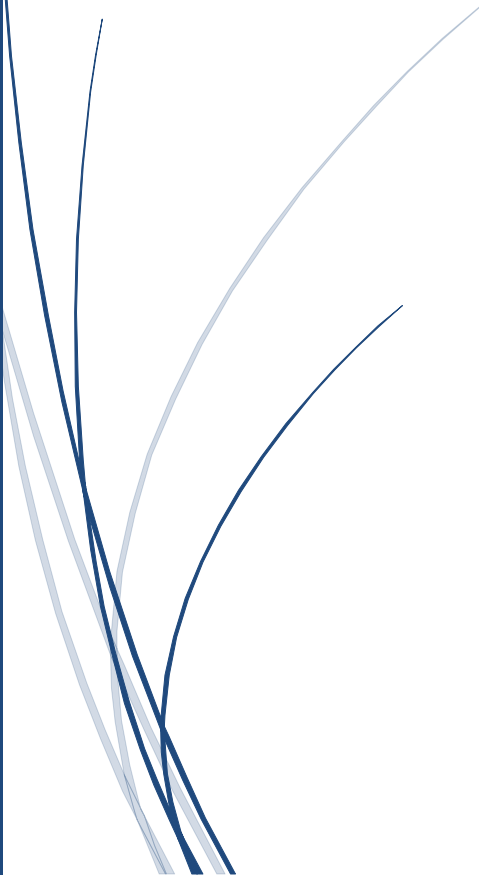




14-11-2022

Plan de proyecto

Aprendizaje automático de lengua de
señas colombiana



Cristian Javier Da Cámara Sousa
Kenneth David Leonel Triana
Juan Pablo Ortiz Rubio
Camilo Andrés Sandoval Guayambuco

Noviembre 2022

Tabla de contenido

1	Introducción.....	3
2	Propuesta del proyecto.....	4
3	Vista general del proyecto.....	5
3.1	Propósito, alcance y objetivos	5
3.2	Supuestos y restricciones	5
3.3	Entregables del proyecto	5
4	Modelo de ciclo de vida	5
5	Lenguajes y herramientas.....	6
6	Funciones del producto	8
6.1	Características del usuario	8
6.1.1	Usuario final o consumidor.....	8
6.1.2	Usuario contribuidor.....	9
8	Plan de trabajo del proyecto	10
9	Administración del proyecto.....	13
9.1	Métodos y herramientas de Estimación	13
9.2	Entrenamiento del equipo	13
9.3	Calendarización	14
10	Procesos de soporte	15
10.1	Ambiente de trabajo.....	15
10.2	Análisis y administración de riesgos	15
10.3	Administración de configuración y documentación	17
10.4	Control de calidad	18
11	Referencias.....	20

Lista de figuras

Figura 1 Modelo Canvas.....	4
Figura 2 Planificación de tareas en Notion	10
Figura 3 Especificación de una tarea	10
Figura 4 Organigrama del proyecto	11
Figura 5 Proceso de cambios a los ítems de configuración	18

Lista de tablas

Tabla 1 Descripción de roles	12
Tabla 2 Estimación tiempos	13
Tabla 3 Mapa de calor.....	15
Tabla 4 Descripción de indicadores	15
Tabla 5 Clasificación de riesgos	16
Tabla 6 Mapa de calor - Riesgos del proyecto.....	16
Tabla 7 Medidas de disminución de riesgos	17
Tabla 8 ítems de configuración.....	17

1 Introducción

En el presente documento se realiza una profundización del plan de proyecto del sistema de aprendizaje automático de lengua de señas colombiana. Para esto se identificó la necesidad por la cual surge la idea y cómo impactaría en el sector de la población. Para ello se presenta una solución mediante diferentes herramientas y técnicas para desarrollar las principales funcionalidades de la aplicación con sus respectivas descripciones. Cabe destacar que el grupo de trabajo usará una metodología ágil llamada *Scrumban* con el que se hace una asignación de tareas que se realizaron en la planeación del proyecto y próximamente en la implementación del sistema.

2 Propuesta del proyecto

El propósito del proyecto es la creación de un Sistema que gracias a sus funcionalidades permita y facilite el reconocimiento de las señas de la lengua de señas colombiana (LSC), gracias a las herramientas para su gestión y planificación como lo es el procesamiento de imágenes y redes neuronales, para lograr consolidar el producto, ofreciendo a la comunidad sorda un instrumento que genere inclusión social. El sistema requerirá a futuro que empresas o entidades públicas o privadas que mediante su motivación por mejorar la comunicación de todos participen, para ampliar el Sistema de reconocimiento creando o ampliando el léxico de señas dejado por el grupo.

El proyecto contará con los siguientes objetivos establecidos para su correcta ejecución:

- Desarrollar un sistema que sea capaz de identificar y traducir la lengua de señas colombiana estáticas mediante el procesamiento de imágenes.
- Diseñar una interfaz que permita la captura de datos de nuevas señas con su traducción.
- Probar el funcionamiento del sistema a través de potenciales clientes.
- Generar una base de datos que contenga la información de las señas seleccionadas para su identificación y traducción.
- Entrenar el modelo resultante y hacer comparativas con procedimientos de ajuste fino.

Los objetivos fueron establecidos para asentar y acotar el problema que se desea solucionar, que supla las necesidades que hoy en día en Colombia la comunidad sorda carece de herramientas que faciliten su aprendizaje o comunicación con los demás conciudadanos, por dicha razón a partir de la ley 1396 de 2009 (Wilmer Leal Pérez, 2021) queremos registrar nuestro proyecto para promocionar la lengua de señas colombiana que mejore la calidad de vida de los individuos en pro de una sociedad inclusiva que prevé el aprendizaje de todos.

