



**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas**

**Proyecto “PDF SOLUTIONS”**

*Curso: Calidad y Pruebas de Software*

Docente: Ing. Patrick Jose Cuadros Quiroga

Integrantes:

- ***Mario Antonio Flores Ramos (2018000597)***
- ***Erick Javier Salinas Condori (2020069046)***
- ***Fiorela Milady Ticahuanca Cutipa (2020068765)***

**Tacna – Perú**  
**2024**

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	10/10/2020	Versión Original

**Sistema *PDF solutions***  
**Documento de Arquitectura de Software**

**Versión *1.0***

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	10/10/2020	Versión Original

## ÍNDICE GENERAL

### Contenido

<b>2. OBJETIVOS Y RESTRICCIONES ARQUITECTÓNICAS</b>	<b>8</b>
1.1.1. Requerimientos Funcionales	8
1.1.2. Requerimientos No Funcionales – Atributos de Calidad	9
<b>3. REPRESENTACIÓN DE LA ARQUITECTURA DEL SISTEMA</b>	<b>10</b>
1.1.3. Diagramas de Casos de uso	10
3.2.1. Diagrama de Subsistemas (paquetes)	11
3.2.2. Diagrama de Secuencia (vista de diseño)	12
3.2.3. Diagrama de Objetos	17
3.2.4. Diagrama de Clases	17
3.2.5. Diagrama de Base de datos (relacional o no relacional)	18
3.3.1. Diagrama de arquitectura software (paquetes)	18
3.3.2. Diagrama de arquitectura del sistema (Diagrama de componentes)	19
<b>4. ATRIBUTOS DE CALIDAD DEL SOFTWARE</b>	<b>19</b>

## 1. INTRODUCCIÓN

### 1.1. Propósito (Diagrama 4+1)

Diagrama de casos de uso:

Este diagrama representa la interacción del actor con las distintas funciones del sistema (Iniciar sesión, Cerrar sesión, Registrar nuevo usuario, Subir un archivo PDF, Fusionar PDFs, Cortar PDFs, Visualizar planes de suscripción, Actualizar suscripción)

Diagrama de secuencia:

Este diagrama muestra la secuencia de interacciones entre los diferentes objetos y componentes del sistema durante la ejecución de un caso de uso.

Diagrama de componentes:

Este diagrama muestra los componentes principales del sistema y las dependencias entre ellos.

Diagrama de despliegue:

Este diagrama representa la distribución física de los componentes del sistema en los diferentes nodos de la red.

+1 Escenario:

Este escenario o vista adicional describe un caso de uso o funcionalidad específica del sistema de manera más detallada, utilizando otros diagramas o descripciones complementarias. Puede ser un diagrama de actividad, un diagrama de clases detallado, una descripción textual, entre otros.

### 1.2. Alcance

Este documento SAD tiene como objetivo proporcionar una descripción detallada de la arquitectura y el diseño del sistema PDF Solutions, centrado en la gestión eficiente de archivos PDF. El sistema será mejorado implementando nuevas funcionalidades y optimizaciones en diversas áreas, con un enfoque en la mejora de la experiencia del usuario y la gestión de documentos.

El alcance de este documento abarca los siguientes aspectos clave:

**Mejoramiento de la gestión de archivos PDF:**

Implementación de funcionalidades que permiten juntar y cortar archivos PDF de manera eficiente y sencilla.

Optimización de las operaciones sobre documentos para proporcionar un manejo más rápido y preciso de los archivos PDF.

#### **Sistema de autenticación y suscripciones:**

Desarrollo e implementación de un sistema de login que permite gestionar usuarios y suscriptores.

Gestión de suscripciones con diferentes niveles de acceso a funcionalidades avanzadas, adaptándose a las necesidades de los usuarios.

#### **Interfaz de usuario optimizada:**

Creación de una interfaz amigable e intuitiva para facilitar la interacción con la aplicación y mejorar la experiencia del usuario.

#### **Administración de suscripciones:**

Descripción de la administración de accesos y permisos de los usuarios según su tipo de suscripción, permitiendo una gestión eficaz de los recursos del sistema.

#### **Seguridad y privacidad:**

Implementación de un sistema seguro de autenticación para garantizar la protección de la información y la privacidad de los datos del usuario.

#### **Vista lógica:**

Descripción de la arquitectura lógica del sistema, incluyendo la gestión de usuarios, manipulación de archivos PDF, administración de suscripciones y control de acceso.

#### **Vista de desarrollo:**

Implementación del patrón MVC con tecnologías como C# y ASP.NET Core, estructuración organizada del código y la adopción de buenas prácticas de programación.

#### **Vista física:**

Despliegue de la aplicación en un entorno web, asegurando la compatibilidad y el rendimiento en los servidores y bases de datos utilizados.

Este documento también aborda aspectos relacionados con la seguridad, la privacidad de los datos y la eficiencia operativa, asegurando que el sistema sea escalable, confiable y fácil de usar.

### **1.3. Definición, siglas y abreviaturas**

- UI (Interfaz de Usuario): Diseño y disposición de los elementos visuales con los cuales los usuarios interactúan en la aplicación.
- UX (Experiencia de Usuario): Conjunto de percepciones y respuestas de un usuario resultantes del uso de la aplicación.
- CSV (Comma-Separated Values): Formato de archivo utilizado para almacenar datos tabulares, donde cada valor está separado por una coma.
- PDF (Portable Document Format): Formato de archivo utilizado para representar documentos de manera independiente del software, hardware o sistema operativo.

## 1.4. Organización del documento

El presente documento se organiza de la siguiente manera:

### **Capítulo 1: Introducción**

En este capítulo se presenta una introducción general al proyecto “PDF SOLUTIONS”, sus objetivos, el contexto y la justificación del trabajo. Se abordan los problemas que resuelve el sistema, la hipótesis planteada, y la importancia del proyecto en el contexto actual de la gestión y procesamiento de documentos PDF. Además, se establece la estructura general del documento.

### **Capítulo 2: Marco Teórico**

En este capítulo se realiza una revisión detallada de los conceptos relacionados con la gestión de documentos electrónicos, el formato PDF, las tecnologías utilizadas para su manipulación y las soluciones actuales en el mercado. Se incluyen antecedentes de investigaciones previas y una discusión sobre las herramientas y tecnologías empleadas en “PDF SOLUTIONS”.

### **Capítulo 3: Metodología**

En este capítulo se describe el enfoque metodológico del proyecto, detallando las fases de desarrollo del sistema. Se explican las herramientas, tecnologías y plataformas utilizadas para implementar “PDF SOLUTIONS”, así como el proceso de diseño y desarrollo. Además, se abordan las técnicas de recolección y análisis de datos, si aplica.

### **Capítulo 4: Desarrollo del Sistema**

Este capítulo describe en detalle la implementación del sistema “PDF SOLUTIONS”, abarcando la arquitectura del sistema, las funcionalidades clave y las tecnologías utilizadas. Se incluyen diagramas de flujo, ejemplos de código y una explicación paso a paso de las principales características del sistema, como la conversión, edición y manipulación de archivos PDF.

### **Capítulo 5: Resultados y Evaluación**

En este capítulo se presentan los resultados obtenidos durante las pruebas y la evaluación del sistema. Se incluyen métricas de desempeño, análisis de usabilidad y rendimiento del sistema, y la validación de las funcionalidades implementadas. Además, se comparan los resultados con los objetivos planteados al inicio del proyecto.

### **Capítulo 6: Conclusiones y Recomendaciones**

Este capítulo resume las conclusiones del proyecto, destacando los logros alcanzados y las implicaciones de “PDF SOLUTIONS”. Se discuten las limitaciones encontradas durante el desarrollo y se proponen recomendaciones para futuras mejoras o investigaciones adicionales en el área de gestión de documentos PDF.

### Referencias Bibliográficas

En este apartado se incluyen todas las fuentes consultadas para la realización del proyecto, como libros, artículos científicos, y documentación técnica, siguiendo las normas de citación correspondientes.

