}1

1. **PARTE I**

| **1. Antecedentes Personales** |
| --- |
| A continuación, se presenta una tabla en la que debes completar la información solicitada. |

| Nombre estudiante | **Javier Ignacio Rivas Gazmuri, Carlos Eduardo Colmenarez, Francisco Eljuri** |
| --- | --- |
| Rut | **19.512.852-9, 27.009.196-2, 25.540.069-K** |
| Carrera | **Ingeniería en Informática** |
| Sede | **Padre Alonso de Ovalle** |

| **2. Descripción Proyecto APT** |
| --- |

| Nombre del proyecto | INTEGRAJOB |
| --- | --- |
| Área (s) de desempeño(s) | * **Desarrollo de soluciones informáticas:** Diseño e implementación de un sistema integral que incluye un chatbot interactivo en el frontend, capaz de guiar al usuario en la creación de un perfil/currículum y facilitar procesos de inclusión laboral automatizados. * **Calidad de software:** Diseño e implementación de pruebas funcionales, unitarias e integrales; aplicación de estándares de ingeniería de software para asegurar la confiabilidad del sistema en cada uno de sus módulos (frontend, backend, servicios de scraping y APIs). * **Gestión de Proyectos Tecnológicos:** Planificación, organización, seguimiento y documentación del proyecto con metodologías tradicionales complementadas con herramientas de control de tareas y roles de equipo. * **Análisis y Evaluación de Soluciones:** Verificación de la factibilidad técnica, escalabilidad en la nube y pertinencia social de la solución propuesta, garantizando impacto en el contexto real de la empleabilidad inclusiva. * **Infraestructura y Arquitectura de Nube:** Diseño y operación de la plataforma en la nube que soporta la solución, incluyendo la definición de ambientes, y optimización de recursos para garantizar escalabilidad, disponibilidad y rendimiento. * **Experiencia de Usuario:** Diseño de interfaces accesibles y usables para un perfil de usuario que fomenten la inclusión laboral, asegurando accesibilidad web * **Integración de Sistemas:** Conexión con fuentes externas de datos (APIs de empleo, servicios públicos, scraping, etc.), asegurando interoperabilidad y consistencia. * **Gestión de Datos y Analítica:** Modelado, almacenamiento y procesamiento de la información obtenida vía scraping; normalización de datos laborales y generación de reportes e indicadores de inserción laboral para la toma de decisiones. |
| Competencias | * **Definición de requisitos funcionales y no funcionales** del sistema. * **Definición de roles, asignación de tareas y seguimiento de avances** en el equipo de trabajo. * **Diseño e implementación de módulos backend**, incluyendo servicios de scraping, integración de APIs y lógica de filtrado de empleos. * **Diseño e implementación de interfaces frontend interactivas**, con un chatbot accesible que genere perfiles laborales de manera automática. * **Gestión y análisis de datos laborales**, mediante normalización, procesamiento y generación de reportes e indicadores. * **Aplicación de metodologías tradicionales de gestión de proyectos**, complementadas con prácticas de control y seguimiento. * **Pruebas de validación y aseguramiento de la calidad** en frontend, backend y servicios de datos. * **Diseño e implementación de infraestructura en la nube** para garantizar seguridad, escalabilidad y disponibilidad de la solución. * **Aplicación de principios de accesibilidad y usabilidad** para fomentar la inclusión laboral mediante el software. * **Implementación de buenas prácticas de seguridad informática**, protegiendo datos personales sensibles de los usuarios. |

