**Universidad Nacional General Sarmiento**

**Laboratorio de Construcción de Software**

**Proyecto: “Master Security System (MSS)”**

1 semestre – Año 2024



**Docentes:**

Ing. Juan Carlos Monteros – [jcmonteros@campus.ungs.edu.ar](mailto:jcmonteros@campus.ungs.edu.ar)

Evelin Aragon – [eve\_aragon@hotmail.com](mailto:eve_aragon@hotmail.com)

**Integrantes del equipo “The Big 6 + 1 Software”:**

Fallatti Franco – [franckfallatti@gmail.com](mailto:franckfallatti@gmail.com)

Castillo Patricio – [patriciojcastillo@gmail.com](mailto:patriciojcastillo@gmail.com)

Gross Pablo Ruben – [pablorubengross@gmail.com](mailto:pablorubengross@gmail.com)

Cañete Ezequiel David – [daviddanielc25@gmail.com](mailto:daviddanielc25@gmail.com)

Hernández Facundo – [facujhernandez46@gmail.com](mailto:facujhernandez46@gmail.com)

Lombardi Lautaro – [lautaro.lombardi.gian@gmail.com](mailto:lautaro.lombardi.gian@gmail.com)

Gonzales Federico – [federico.l.g8195@gmail.com](mailto:federico.l.g8195@gmail.com)

**Repositorio:**

<https://github.com/LautyLombardi/TrabajoPracticoPrincipal>

1. Introducción:

Objetivo del Proyecto:

El objetivo de este trabajo práctico y por ende de este proyecto es que los estudiantes apliquen los conocimientos adquiridos en la asignatura sobre Metodologías Ágiles y Waterfall en el desarrollo de un sistema de autenticación por reconocimiento facial para una Tablet, permitiendo el ingreso a áreas con seguridad. La Tablet captura una serie de imágenes del rostro desde diferentes ángulos y utiliza estos datos para crear un patrón único. Este patrón se almacena en el dispositivo y se utiliza para compararlo con futuras capturas faciales cuando intentamos desbloquear la Tablet.

Objetivo del Documento:

El objetivo de este documento es dar una clara muestra y explicación de cómo se va a llevar a cabo la planificación y desarrollo, como lo vamos a organizar, a construir, a testear entre otras actividades. También tiene como objetivo ser un control de cómo se va desarrollando el proyecto a lo largo de su transcurso, gracias a los indicadores que se agregaran.

1. Equipo de trabajo:

El equipo de trabajo se compone de 7 personas, el scrum master el cual también cumple el rol de analista funcional al que llamaremos “Master de proyecto”, 4 desarrolladores con un líder de desarrollo el cual tendrá conocimientos full stack. Un desarrollador full stack, y otro desarrollador que será también administrador de bases de datos, además contamos con un desarrollador especializado en el backend. Por último, 2 testers los cuales se encargarán de probar todo el producto.

Roles:

* Lombardi Lautaro: Master de proyecto
* Fallatti Franco: Líder de desarrollo
* Castillo Patricio: Desarrollador y administrador de base de datos
* Gross Pablo: Desarrollador backend
* Cañete Ezequiel David: Desarrollador full stack
* Hernández Facundo: Tester
* Gonzales Federico: Tester

El principal objetivo de esta distribución de roles es lograr una mayor eficiencia en la administración de tareas.

Además de estos roles principales, se asignaron roles secundarios para en caso de una emergencia poder agregar personal para ayudar en el área de la misma.

1. Metodología:

La metodología que decidimos usar para el proyecto es la metodología ágil y en específico la Scrum, ya que consideramos que una metodología iterativa incremental es la que mejor se adapta a nuestra forma de trabajo, así como también a los plazos y entregas que tendremos que realizar y respetar a lo largo del desarrollo del proyecto.

Lo que realizaremos será un sprint 0 donde al ser un equipo nuevo definiremos todos los roles, herramientas, estándares y organización del proyecto, además de definir el plan de proyecto. También haremos una estimación y calendarización inicial del mismo.

Luego en sprints los cuales duraran dos semanas realizaremos todas las funcionalidades que se contengan en ese sprint. Los días sábados/domingos realizaremos un sprint Weekly donde analizaremos el trabajo realizado esa semana y diremos que complicaciones encontramos, que funcionalidades completamos y se hará un control de todas las estimaciones y riesgos.

Además de estas reuniones, tendremos los martes y jueves reuniones “Retro-Review” de consulta con los clientes. En estas reuniones lo que haremos será sacarnos las dudas que vayan surgiendo con respecto a diferentes aspectos del producto, para luego poder refinar los requerimientos o en todo caso agregar/quitar requerimientos.

El resto de reuniones como por ejemplo la daily no las realizaremos por falta de tiempo, y por qué no las vemos tan necesarias al ser sprints de dos semanas, y ya tener una reunión semanal, así como las reuniones de consulta con los clientes.

