



Proyecto Profesional I

Plan de Proyecto Actualizado

Datacript

Docentes:

- Juan Carlos Monteros
- Francisco Orozco De La Hoz
- Leandro Dikenstein

Integrantes:

- Benitez Micaela - DNI: 43 473 284
- Benitez Yamila - DNI: 41 199 878
- Clauser Nahuel - DNI: 39 803 927
- Gómez Federico - DNI: 40 743 800
- Perez Giannina – DNI: 43 729 769
- Prieto Lucas - DNI: 43 626 494
- Torrico Franco – DNI: 42 370 140

Introducción

La Inteligencia Artificial (IA) y el Machine Learning (ML) representan una gran revolución en muchos ámbitos, siendo uno de ellos el campo de la medicina. Estas herramientas ofrecen un potencial transformador para la atención médica al mejorar la toma de decisiones clínicas y, de esta forma, beneficiar la salud y el bienestar de los pacientes.

Objetivos del proyecto

El objetivo principal de este proyecto es desarrollar un modelo de clasificación de imágenes médicas mediante el uso de técnicas de Inteligencia Artificial y Machine Learning, con el propósito de mejorar el proceso de diagnóstico en el ámbito de la salud.

Objetivos del Documento

El objetivo de este documento es proporcionar una guía detallada para la planificación y ejecución del proyecto. Esto incluye definir la misión y visión del negocio, establecer el alcance del proyecto, desarrollar un plan de comunicaciones, identificar los requerimientos funcionales y no funcionales, crear una estructura de subdivisión de tareas con WBS, a la cual le asignaremos un diccionario, establecer un calendario que incluya la estimación inicial de pesos y esfuerzo, identificar y realizar la planificación necesaria para posibles riesgos, establecer entregables (Hitos) y describir la administración de bugs y cambios.

Además, se detallarán los indicadores a utilizar para controlar las cuestiones mencionadas anteriormente. Por último, se mencionan las tecnologías y herramientas que se utilizarán para cumplir el objetivo del proyecto.

Equipo de trabajo y roles

- **Product Owner:** Leandro Dikenstein
- **Scrum Master:** Giannina Perez
- **Líder Técnico:** Lucas Prieto
- **Desarrolladora:** Benitez Micaela
- **Desarrolladora:** Benitez Yamila
- **Desarrollador:** Nahuel Clauser
- **Desarrollador:** Franco Torrico
- **Tester:** Federico Gómez

Metodología SAFe

Vamos a estar llevando a cabo la metodología SAFe para gestionar el trabajo entre nuestro equipo de datos, el equipo de UX y los equipos encargados de realizar los modelos para la clasificación de imágenes médicas. De esta manera, vamos a poder agilizar la colaboración entre los equipos, además de aumentar la productividad y calidad de las entregas de trabajo.

Metodología Scrum

En nuestro equipo de trabajo hemos optado por utilizar la metodología ágil Scrum, en donde coordinaremos nuestro espacio de trabajo y tareas a realizar.

Story Points

En nuestro enfoque de gestión ágil de proyectos, hemos decidido implementar "Story Points" (1 story point = 1 hora) como nuestra principal técnica de estimación. Además, facilita la comunicación dentro del equipo, garantizando que todos tengamos una comprensión compartida de la magnitud de cada elemento de trabajo. Al involucrar a todo el equipo en el proceso de estimación, fomentaremos la colaboración y la toma de decisiones conjuntas sobre la capacidad de los sprints, la priorización del backlog y la planificación de nuestro trabajo de manera más precisa y eficiente.

Sprint Planning

Se realizará esta reunión al principio de cada Sprint para armar el Sprint Backlog, estimar y asignar tareas a todos los integrantes del equipo.

Daily Scrum

Hemos decidido implementar reuniones cada 2 días para abordar las necesidades específicas de nuestro proyecto. No haremos reuniones diariamente debido a la disponibilidad de horarios de los integrantes del equipo.

