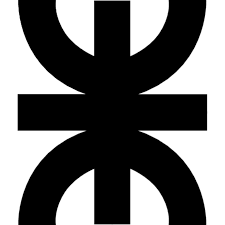
**Universidad Tecnológica Nacional**

**Facultad Regional Rosario**

**Tecnicatura Universitaria en Programación**

****

**Práctica Profesional Supervisada**

Documentación del proyecto:

“Bolsa de Trabajo de la UTN”

Año - 2TUP4

integrantes:

Julian Vazquez - 50496

Matias Napolitano - 50618

Ariel Salinas - 50583

**ÍNDICE**

Tabla de contenido

[**Metodología de Trabajo** 2](#_Toc167459194)

[**Introducción** 2](#_Toc167459195)

[**Nuestros Propósitos** 2](#_Toc167459196)

[**Características Distintivas** 3](#_Toc167459197)

[**Alcance del Proyecto** 3](#_Toc167459198)

[**Algunas cuestiones fuera del alcance:** 3](#_Toc167459199)

[**CURS – Buscar trabajo en la bolsa de trabajo de la UTN** 4](#_Toc167459200)

# 

# **Metodología de Trabajo**

La metodología ágil SCRUM fue la que utilizamos a lo largo de todo el proyecto, nos divididos en roles en la que cada uno de los participantes tenía actividades designadas y una vez a la semana realizamos la daily a cargo del scrum master, en donde por turnos íbamos contando lo que hicimos, siempre respondiendo a tres preguntas: ¿Qué hice? ¿Qué voy a hacer esta semana? ¿Qué problemas o inconvenientes tengo o tendré?, con una duración de no más de 10 minutos.

# **Introducción**

Sabemos lo difícil que puede ser la búsqueda de nuestro primer empleo en este mundo en constante cambio. Con el fin de facilitar este proceso creamos este proyecto que hace de puente entre el estudiante y la empresa.

## **Nuestros Propósitos**

* **Facilitar la búsqueda de empleo:** Nuestro sistema busca agilizar el proceso de búsqueda de empleo los usuarios
* **Conectar empleador con candidato:** Buscamos crear un espacio donde empleadores y candidatos puedan encontrarse de manera eficiente
* **Ofrecer oportunidades:** Nos enfocamos en ofrecer a los usuarios oportunidades laborales personalizadas, basadas en sus habilidades, experiencias y preferencias

## **Características Distintivas**

* **Perfil de Usuario Completo:** Los usuarios tienen la posibilidad de crear perfiles completos y detallados que destacan sus habilidades, experiencia laboral y educación, lo que facilita a los empleadores encontrar candidatos adecuados
* **Curriculum Vitae Automático:** Los usuarios al completar el formulario de perfil se le crea un currículum vitae que podrá imprimir y llevar personalmente a las empresas
* **Filtrado Avanzado:** El sistema ofrece opciones de filtrado avanzado que permite a los usuarios refinar las búsquedas según sus preferencias
* **Notificaciones Personalizadas:** Implementamos un sistema de notificaciones personalizadas que alerta a los usuarios sobre nuevas oportunidades de empleo que coinciden con su criterio de búsqueda.
* **Interfaz fácil de usar:** Diseñamos una interfaz de usuario amigable para que el usuario se sienta cómodo al navegar en la página.
* **Seguridad y Privacidad:** Nos comprometemos a proteger la privacidad y los datos de nuestros usuarios, usando medidas de seguridad para proteger la información sensible

# **Alcance del Proyecto**

El proyecto está pensado para desarrollarse en un periodo aproximadamente de 4 meses. Este alcance incluye:

* **Diseño y desarrollo de la plataforma:** Se dedicará tiempo a diseñar la arquitectura de la plataforma y desarrollar las características principales. Esto implica la creación de la interfaz de usuario, funciones de búsqueda avanzada y la integración de herramientas de perfil de usuario.
* **Funcionalidades esenciales:** se implementarán las funciones esenciales como la creación de perfil de usuario, la publicación de ofertas de empleo, la aplicación a estas ofertas y la gestión de candidatura.
* **Pruebas y Ajustes:** se realizarán todas las pruebas para garantizar que la plataforma sea funcional, fácil de usar y segura.
* **Lanzamiento y despliegue:** Una vez finalizadas todas las pruebas y los ajustes necesarios, se procederá al lanzamiento de la plataforma oficial. se desplegará en un entorno de producción y estará disponible para su uso público.