## 2. OBJETIVOS Y RESTRICCIONES ARQUITECTÓNICAS

[Establezca las prioridades de los requerimientos y las restricciones del proyecto]

### 2.1. Priorización de requerimientos

#### 1.1.1. Requerimientos Funcionales

Tabla 01: Requerimientos Funcionales

ID	Requerimiento Funcional	Descripción	Prioridad
RF01	Registrar Usuario	El sistema debe permitir a los usuarios registrarse con su nombre, correo y contraseña.	Alta
RF02	Iniciar Sesión	El sistema debe permitir a los usuarios iniciar sesión utilizando su correo y contraseña.	Alta
RF03	Seleccionar Tipo de suscripción	El sistema debe permitir mostrar los tipos de suscripción.	Alta
RF04	Pagar suscripción	El sistema debe permitir que un usuario registrado pague su suscripción.	Alta
RF06	Seleccionar Operación a realizar	El sistema debe permitir a los usuarios seleccionar operaciones como cortar y fusionar PDFs	Alta
RF07	Subir Archivo PDF	El sistema debe permitir a los usuarios subir archivos PDF para ser procesados.	Media

RF08	Configurar Parámetros de Operación	El sistema debe permitir a los usuarios configurar parámetros específicos para cada operación.	Alta
RF09	Descargar el pdf generado	El sistema debe permitir a los usuarios descargar el PDF procesado.	Alta
RF10	Mostrar historial de operaciones	El sistema debe permitir a los usuarios ver el historial de operaciones realizadas.	Alta

*Fuente: Elaboración Propia*

### 1.1.2. Requerimientos No Funcionales – Atributos de Calidad

*Tabla 02: Requerimientos No Funcionales*

ID	Requerimiento No Funcional	Descripción	Prioridad
RNF01	Seguridad	El sistema debe asegurar que todos los datos del usuario están protegidos mediante encriptación y otras medidas de seguridad.	Alta
RNF02	Rendimiento	El sistema debe ser capaz de procesar y realizar operaciones en PDFs de manera rápida y eficiente, soportando múltiples usuarios simultáneamente.	Alta
RNF03	Usabilidad	El sistema debe ser fácil de usar y comprender, proporcionando una	Alta



		interfaz de usuario intuitiva.	
RNF04	Compatibilidad	El sistema debe ser compatible con los principales navegadores web	Media

*Fuente: Elaboración Propia*

### 3. REPRESENTACIÓN DE LA ARQUITECTURA DEL SISTEMA

#### 3.1. Vista de Caso de uso

##### 1.1.3. Diagramas de Casos de uso

*Diagrama 03: Diagrama de Caso de Uso*



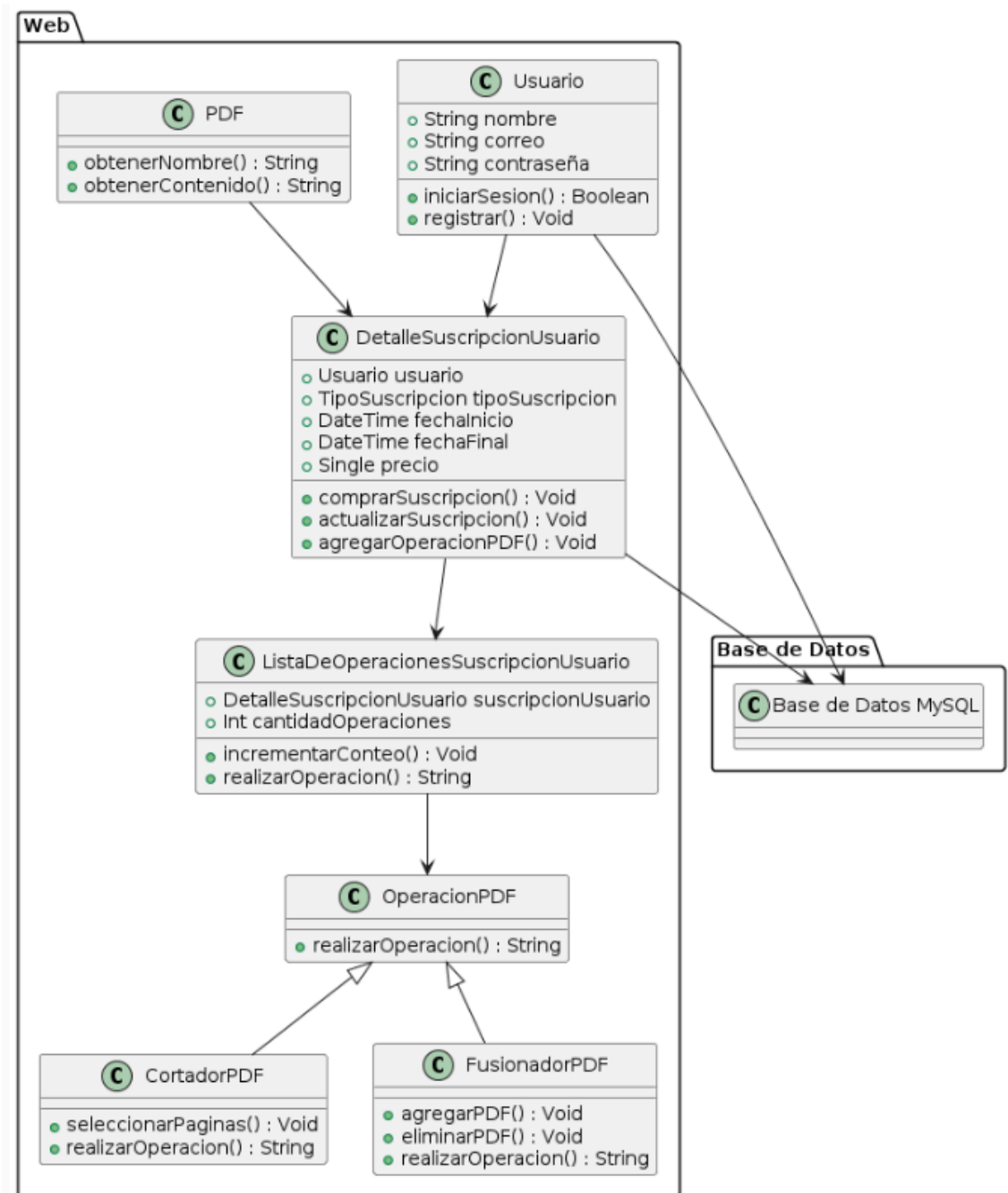
*Fuente: Elaboración Propia*

*Explicación: En este diagrama de caso de uso se puede apreciar como el actor usuario interactúa con los distintos casos del sistema PDF solutions.*

#### 3.2. Vista Lógica

##### 3.2.1. Diagrama de Subsistemas (paquetes)

*Diagrama 04: Diagrama de subsistemas (paquetes)*



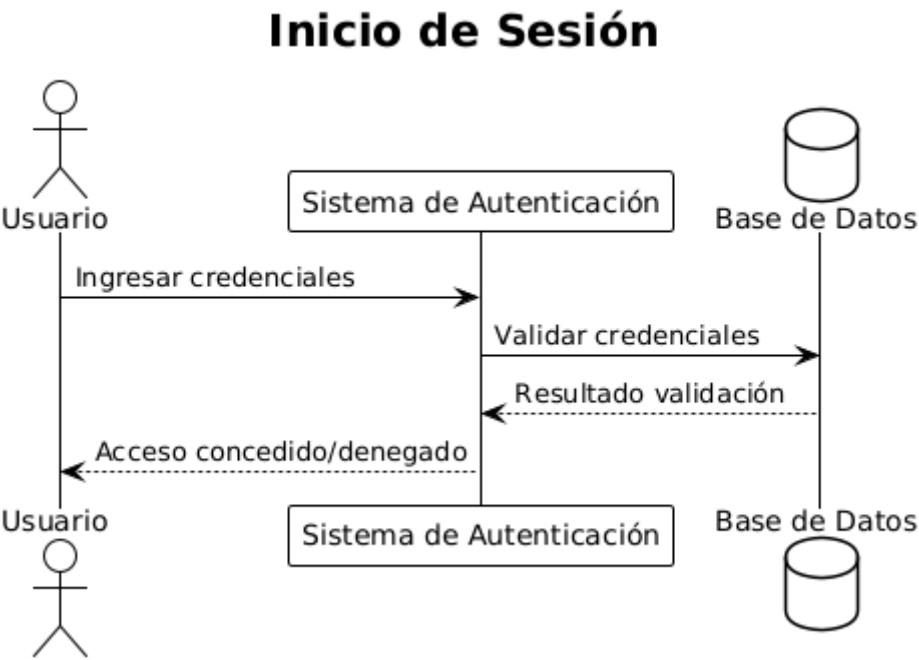
Fuente: Elaboración Propia

Explicación: En este diagrama de paquetes se puede ver como interactúan entre si las distintas clases dentro del proyecto.