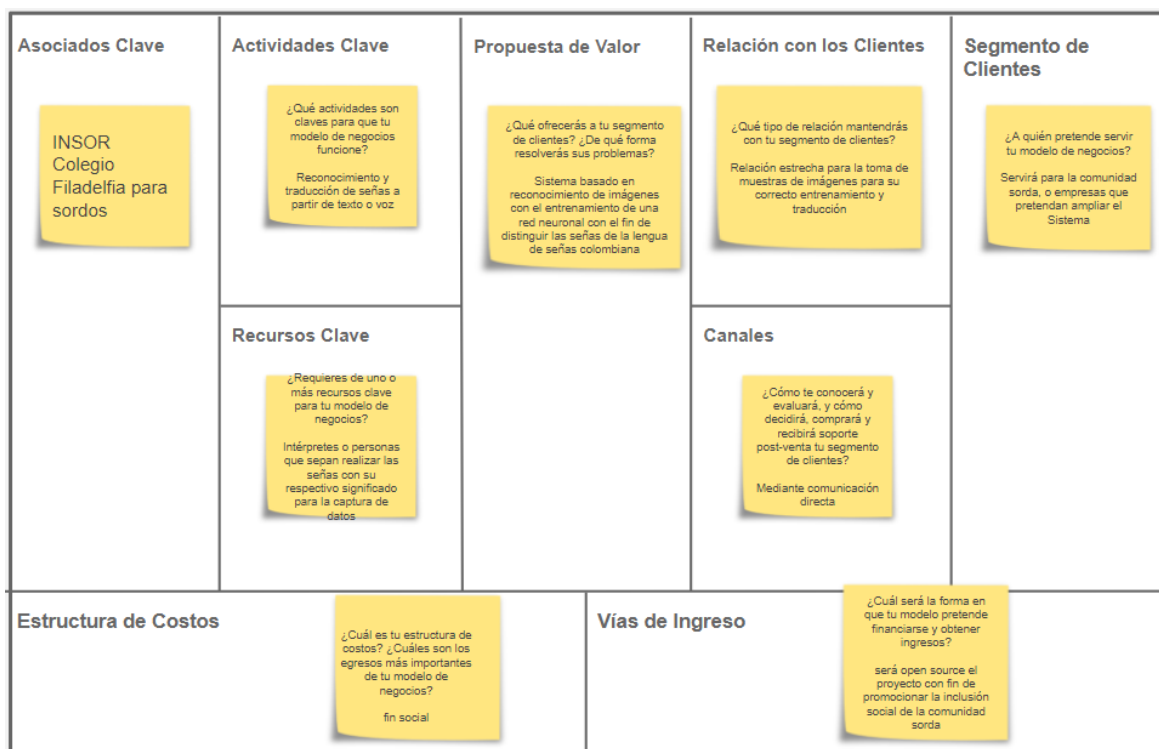


Figura 1 Modelo Canvas

3 Vista general del proyecto

3.1 Propósito, alcance y objetivos

El propósito de este proyecto es solucionar la problemática expuesta en la sección 1.1 del documento final de la memoria de trabajo de grado. También, el proyecto contempla los objetivos descritos en la sección 2.1 y 2.2 del mismo documento. Teniendo en cuenta lo anterior, el alcance del proyecto estará definido en desarrollar un sistema capaz de identificar y traducir la lengua de señas colombiana, para ello se hará énfasis en algunas señas estáticas en las que se pueda alcanzar una precisión como mínimo del 80% y a su vez, capturar nuevas señas (estáticas) para aumentar el diccionario del sistema.

3.2 Supuestos y restricciones

Supuestos:

- El conjunto de datos, es decir las señas, quedarán almacenadas bien sea en una base de datos o en una plataforma online en el que se alojarán en su respectivo conjunto de señas.
- Se podrá acceder a las máquinas virtuales de la Universidad Javeriana para poder realizar el entrenamiento del modelo.
- Futuros colaboradores podrán continuar con el desarrollo del proyecto.

Restricciones:

- Al ser un lenguaje con bastantes señas, se restringe las señas que se van a utilizar en el sistema, manejando únicamente señas estáticas.

3.3 Entregables del proyecto

El proyecto producirá los entregables descritos en la sección 2.3 del documento de la memoria de trabajo de grado. Sin embargo, estos entregables también son descritos en la sección 9.3 del presente documento

4 Modelo de ciclo de vida

El modelo de ciclo de vida que se escogió para la realización del proyecto fueron las metodologías ágiles Scrum y Kanban las cuales son mejor conocidas como *Scrumban*, ya que nos ofrece como equipo, la posibilidad de gestionar, clasificar y priorizar las tareas de acuerdo con su importancia. Asimismo, cada miembro del grupo puede revisar las tareas de cada fase de desarrollo en la herramienta Notion y escoger la tarea con la que se sienta más cómodo de la sección *“To do”* para que cuando esté realizando dicha actividad la mueva a la sección *“Doing”* y cuando la finalice la mueva a la sección *“Do”*. A su vez, que el equipo pueda visualizar los cambios de tal forma que se tenga un control de las tareas faltantes y las que ya se finalizaron, así como también quien realizó dicha tarea [1].

Adicionalmente, el equipo va a realizar un *“Daily Scrum”* el cual consiste en una reunión corta donde cada integrante menciona lo que ha realizado hasta el momento, desde tareas completadas, tareas incompletas e incluso inconvenientes que han tenido. Esta metodología permite estar en un círculo virtuoso donde los procesos se mejoran gracias a que se va ejerciendo una cultura con integrantes motivados donde el flujo de trabajo dinámico y comunicación eficiente es relevante para el desarrollo del proyecto.