En estas reuniones, los miembros del equipo compartirán las actualizaciones sobre su trabajo y discutirán sobre obstáculos que puedan surgir y frenar sus tareas. De esta forma, podremos coordinarnos para enfrentar circunstancias cambiantes y asegurarnos de que la comunicación fluya de manera eficiente.

Sprint Review

Los Sprints que realizaremos tendrán dos semanas de duración para poder llevar a cabo entregas continuas de valor. Al final de cada Sprint, habrá una reunión formal, donde se evaluará el trabajo completado, revisaremos las metas alcanzadas y discutiremos cualquier desviación o ajuste

necesario en nuestra estrategia. Lo valioso de esta reunión es la retroalimentación recibida para el equipo y la información de todos los equipos de trabajo para poder alcanzar el objetivo final.

Sprint Retrospective

Al final de cada Sprint, se hará una retrospectiva en la cual se tendrán en cuenta los aspectos que se manejaron bien y los aspectos que podrían llegar a mejorar para el siguiente Sprint.

Misión

La misión de nuestro equipo es proporcionar un soporte eficaz para la gestión de los datos médicos cruciales para crear el modelo de clasificación de imágenes médicas. De esta manera, garantizamos que los datos médicos sean manejados con la máxima eficiencia y seguridad, y que estén disponibles para impulsar un diagnóstico de salud más preciso y confiable.

Visión

Nuestra visión es facilitar la toma de decisiones informadas en el campo médico, mejorando la calidad de vida de los pacientes y apoyando la investigación médica.

Alcance

Nuestro equipo, enfocado en la funcionalidad de los datos, tiene un papel importante en el éxito de este proyecto. El alcance de nuestras responsabilidades abarca varios aspectos esenciales para garantizar la seguridad y la eficacia del sistema:

Gestión de Datos

- Almacenamiento seguro de conjuntos de datos de imágenes médicas etiquetadas.
- Establecimiento de un proceso de gestión de datos maestros para garantizar la coherencia y la calidad de los datos utilizados en el sistema.

Cifrado

- Implementación de técnicas de cifrado para proteger los datos médicos sensibles durante el almacenamiento y la transmisión de los mismos.

Seguridad de Datos

- Implementar medidas de seguridad de datos para prevenir el acceso no autorizado y los riesgos cibernéticos.

Colaboración Interdisciplinaria

- Colaborar estrechamente con los equipos de desarrollo de IA y ML para garantizar la coherencia de datos y modelos.
- Colaborar con el equipo de UX/UI para disponibilizar los datos de la manera más adecuada para ser consumidos desde la interfaz.
- Facilitar la comunicación y la integración entre diferentes equipos involucrados en el proyecto.
- Colaborar en la definición y desarrollo de la Arquitectura del Sistema.

Nuestro equipo se compromete a garantizar que los datos médicos sean manejados de manera segura, eficiente y ética, lo que es esencial para el éxito del proyecto de clasificación automática de imágenes médicas y para el beneficio de los pacientes y profesionales de la salud.

Plan de Comunicaciones

- Whatsapp: Medio por el cual nos comunicaremos los integrantes del equipo para resolver todo tipo de cuestiones.
- Telegram: Medio por el cual nos comunicaremos con los otros equipos de trabajo, para coordinar tareas en común y/o correlativas. Además, también lo utilizaremos para comunicarnos con el Product Owner.
- JIRA: Para la asignación, seguimiento y control de las tareas a realizar.
- Gather: Utilizaremos esta herramienta para realizar las dailys ya que permite una interfaz interactiva y amigable para desarrollar un buen ambiente entre el equipo.

Requerimientos Funcionales

1. Gestión de Auditores

- El sistema debe crear y enviar un código por e-mail para que el auditor pueda crear su contraseña en su primer ingreso.
- El sistema debe permitirle al auditor el acceso al perfil.
- El sistema debe permitirle al auditor la carga de informes.
- El sistema debe permitirle al auditor la carga de imágenes.
- El sistema debe permitirle al auditor el acceso al historial.
- El sistema debe permitirle al auditor el acceso a los datos cifrados.

2. Gestión de Administradores

- El sistema debe permitir dar de alta, dar de baja y modificar un usuario de auditor, profesional de la salud o médico.