## **Algunas cuestiones fuera del alcance:**

Dada la limitación del tiempo, algunas características y aspectos adicionales quedan fuera del alcance del proyecto final, como:

* **Funcionalidades avanzadas:** características más avanzadas, como la integración de inteligencia artificial para el emparejamiento de candidatos y empleados, pueden requerir más tiempo de desarrollo y pruebas.
* **Optimización de rendimientos a gran escala:** aunque se llevaran a cabo pruebas de rendimientos, la optimización para manejar grandes volúmenes de datos y usuarios simultáneos puede requerir una fase de optimización para dedicar en un futuro.

# **Descripción de los usuarios**

Usuarios:

**Estudiante:** alumno de la Universidad Tecnológica Nacional (UTN) en busca de oportunidades para aplicar sus conocimientos y adquirir experiencia profesional a través de prácticas, pasantías y empleos.

**Empresa:** Entidad organizada que desean reclutar talento capacitado de la UTN para fortalecer sus equipos y fomentar la innovación en sus operaciones.

ADMIN

# **CURN – Bolsa de trabajo de la UTN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel** | **Estructura** | **Alcance** | **Caja** | **Instanciación** | **Interacción** |
| Resumen | Reestructurado | Sistema | Negra | Real | Semántica |

**Meta de caso de uso:** Encontrar un trabajo que se adecue a mis capacidades y preferencias

**Precondiciones:** <vacío> FALTA

**Disparador:** Estudiante desea conseguir trabajo

**Actores:**

**Primarios:** Estudiante (ES)

**Otros:** Empresa (EM), Administrador de la página (AP)

**Flujo de suceso**

**Camino básico:**

1. El ES ingresa al sitio web y se registra con sus datos, el AP le da los permisos.
2. Una vez dentro, el ES completa su perfil con sus datos y el sistema le crea un CV automático.
3. El ES busca y encuentra ofertas de trabajo que se adapten a sus capacidades y preferencias y la EM recibe las solicitudes a su oferta laboral y se contacta con el candidato.

**Camino alternativo:**

**1.a** <>

# **CURS – Buscar trabajo en la bolsa de trabajo de la UTN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel** | **Estructura** | **Alcance** | **Caja** | **Instanciación** | **Interacción** |
| Resumen | Reestructurado | Sistema | Negra | Real | Semántica |

**Meta de caso de uso:** contactar con una empresa para una posible entrevista de trabajo

**Precondiciones:** tener funcionando 100% el sitio web, se debe validar que el usuario sea realmente un estudiante de la UTN con su respectivo legajo como también debe haber ofertas cargadas en el sitio

**Disparador:** Estudiante desea conseguir trabajo

**Actores:**

**Primarios:** Usuario (US)

**Otros:** Empresa (EM), Admin UTN(UTN), Sistema (SI)

**Flujo de suceso**

**Camino básico:**

1. El US entra en la página WEB y selecciona registrarse como estudiante ingresando (datos). El sistema valida los datos y le permite el acceso a la pagina
2. El US carga sus datos para la creación automática del CV. El SI crea el CV.
3. El US realiza una búsqueda laboral en la página y realiza una postulación, el SI comunica a la EM que se realizó una postulación.

**Camino alternativo:**

1. <durante> El US se registra, pero no realiza ninguna acción y se va de la pagina

**1.a.1** se cierra la sesión y su usuario queda guardado en la base de datos para la próxima vez que entre a la pagina

1. <durante> El US ingresa mal los datos o no rellena un campo del registro

**1.b.1** El SI informa al usuario de la situación con un mensaje en la pantalla

1. <durante> El US intenta postularse, pero no tiene su perfil actualizado

**3.a.1** El SI informa al US con un mensaje en pantalla que debe completar su perfil para continuar. Vuelve al paso 2

**3.b** <posterior> La EM está interesada en el candidato

**3.b.1** La EM se comunica con el US mediante la página y coordinan un encuentro

**Postcondiciones:**

**Éxito:** El sistema registro y permitió cargar los datos y postular al usuario satisfactoriamente.

**Fracaso:** El usuario no encontró ninguna postulación que le interese o no se postula a ningún puesto.

**Éxito alternativo:** El sistema registro y permitió cargar los datos y postular al usuario satisfactoriamente y la empresa se interesó y se comunica con el