1. Herramientas:

Para la realización del proyecto usaremos las siguientes herramientas para facilitarnos realizar ciertas tareas.

* Git con GitHub para realizar el control de versiones del proyecto
* Python, Java, JavaScript, CSS, React-native como lenguajes en los que se programara
* SQLite como base de datos donde se guardarán las imágenes de cada persona y otra información
* Springboot para la realización de microservicios en el sistema
* Word, Excel y PowerPoint como herramientas para el control, administración y progreso del proyecto
* Trello como sistema para llevar a cabo la metodología Scrum
* Visual Studio Code como IDE de desarrollo
* Tensor Flow como una de las librerías principales para el desarrollo del reconocimiento facial
* Jest, Junit5 y Robot Framework para el testeo del proyecto
* Draw.io para diagramas
* Figma como sistema de diseño de interfaces
* Google Collab como editor online para el desarrollo de modelos IA
* Whatsapp y Discord como herramientas de comunicación internas

1. Visión de negocio:

Nuestro objetivo con este producto es lograr un sistema de seguridad por reconocimiento facial, el cual sea confiable para los usuarios del mismo, el cual sea de fácil uso e intuitivo para los usuarios primerizos. Así también buscamos que la aplicación use de forma eficiente los recursos del dispositivo.

Este producto busca satisfacer las necesidades de autenticación de distintos usuarios a distintas áreas de forma confiable, rápida y segura. Además de permitir al usuario interactuar con o sin internet con el sistema

1. Alcance:

Definiremos el alcance con la condición de satisfacción del usuario relacionado con los requerimientos del sistema.

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | **Responsabilidad del sistema** | **No contemplado por el sistema** |
| Inicio y cierre de sesión | El sistema permitirá a los usuarios iniciar/cerrar sesión por medio de reconocimiento facial, para luego entrar a ver su pantalla correspondiente | El sistema no podrá iniciar/cerrar sesión, por medio del reconocimiento facial sin internet |
| Manejo de imágenes | El sistema podrá capturar, identificar y comparar imágenes capturadas con la cámara frontal del dispositivo. Además, tendrá la posibilidad de mostrarle la captura al usuario | El sistema no tendrá capacidad de edición de las imágenes |
| Autenticación de personas | El sistema sabrá identificar cual persona es la que intenta autenticarse. Estas personas, que están previamente registradas en la base de datos, tendrán una respuesta de acceso positivo o denegado | El sistema no tendrá la capacidad de diferenciar si la persona a autorizar es una persona real o una foto de esa persona |
| Log de ingresos/egresos | El sistema será capaz de llevar a cabo un log de intentos de ingreso/egreso en el sistema, el cual podrán ver solamente los usuarios catalogados como seguridad de accesos | El sistema no tendrá la capacidad de borrar algún registro del log. |
| Gestión de visitantes | El sistema podrá dar de alta nuevos visitantes, permitiendo a los usuarios administradores de seguridad institucional modificar datos internos de los mismos o incluso borrar visitantes | El sistema no tendrá la capacidad de permitir a los usuarios que no sean administradores de seguridad institucional editar algún dato interno |
| Funcionamiento fuera de la conexión a internet | El sistema será capaz se seguir en funcionamiento fuera de la conexión a internet, teniendo una forma de autenticación offline. Además de poder guardar toda la información que obtenga, para una sincronización posterior | El sistema no tendrá la capacidad funcionar al 100% de sus capacidades, como por ejemplo las autenticaciones de visitantes se guardarán en el log una vez recuperada la conexión |
| Gestión de lugares físicos | El sistema podrá dar de alta, modificar o eliminar lugares físicos a los cuales los visitantes ingresaran una vez autenticados | El sistema no podrá eliminar lugares físicos mientras el día este abierto |
| Inicio/Cierre del día | El sistema tendrá la capacidad de dar inicio al día para empezar a autenticar a los visitantes, a su vez también podrá darlo por cerrado al mismo | Una vez cerrado el día el sistema no podrá autenticar personas |
| Autorización de visitantes | El sistema permitirá a los usuarios administradores de seguridad institucional autorizar que tipo de visitante puede ir a cada lugar físico, como también desautorizarlo | El sistema no podrá rastrear al visitante para verificar que realmente fue al lugar que debería |
| Gestión de usuarios | El sistema permitirá a los usuarios de tipo administración de seguridad institucional dar de alta y dar de baja nuevos usuarios del sistema | Los demás tipos de usuarios no podrán dar de alta o de baja a usuarios |
| Generación de excepciones | El sistema podrá asignar excepciones que ocurrirán en los diferentes lugares físicos, permitiendo que otros tipos de visitantes pudieran ir a los mismos durante esa excepcion | El sistema no contempla la carga de excepciones mientras que el día este abierto |

1. Plan de comunicaciones:

El equipo se reunirá formalmente con el product owner cada 2 semanas, donde realizará una presentación con los avances realizados hasta el momento, a su vez se mostrará por medio de indicadores como es ese avance presentado. En estas reuniones formales se realizarán minutas de reunión la cual será un documento que mostrará de forma ordenada lo hablado en la misma.