3.2.2. Diagrama de Secuencia (vista de diseño)

Inicio Session

Diagrama 05: Inicio Session

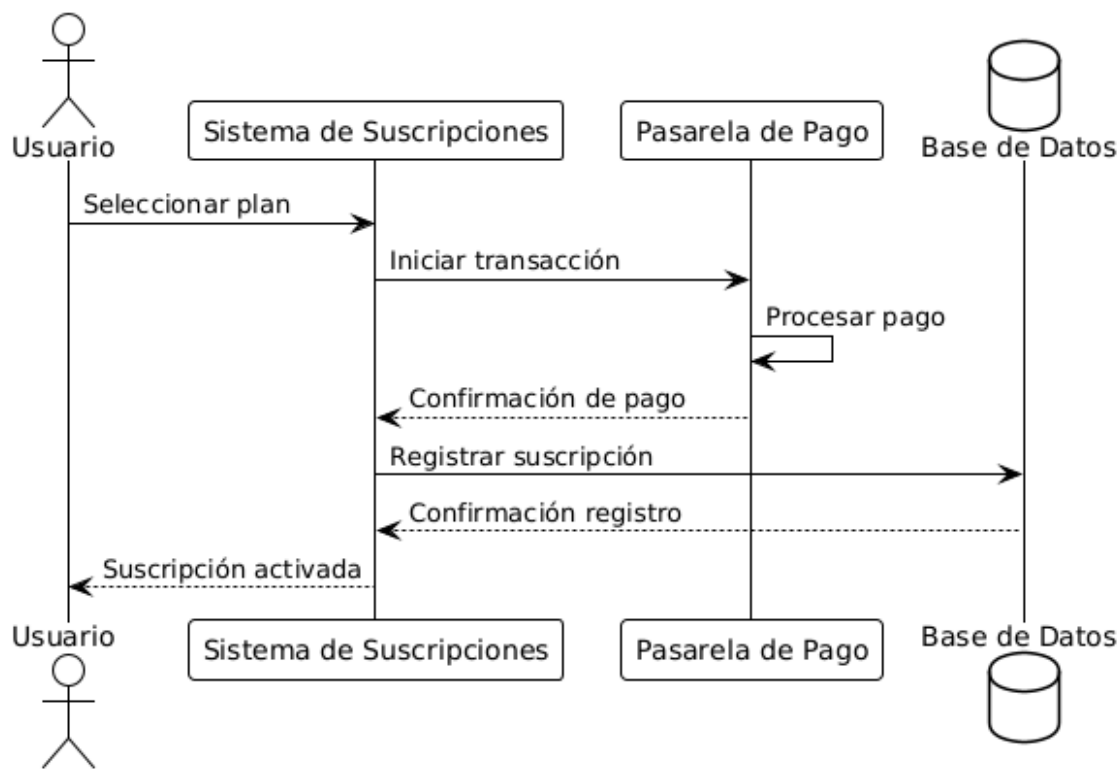


Fuente: Elaboración Propia

Comprar Suscripción Premium

Diagrama 06: Comprar Suscripción

# Comprar Suscripción

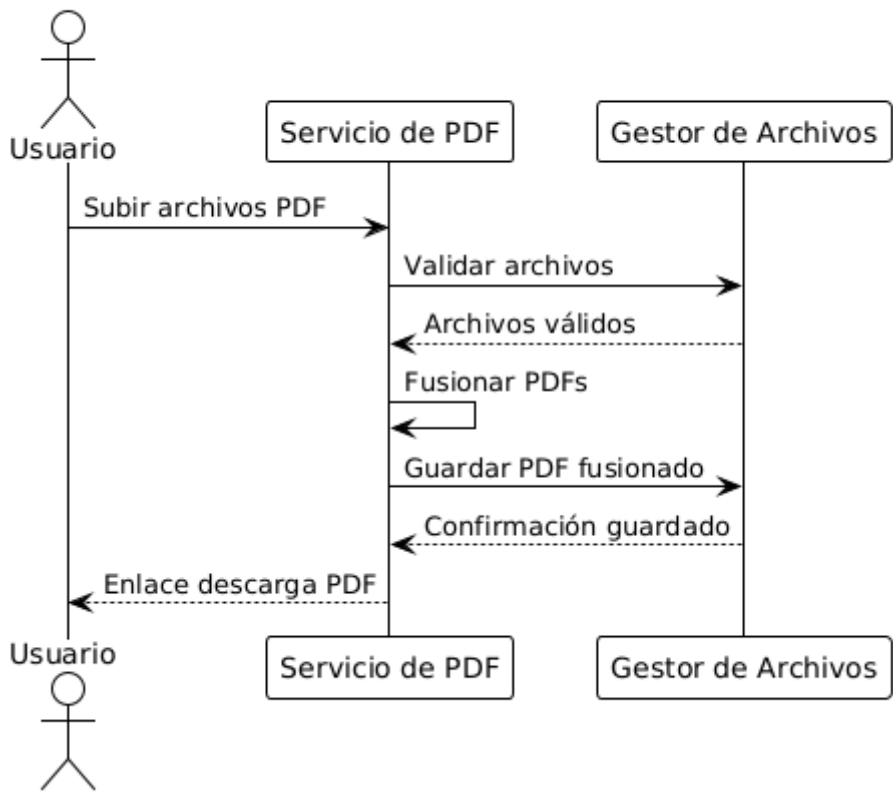


Fuente: Elaboración Propia

Fusionar PDF

Diagrama 07: Fusionar PDF

Fusionar PDF

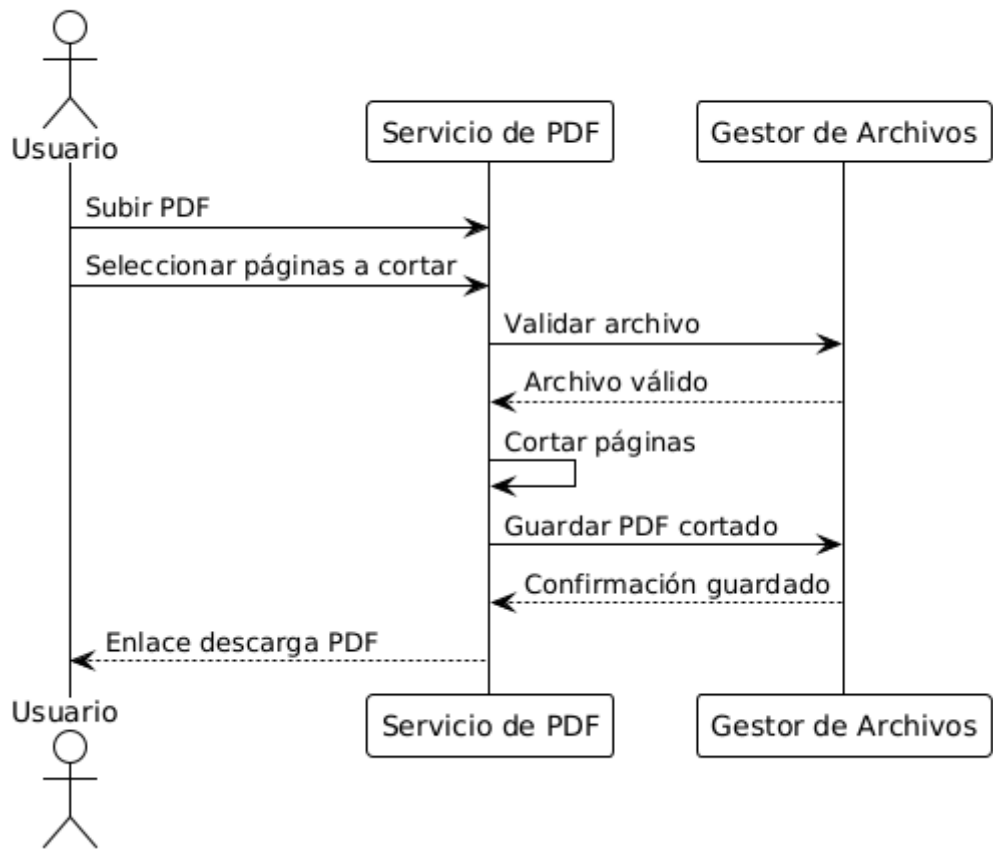


Fuente: Elaboración Propia

Cortar PDF

Diagrama 08: Cortar PDF

Cortar PDF

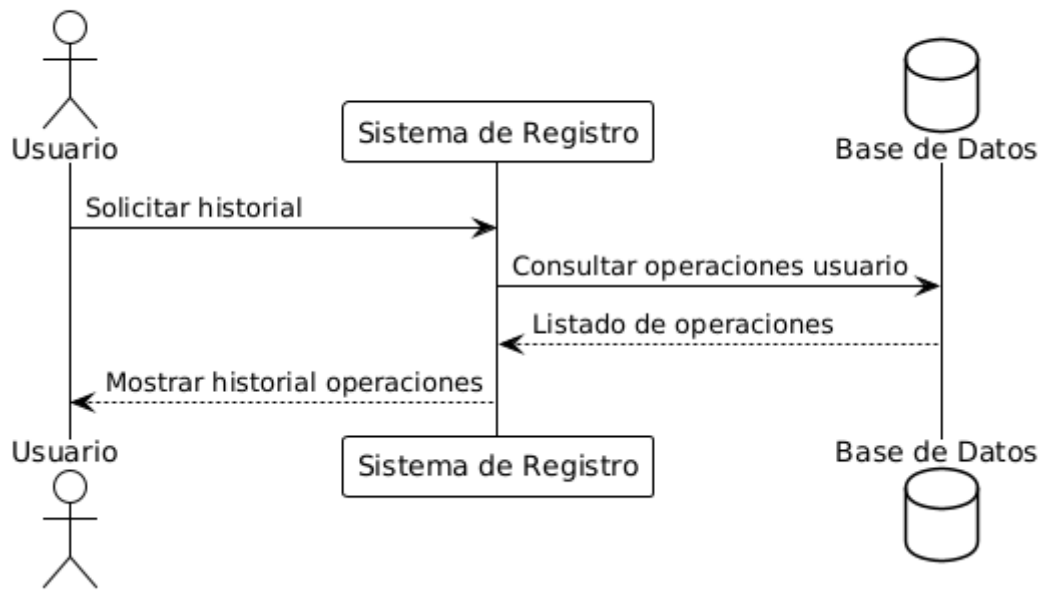


Fuente: Elaboración Propia

Ver Operaciones Realizadas

Diagrama 09: Ver Operaciones Realizadas

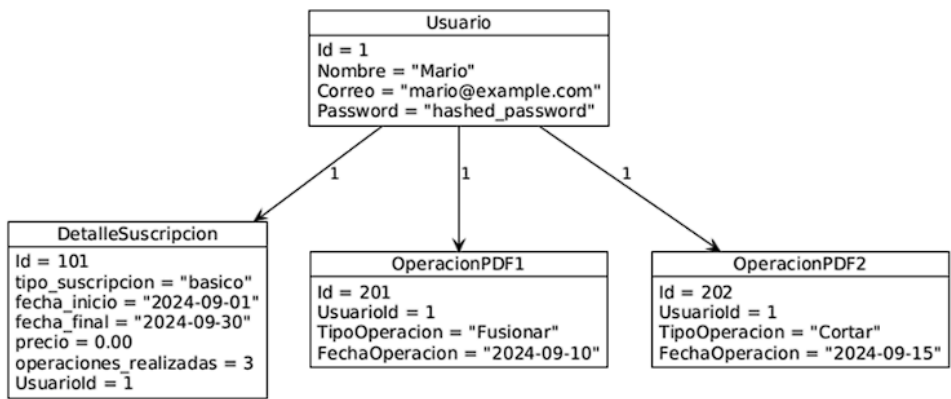
## Ver Operaciones Realizadas



Fuente: Elaboración Propia

### 3.2.3. Diagrama de Objetos

Diagrama 10: Diagrama de Objetos

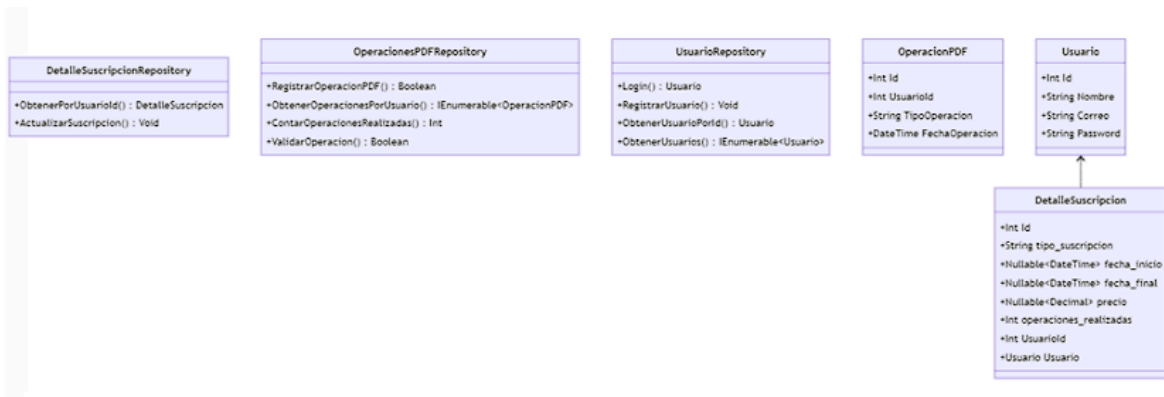


Fuente: Elaboración Propia

Explicación: En este diagrama de objetos se muestran los datos de los objetos mostrados.