3. Gestión de Profesionales de la Salud

- El sistema debe crear y enviar un código por e-mail para que el profesional pueda crear su contraseña en su primer ingreso.
- El sistema debe permitirle al profesional el acceso al perfil.

- El sistema debe permitirle al profesional la carga de informes.
- El sistema debe permitirle al profesional la carga de imágenes.

4. Gestión de Médicos

- El sistema debe crear y enviar un código por e-mail para que el médico pueda crear su contraseña en su primer ingreso.
- El sistema debe permitirle al médico el acceso al perfil.
- El sistema debe permitirle al médico el acceso al historial.
- El sistema debe permitirle al médico el acceso a los resultados de las clasificaciones de imágenes médicas.
- El sistema debe permitirle al médico hacer feedback del modelo, una vez realizada su predicción.

5. Cifrado de datos

- Los datos privados y sensibles de los usuarios y pacientes médicos deben estar cifrados.

6. Gestión de permisos y accesos

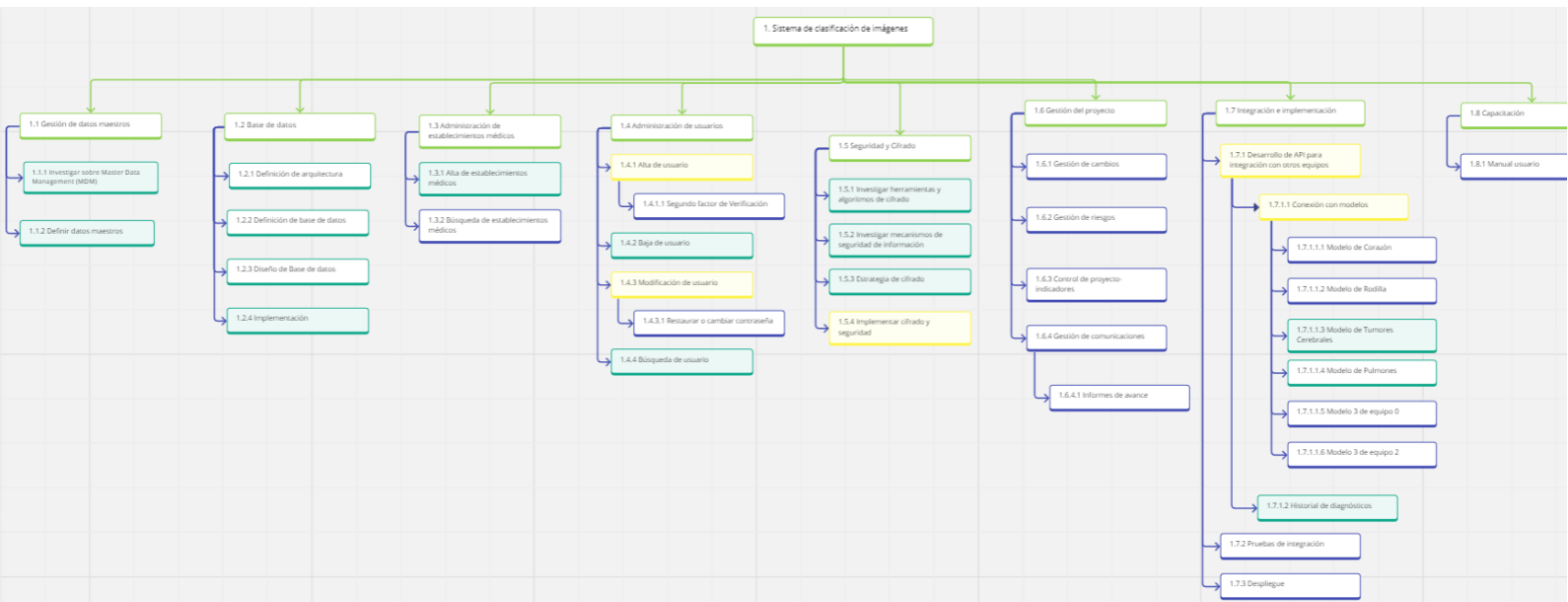
- Definición de roles y permisos de usuario para garantizar el acceso adecuado a la información.
- Restricción de acceso a datos cifrados solo a usuarios autorizados.
- Restricción de acceso a la API solo a usuarios autorizados.