Para estas reuniones se entregará un informe de avances el cual tendrá todas las funcionalidades completadas hasta el momento, así como los problemas y/o observaciones encontradas por cada responsable. Este documento contendrá el indicador de Funcionalidad Completa + el de Nivel de calidad, el cual en cada informe de avance se ira actualizando a partir del anterior.

A su vez los martes y jueves como se especificó en la metodología que utilizaremos, nos reuniremos con el cliente para consultar aspectos sobre el producto y/o el desarrollo del proyecto.

**ACLARACION: PARA CADA REUNION FORMAL SE ENTREGARÁ EN EL MOODLE LAS TRES PLANILLAS DE EXCEL, TANTO LA DE CAMBIOS, COMO LA DE BUGS, COMO LA DE RIESGOS. ESTAS SERAN ADJUNTADAS A LA ENTREGA EN CONJUNTO CON ESTE DOCUMENTO Y LAS ACTUALIZACIONES DEL MISMO**

Cualquier contacto externo con algún miembro del equipo deberá realizarse a través de los mails correspondientes que se indicaron al inicio de este documento, o por canales de comunicación típicos, como por ejemplo Whatsapp.

1. Requerimientos:

Listado de Requerimientos:

Notación: Catalogamos los requerimientos Funcionales, por un lado, y los Requerimientos no funcionales por otro. Clasificamos cada requerimiento como Esencial (E), Importante (I), Deseable (D).

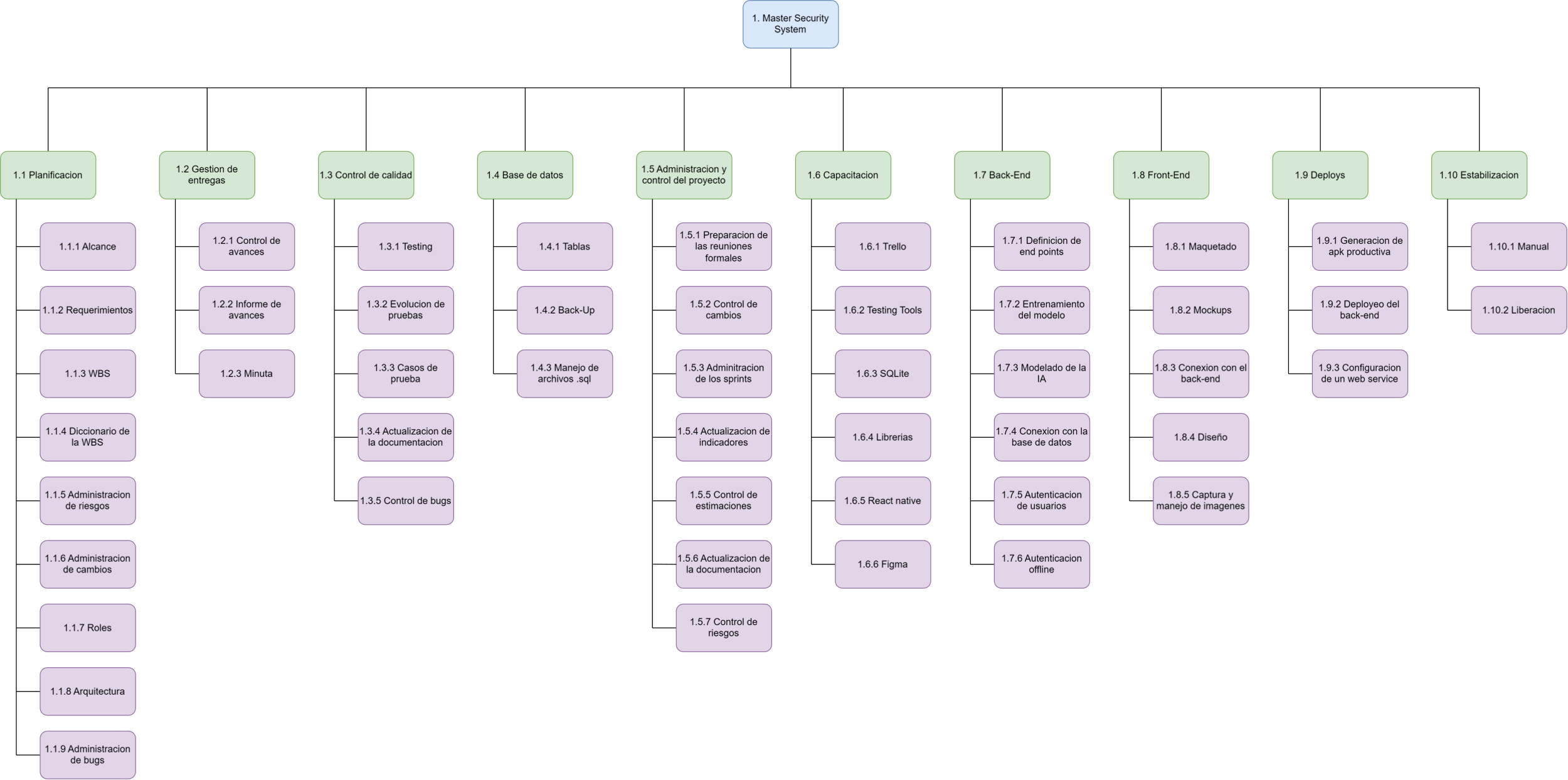
Requerimientos Funcionales:

* RFE1: El sistema debe ser capaz de capturar imágenes de la cámara frontal de la Tablet.
* RFE2: El sistema debe ser capaz de detectar rostros en las imágenes capturadas.
* RFE3: El sistema debe ser capaz de identificar a las personas a partir de sus rostros.
* RFE4: El sistema debe ser capaz de autenticar a los visitantes al momento de ingresa o egresar del lugar, comparando sus rostros con una base de datos de rostros conocidos almacenada en la Tablet.
  + Si no hay conexión a internet el sistema debe ser capaz de autenticar a los visitantes por medio del ingreso de datos predefinidos para cada tipo de visitante.
  + El sistema no permitirá autenticar visitantes si el día ya fue cerrado.
* RFI5: El sistema debe mostrar un mensaje de "Acceso Permitido" en verde o "Acceso Denegado" en rojo en la pantalla de la Tablet en función del resultado de la autenticación. A su vez también mostrara los lugares a los que puede ingresar el visitante en caso de ser un “Acceso Permitido”.
* RFD6: El sistema deberá registrar un log con los datos de ingresos (Fecha, Hora, ID de persona, Nombre de la persona, Nombre del usuario que lo autentico, Acceso permitido o denegado, ingreso o egreso, automático o manual, lugares permitidos).
  + Este log lo verán solamente los usuarios autorizados.
  + Los logs se registrarán en la base de datos.
* RFI7: El sistema deberá permitir una alternativa manual de ingreso ante posibles desconexiones (sin Wifi o datos).
  + Si no hay internet al momento de iniciar/cerrar la sesión en la aplicación, se pedirá el usuario y clave para realizarlo.
  + Si no hay internet al momento de autenticar a un visitante, se lo realizara de forma manual ingresando todos los datos requeridos para ese tipo de visitante.
* RFE8: El sistema deberá permitir el ALTA/MODIFICACIONES/BAJA de Los visitantes a autenticar. Los datos que se le pedirán a cada visitante para registrarlos dependerán de a que categoría pertenece ese visitante. Sin importar que categoría tenga, se le pedirá registrarle el rostro para las autenticaciones por medio de reconocimiento facial.
  + Al querer dar de baja se le pedirá su clave al usuario que lo quiere realizar
  + En la ventana de el registro de un visitante, también deberá haber un pop up que te lleve a la de registro de un nuevo lugar físico/categoría para agilizar el sistema
* RFE9: El sistema deberá permitir el ALTA/MODIFICACIONES/BAJA de los lugares físicos a donde los diferentes tipos de visitantes podrán ingresar.
  + Para registrar un nuevo lugar se pedirá el nombre del lugar, los tipos de visitantes que pueden ingresar, abreviatura, descripción del lugar y horario.
  + El sistema no permitirá que se modifiquen/bajen los lugares físicos mientras el día este considerado como abierto.
* RFI10: El sistema permitirá la carga de excepciones en los lugares físicos, las cuales permitirán que visitantes no autorizados a ese lugar, puedan ingresar al mismo durante su duración.
  + Las excepciones tendrán un nombre, una descripción, lugar donde se realiza, una duración y que tipos de visitantes podrán ingresar.
  + El sistema no permitirá la carga de excepciones mientras el día este considerado como abierto.
* RFE11: El sistema deberá permitir el ALTA/MODIFICACIONES/BAJA de usuarios en el sistema.
  + Al registrarlo se le pedirá nombre, DNI, clave, tipo de usuario y registrar su rostro para el ingreso con reconocimiento facial a la app.
  + Al querer dar de baja se le pedirá su clave al usuario que lo quiere realizar
* RFE12: El sistema deberá poder dar por abierto el día, guardando la fecha y hora en la que se lo realiza, y a su vez deberá poder dar por cerrado un día, también registrando la fecha y hora en la que se lo da por cerrado.
* RFE13: El sistema deberá permitir autorizar a los diferentes tipos de visitantes el ingreso a distintos lugares físicos.
* RFE14: El sistema deberá permitir el ALTA/MODIFICACIONES/BAJA de tipos de visitantes en el sistema. De estas categorías/tipos se registrará el nombre y la descripción del mismo.
* RFI15: El sistema deberá poder sincronizar todos los datos que se obtengan durante el funcionamiento sin conexión, con las bases de datos una vez recuperada la conexión.

