar que el funcionamiento de módulos, perfiles e interfaces sea el mismo en cada uno de los mismos sin afectar el rendimiento y la experiencia de usuario [31] [32].

Estas pruebas de compatibilidad han sido realizadas en tres diferentes navegadores los cuales son Google Chrome, Mozilla Firefox y Opera como se muestra en la **TABLA VIII**. Mientras que total de las pruebas de compatibilidad se encuentra detalladas en el Manual de Técnico – Sección Pruebas de Compatibilidad (pág. 46 - 50).

**TABLA VIII:** Navegadores utilizados en las pruebas de compatibilidad.

NOMBRE	VERSIÓN	OBSERVACIÓN
Google Chrome	95.0.4638.69	Completamente funcional
Mozilla Firefox	94.0.1	Completamente funcional
Opera	81.0.4196.31	Completamente funcional

Tras la ejecución de estas pruebas de compatibilidad y en base a los resultados alcanzados, se determina que el sistema web tiene un correcto funcionamiento en los diferentes navegadores.

### 3.4.4 Ejecución de pruebas de aceptación y resultados

El objetivo principal de las pruebas de aceptación es comprobar si el *software* cumple con el funcionamiento esperado y permitir a los usuarios que intervienen determinar su aceptación desde el punto de vista de su correcto funcionamiento y rendimiento; esto se lleva a cabo con la finalidad de decidir si el sistema web es apto para salir a producción [33].

La **TABLA IX** muestra la primera prueba de aceptación que se ha ejecutado. Mientras que el total de las pruebas de aceptación se encuentra detalladas en el Manual de Técnico – Sección Pruebas de Aceptación (pág. 50 - 68).

**TABLA IX:** Prueba de aceptación 1 - Visualizar página informativa

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA001	<b>Identificador historia de Usuario:</b> HU001
<b>Nombre:</b> Visualizar página informativa.	
<b>Descripción:</b> El usuario administrador puede visualizar la página informativa del sistema.	
<b>Pasos de ejecución:</b> <ul style="list-style-type: none"> <li>• Ir o digitar la URL (<a href="https://esfot-web.web.app">https://esfot-web.web.app</a>) del sistema web en el navegador.</li> </ul>	
<b>Resultado deseado:</b> El sistema web permite visualizar la página informativa al usuario administrador.	
<b>Evaluación de la prueba:</b> Se comprueba el resultado esperado. Aprobación del cliente 100%.	

Tras la ejecución de estas pruebas y en base a los resultados que se han obtenido, se determina que el usuario administrador como el usuario estudiante están conformes que el sistema web actúa cumpliendo todas las funcionalidades definidas al inicio del proyecto.



### 3.5 *Sprint* 4. Despliegue del sistema web

El *Sprint* 4 establece la puesta en producción del sistema web en un *hosting* gratuito.

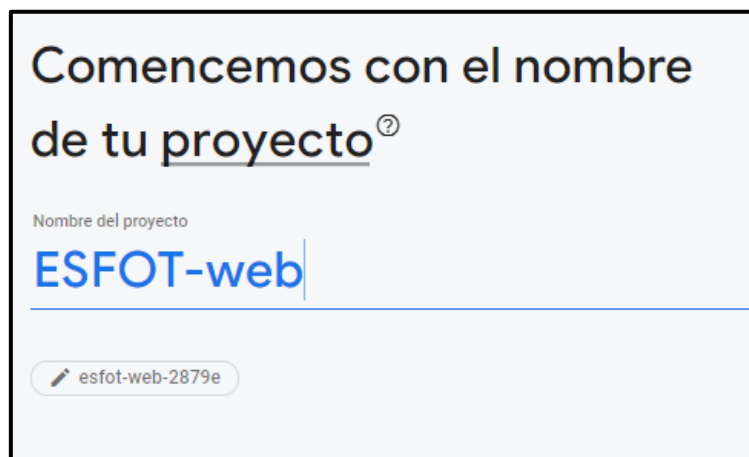
Actividades que forma parte el *Sprint* 4:

- Despliegue a producción en el *hosting* de *Firebase*.