Requerimientos No Funcionales

1. Seguridad

- Cumplimiento de los estándares de cifrado y seguridad para proteger la información de los usuarios y pacientes médicos.

WBS



La WBS se puede observar en más detalle ingresando al siguiente link: [WBS | Miro](#)

Diccionario

ID	NOMBRE	DESCRIPCIÓN	TAREAS
1	Sistema Clasificación de Imágenes	Proyecto de clasificación de imágenes.	Clasificar imágenes médicas para asistir a la medicina en el diagnóstico.
1.1	Gestión de datos maestros	Producto encargado de la gestión de datos maestros.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.1.1	Investigar sobre Master Data Management (MDM)	Producto encargado de la investigación de MDM.	Investigar sobre Master Data Management (MDM).
1.1.2	Definir datos maestros	Producto encargado de analizar y definir datos maestros.	Analizar y definir datos maestros.

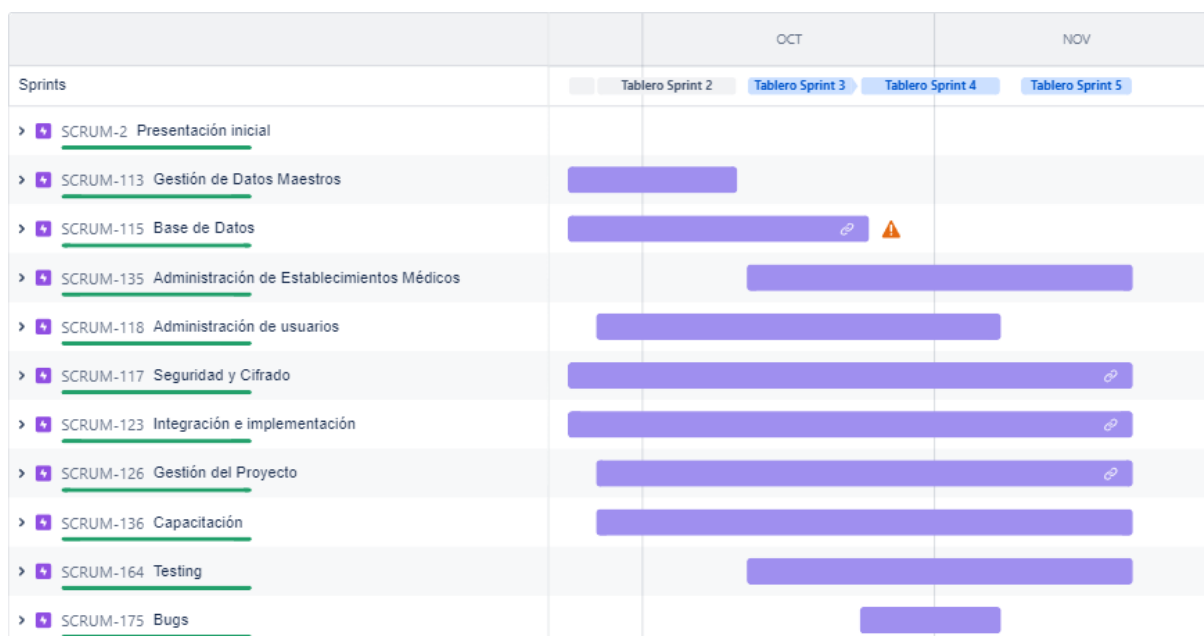
1.2	Base de datos	Producto encargado del modelado de una entidad y de todas las operaciones que lo involucran.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.2.1	Definición de arquitectura	Producto encargado de la definición de la arquitectura de la base de datos.	Definir la arquitectura e infraestructura donde se va a hostear la base de datos.
1.2.2	Definición de base de datos	Producto encargado de la definición de la base de datos a utilizar.	Analizar las necesidades para decidir si usar una base de datos relacional, no relacional, y qué tecnologías.
1.2.3	Diseño	Producto encargado de diseñar la base de datos.	Realizar un DER de la base de datos con las relaciones, entidades y sus datos necesarios.
1.2.4	Implementación	Producto encargado de la implementación de la base de datos.	Implementar la base de datos diseñada utilizando las tecnologías definidas.
1.3	Administración de entidades médicas	Producto encargado del modelado del médico y de todas las operaciones que lo involucran.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.3.1	Alta de entidad médica	Producto encargado de las operaciones de alta de una entidad médica.	Análisis, Diseño, Implementación y pruebas unitarias y modular.
1.3.2	Búsqueda de entidad médica	Producto encargado de las operaciones de búsqueda de una entidad médica.	Análisis, Diseño, Implementación y pruebas unitarias y modular.
1.4	Administración de usuarios	Producto encargado del modelado del usuario y de todas las operaciones que lo involucran.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.4.1	Alta de usuario	Producto encargado de las operaciones de alta de un usuario.	Análisis, Diseño, Implementación y pruebas unitarias y modular.
1.4.1.1	Segundo factor de Verificación	Producto encargado de verificar al usuario.	Enviar mail al usuario, con un código de verificación para poder gestionar claves.
1.4.2	Baja de usuario	Producto encargado de las operaciones de baja de un usuario.	Análisis, Diseño, Implementación y pruebas unitarias y modular.
1.4.3	Modificación de usuario	Producto encargado de las operaciones de modificación de un usuario.	Análisis, Diseño, Implementación y pruebas unitarias y modular.