5 Lenguajes y herramientas

Como equipo buscamos el conjunto de herramientas, entornos, librerías y lenguajes de programación que mejor se adapten a nuestro sistema teniendo en cuenta que deben permitir y facilitar: El trabajo sobre el manejo de datos, procesamiento de imágenes, almacenamiento en base de datos correspondiente, desarrollo de modelos de red neuronal, despliegue de interfaces para interacción con el usuario, entre otros. No solamente con esos factores, sino que también se busca que estas herramientas sean fáciles de adaptar para ser amigables con los usuarios que quieran aportar y dar continuidad en un futuro al sistema.

Una vez aclarado los criterios que seleccionamos para las herramientas y aclarando que en el proceso de implementación pueden ser agregadas o modificadas teniendo en cuenta las propuestas de la metodología SCRUM de iterar y mejorar; se va a listar estos componentes a usar:

- Manejo de contenedores: Necesario para la ejecución de diferentes microsistemas en el entorno local teniendo interacción entre sí, como lo es las dos interfaces mencionadas, el *back-end* y la base de datos.
 - Docker: Es una herramienta que empaqueta software en contenedores, los cuales tienen todo lo necesario para que este se ejecute, incluye: bibliotecas, herramientas de sistema, código y tiempo de ejecución.[2].
 - Docker-Compose: Herramienta para ejecutar Dockers de contenedores múltiples [3].
- Bases de datos: Necesario para almacenar tanto el dataset de las señas, como los datos locales y resultados obtenidos de manera local para ser expuestos globalmente para todos los usuarios.
 - TinyDB: Es un almacén de datos persistente para la aplicación, lo que significa que los datos estarán disponibles cada vez que se abra la aplicación, además mantiene los datos en formato *JSON* siendo almacenamiento simple y no siendo necesario nada de lectura múltiple o paralelismo [4].
 - Kaggle: Página web enfocado a la comunidad de científicos de datos y profesionales del aprendizaje automático con posibilidad de almacenamiento de conjunto de datos[5].
- Back-end: Necesario para realizar toda la gestión de procesamiento de imágenes, comunicación con las bases de datos, entrenamiento y reentrenamiento del modelo, manejo del flujo de datos entre microsistemas.
 - Python: Lenguaje de alto nivel de programación [6].
 - Librerías necesarias en la herramienta Python:
 - Tensorflow:
 - Es una librería implementación aprendizaje automático la cual permite construir y entrenar redes neuronales para detectar patrones y razonamientos usados por los humanos. [7]
 - Keras:
 - Librería implementación aprendizaje automático a alto nivel [8].
 - Numpy:
 - Librería que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones

matemáticas de alto nivel para operar con ellas [9].

- Matplotlib / Seaborn:
 - Librerías para elaborar graficas [10], [11].
- PyMongo:
 - Librería de control de Python para MongoDB [12].
- Front-end: Necesario para mostrar todos los resultados (traducciones) obtenidos por las predicciones del modelo entrenado y para ser intermediario con los usuarios que van a ingresar nuevas señas al sistema.
 - Angular: Es una plataforma de desarrollo para crear aplicaciones web, este está construido sobre *TypeScript* el cual se compila con JavaScript [13].
 - Tensorflow.js: Biblioteca de JavaScript para el entrenamiento y la implementación de modelos de aprendizaje automático en navegadores [14].
- Organización y seguimiento del proyecto: Necesario para mantener un registro de las diferentes actividades que se van a realizar durante el desarrollo del proyecto, asimismo como el control y manejo de versiones del código.
 - Notion: Es un organizador de tareas multiplataforma que permite llevar la planificación del proyecto, así como para agregar actividades, tareas o notas acerca del proyecto [15].
 - GitHub: Es una plataforma de alojamiento, que ofrece a los desarrolladores la posibilidad de crear repositorios de código y guardarlos en la nube de forma segura, usando un sistema de control de versiones [16].

Estas herramientas son presentadas más a detalle en el documento de memoria de trabajo de grado.

6 Funciones del producto

Para la realización del producto se estableció como funciones primordiales las que están presentes en la figura 2, con el fin de que se priorice el desarrollo de las interfaces que satisfagan las necesidades de los usuarios.