| **3. Fundamentación Proyecto APT** |
| --- |

| Relevancia del proyecto APT | **Problema:**  Las ofertas laborales rara vez incorporan **acomodos razonables para personas neurodivergentes** (por ejemplo, instrucciones claras y escritas, agenda predecible, tareas atomizadas, posibilidad de trabajo asincrónico o ambientes silenciosos), ni tampoco consideran de manera explícita la **accesibilidad física y logística** (rampas, ascensores en funcionamiento, baños accesibles, teletrabajo o sedes únicas). Esta carencia dificulta que personas neurodivergentes y con movilidad reducida puedan **identificar vacantes compatibles, postular con confianza y acceder en igualdad de condiciones al mercado laboral**.  **Justificación técnica y formativa:**  El proyecto propone el desarrollo de una plataforma que integra:   * **Extracción, transformación y carga de datos (ETL)** desde distintas fuentes de empleo. * **Modelado relacional y procesamiento de datos**, asegurando consistencia e interoperabilidad. * **Algoritmos de ranking basados en reglas y similitud semántica**, que permiten filtrar vacantes según criterios de inclusión y accesibilidad. * **Arquitectura de servicios mediante API REST**, garantizando escalabilidad e integración con el frontend conversacional. * **Pruebas de calidad de software (QA)** en backend, frontend y servicios de datos, con foco en accesibilidad. * **Despliegue en infraestructura de nube**, asegurando disponibilidad, seguridad y capacidad de crecimiento. * **Interfaz conversacional mediante chatbot inclusivo**, que guía al usuario en la construcción de su perfil y filtra oportunidades de empleo pertinentes.   Esta combinación no solo responde a una **necesidad social real** —la inclusión laboral de personas con neurodivergencia y movilidad reducida, históricamente subatendidas—, sino que también permite al equipo aplicar y fortalecer competencias del perfil de egreso en **desarrollo de software, gestión y análisis de datos, aseguramiento de la calidad y arquitectura de servicios en la nube**. |
| --- | --- |
| Descripción del Proyecto APT | **Solución propuesta:**  INTEGRAJOB es una plataforma digital que incorpora un **chatbot conversacional inclusivo** como punto de interacción principal. Este chatbot:   * **Recolecta y organiza ofertas laborales** desde distintas fuentes, destacando aquellas que explicitan condiciones de accesibilidad física y acomodos razonables para personas neurodivergentes. * **Construye un perfil personalizado de accesibilidad** para cada persona usuaria, a partir de sus preferencias, necesidades y características individuales. * **Entrega recomendaciones explicadas y justificadas**, vinculando las necesidades de la persona con los atributos de la oferta laboral. Ejemplos:   + “Se sugiere está vacante porque solicitaste instrucciones claras y la oferta declara checklist de tareas.”   + “Esta opción es priorizada porque indicaste evitar escaleras y la sede informa contar con ascensor operativo.”   **Arquitectura resumida (cómo funciona el sistema):**  El sistema integra tres componentes principales que trabajan de forma coordinada:   1. **Interfaz conversacional (frontend):** un chatbot que guía al usuario en un diálogo sencillo e inclusivo, permitiendo recopilar información y presentar oportunidades laborales filtradas. 2. **Lógica de procesamiento (backend):** un motor de reglas y análisis semántico que compara las necesidades del usuario con las características de las vacantes, explicando el “porqué” de cada recomendación. 3. **Gestión de datos:** un proceso automatizado que recolecta, transforma y clasifica ofertas laborales, alimentándose en una base estructurada y actualizada continuamente. Este proceso incorpora retroalimentación de los usuarios para mejorar progresivamente la pertinencia de los resultados.   De esta manera, el proyecto no solo responde a una necesidad social urgente vinculada con la inclusión laboral de personas neurodivergentes y con movilidad reducida, sino que también desarrolla un ciclo completo de **interacción, procesamiento y análisis de datos**, orientado a garantizar **accesibilidad, pertinencia y confianza** en el proceso de búsqueda de empleo. |
| Pertinencia del proyecto con el perfil de egreso | El proyecto demuestra una elevada pertinencia con el perfil de egreso de la carrera, puesto que moviliza las competencias clave que se esperan del egresado: diseñar, desarrollar, implementar y desplegar soluciones informáticas para problemas complejos, aplicando estándares tecnológicos, metodologías, innovación, pensamiento crítico, ética profesional y aprendizaje constante, en un entorno digital y colaborativo.  Esta pertinencia radica en que el proyecto exige la puesta en práctica real de estas competencias. Por ejemplo, el diseño y despliegue de soluciones backend robustas obliga al equipo a aplicar estándares de calidad y seguridad, gestionar proyectos de principio a fin y colaborar en equipos técnicos multidisciplinarios. Asimismo, la inclusión de criterios de accesibilidad y ética aplicada fortalece la dimensión humanista del perfil de egreso, reforzando su enfoque centrado en las personas y orientado al bien común |
