**Documento de requerimientos de software**

***Proyecto Mi Visión Académica***

***15/09/2025***

**Tabla de contenido**

Historial de Versiones 3

Información del Proyecto 3

Aprobaciones 3

1. Propósito 4

2. Alcance del producto / Software 4

3. Referencias 4

4. Funcionalidades del producto 5

5. Clases y características de usuarios 5

6. Entorno operativo 5

7. Requerimientos funcionales 6

9.1. (Nombre de la funcionalidad 1) 6

9.2. (Nombre de la funcionalidad 2) 7

9.3. (Nombre de la funcionalidad N) 7

8. Reglas de negocio 8

9. Requerimientos de interfaces externas 9

9.1. Interfaces de usuario 9

9.2. Interfaces de hardware 9

9.3. Interfaces de software 9

9.4. Interfaces de comunicación 9

10. Requerimientos no funcionales 10

11. Otros requerimientos 11

12. Glosario 12

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 09/09/2025 | 1.0 | Diego Ríos |  | Versión inicial del documento de requerimientos de software para el proyecto *Mi Visión Académica*. |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

| Empresa / Organización |  |
| --- | --- |
| Proyecto | Mi Visión Académica – Plataforma de Análisis y Orientación Académica Personalizada |
| Fecha de preparación | Septiembre 2025 |
| Cliente |  |
| Patrocinador principal |  |
| Gerente / Líder de Proyecto | Diego Ríos - Product Owner |
| Gerente / Líder de Análisis de negocio y requerimientos |  |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| Alexis Curiche | Director de Carrera | Escuela de Informática (Duoc UC) |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. **Propósito**

El software se denomina **“Mi Visión Académica – Release 1.0 (MVP)”**.

Este documento cubre la **totalidad del sistema en su versión inicial**, incluyendo:

* Registro e ingreso de notas académicas de los estudiantes.
* Análisis automático de rendimiento y generación de visualizaciones (radar y evolución).
* Identificación de fortalezas y debilidades como puntos de mejoras.
* Recomendación de especialización y orientación académica y laboral.
* Reportes individuales para estudiantes y docentes.

No incluye, en esta versión inicial, integración automática con sistemas institucionales, pero se contempla para releases futuros.

1. **Alcance del producto / Software**

* **Propósito u objetivo general:** Facilitar la autogestión académica y orientación de estudiantes mediante un sistema digital que analice su desempeño académico y entregue reportes visuales y recomendaciones personalizadas.
* **Beneficios para el Área de Negocio y Organización:**
  + Mejora la toma de decisiones de estudiantes en relación con su especialización y trayectoria profesional.
  + Fortalece el acompañamiento docente con herramientas de análisis.
  + Aporta a la retención de estudiantes al detectar riesgos académicos tempranos.
  + Escalable a diferentes niveles educativos (básica, media, universitaria).
* **Objetivos y Metas:**

Desarrollar un MVP en **18 semanas** enfocado en estudiantes de Ingeniería en Informática (Duoc UC).

Generar reportes académicos en un **100% de los usuarios piloto**.

Implementar al menos **2 tipos de gráficos visuales (radar y evolución)**.

Recoger retroalimentación de al menos **50 estudiantes y 5 docentes** en la etapa piloto.

1. **Referencias**

* Guías de gestión de proyectos APT – Duoc UC.
* Estándares IEEE para documentación de software (IEEE 830).
* Documentación institucional de Duoc UC sobre perfiles de egreso.
* Herramientas de análisis y visualización: Python (Matplotlib/Plotly), frameworks web.

1. **Funcionalidades del producto**

* Registro y almacenamiento de notas por estudiante.
* Categorización de asignaturas por Certificados oficiales.
* Cálculo de promedios por área y semestre.
* Generación de gráficos radar y de evolución con escalado de 0,25.
* Reportes personalizados con fortalezas, debilidades y recomendaciones.
* Panel de usuario diferenciado (estudiante, docente, director).
* Cuestionario de intereses para complementar orientación académica.

