Herramientas y Tecnologías

GRCU Manager

Gestión de Requerimientos y Casos de Uso

4 BYTES

ALVAREZ, Abril

BUTTERFIELD, Nicolas

CARRANZA, Cristian

GAGNA, Martina

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.Un dibujo con letras

El contenido generado por IA puede ser incorrecto.

Contenido

[Introducción 4](#_Toc208906364)

[Lenguajes de Programación 4](#_Toc208906365)

[Frontend (Interfaz de usuario) 4](#_Toc208906366)

[HTML5 4](#_Toc208906367)

[CSS3 4](#_Toc208906368)

[JavaScript (ES6+) 4](#_Toc208906369)

[Backend (Lógica del servidor) 5](#_Toc208906370)

[Python 3.13 5](#_Toc208906371)

[Django Framework 5](#_Toc208906372)

[Base de Datos 5](#_Toc208906373)

[PostgreSQL 5](#_Toc208906374)

[Frameworks y Librerías 6](#_Toc208906375)

[Herramientas de Software 6](#_Toc208906376)

[Control de Versiones 7](#_Toc208906377)

[Herramientas de Comunicación 7](#_Toc208906378)

[Comentarios Finales 7](#_Toc208906379)

Herramientas y Tecnologías

# Introducción

El presente documento detalla las herramientas y tecnologías seleccionadas para el desarrollo del sistema GRCU Manager.

El objetivo del proyecto es gestionar requerimientos y casos de uso, mantener un historial de cambios y generar matrices de trazabilidad, lo cual demanda un conjunto de tecnologías que brinden seguridad, escalabilidad, mantenibilidad y facilidad de uso.

La materia propone como base el framework UARGFlow (PHP + MySQL), pero el proyecto innova al migrar hacia un stack más moderno:

* Django (Python) como framework backend.
* PostgreSQL como gestor de base de datos relacional.
* Tecnologías frontend modernas como HTML5, CSS3, JavaScript y Bootstrap.

Esto permite alinearse con estándares industriales actuales, haciendo el sistema más adaptable a escenarios reales.

# Lenguajes de Programación

# Frontend (Interfaz de usuario)

## HTML5

Es el lenguaje de marcado estándar de la web. Permite estructurar la información en páginas, formularios y dashboards. En GRCU Manager se utilizará para definir plantillas que representen casos de uso, requerimientos y reportes de trazabilidad.

Ventaja: soporte universal y semántica clara que facilita accesibilidad.

## CSS3

Se usa para dar estilo y diseño responsivo a la aplicación. Aporta una experiencia de usuario coherente y atractiva, con componentes como formularios, tablas de usuarios y dashboards.

Ventaja: desacopla diseño de la lógica, permitiendo iterar en la interfaz sin tocar la funcionalidad.

## JavaScript (ES6+)

Permite interactividad en tiempo real: validación de formularios sin recargar, filtros dinámicos en listas de usuarios o proyectos, y dashboards con estadísticas gráficas.

Ventaja: otorga fluidez y mejora la experiencia de uso en comparación a páginas estáticas.

# Backend (Lógica del servidor)

## Python 3.13

Lenguaje multipropósito, claro y con amplia comunidad. Provee librerías para análisis, seguridad y conexión con múltiples servicios.

En GRCU Manager se utiliza para implementar reglas de negocio: creación de proyectos, asignación de usuarios y trazabilidad de requerimientos.

Ventaja: código más legible y menor curva de aprendizaje frente a lenguajes como Java o PHP.

## Django Framework

Framework de alto nivel que organiza la aplicación en Modelos, Vistas y Templates (MVT).

Incluye un ORM que simplifica la comunicación con la base de datos, sistema de autenticación integrado y panel administrativo.

En GRCU Manager:

* Administra usuarios y roles.
* Garantiza seguridad en sesiones (prevención de CSRF, XSS, SQL Injection).
* Permite extender modelos para almacenar datos de alumnos (legajo, carrera, fecha de ingreso).
* Ventaja: rapidez de desarrollo + seguridad lista para usar.

# Base de Datos

### PostgreSQL

Base de datos relacional de código abierto. Destaca por su soporte de transacciones ACID, consultas complejas y JSONB para datos semiestructurados.

En GRCU Manager:

* Se almacenan requerimientos, casos de uso y su historial de cambios.
* Se gestionan relaciones entre proyectos, alumnos y roles.
* Se soporta la generación de reportes para matrices de trazabilidad.

Ventaja: más robusto y escalable que MySQL, con mejor soporte para integridad referencial y auditoría.

# Frameworks y Librerías

* Django (Python): núcleo del backend, encargado de manejar autenticación, sesiones, ORM y APIs REST.
* Bootstrap 5: framework CSS que permite diseñar interfaces responsivas rápidamente.
* Uso en GRCU Manager: botones, formularios de creación/edición de usuarios y dashboards con tarjetas.
* jQuery / Vanilla JS: útiles para manipular elementos DOM y añadir pequeñas interacciones, como agregar alumnos a un proyecto de forma dinámica.
* Chart.js o Recharts (opcional): para mostrar KPIs de proyectos y roles asignados en gráficos interactivos.

# Herramientas de Software

* **Visual Studio Code (VSCode):**

Editor con soporte para extensiones de Django, Python y PostgreSQL. Facilita debugging y control de dependencias.

* **PgAdmin 4:**

Herramienta oficial de PostgreSQL, usada para la administración directa de la base de datos. Permite crear tablas, ejecutar queries y verificar integridad.

* **DBeaver:**

Cliente universal que ayuda a visualizar esquemas complejos y realizar ingeniería inversa de la base de datos para generar diagramas ER.

* **Draw.io:**

Usados para diagramas UML y modelos conceptuales. En este proyecto se aplican para casos de uso, clases y matrices de trazabilidad.

* **Docker (opcional en despliegue):**

Permite empaquetar el sistema completo (Django + PostgreSQL) en contenedores. Garantiza que la aplicación corra igual en desarrollo y producción.

# Control de Versiones

* **Git:** sistema distribuido de control de versiones. Permite trabajo en paralelo, revertir errores y mantener un historial detallado.
* **GitHub:** plataforma colaborativa. Se usará para ramas por feature, gestión de issues y documentación en Wiki.

# Herramientas de Comunicación

* **Discord:** para coordinación diaria y compartir avances.
* **Google Meet:** reuniones periódicas de seguimiento con el equipo y docentes.
* **WhatsApp:** comunicación rápida y resolución de dudas.
* **Google Drive**: para almacenar entregables, actas y documentación del sistema.

# Comentarios Finales

El stack Django + PostgreSQL + Bootstrap convierte a GRCU Manager en un sistema:

* **Escalable:** preparado para crecer en cantidad de usuarios y requerimientos.
* **Seguro:** gracias a la gestión de sesiones, permisos y roles integrados en Django.
* **Colaborativo**: aprovechando GitHub y prácticas ágiles para organizar iteraciones.
* **Estandarizado:** utilizando lenguajes y tecnologías reconocidos en la industria.

El proyecto no solo cumple con los objetivos, sino que también se alinea con buenas prácticas profesionales que permitirían su evolución futura como producto real.