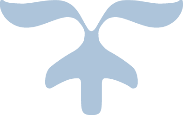


DEFINICIÓN PROYECTO APT

GRADE



**FECHA:** SEPTIEMBRE 2025

**PROFESOR:** RODOLFO SEPULVEDA

**AUTOR:** MARÍA FERNANDA GARAY NÚÑEZ

## Abstract

Español

En esta primera fase del Proyecto APT, el trabajo se centró en la definición del proyecto, evaluando la coherencia entre la problemática identificada y su relevancia, así como la formulación del objetivo general y los objetivos específicos. También se definió el alcance del MVP, la metodología en cascada a utilizar, la planificación por fases (semanas 1–18), la distribución de roles y la factibilidad considerando tiempo, recursos y riesgos.

Durante esta etapa se elaboraron las evidencias comprometidas: la guía de definición, los requerimientos y casos de uso, la arquitectura y el modelo de datos, el plan de trabajo y una carta Gantt preliminar. Para cada una de ellas se consideraron criterios de calidad como la claridad, la trazabilidad, la alineación con el perfil de egreso y la viabilidad técnica.

En la autoevaluación de esta fase se identificaron algunas brechas puntuales, especialmente en el afinamiento de los requerimientos no funcionales, los criterios de aceptación y el plan de pruebas, aspectos que requieren ajustes al inicio de la Fase 2. En términos generales, se concluye que el proyecto es viable y correctamente definido, con compromisos claros para asegurar la continuidad hacia el desarrollo del MVP y la preparación de las evidencias de avance.

Inglés

In this first phase of the APT Project, the work focused on the project definition, evaluating the coherence between the identified problem and its relevance, as well as the formulation of the general objective and the specific objectives. The scope of the MVP was also defined, along with the waterfall methodology to be applied, the phase-based planning (weeks 1–18), the distribution of roles, and the feasibility considering time, resources, and risks.

During this stage, the committed deliverables were developed: the definition guide, the requirements and use cases, the architecture and data model, the work plan, and a preliminary Gantt chart. For each of these, quality criteria such as clarity, traceability, alignment with the graduate profile, and technical feasibility were taken into account.

In the self-assessment of this phase, some specific gaps were identified, particularly in the refinement of non-functional requirements, acceptance criteria, and the test plan—areas that require adjustments at the beginning of Phase 2. In general terms, it is concluded that the project is viable and properly defined, with clear commitments to ensure continuity toward the development of the MVP and the preparation of the progress deliverables.

## Contenido

Contenido

[Abstract 2](#_Toc208871802)

[Contenido 4](#_Toc208871803)

[Descripción de proyecto APT 5](#_Toc208871804)

[Relación del proyecto APT con las competencias del perfil de egreso 6](#_Toc208871805)

[Relación del proyecto con tus intereses profesionales 7](#_Toc208871806)

[Argumento del porqué el proyecto es factible a realizarse dentro de la asignatura 8](#_Toc208871807)

[Propuesta metodológica de trabajo que permita alcanzar los objetivos 10](#_Toc208871808)

[Plan de trabajo para el proyecto APT 13](#_Toc208871809)

[Propuesta de evidencias que darán cuenta del logro de las actividades 15](#_Toc208871810)

## Descripción de proyecto APT

El proyecto en el que estoy trabajando consiste en el desarrollo de GRADE, una plataforma integral de gestión de evaluaciones académicas que centraliza la creación, aplicación y calificación de pruebas. Con esta iniciativa busco dar respuesta a la problemática de procesos fragmentados y poco eficientes en el ámbito educativo, ofreciendo un flujo unificado que permita mejorar la trazabilidad, reducir errores y acelerar la entrega de resultados. El contexto de uso se sitúa en instituciones educativas, teniendo como foco principal a docentes y coordinadores académicos, y generando un impacto indirecto en los estudiantes a través de evaluaciones más consistentes y oportunas.

1. El alcance del MVP que estoy desarrollando considera cuatro capacidades principales:
2. Banco de preguntas reutilizable con metadatos.
3. Generación de evaluaciones y documentos con identificadores únicos/QR.
4. Ingesta de respuestas mediante aplicación móvil (captura fotográfica) y canales alternativos.
5. Calificación automática con registro de resultados y reportes básicos de desempeño.

