**Departamento de Ciencias de la Computación(DCCO)**

**Carrera de Ingeniería de Software**

**Curso de Aseguramiento de la Calidad del Software**

Perfil de Proyecto “TalentFlow”

Presentado por: Bolaños Moreno Mateo, Caiza Tacuri Alisson, Esparza Gallardo Marco, Reinoso Namicela Jerly (Grupo 6)

Director: Ing. Ruiz Robalino, Jenny

Ciudad: Quito

Fecha: 27/11/2024

**Índice Pág.**

PERFIL DE PROYECTO

1. Introducción….

2. Planteamiento del trabajo….

2.1 Formulación del problema….

2.2 Justificación….

3. Sistema de Objetivos….

3.1. Objetivo General…….

3.2. Objetivos Específicos (03)

4. Alcance….

5.1 Metodología (Marco de trabajo 5W+2H) ….

6. Ideas a Defender ….

7. Resultados Esperados

8. Viabilidad (Ej.) …..

8.1 Humana….

8.1.1 Tutor Empresarial ….

8.1.2 Tutor Académico….

8.1.3 Estudiantes….

8.2 Tecnológica….

8.2.1 Hardware….

8.2.2 Software….

9. Cronograma: ….

10. Bibliografía….

1. **Introducción**

En el contexto actual, las empresas enfrentan desafíos significativos para gestionar eficientemente la asistencia y las jornadas laborales de sus empleados. Esta necesidad se hace más evidente en organizaciones como "Chamorro Lara", donde el control manual de las asistencias puede llevar a inconsistencias, errores humanos y una mayor inversión de tiempo y recursos. Además, la falta de herramientas digitales específicas dificulta la generación de reportes precisos y el manejo ágil de permisos, actividades extra y horarios extendidos.

Este panorama pone de manifiesto la necesidad de un sistema que optimice la gestión del talento humano mediante el uso de tecnologías modernas. Un sistema como TalentFlow no solo resolverá problemas de registro y control, sino que también mejorará la productividad al permitir un seguimiento automatizado y centralizado de las actividades laborales.

El proyecto propuesto busca abordar este nicho de mercado con un sistema de control de asistencias que no solo satisfaga las necesidades actuales de la empresa, sino que también esté diseñado para evolucionar, adaptándose a futuros requerimientos tecnológicos, como la integración con distintos sistemas biométricos y la escalabilidad de sus funcionalidades. Esta solución representa una aportación significativa al desarrollo organizacional, ofreciendo herramientas que impulsan la eficiencia y garantizan el cumplimiento de los estándares de calidad en la gestión del talento humano.

1. **Planteamiento del trabajo**

**2.1 Formulación del problema**

La gestión de asistencias y horarios en la empresa "Chamorro Lara" enfrenta retos que las soluciones manuales o parciales existentes no logran superar de manera efectiva. Actualmente, los métodos tradicionales, como hojas de cálculo o registros físicos, generan retrasos, errores y falta de accesibilidad en tiempo real. Estos problemas afectan tanto la productividad del área de talento humano como la transparencia hacia los empleados, especialmente en tareas como la tramitación de permisos, el cálculo de horas extras y la generación de reportes.

TalentFlow se presenta como una solución integral y moderna para resolver estas dificultades. Este sistema digitalizado automatiza los procesos de registro, cálculo y seguimiento de la asistencia y actividades laborales. Al estar diseñado específicamente para las necesidades de la empresa, asegura la precisión en los datos, reduce el tiempo dedicado a tareas repetitivas y mejora la toma de decisiones mediante reportes claros y detallados.

Además, el sistema aborda las limitaciones de las soluciones actuales al integrar funcionalidades como el soporte para hardware biométrico, lo que asegura la autenticidad de los registros, y la capacidad de adaptarse a futuras demandas mediante una arquitectura escalable y flexible. De este modo, *TalentFlow* no solo resuelve los problemas existentes, sino que también prepara a la empresa para enfrentar desafíos futuros en la gestión del talento humano.

**2.2 Justificación**

El desarrollo de TalentFlow como sistema de control de asistencias y gestión de horarios responde a una problemática recurrente en el ámbito organizacional, donde la eficiencia en la administración del talento humano es un factor crítico para el éxito empresarial. Este proyecto no solo aborda las necesidades específicas de la empresa "Chamorro Lara", sino que también genera un impacto significativo en la investigación y desarrollo de herramientas tecnológicas aplicadas a la gestión organizacional.