| Relación con los intereses profesionales | El proyecto se alinea directamente con el perfil de egreso de la carrera al exigir el análisis de requerimientos, la creación de soluciones backend robustas, la aplicación de estándares técnicos y éticos, y el uso de prácticas colaborativas interdisciplinarias en su ciclo de ejecución; estas dinámicas permiten activar competencias tales como innovación tecnológica, gestión de proyectos, comunicación efectiva, enfoque centrado en las personas y compromiso con el bien común.  Esta relación se evidencia en el proceso mismo del proyecto, que implica etapas de ingeniería de requerimientos, diseño de arquitectura, desarrollo e integración, evaluación de calidad y presentación de resultados. Cada fase refuerza competencias definidas en el perfil, como el uso de metodologías formales, la reflexión ética al incorporar accesibilidad, la comunicación efectiva en equipos diversos y el aprendizaje continuo al lidiar con tecnologías emergentes como ETL, automatización y backend con Java/Spring Boot, además del compromiso social contenido en la misión del proyecto. |
| Factibilidad de desarrollo del Proyecto APT | El proyecto **INTEGRAJOB** es técnicamente factible dentro del semestre, considerando las competencias actuales del equipo y los recursos disponibles. **Duración del semestre y horas asignadas** La planificación contempla el desarrollo de un **MVP (producto mínimo viable)** en el período académico, priorizando las funciones esenciales: recolección y procesamiento de datos laborales, construcción de perfiles de accesibilidad, chatbot conversacional y recomendaciones explicadas. El número de horas asignadas a la asignatura, junto con el trabajo autónomo, resulta suficiente para alcanzar un prototipo funcional y evaluable. **Materiales y stack tecnológico** El equipo dispone de los siguientes recursos técnicos y conocimientos:   * **Selenium y Python:** experiencia consolidada en scraping y procesos ETL para recolectar, transformar y normalizar datos laborales. * **Gestión de datos y servidores en la nube:** capacidad para modelar bases de datos, desplegar entornos en cloud y gestionar infraestructura básica. * **QA y aseguramiento de la calidad:** aplicación de pruebas unitarias, funcionales y de validación para garantizar confiabilidad en el sistema.   Áreas en proceso de aprendizaje que representan un desafío, pero que son alcanzables durante el semestre:   * **Spring Boot y Java:** base para la lógica backend y la arquitectura de servicios. * **React y Vite:** desarrollo de la interfaz frontend y el chatbot conversacional. * **Arquitectura de chatbot:** integración de la lógica conversacional con los datos procesados y el motor de recomendaciones.   El equipo ya cuenta con entornos de desarrollo y herramientas colaborativas, por lo que no se requiere inversión adicional significativa en software o hardware. **Factores externos que facilitan el desarrollo**  * Disponibilidad de **fuentes de empleo en línea** que permiten alimentar los procesos de scraping y entrenamiento del motor de recomendaciones. * Existencia de **documentación, librerías y frameworks consolidados**, lo que reduce la curva de aprendizaje en los stacks nuevos. * **Complementariedad de habilidades dentro del equipo**, equilibrando experiencia en scraping/ETL y QA con la disposición a aprender backend y frontend. * Relevancia social del proyecto, que aumenta la motivación y compromiso del equipo.  **Factores que pueden dificultar el desarrollo**  * **Curva de aprendizaje en Spring Boot, Java, React y arquitectura de chatbot**, que podría ralentizar el desarrollo inicial. * **Complejidad en la integración** entre scraping, procesamiento de datos, backend y frontend. * **Tiempo limitado del semestre**, que obliga a priorizar funcionalidades clave sobre mejoras avanzadas.  **Estrategias de mitigación**  * Definir un **alcance claro y realista**, enfocando el semestre en el MVP (scraping + normalización de datos + backend básico + chatbot funcional). * **Dividir roles en el equipo** según fortalezas: quienes dominan scraping/ETL se centran en la base de datos y datos laborales; quienes aprenden React/Java se enfocan en los módulos de interfaz y backend. * **Plan de aprendizaje acelerado**, usando tutoriales, documentación oficial y ejemplos prácticos de Spring Boot y React desde el inicio. * **Integración temprana y continua** para detectar incompatibilidades a tiempo.   **En conclusión**, aunque el proyecto incorpora stacks parcialmente nuevos para el equipo, su factibilidad es alta gracias a la experiencia previa en scraping, ETL, QA y despliegue en nube, sumada a la planificación estratégica y la motivación por resolver una **problemática social relevante**. |