1.4.3.1	Restaurar/cambiar contraseña	Producto encargado del manejo en el cambio de contraseñas	Enviar mail al usuario, con un código de verificación para poder gestionar claves.
1.4.4	Búsqueda de usuario	Producto encargado de las operaciones de búsqueda de un usuario.	Análisis, Diseño, Implementación y pruebas unitarias y modular.
1.5	Seguridad y cifrado	Producto encargado de la seguridad y el cifrado de datos.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.5.1	Investigar herramientas y algoritmos de cifrado	Producto encargado de investigar herramientas y algoritmos de cifrado.	Investigar herramientas y algoritmos de cifrado para decidir qué utilizar en el proyecto.
1.5.2	Investigar mecanismos de seguridad de información	Producto encargado de investigar mecanismos de seguridad de información.	Investigar mecanismos de seguridad de información como reglas ,estándares, tecnologías.
1.5.3	Estrategia de cifrado	Producto encargado de definir la estrategia de cifrado de los datos.	Definir estrategia de cifrado, dónde se va a implementar, de qué manera, con que tecnología, utilizando lo investigado previamente.
1.5.4	Implementar cifrado y seguridad	Producto encargado de la implementación del cifrado.	Implementar el cifrado donde sea necesario siguiendo la estrategia y las tecnologías decididas.
1.6	Gestión del proyecto	Producto encargado de la administración del proyecto.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.6.1	Gestión de cambios	Producto encargado de planificar la gestión de cambios.	Gestionar los cambios a medida que suceden.
1.6.2	Gestión de riesgos	Producto encargado de planificar la gestión de riesgos.	Planificar la gestión de riesgos, Identificar los riesgos, Analizar riesgos. Planificar la respuesta a los riesgos, Controlar los riesgos.
1.6.3	Control de proyecto - Indicadores	Proyecto encargado de controlar la correcta gestión del proyecto en base a los distintos indicadores.	Revisar indicadores, retrasos y/avances de las actividades.
1.6.4	Gestión de comunicaciones	Producto que incluye los procesos requeridos para garantizar que la generación, la recopilación, la distribución, el almacenamiento, la recuperación y la disposición final de la información del proyecto sean adecuados y oportunos.	Gestionar las comunicaciones del proyecto, Controlar las comunicaciones.