Requerimientos No Funcionales:

* RNFI16: Eficiencia: El sistema debe ser eficiente en el uso de la batería, la memoria y el procesador de la Tablet. A su vez debe ser rápido en la autenticación de los visitantes.
* RNFD17: Usabilidad: El sistema debe ser fácil de usar para usuarios con diferentes niveles de experiencia en el uso de tabletas. Colocando ayudas (iconitos “?”) en las diferentes pantallas los cuales explicaran de forma resumida como funciona esa misma pantalla.

1. WBS:



10- Diccionario de la WBS:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Código de paquete** | **Nombre de paquete** | **Descripción de trabajo** | **Responsables** | **Estimación en horas** | **Fecha de completitud** |
| 1.0 | Master Security System (MSS) | Desarrollo completo del sistema MSS | Todos los integrantes del equipo | 375 |  |
| 1.1 | Planificación | Tareas que involucran la planificación de tareas previas al desarrollo del sistema | Lombardi Lautaro | 26 |  |
| 1.1.1 | Alcance | Definición del alcance del proyecto | Lombardi Lautaro | 2 | 20/4/2024 |
| 1.1.2 | Requerimientos | Definición de los requerimientos del proyecto | Lombardi Lautaro | 2 | 20/4/2024 |
| 1.1.3 | WBS | Realización del diagrama WBS | Lombardi Lautaro | 3 | 20/4/2024 |
| 1.1.4 | Diccionario de la WBS | Realización del diccionario del diagrama WBS | Lombardi Lautaro | 6 | 23/4/2024 |
| 1.1.5 | Administración de riesgos | Definición de la administración de riesgos | Lombardi Lautaro | 3 | 20/4/2024 |
| 1.1.6 | Administración de cambios | Definición de la administración de cambios | Lombardi Lautaro | 3 | 20/4/2024 |
| 1.1.7 | Roles | Definición de los roles del equipo de trabajo | Lombardi Lautaro | 2 | 23/4/2024 |
| 1.1.8 | Arquitectura | Realización del diagrama de arquitectura | Lombardi Lautaro | 5 |  |
| 1.1.9 | Administración de bugs | Definición de la administración de bugs | Lombardi Lautaro | 3 | 20/4/2024 |
| 1.2 | Gestión de entregas | Realización de todo lo necesario para las reuniones formales con el cliente | Lombardi Lautaro | 26 |  |
| 1.2.1 | Control de avances | Reunión de todos lo miembro del equipo para controlar los avances realizados | Lombardi Lautaro | 10 |  |
| 1.2.2 | Informe de avances | Realización de los informes de avances para las reuniones formales con su indicador de Funcionalidad completa + nivel de calidad | Lombardi Lautaro | 10 |  |
| 1.2.3 | Minuta | Realización de las minutas después de cada reunión formal | Lombardi Lautaro | 6 |  |
| 1.3 | Control de calidad | Tareas relacionadas con el control de calidad del producto | Hernández Facundo, Gonzales Federico | 61 |  |
| 1.3.1 | Testing | Realización de las pruebas establecidas para verificar que el sistema cumpla adecuadamente con las funcionalidades pautadas | Hernández Facundo | 24 |  |
| 1.3.2 | Evolución de pruebas | Actualización de los indicadores de evolución de la prueba + cobertura de la prueba | Hernández Facundo | 4 |  |
| 1.3.3 | Casos de prueba | Se realiza el documento de casos de prueba para testear | Gonzales Federico | 24 |  |
| 1.3.4 | Actualización de la documentación | Se realizan modificaciones a los documentos del proyecto según corresponda | Hernández Facundo | 3 |  |
| 1.3.5 | Control de bugs | Se lleva a cabo el control de bugs para revisar que se esté respetando la forma de trabajo | Hernández Facundo | 6 |  |
| 1.4 | Base de datos | Tareas relacionadas con las bases de datos del sistema | Castillo Patricio | 25 |  |
| 1.4.1 | Tablas | Realización de todas las tablas del sistema | Castillo Patricio | 16 |  |
| 1.4.2 | Back-Up | Generación de la copia de seguridad de la base de datos del sistema | Castillo Patricio | 6 |  |
| 1.4.3 | Manejo de archivos .sql | Almacenamiento y creación de la base por medio de archivos .sql | Castillo Patricio | 3 |  |
| 1.5 | Administración y control del proyecto | Tareas relacionadas con la administración y el control del correcto avance del proyecto | Lombardi Lautaro | 38 |  |
| 1.5.1 | Preparación de reuniones formales | Se planea quien va a presentar y que se va a presentar en la reunión formal | Lombardi Lautaro | 4 |  |