#### 3.5.1 Despliegue a producción en el *hosting* de *Firebase*

Por último, para que el sistema web entre a producción se ha hecho uso de *Firebase Hosting*, el cual brinda un servicio de alojamiento de contenido web de manera segura, instantánea, simple y gratuita de llevar a cabo [34]. Además, cuenta con un respaldo de almacenamiento para la base de datos y otras características que han beneficiado el desarrollo del sistema web.

La **Fig. 22** muestra la creación del proyecto en *Firebase*, en tanto que la **Fig. 23** ilustra el despliegue del sistema web bajo un nombre de dominio que es <https://esfot-web.web.app>. No obstante, esta etapa tiene un procedimiento de configuración el mismo que se encuentra detallado en el Manual de Instalación – Apartado Despliegue del sistema web.



**Fig. 22:** Creación del proyecto en *Firebase*.



**Fig. 23:** Sistema web en producción.

## 4 CONCLUSIONES Y RECOMENDACIONES

A continuación, se expone una serie de conclusiones y recomendaciones que se ha obtenido en el transcurso del desarrollo del proyecto.

### 4.1 Conclusiones

- Como resultado final se ha obtenido el cumplimiento del alcance y objetivos establecidos en el proyecto. El sistema web permite administrar de una manera eficiente la información sobre los trámites requeridos en la ESFOT, a la vez el mantener informados a los estudiantes en tiempo real sobre cada uno de los requisitos por medio de la tecnología.
- El emplear una metodología como *Scrum* ha facilitado el trabajo en la identificación de las diferentes fases que actúan en el desarrollo del sistema web, en consecuencia, al ser una metodología ágil ha permitido generar entregables progresivamente, facilitando de esta manera que el cliente se encuentre siempre informado acerca de los avances y los entregables. Por último, los avances han permitieron detectar errores y realizar cambios de manera temprana.
- El utilizar la plataforma *Firebase* como Sistema Gestor de Base de Datos, ha permitido obtener una serie de ventajas una de ellas es ofrecer su propio servicio de *hosting* el cual permite un despliegue rápido y sencillo donde se aloja el sistema web, otra ventaja que cabe recalcar es la facilidad con la que permite actualizar la información en tiempo real, obteniendo como resultado final una experiencia agradable al usuario final.
- El realizar las diferentes pruebas al sistema web ha permitido observar resultados positivos, con lo que se comprueba el funcionamiento correcto de los diferentes componentes que intervienen.
- Al aplicar el MVC en el desarrollo del sistema web ha permitido que el proceso se vuelva manejable y más fácil de administrar ya que se ha logrado realizar cambios a cualquiera de sus componentes de una manera mucho más organizada.

## 4.2 Recomendaciones

- Para el correcto desarrollo de cualquier sistema web es recomendable tener claro la utilización de un patrón arquitectónico MVC ya que este facilita la integración de sus componentes de una forma sencilla y práctica, también ayuda a una detección temprana de errores evitando pérdida de tiempo.
- Se recomienda que la ESFOT, implemente nuevas políticas y normas de privacidad de la información para los estudiantes en cuestión.
- Para tener una mejor experiencia con el sistema web se recomienda hacerlo desde un computador de mesa o laptop ya que la interacción es más intuitiva.
- Es recomendable realizar un *backup* de la base de datos, para evitar la pérdida de información por cualquier contingencia que se presente.

