Informe de Trabajo del Equipo

|  |
| --- |
| Nombre del Proyecto: Gestor de credenciales |
| Nombre del Equipo: Mi grupo no tiene nombre |
| Fecha de entrega: 16/05/2025 |

**1. Análisis del Trabajo**

Este proyecto lo hemos interpretado como una introducción práctica a los principios del desarrollo seguro, utilizando la metodología S-TDD (Security Test-Driven Development) para construir un gestor de credenciales que cumpla tanto requisitos funcionales como de seguridad.

**2. Seguimiento y Herramientas de Gestión**

Herramienta Uso principal

GitHub Control de versiones y revisión de código

Trello Gestión de tareas y asignaciones

Google Docs Redacción colaborativa del informe

Discord Comunicación diaria y coordinación

**3. Contribuciones Individuales**

**Estudiante 1: [Esteban Sánchez Gamez]**

Resumen de tareas realizadas, colaboraciones, tiempo dedicado y entregables.

|  |  |
| --- | --- |
| Tarea | Desarrollo de informe |
| Colaboradores | Pablo Jesús Delgado Muñoz |
| Tiempo empleado | 1h |
| Entregables | Informe en pdf |
| Notas | Se revisó en conjunto con el equipo. |

|  |  |
| --- | --- |
| Tarea | Desarrollo de código |
| Colaboradores | Pablo Jesús Delgado Muñoz, Álvaro Pastor |
| Tiempo empleado | 5h |
| Entregables | Código, documentación, pruebas, informes, presentaciones, etc. Incluir enlaces o rutas si se usa un repositorio. |
| Notas | Se programaron las funciones principales del proyecto, se realizaron pruebas unitarias y se documentó el uso de cada módulo. Trabajo en estrecha colaboración para dividir funciones y validar resultados. |

**Estudiante 2: [Pablo Jesús Delgado Muñoz]**

Resumen de tareas realizadas, colaboraciones, tiempo dedicado y entregables.

|  |  |
| --- | --- |
| Tarea | Desarrollo informe |
| Colaboradores | Esteban Sánchez Gamez |
| Tiempo empleado | 2h |
| Entregables | Documento pdf |
| Notas | Se redactó el informe con base en las pruebas y resultados obtenidos durante el desarrollo del código. |

|  |  |
| --- | --- |
| Tarea | Depuración código y testing |
| Colaboradores | Álvaro Pastor, Esteban Sánchez Gamez |
| Tiempo empleado | 2h |
| Entregables | n/a |
| Notas | Se buscó soluciones a ciertos problemas durante el desarrollo y testing del código. |

…

**3. Evaluación del Trabajo en Equipo**

**¿Qué ha funcionado bien?**

Buena organización y coordinación a través de Trello y Discord.

Los roles de programación por parejas rotaban de forma efectiva en cada ciclo.

**¿Qué se puede mejorar?**

Los roles de programación aunque rotamos, pensamos que no era lo más conveniente en todas las ocasiones y perdíamos efectividad en cuanto al desarrollo del proyecto.

Faltó una mejor explicación desde el principio para saber qué probar exactamente en la seguridad.

**4. Documentación breve explicando las decisiones de diseño**

**Principios SOLID:**  
  
 SRP: Separación clara entre cifrado, validación, logging, y lógica de negocio.  
  
 DIP: Uso de interfaces (ICifrador, IValidadorPassword) para facilitar cambios sin afectar dependencias.  
  
 OCP: SecureStrategyFactory permite extender validadores/cifradores según roles sin modificar código base.  
  
**Seguridad:**  
  
 Uso de bcrypt para almacenar contraseñas de forma segura.  
   
 Validación robusta de contraseñas (mínimo de caracteres, tipos, y símbolos).  
  
 LoggerSeguro con hashing encadenado para detectar alteraciones en los logs.  
  
 Protección contra inyección simple (verificación de caracteres peligrosos en el servicio).  
  
**Diseño extensible:**  
  
 Patrón Factory para generar políticas de seguridad adaptadas al rol del usuario.  
  
 Proxy seguro (GestorCredencialesProxy) para controlar el acceso mediante autenticación previa.

**4. Evaluación de la experiencia con S-TDD**

Nos obligó a pensar en seguridad desde el principio, cosa que vemos buena ya que normalmente se ve la seguridad en segundo plano.

Las pruebas automáticas están claras que han sido muy importantes y hemos aprendido mucho con ellas, las implementaremos en futuros proyectos.

Nos ha gustado mucho la forma de aprendizaje basada en Design by Contract, que como hemos dicho antes es muy efectiva para no dejar la seguridad de lado.

Destacamos que este tipo de desarrollo requiere más tiempo de planificación y diseño, pero garantiza seguridad.

**Beneficios:**  
  
 El enfoque S-TDD obliga a pensar en la seguridad desde el inicio del desarrollo.  
  
 Facilita una arquitectura modular y testeable.  
  
 Reduce errores comunes de seguridad, como almacenamiento inseguro de contraseñas o validaciones débiles.

**Desafíos:**  
  
 Mayor tiempo inicial para definir contratos e interfaces.  
   
 La implementación segura de logs y políticas requiere cuidado adicional.

**Valor añadido:**  
  
 Se obtiene un sistema más robusto, trazable (logs seguros) y adaptable a futuros cambios o amenazas.