### 3.2.4. Diagrama de Clases

Diagrama 11: Diagrama de Clases

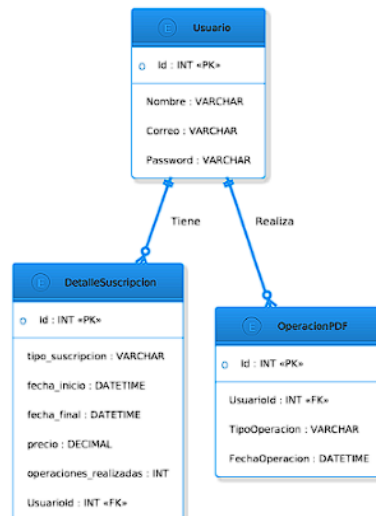


*Fuente: Elaboración Propia*

*Explicación: En este diagrama de clases se aprecia las clases realizadas para el proyecto PDF Solutions.*

### 3.2.5. Diagrama de Base de datos (relacional o no relacional)

*Diagrama 12: Diagrama de Base de datos*



*Fuente: Elaboración Propia*

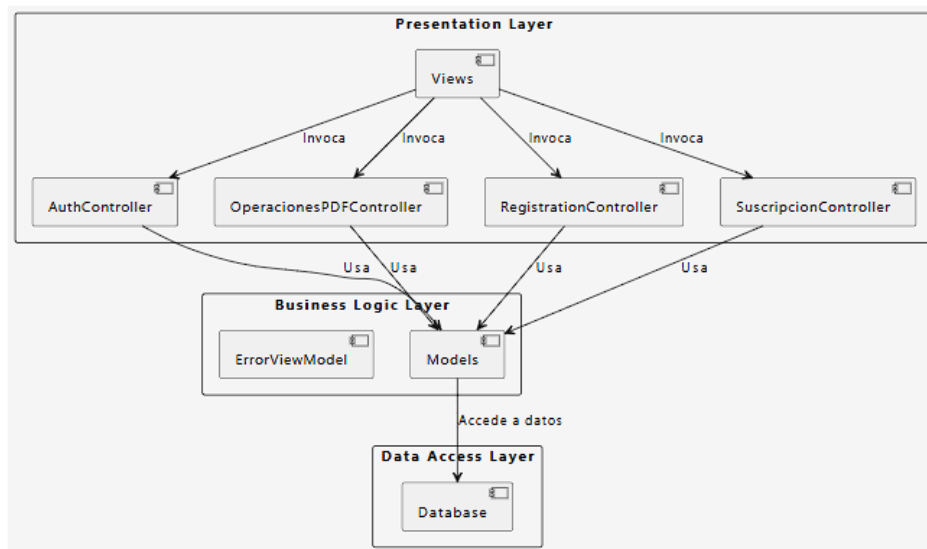
*Explicación: En este diagrama de base de datos se ven las tablas usuarios OperacionesPDF y DetalleSuscripción interactuando entre ellas.*

## 3.3. Vista de Implementación (vista de desarrollo)

### 3.3.1. Diagrama de arquitectura software (paquetes)

*Diagrama 13: Diagrama de arquitectura software*





*Fuente: Elaboración Propia*

*Explicación: En este diagrama de arquitectura de software se pueden ver las distintas capas presentation , Business Logic y Data Acces del proyecto y como interactúan entre ellas.*