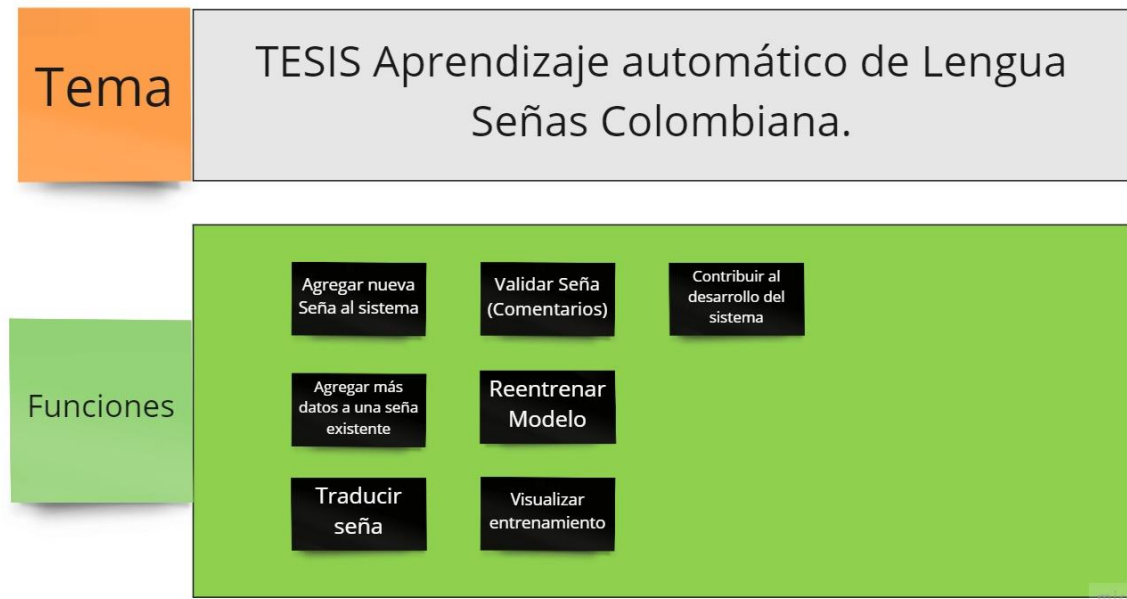


figura 2 Funciones Sistema

En nuestro producto tenemos diferentes historias de usuario que serán introducidas a continuación:

1. Como desarrollador del sistema, quiero capturar los datos (imágenes de la señal) para poder agregarlos al dataset.
2. Como desarrollador del sistema, requiero añadir nuevos datos (imágenes de la señal) a una señal existente en el dataset.
3. Como usuario, requiero una interfaz para realizar la señal mediante el uso de la cámara para obtener la traducción de esta.
4. Como usuario, requiero la opción de reentrenar el modelo con los datos (Señas) que estén presentes en el dataset.
5. Como usuario, necesito visualizar el comportamiento al reentrenar el modelo.
6. Como usuario final, requiero la opción de validar la traducción de la señal, para mejorar el producto ofrecido.
7. Como desarrollador del sistema, requiero tener instrucciones adecuadas para poder contribuir al desarrollo del sistema.

6.1 Características del usuario

Existirán dos clases de usuarios diferentes que utilizarán nuestro sistema, estos se dividen entre usuario contribuidor y usuario final o consumidor.

6.1.1 Usuario final o consumidor

Se refiere al usuario que consume el sistema, que para nuestro caso serán las personas con discapacidad auditiva, nuestro sistema les servirá para poder comunicarse con personas que no tengan el conocimiento suficiente del lenguaje de señas. Este usuario podrá acceder a las interfaces en donde estos mostraran una seña a la cámara y esta será traducida con audio y texto para la persona con la que intenta comunicarse.

6.1.2 Usuario contribuidor

Hace referencia a dos grupos de personas, el primero va destinado a quienes tengan conocimiento en programación, en donde estos tendrán acceso al código y serán capaces de mejorar, continuar o agregar funcionalidades al sistema; El segundo grupo reúne a las personas con conocimiento en el lenguaje de señas, en donde estos tendrán acceso a las interfaces que están hechas para agregar más señas y con esto ampliar el dataset.

7 Entrega del producto

Al finalizar el desarrollo del sistema se entregará la documentación asociada a este, como lo es la memoria de trabajo de grado, plan de gestión de proyecto, especificación de requisitos de software, descripción del diseño de software, manual de usuario, propuesta de proyecto y material filmográfico del funcionamiento del sistema y un documento con las pruebas realizadas a este. En cuanto al sistema, se entregará un repositorio con el versionado del sistema y el código fuente del sistema y su respectiva documentación.

8 Plan de trabajo del proyecto

Para poder realizar el plan de trabajo, se utilizó la herramienta Notion en la que se va a ir poniendo todas las actividades asociadas a las distintas fases de desarrollo del proyecto. En dicho plan se presentan las fechas importantes que hay que tener en cuenta para el desarrollo propicio y estructurado de la aplicación. Se muestra entonces, un tablero en Notion con las actividades relacionadas a la planeación del proyecto.



Figura 2 Planificación de tareas en Notion

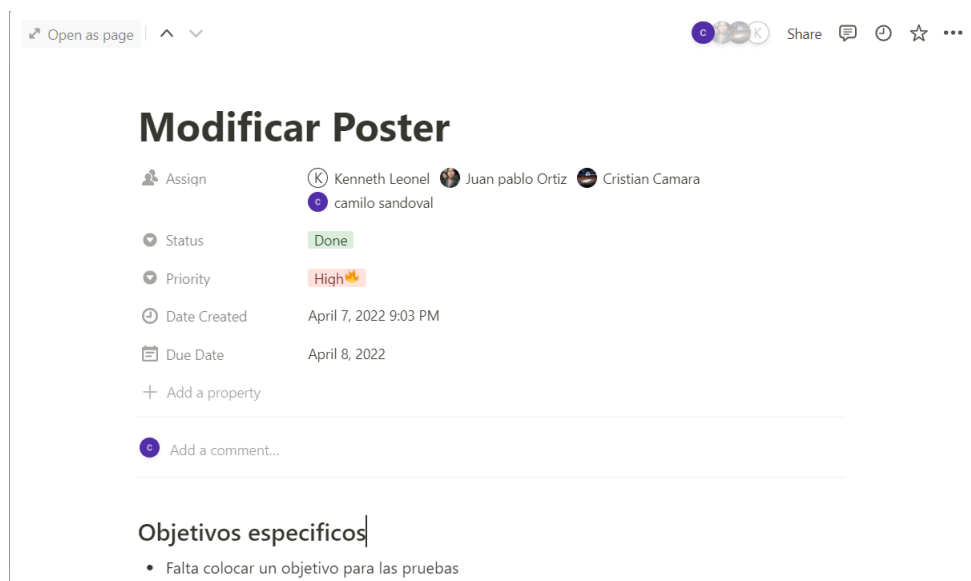


Figura 3 Especificación de una tarea

Como se observa en el tablero, consta de tres secciones donde se van a ir catalogando las tareas dependiendo del estado de estas, cada actividad tiene una prioridad (Alta, media, baja), quienes son los miembros involucrados, la fecha límite de cada tarea y los comentarios correspondientes en caso de que haya.