## 5 REFERENCIAS

- [1] «ATENEA,» EPN, 2002-2008. [En línea]. Available: <https://atenea.epn.edu.ec/>. [Último acceso: 30 agosto 2021].
- [2] I. Invest, «IDB,» American development, 2021. [En línea]. Available: <https://publications.iadb.org/publications/spanish/document/La-educacion-superior-en-tiempos-de-COVID-19-Aportes-de-la-Segunda-Reunion-del-Di%C3%A1logo-Virtual-con-Rectores-de-Universidades-Lideres-de-America-Latina.pdf>. [Último acceso: 30 Agosto 2021].
- [3] Samuel, «Cloud Hispano,» 23 Agosto 2017. [En línea]. Available: <https://www.cloudhispano.com/ventajas-y-desventajas-del-software-a-medida/>. [Último acceso: 30 Agosto 2021].
- [4] «Scielo,» Revista chile de Ingenieria, 2020. [En línea]. Available: [https://scielo.conicyt.cl/scielo.php?pid=S0718-33052017000400599&script=sci\\_arttext&tlng=n](https://scielo.conicyt.cl/scielo.php?pid=S0718-33052017000400599&script=sci_arttext&tlng=n). [Último acceso: 30 Agosto 2021].
- [5] «La Mina Digital,» La Mina Digital, 2020. [En línea]. Available: <https://www.laminadigital.es/ventajas-de-las-aplicaciones-web-para-empresas/>. [Último acceso: 30 Agosto 2021].
- [6] «PMO,» 3 Septiembre 2020. [En línea]. Available: <http://www.pmoinformatica.com/2012/09/ambientes-de-desarrollo-de-software.html>. [Último acceso: 30 Agosto 2021].
- [7] «leBS,» 2021. [En línea]. Available: <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>. [Último acceso: 30 Agosto 2021].
- [8] «RED HAT,» 2021. [En línea]. Available: <https://www.redhat.com/es/devops/what-is-agile-methodology>. [Último acceso: 30 Agosto 2021].
- [9] «ATLASSIAN,» 2020. [En línea]. Available: <https://www.atlassian.com/es/agile/scrum>. [Último acceso: 30 Agosto 2021].

- [10] E. Abellan, «Wam,» 05 Marzo 2020. [En línea]. Available: [https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html#:~:text=La%20metodolog%C3%ADa%20Scrum%20es%20un,equ ipos%20que%20manejan%20proyectos%20complejos.&text=Esto%20permite% 20al%20cliente%2C%20junto,obtener%20ventas%20\(Sales%2.](https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html#:~:text=La%20metodolog%C3%ADa%20Scrum%20es%20un,equ  ipos%20que%20manejan%20proyectos%20complejos.&text=Esto%20permite%  20al%20cliente%2C%20junto,obtener%20ventas%20(Sales%2.) [Último acceso: 30 Agosto 2021].
- [11] «SOFTENG,» 2021. [En línea]. Available: <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html>. [Último acceso: 30 Agosto 2021].
- [12] «Deloitte,» 2021. [En línea]. Available: <https://www2.deloitte.com/es/es/pages/technology/articles/roles-y-responsabilidades-scrum.html>. [Último acceso: 30 Agosto 2021].
- [13] E. Deloitte, «Deloitte,» 15 Agosto 2020. [En línea]. Available: <https://www2.deloitte.com/es/es/pages/technology/articles/artefactos-scrum.html>. [Último acceso: 30 Agosto 2021].
- [14] «tenstep,» 2016. [En línea]. Available: <https://www.tenstep.ec/portal/articulos-boletin-tenstep/45-2-integrar-el-plan/261-siete-tecnicas-para-recopilar-requerimientos>. [Último acceso: 30 Agosto 2021].
- [15] Diego, «ScrumMX,» 02 Agosto 2018. [En línea]. Available: <https://scrum.mx/informate/historias-de-usuario>. [Último acceso: 30 Agosto 2021].
- [16] «ViewNext,» 2021. [En línea]. Available: <https://www.viewnext.com/artefactos-scrum/>. [Último acceso: 30 Agosto 2021].
- [17] «integrait,» 2021. [En línea]. Available: <https://integrait.com.mx/blog/sprint-y-sprint-backlog/>. [Último acceso: 31 Agosto 2021].
- [18] «GetApp,» NinjaMock, [En línea]. Available: <https://www.getapp.es/software/105959/ninjamock>. [Último acceso: 31 Agosto 2021].