1. **Clases y características de usuarios**

* **Tipos de usuarios:**
  + **Estudiantes:** Nivel de experiencia tecnológica variado, desde básico a avanzado.
    - Usan la plataforma para ver su rendimiento y recibir recomendaciones.
  + **Docentes:** Usuarios con experiencia intermedia en herramientas académicas.
    - Usan la plataforma para apoyar tutorías y orientar a estudiantes.
  + **Directores de carrera:** Usuarios con perfil de gestión.
    - interesados en reportes globales y tendencias académicas.
* **Funcionalidades relevantes:**
  + Estudiantes → ingreso de notas, visualización de reportes, cuestionarios de intereses.
  + Docentes → análisis de estudiantes bajo su tutoría, identificación de riesgos.
  + Directores → visión global de cohortes y áreas con más falencias.
* **Frecuencia de uso:**
  + Estudiantes: Semanal / Mensual.
  + Docentes: Mensual / Según calendario académico.
  + Directores: Bimensual / Semestral.

1. **Entorno operativo**

* **Hardware:** PC, notebook o dispositivo móvil con conexión a internet.
* **Software:** Navegador web actualizado (Chrome, Edge, Firefox).
  + Lenguajes de programación: Python para desarrollo de la IA, HTML/CSS para la interfaz web.
  + Bases de datos: MySQL o PostgreSQL para almacenar datos de usuarios y consultas.
* **Sistema Operativo:** Windows (Computadores), Android (Celulares).
* **Otros componentes:** Servidor en la nube para almacenamiento y análisis de datos (ej: AWS, Azure, GCP).

1. **Requerimientos funcionales**

| **ID** | **Descripción** | **Prioridad** | **Acción iniciadora** | **Comportamiento esperado** |
| --- | --- | --- | --- | --- |
| RF-01 | Registrar notas académicas manualmente | Alta | Usuario estudiante ingresa sus notas con PDF de certificados | El sistema almacena las notas en la base de datos |
| RF-02 | Generar gráficos radar con escalado de 0,25 | Alta | Usuario solicita visualización | El sistema genera gráfico dinámico |
| RF-03 | Generar gráficos de evolución de notas | Alta | Usuario solicita visualización | El sistema despliega evolución por semestre |
| RF-04 | Clasificar ramos en áreas académicas | Media | Usuario ingresa asignatura | El sistema etiqueta automáticamente el área |
| RF-05 | Identificar fortalezas y debilidades | Alta | Cálculo automático | El sistema entrega reporte en PDF/visual |
| RF-06 | Recomendación de especialización | Media | Procesamiento IA/algoritmos | El sistema sugiere área de especialización |
| RF-07 | Panel de docente | Media | Login docente | El sistema despliega listado de estudiantes asociados |

1. **Reglas de negocio**

* El sistema debe manejar notas en una escala de 1,0 a 7,0 (con precisión de 0,1).
* Los reportes deben actualizarse cada vez que el usuario ingrese nuevas notas.
* Los datos deben almacenarse bajo normativas de seguridad y privacidad académica.

1. **Requerimientos de interfaces externas**
2. **Interfaces de usuario**

En esta sección se describen las características de las interfaces con las que los usuarios interactúan directamente.

* Interfaz de Estudiante: Esta sería la principal interfaz, accesible para los usuarios que interactúan con la página web para recibir soporte 24/7. Debería estar disponible en aplicaciones móviles y una versión web.
* Interfaz de Docente: Esta sería la interfaz que utilizan los Docentes o Directores de carrera para ver el historial de sus estudiantes, realizar comentarios sobre estos y actualizar la base de conocimientos.