En cuanto a la organización interna del equipo, se busca que todos los integrantes participen en las actividades asociadas al desarrollo del proyecto, se consideró oportuno que

cada integrante asuma roles de liderazgo en áreas específicas.

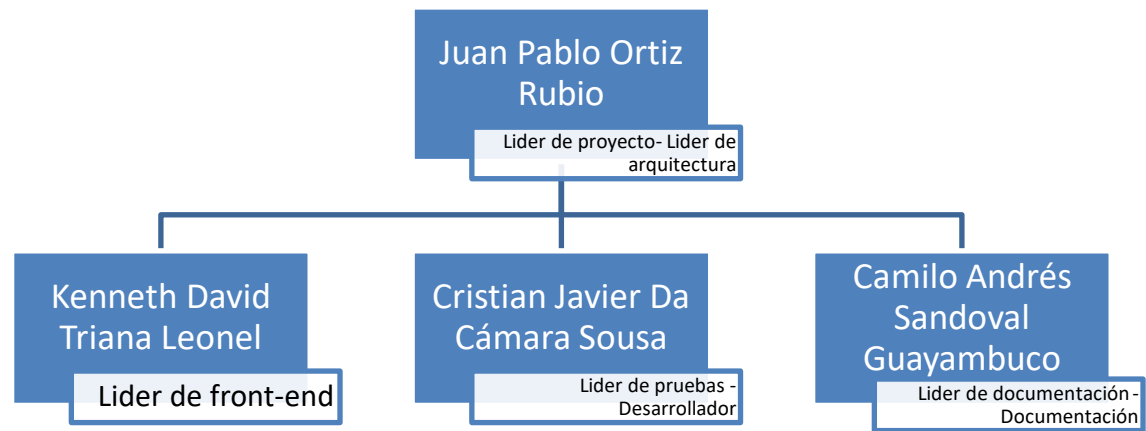


Figura 4 Organigrama del proyecto

Integrante	Rol(es)	Responsabilidades
Juan Pablo Ortiz Rubio	Líder de proyecto	Es el encargado de monitorear el progreso del proyecto, supervisará el correcto desarrollo del trabajo de grado y también supervisará que los demás integrantes cumplan con sus funciones respectivas. Adicionalmente, es el encargado de que en caso de que haya deficiencias en el equipo, deberá proponer medidas correctivas para asegurar el éxito del proyecto.
	Líder de arquitectura	Liderará el diseño de la arquitectura del sistema y velará por el cumplimiento de las directrices del proyecto en cuanto a este aspecto. Además, supervisará que la implementación del sistema cumpla con la arquitectura acordada.
Kenneth David Leonel Triana	Líder de <i>Font-end</i> /	Liderará el diseño e implementación del prototipo de la aplicación web del sistema.
Cristian Javier Da Cámara Sousa	Líder de pruebas / Desarrollador	Es el encargado de liderar y supervisar el diseño, ejecución y documentación de las pruebas del sistema. Además, desarrollará el código necesario para las funcionalidades específicas para el proyecto, también la corrección de bugs y errores y su posterior retroalimentación.
Camilo Andrés	Líder de	Supervisará la correcta documentación de los

Sandoval Guayambuco	documentación / Desarrollador	componentes del sistema de acuerdo con lo establecido. Además, desarrollará el código necesario para las funcionalidades específicas para el proyecto, también la corrección de bugs y errores y su posterior retroalimentación.
------------------------	----------------------------------	--

Tabla 1 Descripción de roles

9 Administración del proyecto

9.1 Métodos y herramientas de Estimación

Para establecer la estimación de nuestro proyecto vamos a estar usando el método de 3 puntos, teniendo en cuenta que ninguno ha tenido una experiencia anterior con proyectos del mismo estilo, por eso estaremos buscando la estimación de duración del proyecto.

Con este método se va a obtener un valor estimado y una desviación típica a partir de 3 diferentes valores:

- Optimista(O): Es la duración que se puede dar en el desarrollo del proyecto en las mejores condiciones.
- Más probable o media(M): Es la duración del desarrollo del proyecto en las condiciones más esperadas.
- Pesimista(P): Es la duración del desarrollo del proyecto en el peor caso.

Estos 3 valores se utilizan en una fórmula que representa la factibilidad, siendo lo más cercano al tiempo dedicado al proyecto con todos los procesos de desarrollo.

$$\text{Formula de factibilidad: } ((4 * M) + O + P) / 6$$

Estimación con tres puntos				
Historia de Usuario	Media (M)	Optimista (O)	Pesimista (P)	Estimación (días)
1	24	17	30	23,833333
2	6	4	8	6
3	24	17	30	23,833333
4	12	8	18	12,333333
5	14	10	20	14,333333
6	14	10	20	14,333333
7	18	14	20	17,666667

Tabla 2 Estimación tiempos

La estimación fue realizada por los 4 integrantes del proyecto dando cada persona su propio criterio de estimación de media(M), optimista(O) y pesimista(P) en semanas para cada historia de usuario. Esta estimación individual se daba según los criterios y los análisis correspondientes realizados a cada historia de usuario, según su complejidad y el proceso que llevaría realizar cada tarea. La suma de cada uno de los resultados (M, O y P) en cada historia de usuario estaría dividida en el número de personas que dieron su estimación (en este caso 4). Dado un resultado final de medida media, optimista y pesimista en cada historia de usuario, se procede a calcular la factibilidad con estos 3 datos dando como resultado la duración más factible en semanas para cada historia de usuario.