Desde el punto de vista científico y tecnológico, TalentFlow aporta en los siguientes aspectos:

1. **Innovación en el diseño de sistemas de gestión**: TalentFlow integra tecnologías avanzadas, como el soporte para hardware biométrico y una arquitectura escalable, estableciendo un estándar de referencia para otros investigadores interesados en la automatización de procesos administrativos.
2. **Optimización de recursos**: La propuesta reduce la dependencia de métodos manuales o parciales, abriendo la puerta a estudios sobre el impacto económico y organizacional de la digitalización en empresas medianas.
3. **Aplicabilidad práctica**: Este proyecto puede servir como caso de estudio para validar enfoques innovadores en la implementación de sistemas de gestión, lo que resulta relevante para académicos y profesionales interesados en el desarrollo de soluciones que equilibren eficiencia y usabilidad.
4. **Escalabilidad y adaptabilidad tecnológica**: La capacidad de integrar funciones adicionales y adaptarse a demandas futuras hace que TalentFlow sea un modelo que puede inspirar investigaciones sobre la evolución y sostenibilidad de los sistemas digitales en diferentes sectores empresariales.
5. **Sistema de Objetivos**

**3.1. Objetivo General**

Desarrollar un sistema de control de asistencias eficiente, seguro y escalable que automatice los procesos de registro, gestión y reporte de jornadas laborales, actividades adicionales y permisos, optimizando la productividad del área de talento humano de la empresa "Chamorro Lara".

**3.2. Objetivos Específicos (03)**

* Garantizar un registro preciso y seguro de las asistencias a través de la integración con hardware biométrico, eliminando errores y manipulaciones comunes en las soluciones manuales. Este nivel de precisión ofrecerá a la empresa un control fiable sobre las jornadas laborales, destacando frente a otros sistemas que carecen de esta capacidad.
* Incorporar la generación automatizada de reportes detallados, personalizables y en tiempo real, adaptados a las necesidades del talento humano y la gerencia. Esto permitirá tomar decisiones informadas de manera más ágil, superando las limitaciones de las soluciones actuales que suelen ser poco intuitivas y están restringidas a formatos básicos.
* Asegurar que el sistema no solo cumpla con los requerimientos actuales, sino que pueda adaptarse a las demandas tecnológicas y organizativas del futuro, diferenciándolo de las soluciones existentes que suelen quedar obsoletas rápidamente.

1. **Alcance**

TalentFlow logrará registrar el ingreso y salida de las diferentes jornadas de los empleados en la compañía “Chamorro Lara”; la gestión de actividades extra para los empleados y también permitirá al empleado tramitar permisos que justifiquen posibles faltas a sus jornadas. Por último, el alcance del sistema permitirá generar reportes de cada una de estas funcionalidades, para el uso del gerente y el talento humano de la compañía.

1. **Marco Teórico**

En el desarrollo del sistema TalentFlow, se emplearán diversas herramientas y entornos de desarrollo integrados (IDEs) para garantizar una solución robusta, eficiente y fácil de mantener. A continuación, se describen las herramientas clave que serán utilizadas en este proyecto:

1. **Flask**  
   Flask es un framework web ligero y flexible para Python, utilizado para construir aplicaciones web de manera rápida y sencilla. Es ideal para proyectos que requieren control sobre la estructura y arquitectura de la aplicación sin la sobrecarga de funcionalidades innecesarias. En el caso de TalentFlow, Flask será utilizado para desarrollar la API REST que gestionará las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para la gestión de los registros de asistencia, permisos, horas extras y reportes. La naturaleza modular de Flask permite crear soluciones escalables y ajustadas a las necesidades del proyecto, facilitando la integración con PostgreSQL y otros componentes del sistema.
2. **PostgreSQL**  
   PostgreSQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto y altamente confiable. Se utilizará para almacenar la información relacionada con los empleados, las jornadas laborales, los permisos, y los reportes generados por el sistema. PostgreSQL es reconocido por su estabilidad y capacidad para manejar grandes volúmenes de datos de manera eficiente, lo que es esencial para el correcto funcionamiento del sistema de control de asistencias. Además, su compatibilidad con SQL y su soporte para transacciones aseguran la integridad de los datos durante las operaciones.
3. **Docker**  
   Docker es una plataforma que permite crear, desplegar y ejecutar aplicaciones dentro de contenedores. Su uso en el proyecto será fundamental para garantizar la portabilidad y consistencia del sistema TalentFlow en diferentes entornos de desarrollo y producción. Mediante Docker, se facilitará la implementación del sistema en diversos servidores, asegurando que funcione de manera coherente sin importar las diferencias entre los entornos de prueba y producción.
4. **Visual Studio Code (VS Code)**  
   Visual Studio Code es un editor de código fuente ligero pero potente que será utilizado como el entorno de desarrollo principal para la creación del sistema. Ofrece soporte para una amplia variedad de lenguajes y frameworks, incluyendo Python y JavaScript, lo que lo convierte en una opción ideal para trabajar en un proyecto que involucra tanto el backend como el frontend. Además, la integración de extensiones en VS Code facilitará tareas como la depuración, la gestión de versiones con Git, y la ejecución de pruebas unitarias.
5. **Git**  
   Git es un sistema de control de versiones distribuido que permite a los desarrolladores gestionar el código fuente del proyecto de manera eficiente. A través de plataformas como GitHub, se podrán realizar seguimientos detallados de los cambios en el código, colaborar de manera efectiva y revertir modificaciones si es necesario. Git es esencial para mantener la calidad del código a lo largo del desarrollo y garantizar la trazabilidad de las modificaciones realizadas por el equipo de trabajo.
6. **PostgreSQL - Herramientas de integración y gestión**  
   La integración con PostgreSQL se realizará mediante herramientas como pgAdmin, que proporcionan una interfaz gráfica para administrar las bases de datos, realizar consultas SQL y gestionar usuarios y permisos. También se utilizarán herramientas de migración y copia de seguridad para asegurar la integridad y recuperación de los datos.

Estas herramientas y tecnologías han sido seleccionadas por su capacidad para integrarse de manera eficiente en el entorno de desarrollo del sistema TalentFlow, y por su probada fiabilidad y escalabilidad en aplicaciones de misión crítica. El uso de estas tecnologías permitirá no solo una solución eficiente a los problemas del control de asistencias de los empleados, sino también una base sólida para futuras mejoras y expansiones del sistema.

**5.1 Metodología (Marco de trabajo 5W+2H)**

#### ¿Qué?

Desarrollar un sistema digital para el control de asistencias y gestión de horarios laborales, que permita automatizar procesos administrativos clave y genere reportes detallados para mejorar la productividad de la empresa.

#### ¿Por qué?

* Optimizar la gestión de talento humano, reduciendo errores y tiempos en la administración de registros manuales.
* Asegurar la precisión y la seguridad en los datos relacionados con las asistencias y jornadas laborales.
* Proveer una herramienta escalable que pueda adaptarse a las necesidades futuras de la empresa.

#### ¿Dónde?

El sistema será implementado en la empresa **Chamorro Lara**, y será accesible desde estaciones de trabajo y dispositivos conectados a la red interna o externa mediante protocolos de seguridad adecuados.

#### ¿Cuándo?

El proyecto se desarrollará en un periodo de **3 meses**, dividido en las siguientes fases:

1. **Planificación**: 2 semanas.
2. **Desarrollo**: 8 semanas.
3. **Pruebas y validación**: 2 semanas.
4. **Implementación y capacitación**: 2 semanas.

#### ¿Quién?

* **Estudiantes desarrolladores** (equipo del proyecto): Encargados del diseño, desarrollo, pruebas e implementación del sistema.
* **Tutor empresarial**: Mediador entre los usuarios finales y el equipo desarrollador.
* **Tutor académico**: Asesor técnico y académico del equipo para garantizar la calidad del proyecto.

#### ¿Cómo?

1. **Análisis de requerimientos**:
   * Identificar necesidades específicas de la empresa mediante reuniones con los stakeholders.
2. **Diseño del sistema**:
   * Crear diagramas UML, arquitecturas y prototipos de UI/UX.
3. **Desarrollo**:
   * Implementar funcionalidades utilizando tecnologías como Python (Flask), PostgreSQL y Docker.
4. **Pruebas**:
   * Ejecutar pruebas unitarias, de integración y de aceptación con herramientas como Pytest y Selenium.
5. **Implementación y capacitación**:
   * Desplegar el sistema en el entorno empresarial utilizando contenedores Docker y capacitar a los usuarios finales.