# **CURSR – Buscar trabajo en la bolsa de trabajo de la UTN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nivel** | **Estructura** | **Alcance** | **Caja** | **Instanciación** | **Interacción** |
| Resumen | Reestructurado | Sistema | Negra | Real | Semántica |

**Meta de caso de uso:** contactar con una empresa para una posible entrevista de trabajo

**Precondiciones:** tener funcionando 100% el sitio web, se debe validar que el usuario sea realmente un estudiante de la UTN con su respectivo legajo como también debe haber ofertas cargadas en el sitio

**Actores:**

**Primarios:** Usuario (US)

**Otros:** Empresa (EM), Admin UTN(UTN), Sistema (SI)

**Disparador:** Estudiante desea conseguir trabajo

**Flujo de suceso**

**Camino básico:**

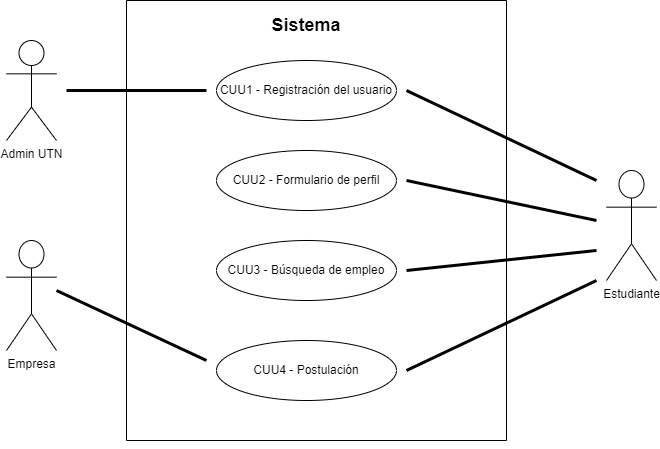
1. El US se registra en el sitio web, invoca al CUU1 – registración del usuario.
2. El US completa su perfil, invoca al CUU2 – formulario de perfil.
3. El US busca empleo, invoca al CUU3 – búsqueda de empleo.
4. El US se postula, invoca CUU4 – postulación.

**Postcondiciones:**

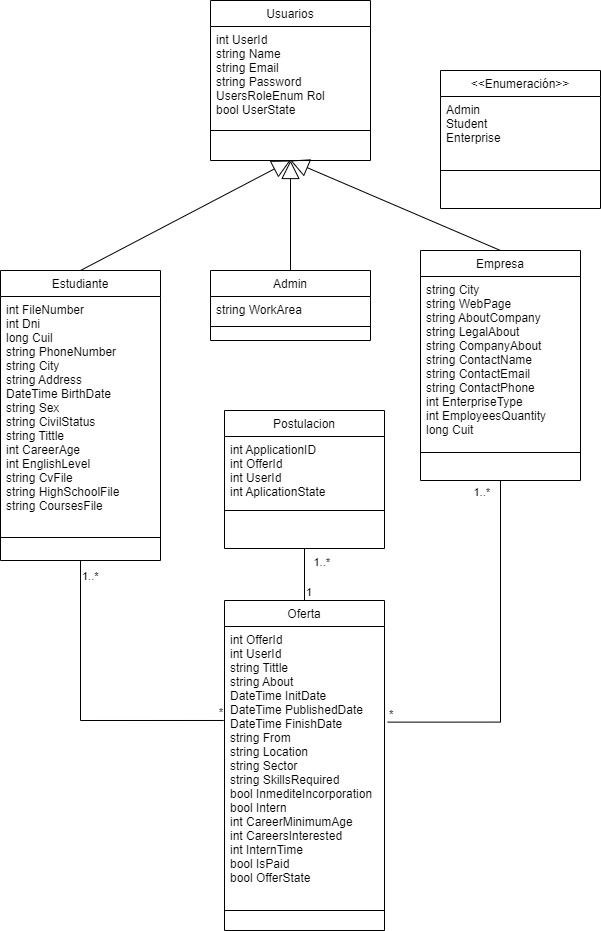
**Éxito:** El sistema registro y permitió cargar los datos y postular al usuario satisfactoriamente.