### 3.3.2. Diagrama de arquitectura del sistema (Diagrama de componentes)

*Diagrama 0X: Diagrama de arquitectura del software*



*Fuente: Elaboración Propia*

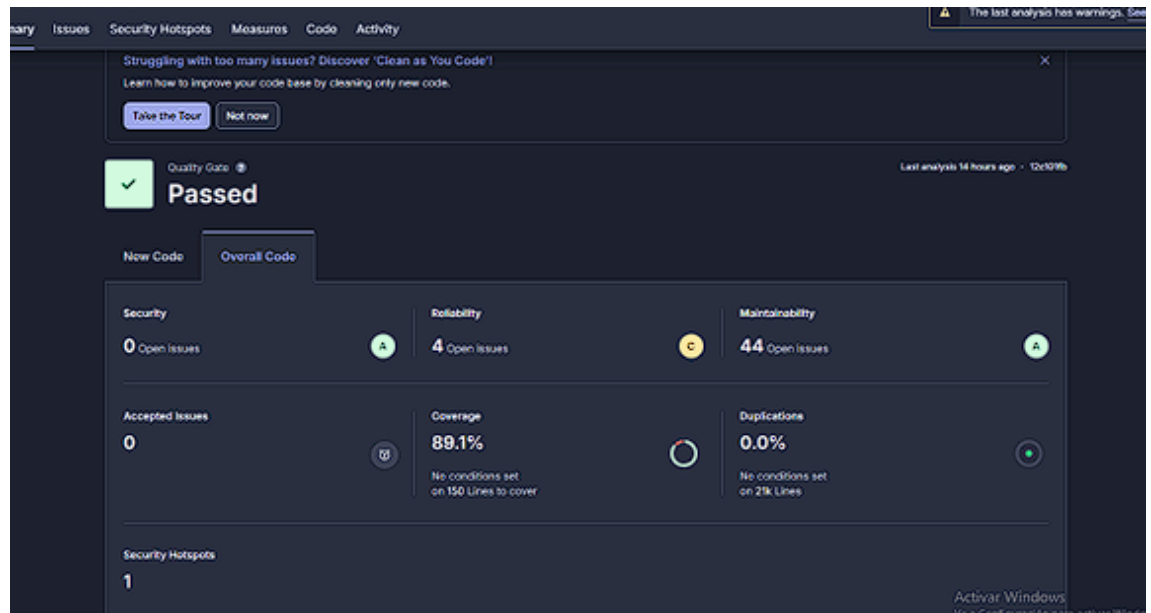
*Explicación: En este diagrama de componentes se pueden ver los distintos componentes que tiene el proyecto PDF solutions.*

## 4. ATRIBUTOS DE CALIDAD DEL SOFTWARE

### 4.1. Informe de cobertura

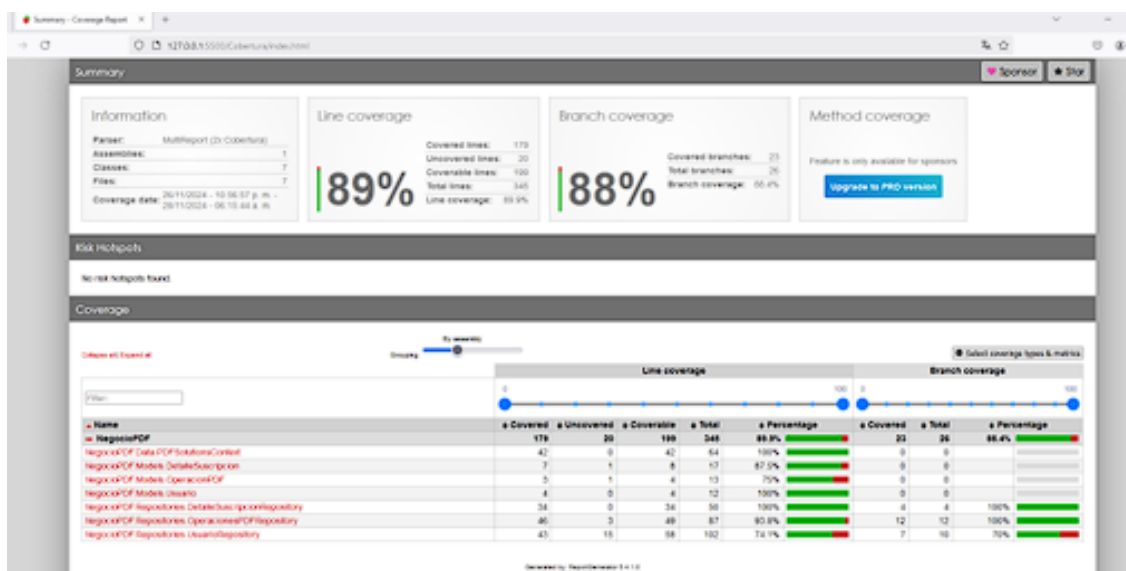
Utilizando cualquier herramienta que pueda visualizar la cobertura, Sonarqube, coverlet, cobertura, etc.

Reporte de cobertura de pruebas

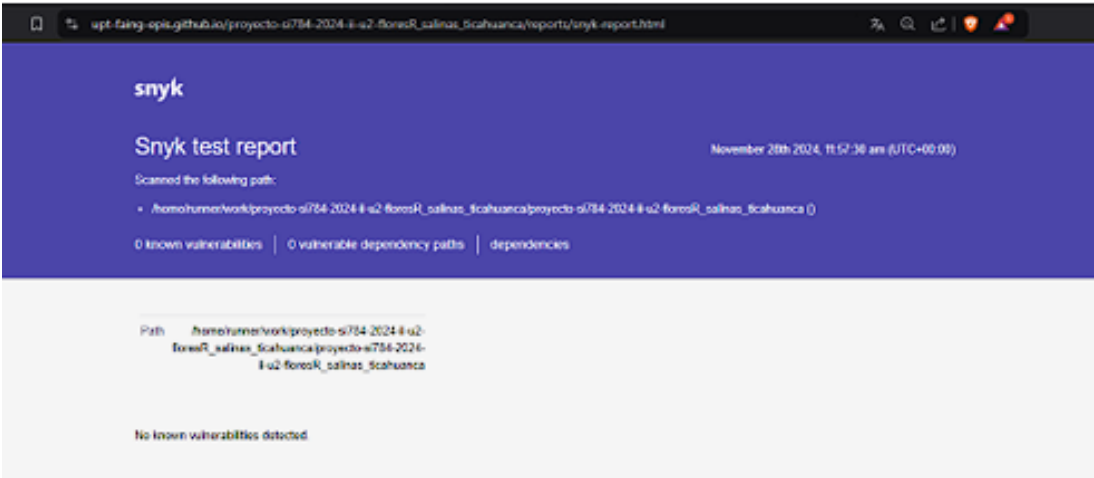


La imagen nos muestra un reporte de pruebas unitarias donde se ha logrado una cobertura del 89.1% en el código, superando el umbral recomendado del 80%. Esto refleja un buen nivel de pruebas aplicadas a los métodos más importantes del sistema, lo que garantiza la validación de su funcionalidad. El análisis también destaca que no se encontraron problemas de seguridad ni duplicaciones de código, lo que contribuye a la mantenibilidad y confiabilidad del proyecto.

## Pruebas de integración utilizando Mocks o Fake Classes



El reporte destaca una cobertura del 89% en líneas de código y del 88% en ramas, lo que indica un alto nivel de pruebas realizadas sobre el código base. Esto incluye detalles de los módulos cubiertos, con porcentajes individuales para líneas y ramas, mostrando áreas bien probadas y algunas con margen de mejora.



El reporte de Snyk muestra que no se detectaron vulnerabilidades conocidas en las dependencias escaneadas dentro del proyecto analizado. Esto indica que las bibliotecas utilizadas actualmente no presentan riesgos de seguridad según la base de datos de vulnerabilidades de Snyk, lo que es una señal positiva en términos de seguridad para las dependencias externas del proyecto.

Informe - Pruebas con mutaciones

The screenshot shows a Stryker.NET Report for "All files". It includes a summary bar with counts: 82 (green), 27 (red), and 25 (orange). Below is a table with columns: File / Directory, Mutation Score (Of total, Of covered), Killed, Survived, Timeout, No coverage, Ignored, Runtime errors, Compile errors, Detected, Undetected, and Total.

File / Directory	Mutation Score	Killed	Survived	Timeout	No coverage	Ignored	Runtime errors	Compile errors	Detected	Undetected	Total
	Of total	Of covered									
All files	61.18 / 75.23	82	27	0	25	10%	0	0	82	82	242
Data/POF/SolutionsContext.cs	34.38 / 34.38	11	21	0	0	4	0	0	11	21	36
Migrations	0.00 / 0.00	0	0	0	14	77	0	0	0	14	81
Models	N/A	0	0	0	0	0	0	0	0	0	0
Repositories	80.44 / 92.31	71	6	0	11	24	0	0	71	17	115

El análisis indica qué partes del código están bien cubiertas por pruebas y cuáles necesitan mayor atención, ayudando a fortalecer el conjunto de pruebas.