- [19] desarrolloweb, «desarrolloweb.com,» 28 Julio 2020. [En línea]. Available: <https://desarrolloweb.com/articulos/que-es-mvc.html>. [Último acceso: 17 septiembre 2021].
- [20] Permalink, «Quality devs,» Quality devs, 19 Septiembre 2019. [En línea]. Available: <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>. [Último acceso: 29 Octubre 2021].
- [21] «Material Angular,» 2021. [En línea]. Available: <https://material.angular.io/guide/getting-started>. [Último acceso: 29 Octubre 2021].
- [22] «Firebase,» Firebase Documentation, 2021. [En línea]. Available: <https://firebase.google.com/docs/functions?hl=es-419#:~:text=Cloud%20Functions%20para%20Firebase%20es,ejecuta%20en%20un%20entorno%20administrado..> [Último acceso: 29 Octubre 2021].
- [23] «Firebase Documentación,» Firebase, 2021. [En línea]. Available: <https://firebase.google.com/docs?hl=es>. [Último acceso: 29 Octubre 2021].
- [24] «Angular Material,» 2021. [En línea]. Available: <https://material.angular.io/>. [Último acceso: 04 Noviembre 2021].
- [25] Firebase, «Firebase,» Firebase, 2021. [En línea]. Available: <https://firebase.google.com/docs?hl=es>. [Último acceso: 04 Noviembre 2021].
- [26] «npm,» Jerosoler, 25 Octubre 2021. [En línea]. Available: <https://www.npmjs.com/package/drawflow>. [Último acceso: 04 Noviembre 2021].
- [27] Sandersn, «Git,» 2021. [En línea]. Available: <https://github.com/microsoft/tslib>. [Último acceso: 04 Noviembre 2021].
- [28] «Angular,» 2021. [En línea]. Available: <https://docs.angular.lat/guide/npm-packages>. [Último acceso: 04 Noviembre 2021].
- [29] yeePLY, «YeePLY,» 18 Mayo 2021. [En línea]. Available: <https://www.yeePLY.com/blog/que-son-pruebas-unitarias/>. [Último acceso: 17 Septiembre 2021].

- [30] «MicroFocus,» 2021. [En línea]. Available: <https://www.microfocus.com/es-es/what-is/performance-testing>. [Último acceso: 17 Septiembre 2021].
- [31] «Compatibly Testing,» 2021. [En línea]. Available: <https://es.myservername.com/what-is-software-compatibility-testing>. [Último acceso: 18 Septiembre 2021].
- [32] admin, «Globe,» 2021. [En línea]. Available: <https://www.globetesting.com/2012/07/pruebas-de-compatibilidad/>. [Último acceso: 18 Septiembre 2021].
- [33] M. Cillero, «Circustancia Digital,» manuel.cillero.es, 25 Enero 2021. [En línea]. Available: <https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/pruebas/acceptacion/>. [Último acceso: 12 Noviembre 2021].
- [34] Firebase, «Firebase,» 2021. [En línea]. Available: [https://firebase.google.com/docs/hosting?hl=es-419#:~:text=Firebase%20Hosting%20es%20un%20servicio,distribuci%C3%B3n%20de%20contenidos\)%20global%20r%C3%A1pidamente..](https://firebase.google.com/docs/hosting?hl=es-419#:~:text=Firebase%20Hosting%20es%20un%20servicio,distribuci%C3%B3n%20de%20contenidos)%20global%20r%C3%A1pidamente..) [Último acceso: 17 septiembre 2021].



## 6 ANEXOS

### 6.1 Manual Técnico

- Índice de Manual Técnico
- Recopilación de requerimientos
- Historias de Usuario
- *Product Backlog*
- *Sprint Backlog*
- Diseño de interfaces
- Diseño de la Base de Datos
- Pruebas unitarias
- Pruebas de rendimiento
- Pruebas de compatibilidad
- Pruebas de aceptación

### 6.2 Manual de Usuario

<https://youtu.be/WDcbzwYYWjE>

### 6.3 Manual de Instalación

- Despliegue del sistema web en el *hosting* de *Firebase*