1.6.4.1	Informes de avance	Producto encargado de la creación de informes cuyo objetivo es informar sobre los avances realizados en el proyecto.	Realizar reuniones o puntos de control para ver avances y realizar el informe.
1.7	Integración e implementación	Producto encargado del modelado de la integración e implementación del proyecto.	Integración de las sub-categorías que lo componen y pruebas integrales.
1.7.1	Desarrollo de API para integración con otros equipos	Producto encargado del desarrollo de servicios para realizar la interacción con los demás componentes del sistema.	Desarrollar API que se encargue de las funciones de conexión.
1.7.1.1	Conexión con modelos	Producto encargado de implementar la conexión con los modelos.	Integración de la API con los modelos.
1.7.1.1.1	Modelo de Corazón	Producto encargado de realizar predicciones de enfermedades cardíacas.	Integración de la API con el modelo de corazón.
1.7.1.1.2	Modelo de Rodilla	Producto encargado de realizar predicciones de enfermedades de rodilla.	Integración de la API con el modelo de rodilla.
1.7.1.1.3	Modelo de Tumores Cerebrales	Producto encargado de realizar predicciones de tumores cerebrales.	Integración de la API con el modelo de tumores cerebrales.
1.7.1.1.4	Modelo de Pulmones	Producto encargado de realizar predicciones de neumonía.	Integración de la API con el modelo de neumonía.
1.7.1.1.5	Modelo 3 de equipo 0	Producto encargado de realizar predicciones de enfermedad indefinida.	Integración de la API con el modelo 3 del equipo 0.
1.7.1.1.6	Modelo 3 de equipo 2	Producto encargado de realizar predicciones de enfermedad indefinida.	Integración de la API con el modelo 3 del equipo 2.
1.7.1.2	Historial de diagnósticos	Producto encargado de visualizar los resultados correspondientes al usuario que solicita el historial.	Diseño, implementación y permisos de historial.
1.7.2	Pruebas de integración	Producto encargado de la comprobación de que elementos del software que interactúan entre sí, funcionan de manera correcta.	Pruebas de integración de todos los componentes y funcionalidades del sistema.

Calendario

Realizado en Jira, plataforma que utilizamos para la asignación y el control de tareas:



El calendario se puede observar en más detalle ingresando al siguiente link:

[Calendario | Jira](#)

Estimaciones

En nuestras estimaciones iniciales del proyecto consideramos que cada integrante del equipo trabajaría un aproximado de dos horas diarias, por lo que se realizó el siguiente cálculo; *Cantidad de integrantes del equipo * Cantidad de días del Sprint * 2*

Hicimos este cálculo para los cinco Sprints de este proyecto y, de esta forma, poder asignar un total de tareas adecuado a cada Sprint teniendo en cuenta sus story points estimados.

El resultado final de la suma de todos los Sprints fue de 784 story points estimados.

Al final del proyecto, se registraron 925 horas reales.

Hubo una diferencia de 141 horas.

Matriz de riesgos

	Descripción	Prob. de ocurrencia	Impacto del riesgo	Exposición al riesgo	Acciones para mitigarlo	Contingencia	Responsable
1	Fallos en la integración con modelos y/o UX	1	3	3	Realizar pruebas de integración y todos los tests necesarios.	Reportar y asignar bug al responsable de la tarea para su resolución.	Gómez, Federico (Tester)
2	Modificación de endpoints	1	3	3	Estimar las tareas de endpoints teniendo en cuenta posibles modificaciones.	Designar al responsable de dicha tarea para realizar la modificación necesaria.	Perez, Giannina (SM)
3	Falta de conocimiento y experiencia sobre las tareas asignadas	1	2	2	Capacitación de la herramienta que se utilizará en el siguiente Sprint.	El responsable ayudará a capacitar al integrante asignado a dicha tarea.	Perez, Giannina (SM)
4	Pérdida de datos maestros	1	3	3	Realizar copias de seguridad semanalmente.	Restauración de la copia de seguridad.	Clauser, Nahuel (Dev)
5	Problemas en la API	1	2	2	Deployar una API auxiliar que funcione igual a la principal.	Utilizar API auxiliar.	Prieto, Lucas (LT)
6	Plazos incumplidos	1	2	2	Estimar fechas de entrega para cada tarea.	Priorizar la finalización de la tarea para que la brecha entre la fecha estimada y la fecha real no sea demasiado grande.	Perez, Giannina (SM)