1. **PARTE II**

| **4. Objetivos** |
| --- |

| Objetivo general | Diseñar e implementar una plataforma digital inclusiva que, a través de un **chatbot conversacional en una interfaz web**, facilite la **búsqueda y recomendación personalizada de ofertas laborales** para personas neurodivergentes y con movilidad reducida. |
| --- | --- |
| Objetivos específicos | 1. **Realizar un levantamiento de requerimientos funcionales y no funcionales** mediante entrevistas, análisis documental y revisión de normativa vigente, incorporando criterios de accesibilidad y necesidades de inclusión laboral para orientar el diseño del sistema. 2. **Definir una taxonomía de accesibilidad y acomodos razonables** que considere necesidades de personas neurodivergentes y con movilidad reducida, para orientar el filtrado de ofertas laborales. 3. **Diseñar e implementar procesos de ingesta y transformación de datos (ETL)** a partir de fuentes laborales seleccionadas, normalizando la información en una base de datos estructurada. 4. **Construir un motor de correspondencia (matching)** que combine reglas de accesibilidad con técnicas de análisis semántico, capaz de priorizar vacantes compatibles y justificar el “por qué” de cada recomendación. 5. **Desarrollar una interfaz conversacional (chatbot web inclusivo)** que recoja necesidades y preferencias de las personas usuarias y entregue recomendaciones explicadas en un lenguaje claro y accesible. 6. **Diseñar y documentar una arquitectura de servicios escalable**, que permita la comunicación entre la interfaz conversacional, el backend de procesamiento y la base de datos laboral. 7. **Implementar un sistema de retroalimentación de usuarios**, para ajustar y mejorar progresivamente la pertinencia de las recomendaciones entregadas. 8. **Aplicar pruebas de calidad, accesibilidad y usabilidad** con un grupo reducido de personas del público objetivo, evaluando métricas de compatibilidad, confianza y facilidad de uso. 9. **Gestionar el proyecto mediante planificación, roles y seguimiento**, asegurando el cumplimiento de plazos, la integración de componentes y la documentación del proceso de desarrollo. |