**Fracaso:** <vacío>

**Éxito alternativo:** <vacío>



# **Diagrama de Clase**



# **Módulos del Sistema**

* Módulo de Usuarios
* Módulo de Estudiantes
* Módulo de Empresas
* Módulo de Administradores
* Módulo de Ofertas
* Módulo de Postulación
* Módulo de Autenticación y Autorización
* Módulo de Notificaciones
* Módulo de Gestión de Archivos

**Descripción**

**Módulo de Usuarios**

Gestiona la información básica de los usuarios del sistema, incluyendo su registro, perfil, y roles.

**Módulo de Estudiantes**

Maneja las funcionalidades específicas para los estudiantes como gestionar su perfil, subida y gestión de archivos (CV, certificados, etc.) y la creación automática del CV.

**Módulo de Empresas**

Gestiona las funcionalidades específicas para las empresas como gestionar perfil de la empresa, publicación y gestión de ofertas de trabajo y visualización de postulaciones recibidas.

**Módulo de Administradores**

Proporciona herramientas para los administradores del sistema para gestionar y supervisar el funcionamiento del sistema. Ejemplo: la gestión de usuarios (creación, actualización, eliminación) supervisión de ofertas y aplicaciones, y gestión de contenido y configuración del sistema.

**Módulo de Ofertas**

Maneja la creación, publicación y gestión de ofertas de trabajo.

**Módulo de Postulación**

Gestiona las postulaciones de los estudiantes a las ofertas de trabajo como también el registro de postulación, actualización del estado de las postulaciones y su notificación a las empresas y estudiantes.

**Módulo de Autenticación y Autorización**

Gestiona la seguridad del sistema, incluyendo el inicio de sesión y control de acceso.

**Módulo de Notificaciones**

Maneja el envío de notificaciones a los usuarios del sistema, ya sea por correo electrónico o mensajes dentro de la plataforma como también la configuración de preferencias de notificación.

**Módulo de Gestión de Archivos**

Gestiona la subida, almacenamiento y acceso a archivos de los usuarios.

# **Compatibilidad y versiones de tecnologías utilizadas**

**Frontend:**

Librerías y Frameworks:

React y relacionados:

"react": "^18.2.0"

"react-dom": "^18.2.0"

"react-router-dom": "^6.22.3"

"@vitejs/plugin-react": "^4.2.1"

"@emotion/react": "^11.11.4"

"@emotion/styled": "^11.11.5"

"react-simple-chatbot": "^0.6.1"

"react-toastify": "^10.0.5"

"formik": "^2.4.5"

"yup": "^1.4.0"

Material-UI:

"@mui/base": "^5.0.0-beta.40"

"@mui/icons-material": "^5.15.18"

"@mui/material": "^5.15.18"

Otras Librerías de UI y estilos:

"tailwindcss": "^3.4.3"

Utilidades y Herramientas:

"axios": "^1.6.8"

"js-base64": "^3.7.7"

"jwt-decode": "^4.0.0"

"localforage": "^1.10.0"

"match-sorter": "^6.3.4"

"sort-by": "^1.2.0"

Herramientas de Desarrollo:

"vite": "^5.2.0"

"eslint": "^8.57.0"

"eslint-plugin-react": "^7.34.1"

"eslint-plugin-react-hooks": "^4.6.0"

"eslint-plugin-react-refresh": "^0.4.6"

"@types/react": "^18.2.66"

"@types/react-dom": "^18.2.22"

"autoprefixer": "^10.4.19"

"postcss": "^8.4.38"

Requisitos de Node.js:

Node.js v20.14.0

Npm 10.8.015

**Backend:**

[net6.0]:

ASP.NET Core y Entity Framework:

Microsoft.AspNetCore.Authentication.JwtBearer v6.0.0

Microsoft.EntityFrameworkCore v6.0.28

Microsoft.EntityFrameworkCore.Design v6.0.28

Microsoft.EntityFrameworkCore.Sqlite v6.0.28

Microsoft.EntityFrameworkCore.Tools v6.0.0

Microsoft.Extensions.Configuration.FileExtensions v6.0.0

Microsoft.Extensions.Configuration.Json v6.0.0

Otras Librerías y Utilidades:

MailKit v4.5.0

Newtonsoft.Json v6.0.8

Swashbuckle.AspNetCore v6.5.0

Requisitos de .NET SDK:

.NET 6.0 SDK

**Base de Datos:**

Motor de Base de Datos: SQLite

Librerías de Conexión: vite

**Herramientas de Desarrollo:**

IDE: Visual Studio 2022

Control de Versiones: Git 2.45.2

**Requisitos para Desplegar el Proyecto**

Primero necesitaremos descargar [visual code](https://code.visualstudio.com/download), [visual studio 2022](https://visualstudio.microsoft.com/es/vs/community/) y [SQLite](https://sqlite.org/download.html).