7	Caída o problemas de la BBDD	1	2	2	Mantenimiento del equipo utilizado para la BBDD.	Ver los logs de la bbdd para encontrar el error de conexión. Si no se encuentra, utilizar la alternativa levantada en Render.	Clauser, Nahuel (Dev)
8	Falta de pruebas adecuadas	1	2	2	Diseñar un plan de pruebas y realizar la cantidad de tests necesarios.	Reasignar las tareas de testing para hacer testing cruzado.	Gómez, Federico (Tester)
9	Problemas de comunicación con los demás equipos	2	1	2	Mantener comunicación constante con los SMs y LTs de los demás equipos.	El responsable se encargará de gestionar la comunicación para resolver el problema que surja.	Perez, Giannina (SM)
10	Errores de validación de JWT	1	1	1	Documentar manual de usuario.	Ver manual de usuario.	Torrico, Franco (Dev)

Hitos

Los Hitos nos permitirán realizar un seguimiento del progreso, verificar que los componentes claves del proyecto se implementen con éxito y garantizar la alineación con los objetivos del proyecto.

Hito de infraestructura de almacenamiento seguro de datos

Hito: Implementación exitosa de la infraestructura de almacenamiento seguro de datos.

Criterios de éxito: el sistema de almacenamiento seguro de datos está operativo y puede almacenar de forma segura conjuntos de datos de imágenes médicas.

Hito de gestión de datos

Hito: Implementación de políticas y procedimientos de gestión de datos.

Criterios para el éxito: existen prácticas de gestión de datos, que incluyen validación, duplicación y controles de calidad.

Hito de implementación del cifrado

Hito: Implementación de técnicas de cifrado para la protección de datos.

Criterios de éxito: el cifrado se aplica con éxito a los datos en tránsito y en reposo, lo que garantiza la seguridad de los datos médicos.

Hito de gestión de datos maestros (MDM)

Hito: Establecimiento del proceso MDM.

Criterios de éxito: las políticas y directrices de MDM se definen e implementan para garantizar la coherencia y la calidad de los datos.

Hito de implementación de seguridad de datos

Hito: Implementación de medidas de seguridad de datos.

Criterios para el éxito: existen controles de acceso, mecanismos de autenticación, auditoría y herramientas de monitoreo para evitar el acceso no autorizado y proteger contra riesgos cibernéticos.

Hito de cumplimiento de privacidad de datos

Hito: Garantizar el cumplimiento de la normativa de privacidad de datos.

Criterios de éxito: el proyecto cumple con las leyes de privacidad de datos pertinentes.

Administración en el Manejo de Bugs

Para lograr una mejor gestión sobre los errores que se van presentando en el proyecto utilizaremos la funcionalidad de Jira en la cual podremos generar reportes específicos y vincularlos a las funcionalidades correspondientes. Para ello, tendremos en cuenta los siguientes:

1. **Registro de Bugs:** Inicia el proceso registrando todos los bugs detectados de manera detallada en Jira. Esto incluye descripciones precisas, pasos para reproducir el problema y, si es posible, capturas de pantalla.
2. **Priorización:** Clasifica los bugs según su gravedad y su impacto en el proyecto. Se utilizarán los campos de prioridad en Jira para asignar niveles de urgencia y gravedad.
3. **Asignación de Responsabilidades:** Asigna cada bug a un miembro del equipo responsable de su resolución.
4. **Seguimiento y Comentarios:** Para llevar un registro del estado de cada bug y fomentar la colaboración. Cada miembro del equipo puede agregar comentarios, actualizaciones y notas relevantes en cada bug.
5. **Documentación de Soluciones:** Una vez que se resuelve un bug, documenta la solución en Jira. Esto crea un registro histórico útil y facilita la revisión de problemas similares en el futuro.
6. **Cierre y Análisis:** Cuando un bug se ha resuelto de manera satisfactoria, se cerrará y esto será de utilidad para identificar las causas raíces y prevenir problemas similares en proyectos futuros.