| **5. Metodología** |
| --- |

| Descripción de la Metodología |
| --- |
| El desarrollo del proyecto **INTEGRAJOB** se abordará mediante la **metodología tradicional en cascada**, dado que permite una planificación estructurada y una ejecución secuencial de las fases del ciclo de vida del software. Este enfoque es adecuado para el contexto académico del semestre, ya que facilita la definición anticipada de entregables, la asignación de responsabilidades y el control de avances en cada etapa.  Las fases consideradas son:   1. **Levantamiento de requisitos:** identificación de necesidades funcionales y no funcionales, con énfasis en criterios de accesibilidad y pertinencia social. 2. **Diseño:** elaboración de la arquitectura general, definición de la base de datos, estructura del chatbot y flujo de interacción. 3. **Implementación:** desarrollo de los módulos backend, frontend y procesos de scraping/ETL, de manera secuencial según lo planificado. 4. **Pruebas:** aplicación de pruebas funcionales, de accesibilidad y usabilidad, con retroalimentación de usuarios simulados del público objetivo. 5. **Despliegue:** instalación del sistema en un entorno en la nube y entrega de un prototipo funcional. 6. **Documentación y cierre:** registro de decisiones técnicas, resultados obtenidos y evaluación del cumplimiento de objetivos.  **Organización del trabajo grupal** El proyecto será desarrollado en equipo, dividiendo responsabilidades de acuerdo con las competencias y fortalezas de cada integrante:   * **Scraping y ETL:** responsables de la recolección, transformación y normalización de datos laborales. * **Backend:** responsables de la lógica de matching, APIs y procesamiento de datos. * **Frontend y chatbot:** responsables de la interfaz conversacional y la experiencia de usuario. * **QA y documentación:** responsables de la validación del sistema, control de calidad y consolidación de avances.  **Reuniones y control de entregables** Para garantizar la coordinación, se establecerán **cuatro reuniones formales**, distribuidas en el semestre, bajo dos modalidades:   * **Dos reuniones presenciales**, orientadas a la planificación inicial y revisión de hitos críticos. * **Dos reuniones virtuales**, destinadas a seguimiento de avances y resolución de contingencias.   En cada reunión se definirán y revisarán **entregables específicos**, asegurando que los resultados de una fase estén completos antes de pasar a la siguiente. Además, se registrarán acuerdos y compromisos en actas de reunión, lo que permitirá un control transparente del proceso y la responsabilidad de cada integrante.  **En síntesis**, la metodología en cascada, complementada con reuniones periódicas y división clara de roles, asegura que el desarrollo de INTEGRAJOB avance de manera organizada, verificable y orientada al cumplimiento de los objetivos planteados. |

| **6. Evidencias** |
| --- |
| A continuación, describe qué evidencias serán evaluadas en el informe de avance y en el informe final de tu proyecto APT. Estas evidencias deben ser acordadas con tu docente. Se entenderá por evidencia los productos que se desarrollen durante el proyecto y cuyo propósito sea visibilizar o documentar cómo se ha implementado el trabajo. |

| **Tipo de evidencia**  **(avance o final)** | **Nombre de la evidencia** | **Descripción** | **Justificación** |
| --- | --- | --- | --- |
|  |  | Describe las evidencias acordadas con tu docente, siempre teniendo en mente que estas deben dar cuenta del desarrollo de tu Proyecto APT. |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