Dentro de este alcance también se contemplan aspectos de seguridad y trazabilidad (roles, auditoría de acciones) y un modelo de datos relacional. Quedan fuera del MVP funcionalidades avanzadas, como portales para estudiantes, analítica avanzada o integraciones externas completas, las cuales se proyectan para etapas posteriores.

La metodología seleccionada para el desarrollo es clásica en cascada, alineada con las fases del APT: Fase 1 (definición, requerimientos, arquitectura y planificación), Fase 2 (diseño detallado, construcción, integración y pruebas del MVP) y Fase 3 (validación final, documentación y presentación). La distribución del trabajo que llevo adelante contempla frentes de desarrollo web/backend y aplicación móvil/OMR, bajo una coordinación responsable de la integración técnica, el control de avance y el aseguramiento de la calidad.

Como resultado esperado, este proyecto entregará un MVP funcional acompañado de la documentación respectiva (informe, manual de usuario y guía técnica), evidenciando así la pertinencia con mi perfil de egreso: análisis y levantamiento de requerimientos, diseño y construcción de software, gestión de proyectos, aseguramiento de calidad y uso de datos para apoyar la toma de decisiones en el ámbito educativo.

## Relación del proyecto APT con las competencias del perfil de egreso

El proyecto que estoy desarrollando se alinea directamente con las competencias del perfil de egreso, ya que integra el análisis, diseño, construcción y validación de una solución tecnológica con impacto en el ámbito educativo. En particular:

* Análisis y levantamiento de requerimientos. Me ha permitido aplicar técnicas de levantamiento, priorización y trazabilidad para definir el alcance del MVP, los casos de uso y los requisitos no funcionales, asegurando coherencia entre la problemática y la solución propuesta.
* Diseño y arquitectura de software. He debido modelar una arquitectura modular (web, backend, móvil/OMR) y un modelo de datos relacional, documentando decisiones técnicas y criterios de calidad que habilitan mantenibilidad y escalabilidad.
* Desarrollo de software (web, backend y móvil). El proyecto demanda implementar funcionalidades núcleo (banco de preguntas, generación de evaluaciones, ingesta y calificación), integrando interfaces, servicios y persistencia con buenas prácticas de programación.
* Gestión de proyectos y trabajo en equipo. Me ha exigido planificar por fases, asignar roles, gestionar riesgos y controlar avances, en alineación con la metodología seleccionada (cascada) y los hitos formales del APT.
* Aseguramiento de calidad y pruebas. He tenido que definir criterios de aceptación y ejecutar pruebas unitarias, de integración y de validación funcional del MVP, resguardando la confiabilidad y la corrección de los resultados.
* Datos, reportabilidad y toma de decisiones. El proyecto me ha llevado a diseñar consultas y reportes de desempeño, transformando datos de evaluación en información útil para la gestión académica.
* Seguridad, ética y cumplimiento. He debido incorporar control de acceso, trazabilidad y resguardo de información sensible, promoviendo prácticas éticas en el manejo de datos educativos.

En conjunto, este proyecto evidencia mi capacidad de resolver un problema real mediante ingeniería de software, articulando las competencias técnicas y de gestión que se esperan en el perfil de egreso.

## Relación del proyecto con tus intereses profesionales

Desde mis intereses profesionales, este proyecto me ofrece un espacio concreto para construir software integral y escalable. Al planificar y ejecutar un MVP bajo una metodología clásica en cascada, puedo profundizar en ingeniería de requisitos y diseño de sistemas, definir la arquitectura y materializarla en desarrollo full-stack (web/API y móvil/OMR). Esto me permite aplicar buenas prácticas de ingeniería y mantener trazabilidad entre requisitos, casos de uso y pruebas, asegurando coherencia técnica desde el diseño hasta la validación.

Además, aspiro a fortalecer mi rol en la gestión del ciclo de vida del software (SDLC): asumir planificación, coordinación de equipos e integración de componentes; elevar la calidad mediante pruebas unitarias, de integración y UAT; y resguardar seguridad y auditoría con controles de acceso y registro de acciones. La gestión de datos y reportes que apoyan decisiones pedagógicas conecta mi motivación técnica con un impacto social tangible en educación, consolidando un perfil capaz de traducir necesidades reales en soluciones robustas, mantenibles y verificables.