4.2. Informe de ejecución de Pruebas

Utilizando cualquier herramienta de visualización de resultados de ejecución, Azure Devops, Github actions, Aws Code Build, Specflow LivinDoc

## Reporte de cobertura de pruebas

PROYECTO\_PDF/NegocioPDF/Repositories/IUsuarioRepository.cs

**Adaptability**  
Move 'IUsuarioRepository' into a named namespace. No tags +

Open ☒ Mario Antonio Flores Ramos **Reliability** Bug Major 5min effort · 4 days ago

PROYECTO\_PDF/PROYECTO\_PDF/Views/Registration/Registrarse.cshhtml

**Intentionality**  
A form label must be associated with a control. accessibility +

Open ☒ Mario Antonio Flores Ramos **Reliability** Code Smell Minor 5min effort · 4 days ago

**Intentionality**  
A form label must be associated with a control. accessibility +

Open ☒ Mario Antonio Flores Ramos **Reliability** Code Smell Minor 5min effort · 4 days ago

**Intentionality**  
A form label must be associated with a control. accessibility +

Open ☒ Mario Antonio Flores Ramos **Reliability** Code Smell Minor 5min effort · 4 days ago

4 of 4 shown

Adaptability | Not modular

**Move 'IUsuarioRepository' into a named namespace.** [🔗](#)

Types should be defined in named namespaces [csharpguid:S3903](#)

Software qualities impacted: **Reliability**

Open ☒ Mario Antonio Flores Ramos **Bug** Major

Where is the issue? Why is this an issue? Activity [Open in IDE](#)

PROYECTO\_PDF/NegocioPDF/Repositories/IUsuarioRepository.cs [See all issues in this file](#)

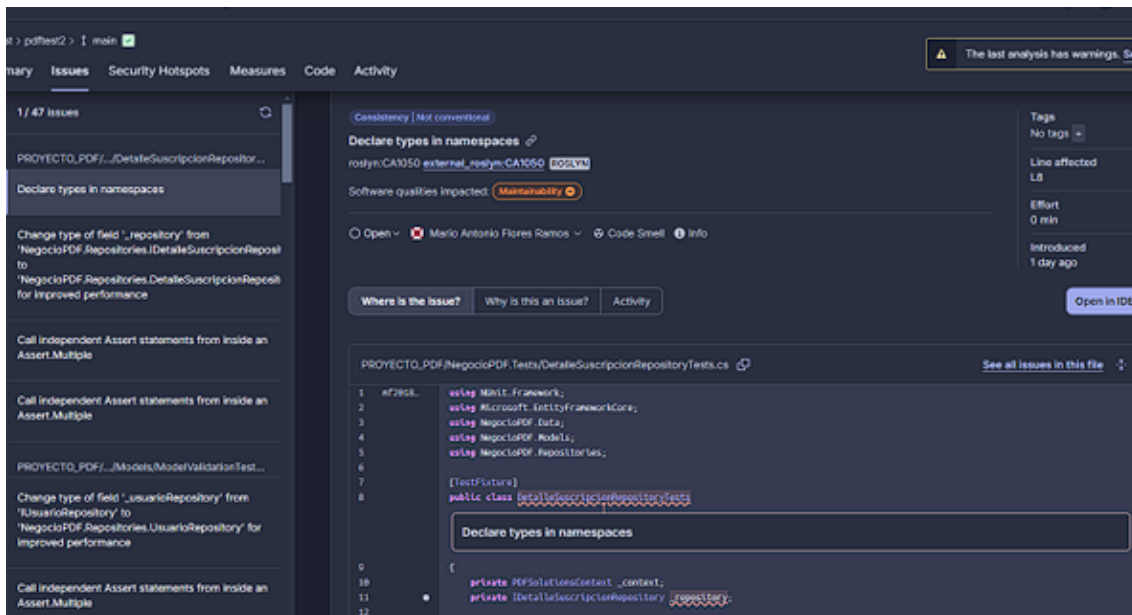
```

1  nf2018...
2      using System.Collections.Generic;
3      using NegocioPDF.Models;
4
5
6      public interface IUsuarioRepository
7      {
8          Usuario Login(string correo, string password);
9          void RegistrarUsuario(Usuario usuario);
10         IEnumerable<Usuario> ObtenerUsuarios();
11         Usuario ObtenerUsuarioPorId(int idUsuario);
12     }

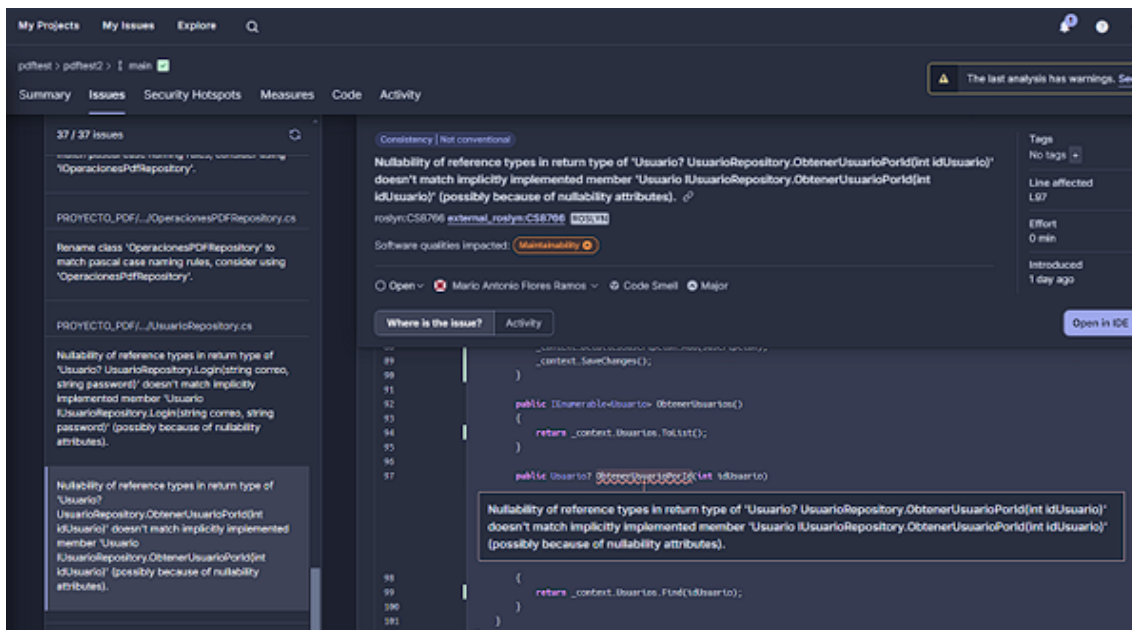
```

Move 'IUsuarioRepository' into a named namespace.