| 1.5.2 | Control de cambios | Se lleva a cabo el control de cambios para revisar que se esté respetando la forma de trabajo | Lombardi Lautaro | 6 |  |
| 1.5.3 | Administración de los sprints | Se establece como va a trabajar el equipo en el siguiente sprint y se actualiza el indicador de Burdown chart | Lombardi Lautaro | 8 |  |
| 1.5.4 | Actualización de indicadores | Se hace una actualización de todos los indicadores | Lombardi Lautaro | 3 |  |
| 1.5.5 | Control de estimaciones | Se hace una revisión de todas las estimaciones para controlar que se esté respetando lo estimado o haya que actualizarlo | Lombardi Lautaro | 6 |  |
| 1.5.6 | Actualización de la documentación | Se realizan modificaciones a los documentos del proyecto según corresponda | Lombardi Lautaro | 3 |  |
| 1.5.7 | Control de riesgos | Se lleva a cabo el control de riesgos para revisar que se esté respetando la forma de trabajo | Lombardi Lautaro | 8 |  |
| 1.6 | Capacitación | Tiempo consumido en el aprendizaje de nuevas herramientas | Lombardi Lautaro, Castillo Patricio, Hernández Facundo, Gross Pablo, Fallatti Franco, Cañete Ezequiel, Gonzales Federico | 38 |  |
| 1.6.1 | Trello | Herramienta para llevar a cabo la ejecución de la metodología ágil | Lombardi Lautaro | 6 |  |
| 1.6.2 | Testing Tools | Herramientas para llevar la ejecución del testeo | Hernández Facundo | 8 |  |
| 1.6.3 | SQLite | Herramientas para el uso de base de datos | Castillo Patricio | 6 |  |
| 1.6.4 | Librerías | Librerías para la el desarrollo del producto | Gross Pablo | 8 |  |
| 1.6.5 | Reac native | Lenguaje para el desarrollo de interfaces | Fallatti Franco | 6 |  |
| 1.6.6 | Figma | Herramienta de diseño de interfaces | Cañete Ezequiel | 4 |  |
| 1.7 | Back-End | Tareas relacionadas con el desarrollo del producto | Castillo Patricio, Gross Pablo, Fallatti Franco | 70 |  |
| 1.7.1 | Definición de end points | Definición de los puntos de acceso de la API del producto | Fallati Franco | 6 |  |
| 1.7.2 | Entrenamiento del modelo | Entrenamiento del modelo de la IA de reconocimiento facial | Gross Pablo | 8 |  |
| 1.7.3 | Modelado de la IA | Realización del modelado de la IA de reconocimiento facial | Gross Pablo | 24 |  |
| 1.7.4 | Conexión con la base de datos | Realización de la conexión del back-end con las bases de datos | Catillo Patricio | 6 |  |
| 1.7.5 | Autenticación de usuarios | Validación de las personas a partir de los rostros de los mismos | Castillo Patricio | 20 |  |
| 1.7.6 | Autenticación offline | Validación de los usuarios por medio de sus credenciales | Fallati Franco | 6 |  |
| 1.8 | Front-End | Tareas relacionadas con la creación de interfaces de producto | Cañete Ezequiel, Fallatti Franco | 60 |  |
| 1.8.1 | Maquetado | Creación del maquetado de las interfaces | Cañete Ezequiel | 24 |  |
| 1.8.2 | Mockups | Emulación del front-end con un server de prueba | Cañete Ezequiel | 12 |  |
| 1.8.3 | Conexión con el back-end | Realización de la conexión del front con el back | Fallati Franco | 10 |  |
| 1.8.4 | Diseño | Realización del diseño de las interfaces | Cañete Ezequiel | 6 |  |
| 1.8.5 | Captura y manejo de imágenes | Creación de la funcionalidad de capturar los rostros de los usuarios a través de las cámaras frontales de sus dispositivos | Cañete Ezequiel | 8 |  |
| 1.9 | Deploy | Tareas relacionadas con el deploy del producto | Fallatii Franco, Cañete Ezequiel | 20 |  |
| 1.9.1 | Generación de apk productiva | Creación de la apk del producto | Cañete Ezequiel | 6 |  |
| 1.9.2 | Deployeo del back-end | Puesta en marcha del back-end en la nube | Fallatii Franco | 6 |  |
| 1.9.3 | Configuración de un web service | Configuración de la nube donde se almacenará el back-end | Fallatii Franco | 8 |  |
| 1.10 | Estabilización | Tareas finales para la estabilización final del producto y proyecto | Lombardi Lautaro | 11 |  |
| 1.10.1 | Manual | Documento en el cual se explica correctamente como usar el sistema | Lombardi Lautaro | 8 |  |
| 1.10.2 | Liberación | Etapa en la cual se presentará el proyecto completo al cliente | Lombardi Lautaro | 3 |  |

Para la realización de las estimaciones utilizamos la técnica de Planning Póker.