| **7. Plan de Trabajo** |
| --- |

| **Plan de Trabajo Proyecto APT** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Competencia o unidades de competencias** | **Nombre de Actividades/Tareas** | **Descripción Actividades/Tareas** | **Recursos** | **Duración de la actividad** | **Responsable** | **Observaciones** |
| **Requisitos** | | | | | | |
| **Análisis de requisitos, modelado de datos** | Definir alcance y taxonomía de accesibilidad | Reunir requisitos, crear taxonomía (acomodos razonables, accesibilidad física). | Google Docs, entrevistas ligeras, papers | 2 semanas | Javier Rivas  Carlos Colmenarez  Francisco Eljuri | Riesgo: falta de referentes; usar fuentes oficiales (Ley de Inclusión, SENADIS). |
| **Diseño** | | | | | | |
| **Java Spring Boot**  **Backend / API REST** | Diseñar e implementar API básica | Endpoints,en Spring Boot. | Spring Boot, Java, Postman | 4 semanas | Javier Rivas  Carlos Colmenarez  Francisco Eljuri | Riesgo: curva de aprendizaje; mitigar con tutoriales y ejemplos. |
| **Infraestructura y despliegue** | Configurar entorno en la nube y Docker | Montar API y frontend en contenedores; BD en servicio cloud. | Docker, GitHub Actions | 2 semanas | Francisco Eljuri  Carlos Colmenarez | Priorizar un único entorno |
| **Implementación** | | | | | | |
| **Scraping Java con Selenium.** | Scraping | Desarrollo de scripts Selenium/Java para extraer y normalizar ofertas de empleo. | Java, Selenium | 3 semanas | Javier Rivas | Dependencia de cambios en sitios web; mitigar con logs y fallback. |
| **Chatbot Frontend** | Desarrollar interfaz web + chatbot | Chat simple con flujo guiado para perfil y recomendaciones. | React/Vite, BotUI | 4 semanas | Francisco Eljuri  Carlos Colmenarez | Curva de aprendizaje alta; iniciar pronto con prototipo mínimo. |
| **Motor de correspondencia** | Motor de correspondencia (matching) | Implementar el módulo backend que compara el perfil de accesibilidad del usuario con las ofertas laborales, aplicando reglas y análisis semántico para generar un ranking con explicaciones. | Spring Boot, MySql, Python (ETL + NLP), Docker, Postman. | 3 semanas | Javier Rivas  Carlos Colmenarez  Francisco Eljuri | Dificultades: Curva de aprendizaje en NLP e integración Java–Python;  facilitadores: taxonomía definida, experiencia previa en scraping y QA. |
| **Pruebas** | | | | | | |
| **QA y validación** | Pruebas funcionales y de accesibilidad | Unitarias, integración, validación con herramientas. | Selenium, Testng | 2 semanas | Javier Rivas | Riesgo: usar pruebas automáticas si es necesario. |
| **Documentación y Cierre** | | | | | | |
| **Documentación y cierre** | Manual de usuario + retrospectiva | Manual sencillo, guía de instalación, lecciones aprendidas. | Word, GitHub | 1 semana   |  | | --- | | Javier Rivas  Carlos Colmenarez  Francisco Eljuri | Puede hacerse paralelo al despliegue final. |

| **8. Carta Gantt** |
| --- |

| **Actividad** | **Fase 1** | | | | **Fase 2** | | | | | | | | | | | **Fase 3** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S 1** | **S 2** | **S 3** | **S 4** | **S 5** | **S 6** | **S 7** | **S 8** | **S 9** | **S 10** | **S 11** | **S 12** | **S 13** | **S 14** | **S 16** | | **S 17** | **S 18** |
| Definir alcance y taxonomía de accesibilidad. | **X** | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Análisis de requisitos y modelo de datos. |  | **X** | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Infraestructura y despliegue en nube/Docker (primer entorno de pruebas). |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Diseño inicial del Chatbot Frontend (React/Vite). |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Backend API básica en Spring Boot. |  |  |  |  | **X** | **X** | **X** |  |  |  |  |  |  |  |  | |  |  |
| Desarrollo del motor de correspondencia (matching + reglas).  Scraping inicial (Scraping + Python). |  |  |  |  | **X** | **X** | **X** | **X** | **X** | **X** | **X** |  |  |  |  | |  |  |
| Avance en Chatbot + integración con API (BNE). |  |  |  |  |  |  |  |  | **X** | **X** | **X** |  |  |  |  | |  |  |
| QA y validación (unitarias, accesibilidad). |  |  |  |  |  |  |  |  |  | **X** | **X** | **X** |  |  |  | |  |  |
| Integración final, pruebas con usuarios y métricas. |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  | |  |  |
| Documentación (manual de usuario, guía de instalación). |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  | |  |  |
| Retrospectiva y entrega final. |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  | |  |  |