#### ¿Cuánto?

El presupuesto del proyecto incluye:

* **Recursos Humanos**: Tiempo de los desarrolladores, tutores y usuarios finales para pruebas y validación.
* **Tecnología**: Adquisición de hardware y licencias necesarias para el despliegue del sistema.
* **Software**: Herramientas de desarrollo y gestión, todas seleccionadas para minimizar costos al usar tecnologías de código abierto.

1. **Ideas a Defender**

En el contexto de este proyecto de desarrollo de un sistema de control de asistencias para la empresa "Chamorro Lara", las ideas a defender combinan conocimientos clave de Fundamentos de Ingeniería de Software y Fundamentos de Programación. A continuación, se presentan las principales ideas:

1. **Enfoque en la calidad del software**

El uso de estándares como el IEEE 830 y el IEEE 1012 nos ha permitido establecer una base sólida en la documentación de requisitos de software y en la validación del sistema (IEEE, 1998; IEEE, 2012). Estos estándares aseguran que el desarrollo de TalentFlow cumpla con los criterios de calidad, fiabilidad y desempeño, lo que es fundamental para cualquier sistema crítico como el de gestión de recursos humanos. En particular, la verificación y validación continuas garantizarán que el sistema cumpla las expectativas del usuario final y que su funcionamiento sea consistente y sin errores. Esto demuestra el compromiso con las buenas prácticas de ingeniería de software.

1. **Desarrollo modular y escalable**

El diseño modular de TalentFlow asegura que el sistema sea fácilmente escalable y mantenible. Esta idea se basa en el principio de modularidad, que se aprende tanto en Fundamentos de Ingeniería de Software como en Programación, ya que permite la creación de componentes independientes que pueden modificarse sin afectar al sistema global. La estructura modular del software también facilitará su actualización con nuevas funcionalidades, como la integración con otros sistemas biométricos en el futuro (IEEE, 1998; IEEE, 2012).

1. **Cumplimiento de normativas laborales**

Una de las áreas clave en el proyecto es el cumplimiento con el Código del Trabajo de Ecuador. El sistema debe ajustarse a las normativas sobre jornada laboral, remuneración por horas extraordinarias, y permisos laborales, lo que se logra mediante la automatización de procesos como el cálculo de horas extra y la gestión de permisos. Estas funcionalidades están alineadas con las leyes mencionadas en el Código del Trabajo, lo que asegura que TalentFlow sea legalmente conforme y ofrezca un servicio útil para la administración de personal (R.O. 167, Art. 47, 2023).

1. **Implementación de interfaces de usuario y software amigables**

En cuanto a la interfaz de usuario (UI), se ha hecho énfasis en el diseño de una interfaz clara, fácil de usar y eficiente. Este concepto, aprendido en Fundamentos de Programación, se pone en práctica para garantizar que los empleados y los administradores puedan interactuar con el sistema de manera efectiva. El sistema también está diseñado para ser compatible con diferentes plataformas (Windows, macOS, Linux), lo cual refleja un enfoque en la portabilidad, uno de los aspectos fundamentales de la ingeniería de software (IEEE, 2006).

1. **Seguridad en la gestión de datos**

TalentFlow incluye mecanismos de seguridad como el cifrado de contraseñas y protección contra accesos no autorizados, siguiendo los principios de seguridad aprendidos en Fundamentos de Ingeniería de Software. Dado que se maneja información sensible sobre los empleados, como sus jornadas laborales, permisos y horas extra, la seguridad es una prioridad. Además, el sistema está diseñado para garantizar la disponibilidad de los datos, lo que permite a los usuarios acceder al sistema de manera continua (IEEE-CS/ACM, s.f.).

1. **Optimización del rendimiento**

La optimización de un diseño que optimiza el rendimiento del sistema, utilizando técnicas de programación eficiente para garantizar que el sistema pueda manejar grandes volúmenes de datos sin afectar su desempeño. Esto se logra mediante la correcta elección de algoritmos y la arquitectura de software adecuada (IEEE 1061, 1998).