Gestión de Cambios

Notamos que nuestra primer idea sobre la gestión de cambios era muy poco ágil, pues demandaba muchas instancias para actualizar el expediente que contenía los siguientes campos:

ID petición	Solicitante	Resumen de la petición	Impacto Esperado	Estado	Impacto Real
-------------	-------------	------------------------	------------------	--------	--------------

Al día de hoy, al momento de presentar una solicitud de cambio por parte de un integrante del equipo, esta petición se discutirá con los desarrolladores, el tester, el Líder Técnico y la Scrum Master como

primera instancia de aprobación. Si se llega a un acuerdo, el próximo paso será comunicar el cambio al Product Owner por contacto directo.

Si el cambio es aprobado por el Product Owner, se eliminará el elemento relacionado a la tarea anterior y, por otro lado, se agregará un nuevo elemento en el backlog relacionado a esta nueva tarea, definiendo posteriormente sus Story Points y prioridad. Si el cambio es rechazado por el Product Owner, se descarta la idea o se modifica en caso de ser necesario.

Por otro lado, durante la Sprint Review puede ocurrir que se propongan cambios por parte del Product Owner, en ese caso se discutirá y se evaluará el cambio en el momento, para luego modificar el backlog como se mencionó anteriormente.

Indicadores a utilizar

- **Burndown Chart:** Ayudará al equipo a rastrear y gestionar eficazmente el progreso en la resolución de problemas, lo que contribuye a la calidad del software y al cumplimiento de los objetivos del proyecto, así también como una herramienta adicional para ver el progreso del desarrollo.
- **Evolución de la prueba:** Servirá para llevar un control de los bugs encontrados durante los Sprints. Este indicador tendrá en cuenta la cantidad de bugs encontrados, cerrados, pendientes y cancelados.
- **Historias terminadas:** Este indicador nos ayudará a considerar la cantidad de tareas terminadas críticas (3), esenciales (2) y deseables (1).
- **Horas imputadas:** A partir de un presupuesto fijo, se gestionará la cantidad de horas imputadas por roles. Gracias a este indicador, podremos obtener un acercamiento a lo que debería costar nuestras horas trabajadas y en general, el proyecto entero.
- **Horas trabajadas por integrante:** Este indicador se agregó para tener datos reales sobre lo que trabaja cada miembro del equipo, y en relación a ello, enfocarse en las diferencias para que éstas se reduzcan.
- **Índice de Mitigación:** Se agregó este indicador para mostrar resultados efectivos de las medidas de mitigación generadas, medirlos y reflejar el nivel de exposición inicial y el nivel de exposición reducido.
- **Índice de Probabilidad:** En gestión de riesgo, este indicador se agregó para indicar por semana el nivel de riesgo, el cual se refleja de la suma de exposición por riesgo en semana.
- **Nivel de Riesgo:** Se utilizará para evaluar y cuantificar el grado de incertidumbre o posibles problemas que pueden afectar al proyecto. Contribuye a una mayor calidad en la gestión del proyecto y a la reducción de problemas que puedan afectar la calidad final del entregable.

Tecnologías

- Flask: Utilizaremos Flask para desarrollar una API Web en el caso de necesitar disponibilizar datos. Además Flask tiene funcionalidades para conectar con bases de datos, de donde podremos obtener los datos normalizados.
- JIRA: Para la asignación, seguimiento y control de las tareas a realizar.
- Google Drive: Lugar donde estará la carpeta “Datacrypt” con toda la documentación realizada durante el proyecto.
- GitHub: Como repositorio de código para nuestros desarrollos.
- Algoritmo AES-128 (Advanced Encryption Standard): Utilizaremos este algoritmo para realizar el cifrado de datos sensibles en la base de datos.
- OWASP ZAP: Herramienta que se utilizará para probar las vulnerabilidades del sistema.

Herramientas y Repositorios

[Datacrypt | Google Drive](#)

[Repositorio Datacrypt | GitHub](#)

[Repositorio API | GitHub](#)

[API | Render](#)