## Argumento del porqué el proyecto es factible a realizarse dentro de la asignatura

Se considera factible desarrollar el proyecto GRADE dentro de la asignatura, ya que se ajusta explícitamente a la estructura del APT. En la Fase 1 (semanas 1–4) se definió el proyecto (problema, alcance, objetivos y metodología) y se avanzó en discovery, requerimientos, casos de uso y arquitectura preliminar. La Fase 2 (semanas 5–15) concentra un único ciclo de desarrollo en cascada (diseño detallado → construcción → integración → pruebas) hasta la entrega del MVP, mientras que la Fase 3 (semanas 16–18) contempla la validación final, la documentación y la presentación.

La viabilidad operativa se sustenta en un alcance acotado a un MVP con cuatro capacidades esenciales (banco de preguntas, generación de evaluaciones con identificador, ingesta de respuestas y calificación automática), con exclusiones explícitas de funcionalidades avanzadas. Esto permite reducir la complejidad y ajustar los esfuerzos al tiempo disponible de la asignatura, combinando horas de cátedra y trabajo autónomo.

En cuanto a recursos, solo se requiere equipamiento estándar (computadores, IDE, repositorio y base de datos) y un dispositivo móvil para pruebas de captura. Todas las herramientas son de bajo costo o ya disponibles, y se cuenta con plantillas institucionales para los entregables.

La organización del equipo asegura una ejecución controlada: un frente dedicado a desarrollo web/backend, otro a móvil/OMR, y una coordinación responsable de la integración y control del avance. Aunque el enfoque es en cascada, el diseño común permite trabajo en paralelo durante la construcción, con hitos y revisiones en momentos clave.

Se han identificado riesgos y mitigaciones, incluyendo:

Precisión del OMR → pruebas tempranas con set controlado y plan de respaldo de ingreso manual en casos excepcionales.

Tiempos de integración → buffer de consolidación previo al cierre.

Requerimientos no funcionales de seguridad y calidad → checklists, criterios de aceptación y plan de pruebas (unitarias, de integración y UAT).

Finalmente, la trazabilidad y el control están garantizados por las evidencias por fase (Guía Fase 1, Informe de Avance, Informe Final, manuales y presentación), que permiten recibir retroalimentación docente oportuna y ajustar desvíos. En conjunto, el encuadre temporal, el alcance realista, los recursos disponibles, la organización del trabajo y las mitigaciones previstas hacen viable la realización del proyecto dentro del semestre y conforme a los lineamientos de la asignatura.

Objetivos claros y coherentes

Los objetivos del proyecto se presentan con claridad, coherencia interna y trazabilidad. El objetivo general define el resultado esperado: una plataforma que unifique la creación, aplicación y calificación de evaluaciones. Los objetivos específicos desagregan este resultado en acciones concretas y secuenciales (requisitos/diseño → construcción web/backend y móvil/OMR → calificación automática → seguridad/calidad → documentación y presentación), alineadas a las fases del APT y al alcance del MVP. Se consideran pertinentes al perfil de egreso (levantamiento de requisitos, arquitectura, desarrollo, aseguramiento de calidad y gestión de proyectos), viables en el tiempo definido y medibles mediante indicadores y evidencias.

Objetivo Específicos

1. Definir los requisitos funcionales y no funcionales, acotando el alcance del MVP.
2. Diseñar la arquitectura del sistema y el modelo de datos que soporten el MVP.
3. Implementar el banco de preguntas y el módulo web/API para crear y administrar evaluaciones, incluyendo la generación de PDF con identificador/QR.
4. Construir la aplicación móvil de captura y el servicio OMR para la ingesta de respuestas.
5. Desarrollar el motor de calificación automática y los reportes básicos de resultados.
6. Asegurar seguridad y trazabilidad (autenticación, roles, auditoría) y calidad mediante pruebas unitarias, de integración y validación funcional.
7. Integrar los componentes (web, backend, móvil/OMR) y consolidar el MVP listo para demostración.
8. Elaborar la documentación (informe de avance, informe final, manuales) y preparar la presentación ante la comisión.

## Propuesta metodológica de trabajo que permita alcanzar los objetivos

**Enfoque general y plan de trabajo del proyecto GRADE**

El proyecto adoptará una metodología clásica en cascada, alineada a las fases del APT y a un único ciclo de construcción del MVP. Cada etapa concluye con entregables definidos y revisión formal, garantizando trazabilidad desde los requisitos hasta la presentación final.