1. **Resultados Esperados**
2. Cumplimiento de los Requisitos Funcionales y No Funcionales

Uno de los principales resultados esperados es que el sistema desarrollado cumpla con los requisitos establecidos en la especificación. Esto incluye tanto los requisitos funcionales, como la correcta implementación de las funcionalidades del sistema (como la gestión de empleados, el control de horas de trabajo, permisos, etc.), como los no funcionales (por ejemplo, la seguridad, la escalabilidad, y el rendimiento). La correcta integración de estos requisitos será validada durante el proceso de verificación y validación.

1. Aplicación de Buenas Prácticas de Ingeniería de Software

Se espera que el proyecto demuestre el uso adecuado de las buenas prácticas de ingeniería de software, como la correcta elaboración de la documentación del proyecto (por ejemplo, especificaciones, diagramas UML, pruebas de software), la aplicación de principios de diseño como el modularidad, y la implementación de una arquitectura de software adecuada y escalable. Estas prácticas deberían garantizar la calidad del software y su facilidad de mantenimiento.

1. Desarrollo de un Código Limpio y Eficiente

Se espera que, al aplicar los principios de los Fundamentos de Programación, el código resultante sea limpio, bien estructurado y fácil de entender. Esto incluye el uso adecuado de técnicas de refactorización, el cumplimiento de las mejores prácticas de programación (como el uso de patrones de diseño cuando sea necesario) y la creación de código modular que facilite futuras modificaciones o extensiones.

1. Implementación de Funcionalidades de Software Seguro

Dado que en el proyecto se están abordando temas de desarrollo de software seguro, se espera que el sistema implemente medidas de seguridad efectivas, como la autenticación de usuarios, la protección contra ataques comunes (como inyecciones de SQL o XSS), y la implementación de políticas adecuadas de control de acceso.

1. Despliegue y Gestión en un Entorno de Producción

Un resultado esperado es que el sistema esté correctamente desplegado en un entorno controlado utilizando tecnologías como Docker, lo que garantizaría su portabilidad y capacidad de ejecución en diferentes plataformas sin problemas. Esto incluiría la creación de scripts de automatización para la gestión y despliegue del software en un servidor.

1. Entrega de un Producto de Alta Calidad

El resultado final debería ser un sistema robusto y libre de defectos, que no solo cumpla con los requisitos técnicos y funcionales, sino que también sea fácil de usar para los usuarios finales, asegurando la satisfacción del cliente. El sistema también debe ser confiable, con una tasa de fallos mínima y una experiencia de usuario fluida.

1. Documentación Completa y Clara

Se espera que toda la documentación asociada al proyecto esté completa y clara, incluyendo desde los requisitos y la arquitectura del sistema, hasta los planes de prueba y los manuales de usuario y de instalación. Esta documentación debe ser útil tanto para futuros desarrolladores como para los usuarios del sistema.

1. Evaluación de la Calidad mediante Pruebas de Software

Como parte integral del proyecto, se espera realizar pruebas exhaustivas del sistema para garantizar su funcionalidad, rendimiento y seguridad. Las pruebas unitarias, de integración y de aceptación del usuario serán realizadas y documentadas, y cualquier problema o fallo detectado será solucionado antes de la entrega final.

1. Desarrollo de Habilidades Técnicas y de Gestión de Proyectos

A nivel personal, el proyecto permitirá desarrollar y mejorar mis habilidades tanto en programación como en gestión de proyectos de software. A través de la implementación de metodologías ágiles y el trabajo en equipo, espero mejorar mis capacidades de planificación, ejecución y seguimiento del proyecto, así como también fortalecer mi capacidad de liderazgo y resolución de problemas.