9.2 Entrenamiento del equipo

Si bien, para la realización del proyecto es necesario realizar una correcta capacitación de todos los integrantes frente a las diferentes herramientas y conceptos que se utilizarán a lo largo del desarrollo del proyecto. Teniendo en cuenta lo anterior, para realizar el sistema capaz de identificar las señas, fue necesario tomar cursos de aprendizaje de máquina, también leyendo documentación respectiva, artículos y videos de los diferentes métodos y herramientas necesarias para el desarrollo del proyecto. Una vez hecho esto, los integrantes empezaron a utilizar el lenguaje de programación Python, enfocado al aprendizaje de maquina y a su vez, entrenarnos aún más en cuanto al desarrollo de la interfaz.

9.3 Calendarización

De acuerdo con el modelo de ciclo de vida planteado y la metodología SCRUMBAN, todas las actividades que se establecen comienzan con una reunión los días martes, en la que se realiza una retroalimentación de las actividades realizadas y del avance de los participantes durante la respectiva iteración. A su vez, se realiza una reunión todos los viernes con la directora de trabajo de grado para comentar avances y retroalimentación al respecto.

10 Procesos de soporte

10.1 Ambiente de trabajo

El trabajo del equipo estará regulado por unas normas que permitan una correcta convivencia y sano desarrollo del proyecto a lo largo del tiempo en el que se llevará a cabo. Por tal razón se establecieron reglas establecidas y aprobadas por todos los miembros del equipo, las cuales son:

1. Se realizará una reunión semanal (mínimo una) ya sea para mostrar avances del proyecto o comunicar algo en particular acerca del mismo.
2. Cada reunión deberá ser planeada por lo menos con un día de anticipación.
3. El desarrollo del proyecto se regirá por una sana convivencia, donde cada miembro del equipo se sienta cómodo en el proceso del proyecto ya sea en la implementación como dando ideas garantizando así un ambiente de trabajo sano.
4. La asignación de tareas debe ser lo más equitativo posible, de forma de que cada integrante pueda desempeñarse en lo más adecuado a sus habilidades.
5. Puntualidad y disposición para todas las tareas relacionadas con el proyecto.
6. Si algún miembro del equipo no contribuye en el desarrollo del proyecto o en asistencia a las reuniones, esto se tomará en cuenta para las entregas del mismo.

10.2 Análisis y administración de riesgos

Para realizar un correcto análisis y gestión de riesgos, es de vital importancia establecer el origen del riesgo, dado que puede variar la posibilidad que este se materialice y las acciones a realizar para disminuir el impacto del mismo. De igual forma, se espera establecer algunas medidas que pueden ser tomadas para reducir la probabilidad de que el riesgo ocurra.

También para una correcta clasificación es necesario establecer el impacto que puede llegar a tener el proyecto en cada riesgo asociado. Por ende, un mapa de calor es lo idóneo para realizar una correcta gestión de riesgos como lo es el siguiente:

Probabilidad /Impacto	1	2	3	4	5
1					
2					
3					
4					
5					

Tabla 3 Mapa de calor

Nivel	Probabilidad (%)	Impacto
1	0-5	Despreciable
2	5-20	Menor
3	20-50	Moderado
4	50-80	Alto
5	80-100	Catastrófico

Tabla 4 Descripción de indicadores

Teniendo en cuenta los indicadores previamente descritos, a continuación, se realizará una identificación de los riesgos que pueden afectar al proyecto teniendo en cuenta las siguientes notaciones:

- E: Riesgo externo
- I: Riesgo interno
- T: Riesgo tecnológico

ID	Descripción	Origen	Probabilidad de ocurrencia	Impacto
E1	Demora en obtención del dataset de las señas	Demoras por parte del equipo en contactar a la persona capacitada en el lenguaje de señas	2	4
T1	Pérdida importante de información	Los equipos sufren retrasos por pérdida total o parcial del trabajo.	2	5
I1	Incremento en la duración de tiempo de desarrollo por desconocimiento del equipo.	Desconocimiento por parte del equipo	2	4
E2	Algún(os) integrante(s) se enferma(n) gracias al COVID 19	COVID 19	3	3
T2	Dificultad en el préstamo de máquinas virtuales por parte de la universidad	Problemas con la universidad en cuanto al préstamo	1	1
I2	Propagación constante de errores a lo largo del proyecto	Una mala / nula revisión de los avances realizados del proyecto	1	4
I3	Disolución del equipo por diferencias de los integrantes del proyecto	Incapacidad de solucionar problemas del equipo	1	5

Tabla 5 Clasificación de riesgos

Probabilidad /Impacto	1	2	3	4	5
1	T2				
2					
3			E2		
4	I2	E1 – I1			
5	I3	T1			

Tabla 6 Mapa de calor - Riesgos del proyecto

Como se pudo observar en las tablas anteriores, los riesgos van asociados junto con la probabilidad de que estos ocurran. Sin embargo, es necesario establecer una medida para disminuir la probabilidad de ocurrencia, por tal razón, a continuación, se presentarán medidas a tomar para disminuir dicha probabilidad:

ID riesgo	Medida correctiva
E1	Investigar no solo en la universidad sino en organizaciones externas algún interprete / traductor de lengua de señas
T1	Para evitar la pérdida de información, los integrantes estarán realizando un guardado

	constante de los avances hechos, estos avances se subirán a GitHub para evitar la pérdida de los mismos.
I1	Los integrantes estarán en constante lectura y aprendizaje del tema, haciendo uso de la información asociada al proyecto.
E2	El (Los) integrante(s) intentarán permanecer en reposo en sus casas y en caso de que estos se encuentren hábiles pueden realizar labores como la documentación. Y en caso de que este se encuentra bastante enfermo, deberá guardar reposo y los demás integrantes ayudarán a las labores asignadas al estudiante afectado.
T2	Cuando se requiera no solo las máquinas virtuales, sino recursos extra de la universidad, se le enviará una solicitud a la universidad mediante un correo con al menos una semana de anticipación para evitar demoras en la entrega.
I2	Los integrantes realizarán revisiones constantes de los avances grupales, dichas revisiones se harán en cada sprint
I3	Siempre que haya una diferencia en el equipo, hablar lo más pronto para intentar solucionar los problemas presentados y si es necesario, hablar con la psicóloga asociada para resolver los inconvenientes.

Tabla 7 Medidas de disminución de riesgos

10.3 Administración de configuración y documentación

Con el fin de realizar una descripción detallada de los ítems de configuración de nuestro proyecto se tomó la decisión de repartir ítems según se requiera para cada entrega. Las entregas están asociadas con los siguientes documentos:

Ítem de configuración	Descripción
Memoria de trabajo	Es el documento final de trabajo donde se tiene la información esencial de la tesis desarrollada por el grupo de trabajo.
Plan de proyecto	Es un documento de control para administrar un proyecto de software. En él se establece las políticas, procedimientos, normas, tareas, horarios y recursos necesarios para completar el proyecto.
Especificación de requisitos	Es un conjunto de recomendaciones para la especificación de los requerimientos o requisitos de software el cual tiene como producto final la documentación de los acuerdos entre el cliente y el grupo de desarrollo para así cumplir con la totalidad de exigencias estipuladas.
Especificación del diseño	Es un documento que describe el diseño de software en cuanto a la arquitectura del proyecto y las características de cada componente del sistema.
Propuesta del diseño	Es el documento el cual contiene la descripción de la solución propuesta asociada al diseño
Manual de usuario	Es un conjunto de instrucciones y recomendaciones para el usuario final, en el que se explica el funcionamiento del aplicativo y las diversas funciones que el usuario puede hacer.

Tabla 8 Ítems de configuración

Para realizar los cambios a los diferentes ítems de configuración se deben evaluar dos tipos de cambios: El primero de ellos es cuando se presentan cambios menores y el integrante del grupo puede realizar dicho cambio sin afectar notablemente la línea base del proyecto, generalmente estos cambios están relacionados con aspectos de la documentación del código u otros ítems de configuración cuyos cambios no impliquen un efecto sobre el trabajo de los demás integrantes. El segundo tipo de cambio es asociado cuando un miembro del proyecto desea realizar un cambio trascendental sobre algún ítem de configuración que puede afectar notablemente la línea base del proyecto, para realizar este tipo de cambio se debe realizar el siguiente proceso:

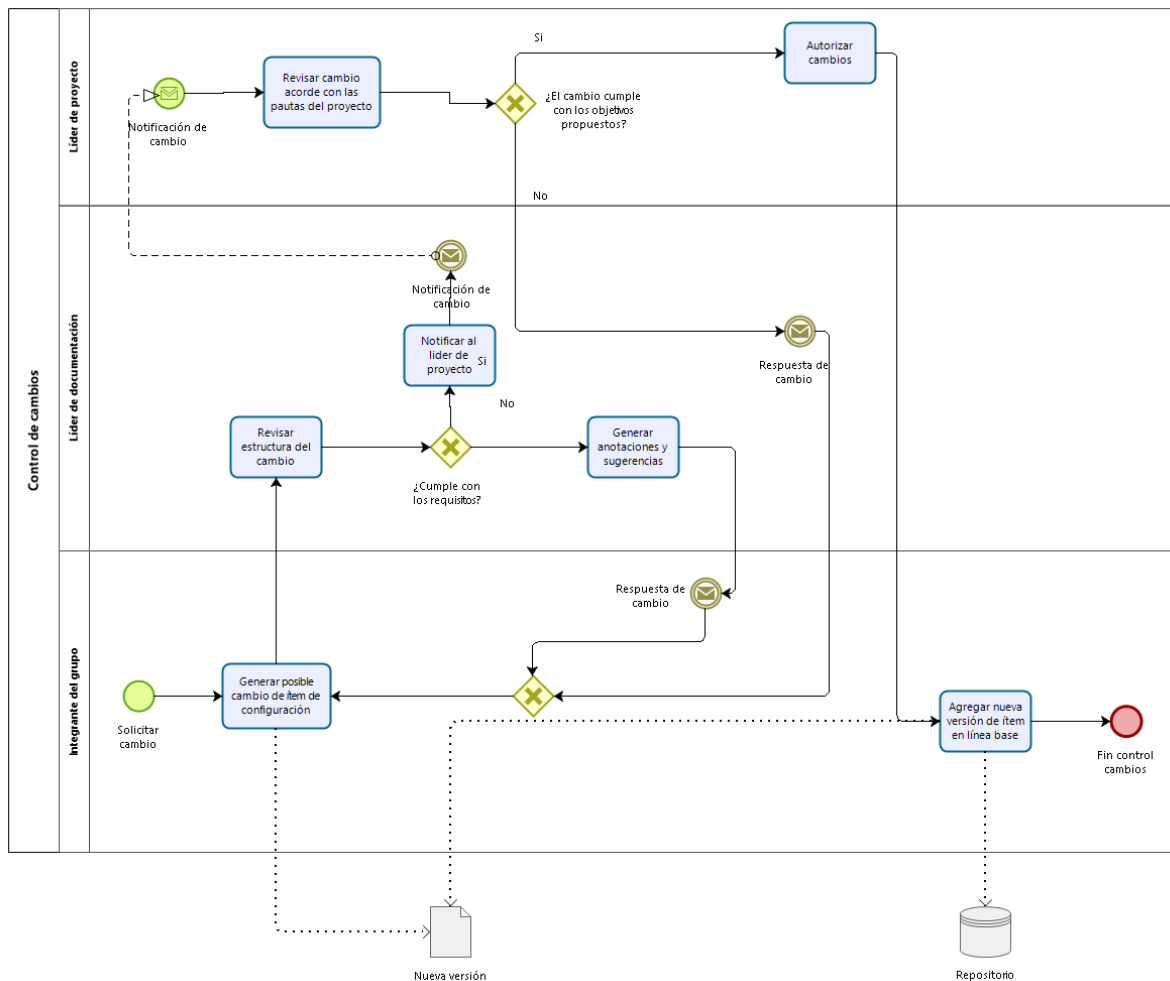


Figura 5 Proceso de cambios a los ítems de configuración

Como se pudo observar en la ilustración anterior, el proceso inicia con la generación de algún cambio en el ítem de configuración por parte de un integrante del grupo, este cambio será presentado al líder de documentación el cual se encargará de revisar si el cambio está correctamente documentado y cumple las pautas requeridas, el líder de documentación aprobará este cambio y se lo presentará al líder del proyecto y este revisará que el cambio propuesto se alinee con los objetivos del grupo, que aporte valor y que no perjudique al equipo. Finalmente, si estos requisitos se cumplen, el líder del proyecto aprueba el cambio propuesto y se procede a agregar una nueva versión del ítem a la línea base.

10.4 Control de calidad

Las actividades de control de calidad deberán realizarse tanto para el sistema como para la documentación realizada para el proyecto, para ello se realizará control de calidad de los avances

mediante verificación, validación y revisión, todo esto con el fin de supervisar y mejorar la calidad del proyecto. A continuación, se describe cada técnica de control de calidad para tener una mayor claridad en cuanto a estas.

- Verificación: Cada actividad a realizar tendrá uno o más integrantes responsables de realizarla y deberá tener al menos un integrante adicional que deberá revisar ya sea su forma, contenido o funcionalidad, dependiendo de la actividad.
- Validación: Después de la verificación de cada actividad, se realizará la validación del contenido de dicha actividad, con el fin de asegurar de que el contenido e información (En el caso de la documentación) y funcionamiento del software correspondan a lo especificado para dicha actividad.
- Revisión: Luego de que se haya verificado la realización de una actividad y validado su funcionamiento, se realizará una revisión de que dicha actividad este alineada con los principios, objetivos, requisitos y restricciones del proyecto, con el objetivo de asegurar un óptimo resultado entre las actividades.

A estos métodos de control de calidad también hay que tener en cuenta que se realizarán reuniones entre los integrantes y la directora de grupo en la que esta última puede sugerir modificaciones bien sea al código o a la documentación.

11 Referencias

- [1] I. Germanov, "Kanban vs Scrum vs Scrumban: Cuales son las diferencias?" <https://ora.pm/es/blog/scrum-vs-kanban-vs-scrumban/> (accessed May 18, 2022).
- [2] "Docker." <https://www.docker.com/> (accessed May 19, 2022).
- [3] "Docker Compose." <https://docs.docker.com/compose/> (accessed May 19, 2022).
- [4] P. Jezabel, M. Gil, C. Gil, and L. Santiago, "Desarrollo de APPs sin saber Programación." Accessed: Nov. 14, 2022. [Online]. Available: <https://campusvirtual.ull.es/ocw/mod/resource/view.php?id=8680#:~:text=TinyDB%20es%20un%20almac%C3%A9n%20de,cada%20vez%20que%20se%20juega.>
- [5] "Kaggle." <https://www.kaggle.com/> (accessed May 19, 2022).
- [6] "Python." <https://www.python.org/> (accessed May 19, 2022).
- [7] "TensorFlow." <https://www.tensorflow.org/> (accessed May 19, 2022).
- [8] "Keras." <https://keras.io/> (accessed May 19, 2022).
- [9] "Numpy." <https://numpy.org/> (accessed May 19, 2022).
- [10] "Matplotlib." <https://matplotlib.org/> (accessed May 19, 2022).
- [11] "Seaborn." <https://seaborn.pydata.org/> (accessed May 19, 2022).
- [12] "PyMongo." <https://www.mongodb.com/docs/drivers/pymongo/> (accessed May 19, 2022).
- [13] M. J. Goncalves, "¿Qué es Angular y para qué sirve?," Oct. 13, 2021. <https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/> (accessed Nov. 14, 2022).
- [14] "TensorFlow for JavaScript." <https://www.tensorflow.org/js> (accessed May 19, 2022).
- [15] L. Paez, "¿Qué es Notion? La mejor app de productividad que organizará tu vida," *Crehana*, Dec. 21, 2021. <https://www.crehana.com/blog/negocios/que-es-notion/> (accessed Nov. 14, 2022).
- [16] D. Camacho, "Qué es GitHub y cómo usarlo para aprovechar sus beneficios," *Platzi*, 2021.