**Fase 1 – Definición (Semanas 1–4)**

* Actividades: levantamiento y priorización de requisitos funcionales y no funcionales; elaboración de casos de uso; definición de arquitectura de alto nivel y modelo entidad-relación; creación de prototipos de baja fidelidad; planificación del trabajo y carta Gantt.
* Controles de calidad: revisión de requisitos (completitud, claridad y no ambigüedad), validación del alcance del MVP, checklist de arquitectura.
* Entregables: Guía Fase 1; Documento de Requerimientos y Casos de Uso; Arquitectura/MER; Plan de trabajo y Gantt; presentación F1; evidencias individuales (autoevaluaciones y diario de trabajo).

**Fase 2 – Desarrollo (Semanas 5–15)**

* Diseño detallado: contratos de API, modelos de datos finales, diagramas de componentes.
* Construcción:
  + Web/Backend: banco de preguntas, creación de evaluaciones, generación de PDF/QR y servicios de calificación.
  + Móvil/OMR: captura de respuestas y servicio de reconocimiento.
* Integración: pruebas de integración entre cliente, API y OMR; estabilización del entorno.
* Pruebas: unitarias (módulos críticos), de integración (flujos end-to-end) y UAT interna del MVP.
* Documentación: borradores de Informe de Avance, Manual de Usuario y Guía Técnica.
* Controles de calidad: code review por pares; checklist de seguridad (autenticación, roles, trazabilidad); matriz de trazabilidad RF→CU→Pruebas.
* Entregables: Informe de Avance; MVP funcional; resultados de pruebas; borradores de manuales.

**Fase 3 – Presentación (Semanas 16–18)**

* Validación final: hardening del MVP, pruebas de regresión y performance básica.
* Cierre documental: Informe Final, Manual de Usuario y Guía Técnica definitivos.
* Comunicación: preparación de demo y defensa; presentación ante comisión.
* Entregables: Informe Final; manuales; presentación y demo.

**Roles y responsabilidades**

* Coordinación/Integración: planificación, control de avance, gestión de riesgos e integración técnica.
* Desarrollo Web/Backend: implementación de módulos web y API, base de datos, seguridad y auditoría.
* Desarrollo Móvil/OMR: aplicación de captura y servicio de reconocimiento, integración con API.
* QA/Documentación (rotativo): diseño y ejecución de pruebas, trazabilidad, informes y manuales.

**Gestión de calidad**

* Criterios de aceptación por objetivo (definidos en Fase 1 y verificados en Fases 2 y 3).
* Plan de pruebas con casos críticos por flujo y evidencias (logs, capturas, datos de prueba).
* Revisiones formales: puerta de salida por etapa (Requisitos, Diseño, Integración, UAT).
* Métricas básicas: cumplimiento de hitos, porcentaje de casos críticos aprobados, incidencias abiertas/cerradas.

**Gestión de riesgos (síntesis)**

* OMR insuficiente: pruebas tempranas con set controlado, umbrales de calidad, alternativa de ingreso manual acotado.
* Desviaciones de alcance/tiempo: MVP estricto, control semanal y re-priorización de tareas no críticas.
* Integración tardía: hitos de integración intermedia, ambientes reproducibles, contratos de API.
* Seguridad/datos: checklist de autenticación/roles, auditoría de acciones y manejo de datos sensibles.

**Gestión de configuración y cambios**

* Control de versión: Git con ramas main/develop y feature branches; pull requests con revisión.
* Versionado: etiquetas por hito (F1, MVP, RC, Final).
* Control de cambios: registro de decisiones (ADR) y tickets vinculados a requisitos.

**Comunicación y seguimiento**

* Ritmo: reuniones breves asincrónicas (dailies) y tablero Kanban para seguimiento de tareas.
* Documentación viva: repositorio + wiki para requisitos, diseño, pruebas y manuales.

## Plan de trabajo para el proyecto APT

El plan de trabajo se alinea con las fases del APT:

* Fase 1 (Semanas 1–4): definición, levantamiento de requerimientos y arquitectura.
* Fase 2 (Semanas 5–15): construcción web/backend y móvil/OMR, integración y pruebas.
* Fase 3 (Semanas 16–18): preparación de la presentación final.

A continuación, se detallan las líneas de trabajo con actividades, recursos, duración, responsables y consideraciones:

Levantamiento de requerimientos y análisis de sistemas – Definición del proyecto y alcance

* Actividad: Elaboración de la guía de estudiante con antecedentes, relevancia, objetivos y metodología.
* Recursos: Plantilla institucional, repositorio, herramientas de documentación.
* Duración: Semanas 1–2.
* Responsable: Equipo completo.
* Observaciones: Posible falta de alineación inicial; mitigada con reuniones de coordinación.