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Tarea de la WBS** | **Peso de horas** | **Peso de hito** |
| 23/4/2024 | 1.1.1 | 1 | 9 |
| 1.1.2 | 1 |
| 1.1.3 | 1 |
| 1.1.4 | 2 |
| 1.1.5 | 1 |
| 1.1.6 | 1 |
| 1.1.7 | 1 |
| 1.1.9 | 1 |
| 30/4/2024 | 1.1.8 | 1 | 8 |
| 1.6 | 5 |
| 1.8.4 | 2 |
| 7/5/2024 | 1.3.3 | 3 | 18 |
| 1.4.1 | 3 |
| 1.7.3 | 5 |
| 1.7.1 | 2 |
| 1.8.1 | 5 |
| 14/5/2024 | 1.4.2 | 2 | 9 |
| 1.4.3 | 1 |
| 1.7.2 | 2 |
| 1.7.4 | 2 |
| 1.8.5 | 2 |
| 21/5/2024 | 1.3.1 | 5 | 16 |
| 1.7.5 | 4 |
| 1.7.6 | 2 |
| 1.8.2 | 3 |
| 1.8.3 | 2 |
| 28/5/2024 | 1.9 | 4 | 4 |
| 4/6/204 | 1.10 | 2 | 2 |

11- Calendario:

Tareas 1.2 y 1.5 se realizan cada semana hasta el comienzo de la etapa de estabilización

|  |  |  |
| --- | --- | --- |
| **Peso** | **Min hs** | **Max hs** |
| 1 | 1 | 5 |
| 2 | 6 | 11 |
| 3 | 12 | 17 |
| 4 | 18 | 23 |
| 5 | 24 | 30+ |

|  |  |  |
| --- | --- | --- |
| **Entrega** | **Semanas** | **Baseline** |
| Reunión formal 1 | 15/04/2024 - 30/04/2024 |  |
| Reunión formal 2 | 30/04/2024 - 14/05/2024 |  |
| Reunión formal 3 | 14/05/2024 - 28/05/2024 |  |
| Reunión formal 4 | 28/05/2024 - 11/06/2024 |  |
| Reunión formal 5 | 11/06/2024 - 25/06/2024 |  |

12- Indicadores:

Para este proyecto decidimos utilizar los siguientes indicadores:

* Funcionalidad completa + Nivel de calidad
* Indicador de riesgo
* Tiempo X Recurso

13- Administración de riesgos:

Para identificar los riesgos que se presentan en el proyecto, inicialmente se realizará un análisis de riesgos al inicio del proyecto y se identificaran los que se presentan. Una vez completa la lista inicial de riesgos, se procederá a calcular la probabilidad de ocurrencia de cada riesgo, su impacto en el proyecto y su nivel de exposición como resultado de multiplicar los valores de ocurrencia e impacto. La forma de clasificación de cada riesgo será la siguiente:

|  |  |
| --- | --- |
| **Probabilidad** | |
| Alta | 3 |
| Media | 2 |
| Baja | 1 |

|  |  |
| --- | --- |
| **Impacto** | |
| Alta | 3 |
| Media | 2 |
| Baja | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Definición de exposiciones** | **Probabilidad Alta** | **Probabilidad Media** | **Probabilidad Baja** |
| **Impacto Alta** | 9 | 6 | 3 |
| **Impacto Media** | 6 | 4 | 2 |
| **Impacto Baja** | 3 | 2 | 1 |

Una vez definida la exposición a cada riesgo, de la lista total de riesgos, se seleccionarán aquellos 5 (TOP 5) que tengan mayor exposición y se les llevarán a cabo sus correspondientes acciones de mitigación y acciones de contingencia. Por último, se calculará el riesgo total de del proyecto y se anotará y actualizará de manera constante en cada reunión formal.

Forma de trabajo:

* Se dedicará un tiempo a la evaluación de los riesgos en cada ciclo de desarrollo, de esta evaluación se actualizará la planilla de Excel de riesgos con nuevos riesgos si los hubiere o actualizando los valores de probabilidad e impacto de riesgos ya existentes.
* Luego de cada evaluación se volverá a definir el TOP 5.
* Se trabajará en realizar planes de mitigación y contingencia de los riesgos conformados en este TOP 5. No se evaluarán los demás debido a que puede llegar a ser una lista muy larga y muy costosa de analizar. Si ya existen planes de mitigación o contingencia para alguno de ellos, se pensará en cómo mejorarlos.
* Una vez realizado esto se actualizará el indicador de riesgo

Listado de riesgos:

|  |  |  |  |
| --- | --- | --- | --- |
| **Descripción** | **Probabilidad de ocurrencia** | **Impacto del riesgo** | **Exposición al riesgo** |
|  |  |  |  |
|  |  |  |  |

Top 5 riesgos:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Descripción** | **Probabilidad de ocurrencia** | **Impacto del riesgo** | **Exposición al riesgo** | **Plan de mitigación** | **Plan de contingencia** | **Responsable** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Estos son los formatos de las tablas que se usan en las planillas.