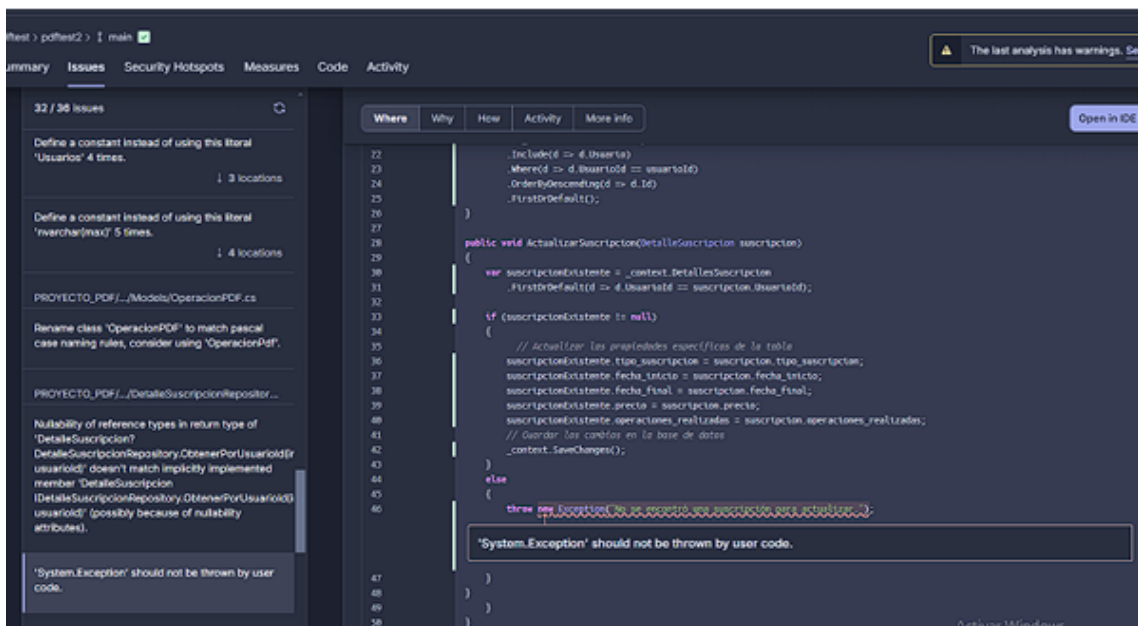
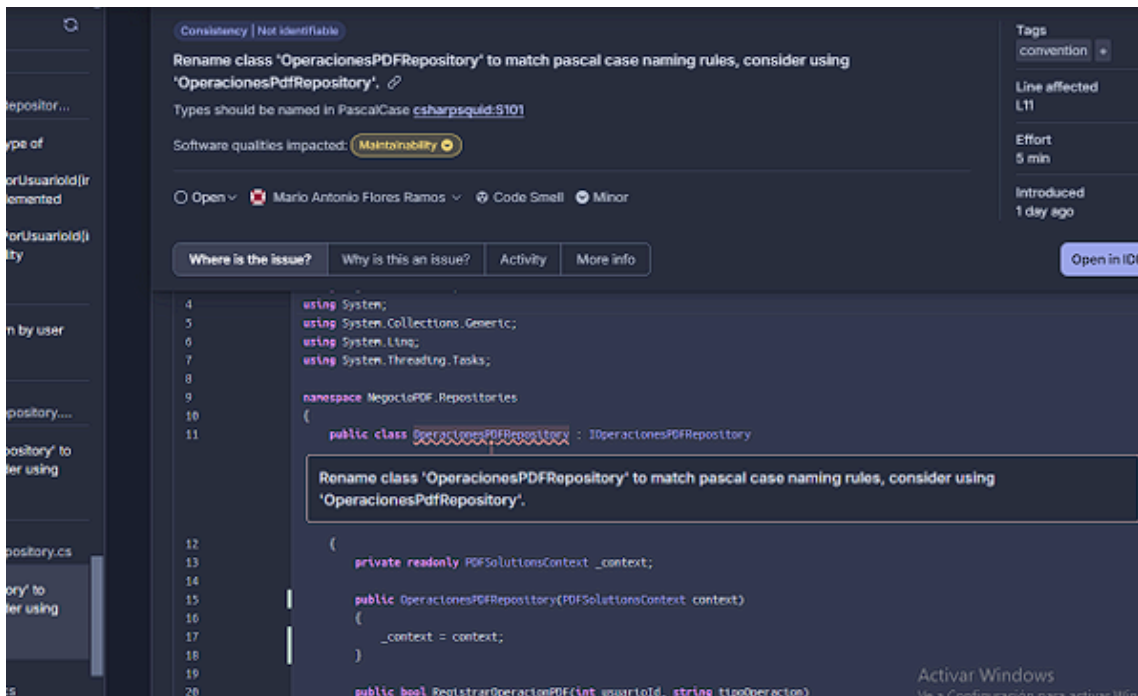
Se detectó un problema en la organización del código, donde una interfaz no estaba ubicada en el namespace adecuado, lo que afectaba la mantenibilidad del proyecto.



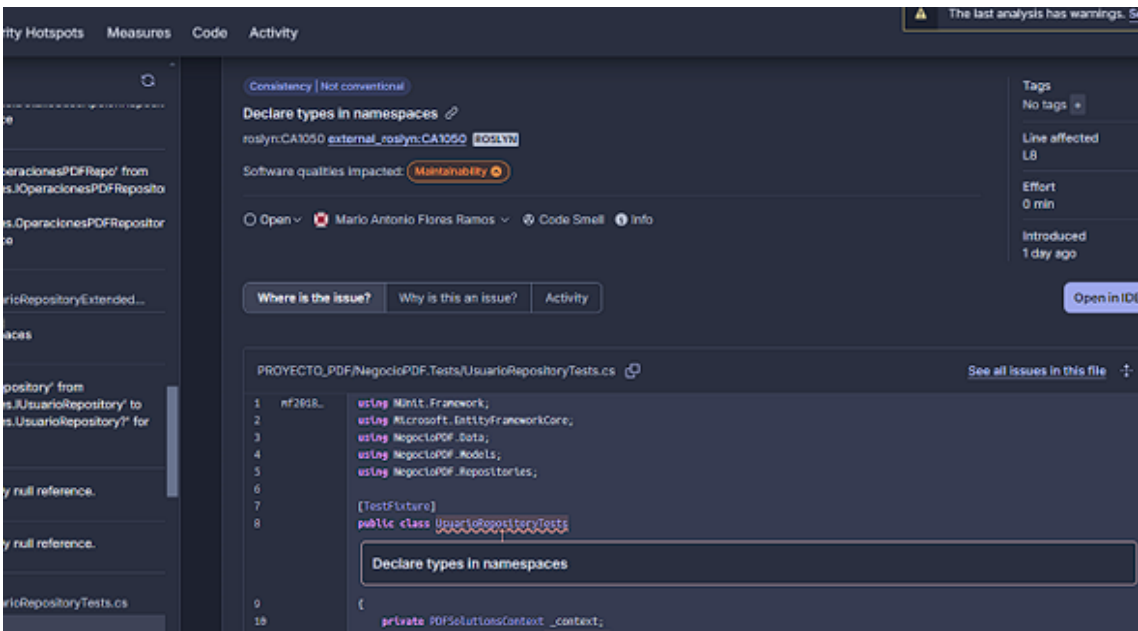
Se identificó un problema de consistencia en la organización de las pruebas, ya que la clase `DetalleSuscripcionRepositoryTests` no estaba correctamente declarada en un namespace adecuado, lo que afectaba la claridad y mantenibilidad del código.



El análisis estático detectó un problema de anulabilidad en el método `ObtenerUsuarioPorId` dentro del repositorio `UsuarioRepository`, ya que el tipo de retorno no manejaba adecuadamente los valores nulos.

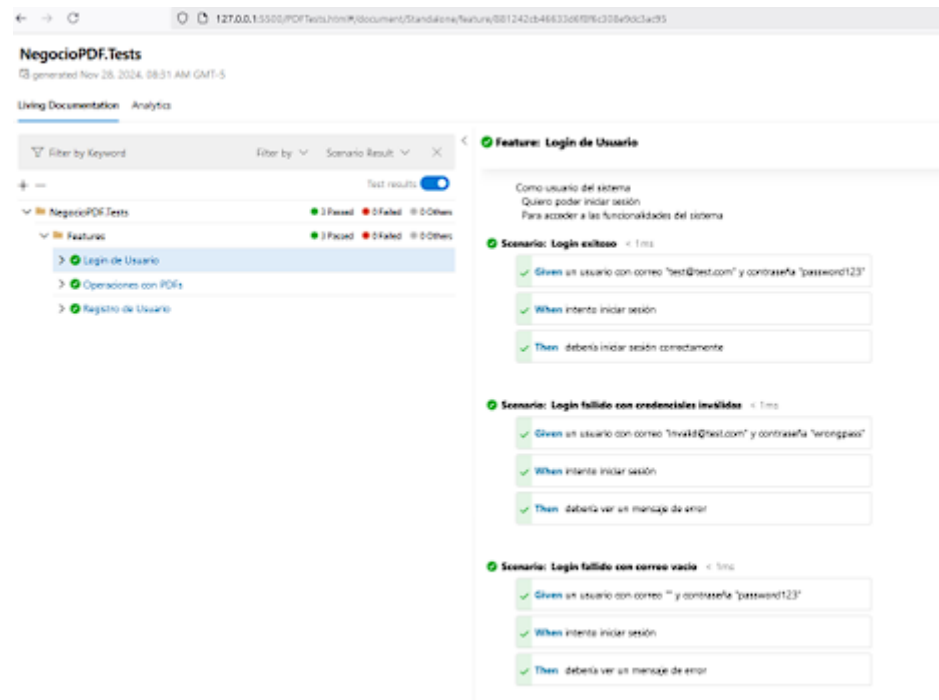


El análisis estático destaca un problema en el método ActualizarSuscripcion, donde se utiliza System.Exception, lo cual no es recomendable porque dificulta la identificación precisa de los errores.