Ingeniería de software y requisitos – Especificación de requerimientos y casos de uso

* Actividad: Documentación de RF/RNF y diseño de casos de uso principales.
* Recursos: Herramientas de modelado (Mermaid, Draw.io), repositorio.
* Duración: Semanas 2–4.
* Responsable: Subgrupo de análisis/diseño.
* Observaciones: Riesgo de sobreextensión del alcance; mitigado priorizando el MVP.

Arquitectura de software – Diseño de arquitectura y modelo de datos

* Actividad: Elaboración de diagramas de arquitectura y modelo entidad–relación para guiar el desarrollo.
* Recursos: Herramientas de diagramación, repositorio.
* Duración: Semanas 3–4.
* Responsable: Subgrupo de arquitectura.
* Observaciones: Puede requerir ajustes en la fase de desarrollo.

Desarrollo web y backend – Implementación módulo web y API

* Actividad: Construcción del banco de preguntas, gestión de evaluaciones y generación de PDF con QR.
* Recursos: IDEs (VSCode, IntelliJ), repositorio, base de datos.
* Duración: Semanas 5–10.
* Responsable: Subgrupo web/backend.
* Observaciones: Riesgo de retraso técnico; mitigado con control de versiones y revisiones periódicas.

Desarrollo móvil y OMR – Implementación aplicación móvil y servicio de calificación

* Actividad: Creación de la app de captura de respuestas e integración del servicio de reconocimiento óptico de marcas.
* Recursos: Dispositivo móvil, librerías OCR, repositorio.
* Duración: Semanas 6–11.
* Responsable: Subgrupo móvil/OMR.
* Observaciones: Riesgo en precisión de OCR; mitigado con pruebas tempranas y ajustes.

Aseguramiento de calidad – Pruebas parciales y validación del MVP

* Actividad: Ejecución de pruebas unitarias, de integración y validaciones internas del MVP.
* Recursos: Herramientas de testing, dataset de prueba.
* Duración: Semanas 9–13.
* Responsable: Equipo completo.
* Observaciones: Posible falta de tiempo para pruebas; priorización en funcionalidades críticas.

Gestión de proyectos – Elaboración de informes y evidencias

* Actividad: Redacción de informe de avance, informe final, manual de usuario y guía técnica.
* Recursos: Plantillas institucionales, editor de texto.
* Duración: Semanas 5–15.
* Responsable: Subgrupo de documentación.
* Observaciones: Riesgo de retraso en consolidación; mitigado con entregas parciales por fase.

Comunicación y presentación – Presentación del proyecto

* Actividad: Preparación de exposición grupal con demo del MVP y síntesis del proceso.
* Recursos: Presentación (PPT), demo del sistema.
* Duración: Semanas 16–18.
* Responsable: Equipo completo.
* Observaciones: Riesgo de falta de claridad; mitigado con ensayos previos.