14- Administración de cambios:

Para la documentación y gestión de cambios utilizaremos la siguiente plantilla:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fecha aparición** | **Resumen** | **Detalle** | **Impacto** | **Beneficio** | **Análisis** | **Validación** | **Fecha aprobación** | **Fecha conclusión** |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Los valores de dicha planilla serán:

* Fecha Aparición: Fecha en la que se propuso el cambio.
* Resumen: Se detalla brevemente en qué consiste el cambio, lo utilizamos para identificar rápidamente el cambio.
* Detalle: Se detalla el cambio para luego ser analizado por el equipo de trabajo y evaluar los detalles del cambio y cómo poder realizarlo.
* Impacto: Cuánto impacta el cambio en el proyecto. Utilizaremos la siguiente escala
  + Alto = Es un cambio que por diversos motivos será muy laborioso de trabajar y a su vez puede conllevar a futuros problemas al implementarlo.
  + Medio = Es un cambio que, si bien no desestabiliza el proyecto, su implementación traerá un costo considerable para implementar.
  + Bajo = Es un cambio que no afectara la estructura del proyecto, además su implementación no debería traer costos considerables de implementación.
* Beneficio: Que tanto beneficia al usuario la implementación de dicho cambio, utilizamos la siguiente escala:
  + Alto = Grandes beneficios para el sistema al aplicarlos.
  + Medio = Cambios útiles para el sistema.
  + Bajo = Cambios poco significativos para el sistema.
* Análisis: Detalla los pasos necesarios para aplicar el cambio.
* Validación: Detalla si el cambio va a ser aplicado o no.
* Fecha aprobación: Fecha en la que se aprueba el cambio
* Fecha de conclusión: Fecha en la que el cambio ya está implementado en el sistema.

Forma de trabajo:

* Los cambios serán relevados en cada reunión formal por medio de un documento de solicitud de cambios, de ese relevamiento, se registrarán estos cambios en la planilla de Excel de cambios.
* Durante la primera semana al registro del cambio se lo analizará para determinar si se acepta o se rechaza.
* En la próxima reunión sea formal o no con el cliente, ya una vez analizado el cambio podrán ocurrir 2 escenarios:
  + Se negociará con el cliente sobre las condiciones de aceptación del cambio.
  + Se explica al cliente porque no debe aceptarse el cambio y se negocia si debería ser reevaluado.

15- Administración de bugs:

El procedimiento para la elaboración de la planilla de bugs consistirá en lo siguiente:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **BUG-ID** | **Categoría** | **Ciclo de aparición** | **Fecha** | **Descripción** | **Estado** | **Fecha de corrección** | **Responsable** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

* BUG-ID: Identificador del bug, consiste en un número único referente al bug, si llega a ser un bug recurrente (bug resuelto previamente que reapareció) se agregara una R por cada aparición del mismo.
* Categoría: Categoría del bug en cuestión, puede ser:
  + Crítico: Bug que impide la realización de una funcionalidad en cuestión, están relacionados con los criterios de aceptación de las pruebas.
  + Importante: Bug que dificulta la realización de la funcionalidad en cuestión. Por ej.: Ausencia de mensaje o mensaje poco descriptivo en la validación de un campo de un formulario.
  + Leve: Bugs referidos a pequeños desperfectos del sistema. Por ej.: Error ortográfico en algún mensaje, Texto en botones escondido, Errores visuales que provengan de la interfaz, etc.
* Ciclo de aparición: Número de ciclo de la 1ra aparición del bug.
* Fecha: Fecha exacta de aparición del bug
* Descripción: Detalles sobre el bug, el mismo deberá proveer la suficiente información para poder ser resuelto.
* Fecha de corrección: Fecha exacta de cierre del bug
* Estado: Estado actual del bug, los estados pueden ser.
  + Abierto: Todavía no se asignaron recursos para que se arregle dicho bug
  + En corrección: Se asignaron recursos para resolver el bug.
  + Pendiente de aprobación: Se está esperando una aprobación del bug después de haber estado en corrección.
  + Corregido: El bug ya se encuentra solucionado.
* Responsable: responsable del seguimiento del bug.

Forma de trabajo:

* Se encuentra un bug, y lo primero que se hará es agregarlo a la planilla de Excel correspondiente para los bugs.
* Una vez asignado, se le avisara al responsable del área del bug.
* El responsable del área asignara la responsabilidad a quien le parezca más capacitado para resolverlo en su sector correspondiente.
* El responsable asignado realizará el seguimiento del bug y una vez resuelto se lo dará como corregido en la planilla de bugs.