**Características principales de la interfaz de usuario**

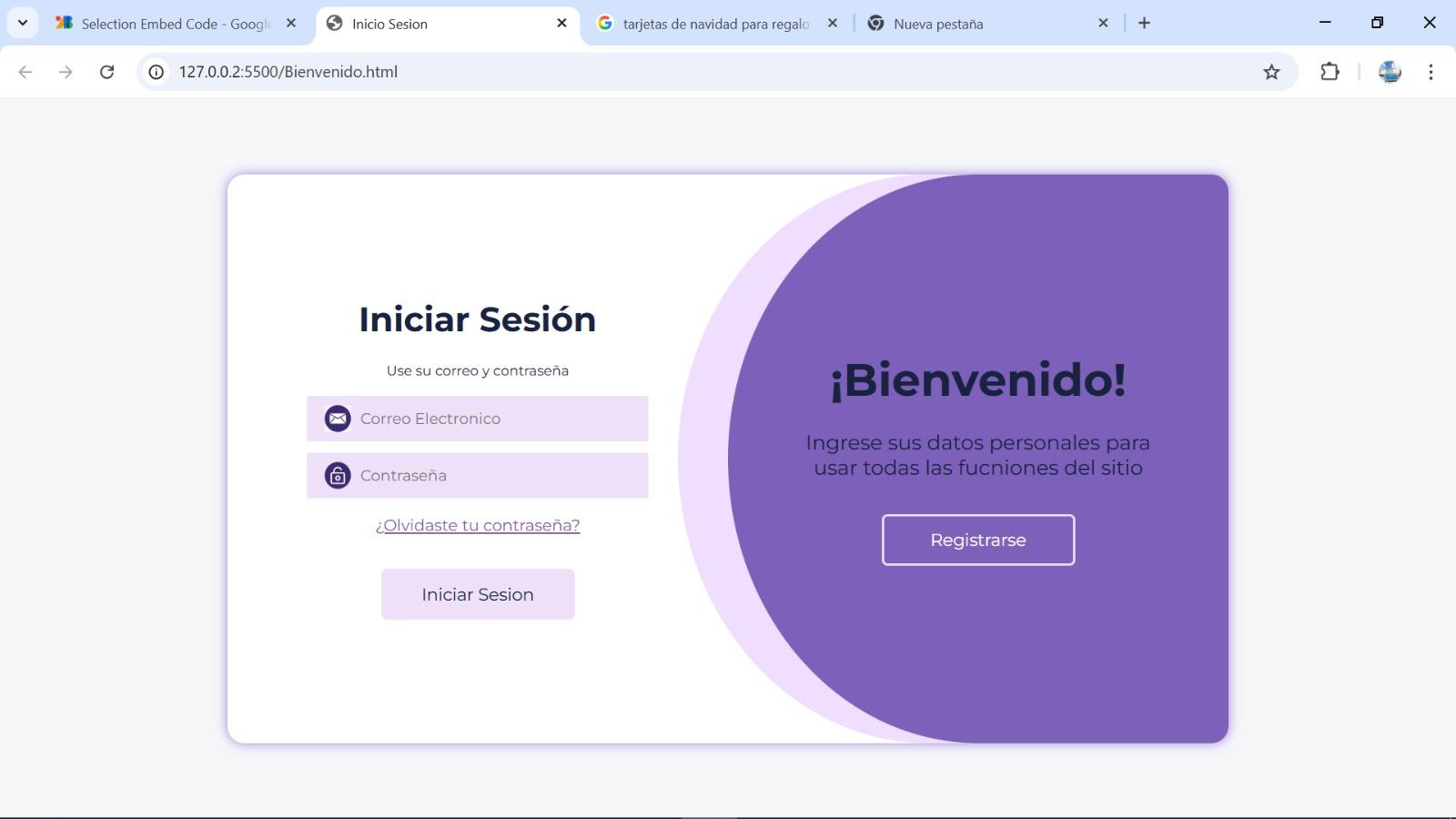
Pantallas principales

* Informes detallados del estudiante
* Menú principal

Estándares a considerar

1. Diseño de interfaz
2. Paleta de colores a utilizar
3. Responsivo
4. Botones de navegación (navegación)
5. Consistencia
6. Responsividad y adaptabilidad
7. Tipografía (legibilidad)
8. Rendimiento
9. Accesibilidad

Mockup de prueba

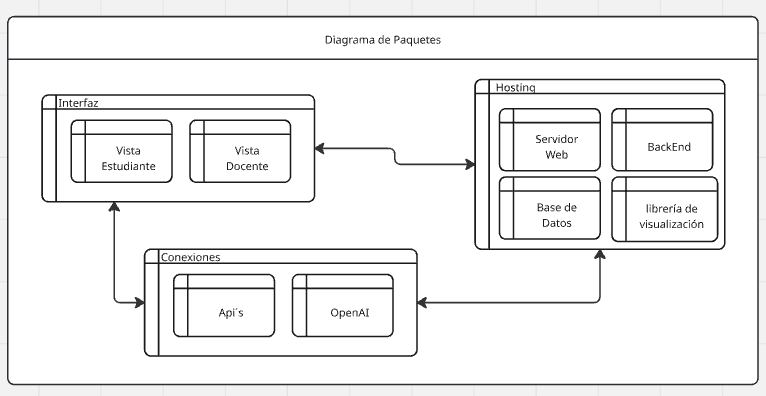


1. **Interfaces de hardware**

El sistema está concebido para ser accesible desde una variedad de dispositivos, facilitando así que los usuarios usen la página web de forma fluida y sin limitaciones. Las categorías de hardware que se admiten son:

* 1. **Tipos de dispositivos soportados**: PCs, tablets, Celulares.
  2. **Protocolos de comunicación**: HTTP/HTTPS para acceso web.
  3. **Interacciones entre hardware y software**: Entrada y salida de datos, interacción con API REST, actualizaciones automáticas.

Diagrama de paquetes

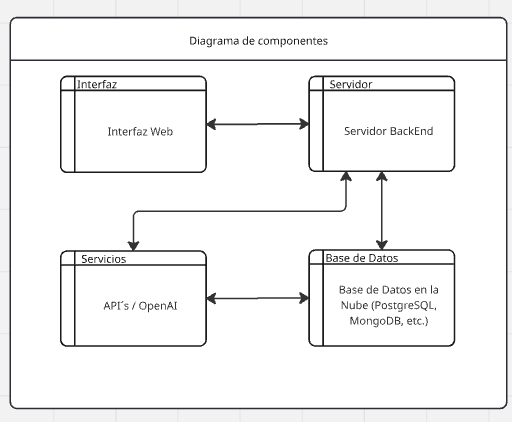


1. **Interfaces de software**

El sistema interactúa con varios componentes de software. Se detallan a continuación las interacciones clave:

* **Base de Datos:** Guarda todas las notas de los estudiantes y su historial académico. Está alojada en la nube para que siempre sea accesible desde internet.
* **Uso de librerías de visualización**: Plotly, D3.js, Chart.js.
* **Plataformas de terceros:** Puede usar librerías de Machine Learning o conectarse con APIs como OpenAI.
* **Hosting (Vercel / Heroku / Supabase):** Es el servidor en la nube donde se aloja la página web. Permite que cualquier usuario con internet pueda acceder a la plataforma. Mantiene funcionando tanto el frontend como el backend.

Diagrama de componentes



1. **Interfaces de comunicación**

El sistema requiere de varias interfaces de comunicación que se detallan a continuación:

* **Protocolos:** El sistema utilizará protocolos HTTP/HTTPS para la comunicación web y móvil.
* **Formatos de Mensajería:** Se emplearán mensajes en formato JSON para la interacción entre el frontend y el backend del sistema.
* **Requerimientos de Encriptación y Seguridad:** Todas las comunicaciones estarán encriptadas utilizando HTTPS y certificados SSL para proteger la privacidad de los usuarios. Se implementará autenticación basada en tokens (JWT) para asegurar el acceso seguro a la plataforma.
* **Base de datos SQL/NoSQL para persistencia.**

1. **Requerimientos no funcionales**

* **Seguridad:** Cifrado de datos en tránsito y en reposo.
* **Usabilidad:** Interfaz intuitiva con navegación simple.
* **Escalabilidad:** Capacidad de soportar más usuarios e instituciones en el futuro.
* **Disponibilidad:** Uptime del 95% en versión MVP.
* **Portabilidad:** Compatible con navegadores modernos y dispositivos móviles.

1. **Otros requerimientos**

* Incluir módulo de retroalimentación de estudiantes y docentes.
* Generar reportes exportables en PDF.
* Mantener logs de actividad para auditoría básica.

1. **Glosario**

* **MVP (Minimum Viable Product):** Versión inicial del software con funcionalidades básicas.
* **Radar chart:** Gráfico que muestra múltiples variables en un plano radial.
* **Evolución académica:** Seguimiento del rendimiento por semestre.
* **Stakeholders:** Personas o grupos interesados en el proyecto (estudiantes, docentes, directores).
* **Retroalimentación:** Comentarios de los usuarios sobre el funcionamiento y utilidad de la herramienta.