## Propuesta de evidencias que darán cuenta del logro de las actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo de evidencia (avance o final)** | **Nombre de la evidencia** | **Descripción** | **Justificación** |
| Avance (Fase 1) | Autoevaluación de Competencias (Individual) | Documento individual donde cada estudiante reflexiona sobre sus competencias iniciales frente al proyecto. | Permite al docente conocer el nivel de preparación y autopercepción de cada integrante. |
| Avance (Fase 1) | Diario de Reflexión (Individual) | Registro escrito de aprendizajes y reflexiones personales durante la fase inicial. | Favorece la autoevaluación y evidencia la apropiación progresiva del proyecto. |
| Avance (Fase 1) | Autoevaluación Global (Individual) | Documento de autoevaluación que recopila percepciones y aprendizajes de la primera fase. | Facilita el seguimiento del desarrollo de competencias a nivel individual. |
| Avance (Fase 1) | Guía Estudiante – Fase 1: Definición Proyecto APT | Documento grupal con la definición inicial del proyecto: antecedentes, relevancia, objetivos y metodología. | Asegura que el proyecto esté correctamente planteado y validado con el docente. |
| Avance (Fase 1) | Presentación Proyecto (Fase 1) | Exposición grupal en la que se comunica la propuesta inicial del proyecto. | Permite evaluar la capacidad de síntesis, comunicación y justificación de la propuesta. |
| Avance (Fase 1) | Formativa Fase 1 | Documento grupal con el desarrollo formativo de la definición del proyecto. | Complementa la validación escrita con un formato académico requerido por la asignatura. |
| Avance (Fase 1) | Planilla de Evaluación Fase 1 | Registro enviado por el equipo como evidencia administrativa. | Refleja cumplimiento formal de la entrega en la asignatura. |
| Final (Fase 2) | Informe de Avance | Documento que reporta el progreso en el desarrollo del sistema, incluyendo discovery, requerimientos, casos de uso y primeras pruebas. | Permite retroalimentación docente y ajustar el rumbo del proyecto antes de la entrega final. |
| Final (Fase 2) | Prototipo Funcional / MVP | Sistema en su versión mínima viable con los módulos principales (banco de preguntas, evaluaciones, ingesta de respuestas, calificación automática). | Evidencia el logro técnico y la aplicación de las competencias de la carrera. |
| Final (Fase 2) | Resultados de Pruebas Iniciales | Evidencias de casos de prueba ejecutados y validación parcial de funcionalidades. | Asegura calidad y funcionamiento antes de la entrega final. |
| Final (Fase 3) | Informe Final del Proyecto | Documento que consolida la experiencia completa: problema, diseño, desarrollo, pruebas, resultados y reflexión. | Permite la evaluación integral del trabajo realizado por el equipo. |
| Final (Fase 3) | Manual de Usuario y Guía Técnica | Documentación para usuarios finales y técnicos, con instrucciones de uso e instalación. | Garantiza la continuidad y aplicabilidad práctica del proyecto. |
| Final (Fase 3) | Presentación Proyecto | Exposición grupal ante la comisión evaluadora, mostrando funcionamiento del MVP y principales evidencias. | Permite evaluar la capacidad de comunicación y defensa técnica del equipo. |

.

## Conclusions

From the initial “Definición del Proyecto APT” through the planning artifacts completed to date, the work demonstrates a coherent, feasible, and educationally meaningful path to a high-impact MVP. The team articulated a well-bounded problem, justified its relevance to the educational domain, and translated it into a clear objective with specific, verifiable sub-objectives. A classical waterfall methodology—requirements → design → construction → integration/testing → presentation—proved appropriate for a single development cycle, and it anchored traceability across deliverables (Requirements & Use Cases → Architecture & Data Model → Work Plan & Gantt → Testing and Evidence).

The process produced tangible evidence: the definition guide, the requirements catalog with use cases, the architectural view and relational model, the phased work plan, the Mermaid Gantt, the quality approach (acceptance criteria, RF→UC→Tests matrix), and a reasoned feasibility argument considering time, resources, risks, and mitigations. This sequence validated the project’s viability within the semester and strengthened discipline around scope (MVP), security and auditability, and quality assurance practices (unit, integration, and UAT).

Beyond compliance, the experience was enriching. It consolidated project leadership, full-stack engineering (web/API and mobile/OMR), and data/reporting capabilities, while connecting technology decisions to pedagogical impact and user trust. It also aligned with professional interests across the team—management, BI/QA/security/cloud, and scalable software design—turning individual goals into shared outcomes.

In sum, the work to date confirms both feasibility and readiness: the problem–solution fit is solid, objectives are measurable, risks are understood with concrete mitigations, and the plan is synchronized with the APT phases. This foundation positions the project to enter development confidently and to deliver a credible, socially relevant MVP, making the journey thus far both successful and deeply formative.

## Project Reflection

The GRADE project showed a clear connection between the identified problem and the proposed solution. Focusing on a well-defined MVP and following a waterfall approach allowed structured planning from requirements to presentation, ensuring traceability and a functional prototype with practical educational impact.

Key lessons included the importance of precise scope, early definition of non-functional requirements, and careful management of technical risks, especially in the OMR component. Clear communication and documented decisions helped reduce ambiguity and streamline integration.

Strengths observed were the structured methodology, modular architecture, and consistent attention to quality and traceability. Areas for improvement include refining acceptance criteria, enhancing security measures, and establishing performance baselines for grading and OMR processes.

The next phase focuses on finalizing MVP features, performing OMR testing, integrating components smoothly, and completing user and technical documentation incrementally.

Ethically, handling educational data reinforced the need for controlled access, auditability, and transparency, ensuring trust and integrity while delivering tangible benefits for educators.