**Frontend**

Instalar Node.js y npm:

* Asegúrate de tener Node.js v20.14.0 y npm 10.8.0 instalados.
* Puedes descargarlos desde [Node.js](https://nodejs.org/en).

Instalar dependencias del proyecto:

* Navega al directorio del proyecto frontend.
* Ejecuta el siguiente comando para instalar las dependencias:

npm install



Construir el proyecto:

* Ejecuta el comando de construcción:

npm run dev



**Backend**

Instalar .NET SDK:

* Asegúrate de tener .NET 6.0 SDK instalado.
* Puedes descargarlo desde [dotnet.microsoft.com](https://dotnet.microsoft.com/es-es/download).

Restaurar las dependencias del proyecto:

* Navega al directorio del proyecto backend.
* Ejecuta el siguiente comando para restaurar las dependencias:

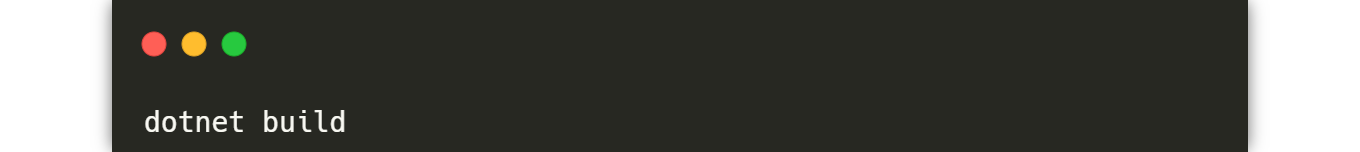
dotnet restore



Construir el proyecto:

* Ejecuta el comando de construcción:

dotnet build



Configurar la base de datos:

* Asegúrate de que SQLite esté correctamente configurado en tu aplicación.
* Ejecuta las migraciones para crear la base de datos:

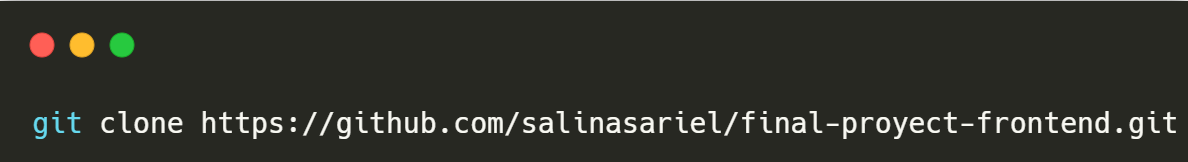
dotnet ef database update



**Despliegue Paso a Paso**

Clonar el repositorio de frontend:

git clone <https://github.com/salinasariel/final-proyect-frontend.git>



cd final-proyect-frontend



Navegar al directorio del frontend:

cd pps\_front



Instalar dependencias y construir el proyecto:

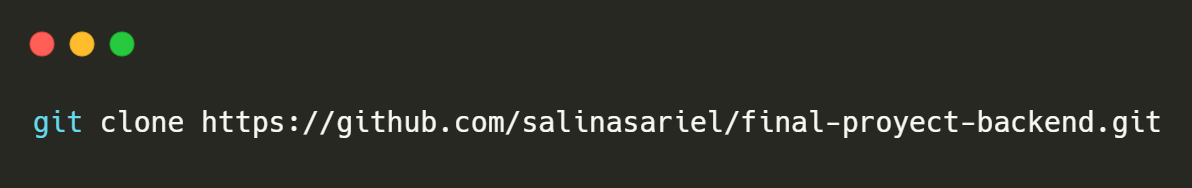
* npm install
* npm run dev



**Backend:**

Clonar el repositorio:

git clone <https://github.com/salinasariel/final-proyect-backend.git>



Navegar al directorio del backend:

cd final-proyect-backend



Restaurar dependencias y ejecutar migraciones:

* dotnet restore
* dotnet ef database update