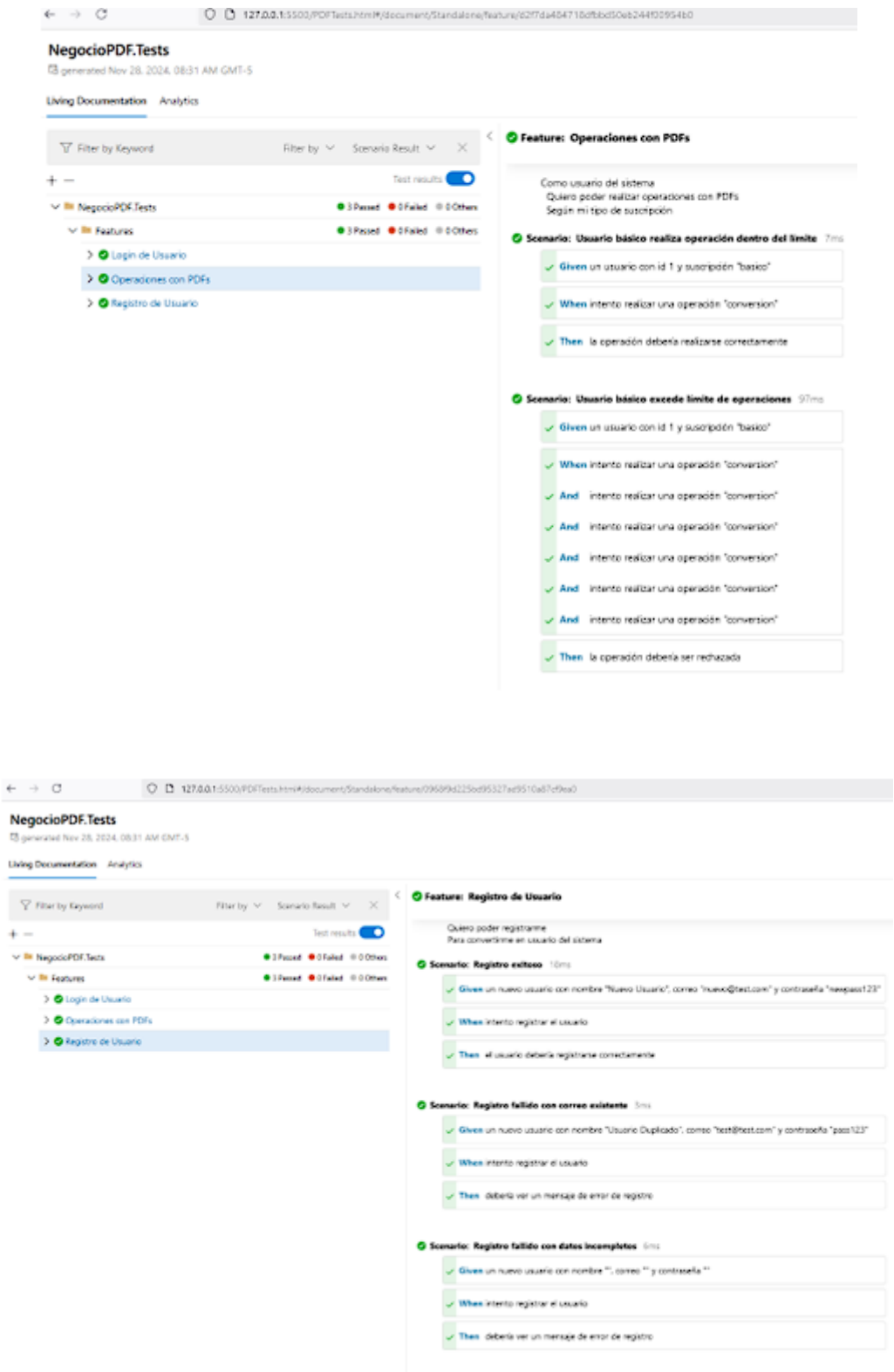


El análisis estático señala un problema de consistencia debido a la ausencia de un namespace definido, lo que afecta la mantenibilidad del código.

Reporte de Pruebas guiadas por el comportamiento (BDD Given When Then)





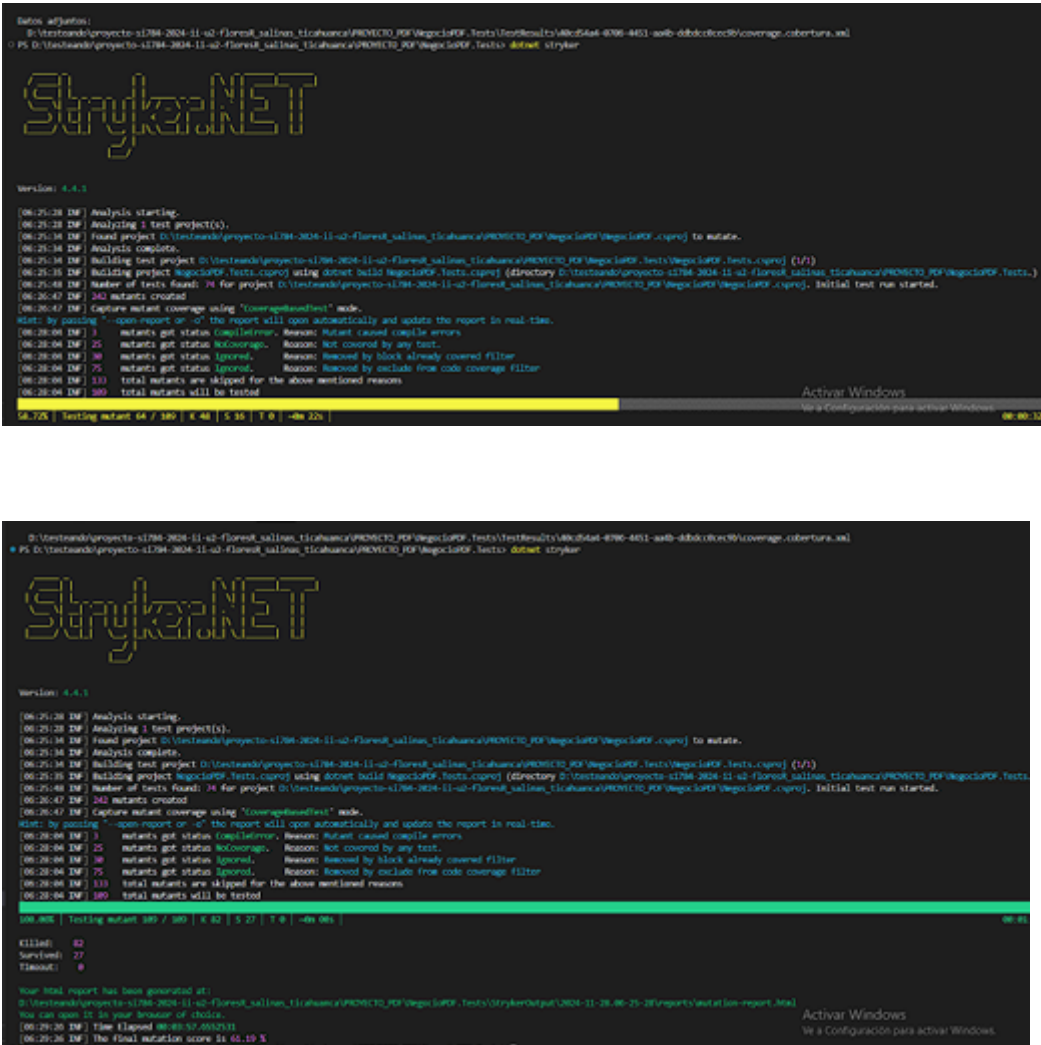


El reporte muestra pruebas guiadas por el comportamiento (BDD) utilizando el formato Given-When-Then, específicamente para el caso de uso de inicio de sesión. Cada prueba representa un escenario distinto basado en una historia de usuario: un inicio de sesión exitoso, un intento fallido por credenciales inválidas y otro fallido por un correo vacío.



Esto asegura que las funcionalidades clave se validan según las expectativas del usuario y los criterios de aceptación.

Reporte de Pruebas mutantes



Reporte de Stryker.NET - Análisis de Pruebas Mutantes

Barra Superior Colorida

Verde (82): Mutantes eliminados (¡Bueno!)

Rojo (27): Mutantes sobrevivieron (¡Malo!)

Naranja (25)\*\*: Sin cobertura

## Columnas Principales

### Mutation Score

Of total: 61.19% - Porcentaje general de mutantes eliminados

Of covered: 75.23% - Porcentaje de mutantes eliminados en código cubierto por pruebas

### Estado de los Mutantes

Killed (82): Mutantes que tus pruebas detectaron y fallaron (¡Bueno!)

Survived (27): Mutantes que tus pruebas no detectaron (¡Malo!)

No coverage (25): Código sin pruebas unitarias

Ignored (105): Mutantes que se ignoraron

### Análisis por Carpetas

Data/PDFSolutionsContext.cs: 34.38% de efectividad

Migrations: 0% (Normal, no se suelen probar)

Models: N/A

Repositories: 80.68% (Bastante bueno)

### Bibliografía:

- Semgrep | Homepage. (s. f.). Semgrep. <https://semgrep.dev>
- Datadog. (2016, 14 julio). DataDog Code Monitoring | DataDog. Datadog. <https://www.datadoghq.com>
- Snyk. (s. f.). Developer security | Snyk. <https://snyk.io>