1. **Viabilidad**

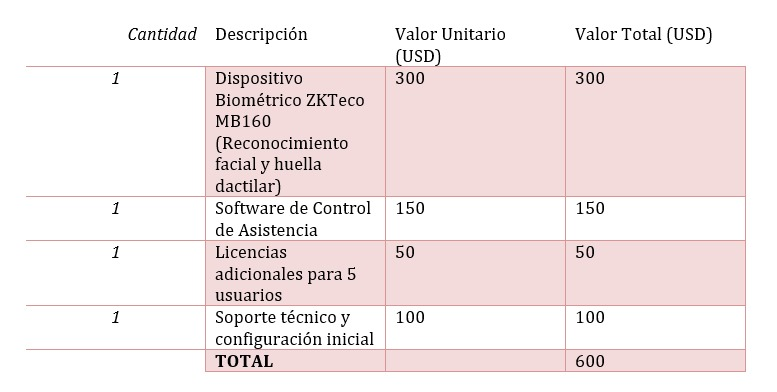


Tabla 1 Presupuesto del proyecto

Debe explicar los recursos necesarios para su proyecto y adicionalmente la viabilidad del punto 8.1. y 8.2

**8.1 Humana**

**8.1.1 Tutor Empresarial**

|  |  |
| --- | --- |
| Nombre | Fdo. D. Robert Chamorro |
| Rol | Cliente |
| Categoría profesional | Dueño de la empresa Chamorro Lara |
| Responsabilidades | Ser mediador entre los usuarios finales y los desarrolladores. Ayudar a mantener el enfoque en los objetivos del proyecto y asegurar que todas las voces sean escuchadas. |
| Información de contacto |  |
| Aprobación | NA |

**8.1.2 Tutor Académico**

|  |  |
| --- | --- |
| Nombre | Ing. Jenny Ruiz |
| Rol | Tutor académico |
| Categoría profesional | Docente de Ingeniería de la Seguridad. |
| Responsabilidades | Gestionar los grupos de trabajo de los empleados e implementar la funcionalidad de importación de las horas de inicio y fin de cada una de las actividades de los empleados. |
| Información de contacto |  |
| Aprobación | NA |

**8.1.3 Estudiantes**

|  |  |
| --- | --- |
| Nombre | Bolaños Sebastián |
| Rol | Desarrollador |
| Categoría profesional | Estudiantes de ingeniería de software |
| Responsabilidades | Gestionar los grupos de trabajo de los empleados e implementar la funcionalidad de importación de las horas de inicio y fin de cada una de las actividades de los empleados. |
| Información de contacto |  |
| Aprobación | NA |

|  |  |
| --- | --- |
| Nombre | Caiza Alisson |
| Rol | Desarrollador |
| Categoría profesional | Estudiantes de ingeniería de software |
| Responsabilidades | Gestionar los grupos de trabajo de los empleados e implementar la funcionalidad de importación de las horas de inicio y fin de cada una de las actividades de los empleados. |
| Información de contacto |  |
| Aprobación | NA |

|  |  |
| --- | --- |
| Nombre | Esparza Marco |
| Rol | Desarrollador |
| Categoría profesional | Estudiantes de ingeniería de software |
| Responsabilidades | Gestionar los grupos de trabajo de los empleados e implementar la funcionalidad de importación de las horas de inicio y fin de cada una de las actividades de los empleados. |
| Información de contacto |  |
| Aprobación | NA |

|  |  |
| --- | --- |
| Nombre | Reinoso Jerly |
| Rol | Desarrollador |
| Categoría profesional | Estudiantes de ingeniería de software |
| Responsabilidades | Gestionar los grupos de trabajo de los empleados e implementar la funcionalidad de importación de las horas de inicio y fin de cada una de las actividades de los empleados. |
| Información de contacto |  |
| Aprobación | NA |

**8.2 Tecnológica**

**8.2.1 Hardware**

Para implementar y operar el sistema TalentFlow, se requieren los siguientes componentes de hardware:

1. Servidor de Aplicaciones:
   * Procesador: Intel Xeon o AMD Ryzen (equivalente a 6 núcleos o superior).
   * Memoria RAM: Mínimo 8 GB.
   * Almacenamiento: SSD con al menos 125 GB para garantizar rapidez en la lectura/escritura de datos.
   * Conectividad: Puerto Ethernet Gigabit y soporte para conexiones redundantes.
2. Simulador de Biométricos:
   * Conexión: USB o IP para integración con el sistema central.
3. Estaciones de Trabajo:
   * Procesador: Intel Core i5 o equivalente.
   * RAM: 8 GB mínimo.
   * Periféricos: Ratón, teclado y webcam para soporte técnico.
4. Equipos de Red:
   * Switches y routers con capacidad para manejar tráfico de datos del servidor y dispositivos biométricos.

**8.2.2 Software**

Backend:

* Lenguaje: Python
* Framework: Flask para desarrollo de la API REST.
* Base de Datos: PostgreSQL.

Frontend:

* Lenguajes: HTML5, CSS3 y JavaScript, SpringBoot.

Sistemas de Control de Versiones:

* GitHub.

Herramientas de Desarrollo:

* IDE: Visual Studio Code con extensiones para JavaScript y PostgreSQL.

Pruebas de Software:

* Herramientas: Pytest (pruebas unitarias) y Selenium (pruebas automatizadas de UI).

Integración y Gestión:

* pgAdmin para administrar bases de datos PostgreSQL.
* Jenkins o GitHub Actions para integración continua y despliegue automatizado (CI/CD).

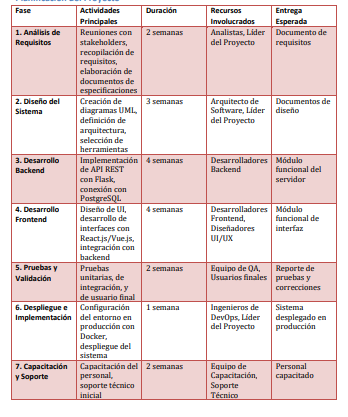
1. **Conclusiones y recomendaciones**

## **9.1 Conclusiones**

* El desarrollo del sistema TalentFlow abordará eficazmente los retos actuales en la gestión de asistencias y horarios de la empresa "Chamorro Lara", optimizando procesos administrativos críticos.
* La implementación de buenas prácticas de ingeniería de software asegura la calidad del producto, cumpliendo con estándares internacionales y normativas locales aplicables.
* La digitalización de los procesos reducirá significativamente los errores manuales, mejorará la precisión de los datos y permitirá una gestión ágil del talento humano.

## **9.2 Recomendaciones**

* Capacitación Continua: Capacitar al personal administrativo en el uso de TalentFlow y su interfaz para garantizar una transición fluida del sistema manual a la digital.
* Monitoreo y Mantenimiento: Establecer protocolos regulares para el mantenimiento de hardware biométrico y actualizaciones del sistema.
* Escalabilidad a Futuro: Considerar la integración de nuevas funcionalidades, como el uso de inteligencia artificial para análisis predictivos de desempeño y patrones de asistencia.

1. **Planificación para el Cronog rama:**
2. **Referencias**

* IEEE. (1998). *Guía para la Especificación de Requisitos de Software*. IEEE. Recuperado de <https://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf>
* IEEE. (2012, marzo 29). *Estándar para la Verificación y Validación de Software (IEEE 1012)*. IEEE. Recuperado de <https://standards.ieee.org/standard/1012-2012.html>
* IEEE. (2006, marzo 30). *Estándar para la Documentación de Arquitectura de Software (IEEE 1074)*. IEEE. Recuperado de <https://standards.ieee.org/standard/1074-2006.html>
* IEEE. (1998). *Estándar para Descripciones de Procesos de Ingeniería de Software (IEEE 1061)*. IEEE. Recuperado de <https://standards.ieee.org/standard/1061-1998.html>
* IEEE-CS/ACM. (n.d.). *Código de Ética y Práctica Profesional en Ingeniería de Software*. Recuperado de <https://ethics.acm.org/wp-content/uploads/2016/07/SE-code-spn.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 47. De la jornada máxima*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 49. Jornada Nocturna*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 42 - 11. Obligaciones del empleador en cuanto a permitir a los trabajadores faltar o ausentarse al trabajo*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 42 - 18. Obligaciones del empleador en cuanto a pagar al trabajador la remuneración correspondiente al tiempo perdido cuando se vea imposibilitado de trabajar por culpa del empleador*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 54. Pérdida de Remuneración*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 55. Remuneración por horas suplementarias y Extraordinarias*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 57. División de la Jornada*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 59. Indemnización al empleador*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>
* Código del Trabajo. (2023, febrero 11). *R.O. 167. Art 60. Recuperación de horas de trabajo*. Recuperado de <https://www.gob.ec/sites/default/files/regulations/2018-10/C%C3%B3digo-del-Trabajo.pdf>