

Analisis de requerimientos para el sprint #1

David Andres Monroy cod: 160005019, Alexander Auner Vallejos R cod: 160005498, Daniel Felipe Vega cod: 160005039, Juan Diego Vélez cod: 16005032, Lucero Castillo cod: 160005499, Juliana Morales: 160005020

I. INTRODUCCIÓN

El presente documento tiene como objetivo identificar y especificar los requerimientos del sistema de gestión académica enfocado en la administración del pensum y sus componentes curriculares. Esta documentación es fundamental para establecer una visión clara y compartida entre los interesados del proyecto, sirviendo como base para el diseño, desarrollo e implementación del sistema.

En primer lugar, se presentan los requerimientos funcionales, los cuales detallan las funciones esenciales que el sistema debe cumplir para satisfacer las necesidades de los usuarios. A continuación, se describen los requerimientos no funcionales, que establecen los criterios de calidad relacionados con el rendimiento, la seguridad, la usabilidad, entre otros aspectos técnicos del sistema.

Posteriormente, se incluye la especificación mediante historias de usuario, donde se representan las necesidades del sistema desde la perspectiva de cada rol involucrado. Estas historias facilitan la comprensión y validación de los requerimientos, permitiendo al equipo de desarrollo enfocar sus esfuerzos en entregar valor a los usuarios finales.

II. IDENTIFICACION DE REQUERIMIENTOS

La encuesta realizada a los estudiantes, complementada con entrevistas al cliente, permitió los siguientes requerimientos, riesgos y restricciones:

REQUERIMIENTOS FUNCIONALES	REQUERIMIENTOS NO FUNCIONALES
1. Gestión de Roles y Accesos <ul style="list-style-type: none">Decano (Superusuario):<ul style="list-style-type: none">Asignar y administrar roles, en especial los de directores de programa para cada carrera.Recibir notificaciones e información sobre cambios en los planes de estudio.Director de Programa:<ul style="list-style-type: none">Administrar el currículum de la carrera.Cargar información detallada de cada curso, incluyendo expectativas, guías de laboratorio y descripción.Designar roles auxiliares (secretarios, directivos y miembros del comité de programa).Secretario del director de Programa:<ul style="list-style-type: none">Asistir en las funciones administrativas y operativas del currículum.Comité de programa:<ul style="list-style-type: none">Recibir y evaluar propuestas de modificación de cursos o planes de estudio.Aprobar, rechazar o solicitar ajustes en las modificaciones de cursos o planes de estudios antes de enviarlas al director de programa	1. Seguridad <ul style="list-style-type: none">Mecanismos robustos de autenticación, incluyendo la verificación mediante notificación al celular.Registro y monitoreo de auditorías de operaciones para asegurar la trazabilidad. 2. Usabilidad <ul style="list-style-type: none">Diseño de una interfaz intuitiva, fácil de usar y adaptada a los distintos perfiles de usuario.Experiencia de usuario optimizada en dispositivos de escritorio 3. Rendimiento <ul style="list-style-type: none">Tiempos de carga optimo al procesamiento de información curricular.Capacidad para soportar múltiples usuarios concurrentes sin afectar el desempeño del sistema. 4. Escalabilidad <ul style="list-style-type: none">Arquitectura modular que permita la incorporación de nuevas funcionalidades, roles y la expansión de la base de datos sin degradar el rendimiento. 5. Mantenibilidad <ul style="list-style-type: none">Código bien documentado y modular que facilite futuras actualizaciones y correcciones.

<ul style="list-style-type: none">Estudiantes:<ul style="list-style-type: none">Ingresar al sistema para visualizar el pensum de la carrera.Consultar la información detallada de cada materia al hacer clic sobre ella. 2. Módulo de Gestión Curricular <ul style="list-style-type: none">Creación y actualización de planes de estudio para cada carrera.Funcionalidad para subir un archivo Excel que, al ser procesado, genere automáticamente el pensum.Almacenamiento y consulta de pensums anteriores para mantener un historial. 3. Operaciones CRUD y Auditoría <ul style="list-style-type: none">Implementación de operaciones de creación, lectura, actualización y eliminación (CRUD) para cursos, planes de estudio y demás información.Registro de auditoría que guarde:<ul style="list-style-type: none">Usuario que realizó la operación.Dirección IP de origen.Fecha y hora de la operación. 4. Seguridad y Autenticación <ul style="list-style-type: none">Acceso al sistema mediante usuario y contraseña.Implementación de autenticación de dos factores: envío de notificación al celular del usuario para aprobar el acceso. 5. Interfaz de Usuario e Interacción <ul style="list-style-type: none">Diseño de una interfaz intuitiva y amigable que se adapte a las necesidades de los diferentes roles (decano, director, secretarios y estudiantes).Visualización interactiva del pensum y detalle de cada curso al seleccionarlo. 6. Notificaciones y Alertas <ul style="list-style-type: none">Envío de notificaciones automáticas al decano sobre cambios en los planes de estudio.Notificación para la autenticación de acceso y otras alertas relevantes según el rol. 7. Integración y Procesamiento de Archivos Excel <ul style="list-style-type: none">Funcionalidad para la importación de archivos Excel que contengan la información del plan de estudios.Procesamiento automático del Excel para generar o actualizar el pensum en el sistema.	<ul style="list-style-type: none">Mecanismos de registro (logging) detallado para facilitar la identificación y resolución de incidencias. 6. Fiabilidad <ul style="list-style-type: none">Estrategias de respaldo periódico de datos y mecanismos de recuperación ante fallos.
---	--

OTROS ELEMENTO A CONSIDERAR: RIESGOS Y REQUERIMIENTOS DE PROYECTO
Restricciones <ul style="list-style-type: none">Presupuesto limitado: El desarrollo debe ajustarse a recursos financieros restringidos.Compatibilidad tecnológica: El sistema debe ser compatible con la infraestructura y software existentes en la facultad (por ejemplo, servidores, sistemas operativos, navegadores).Límites de tiempo: El proyecto debe completarse antes del inicio del próximo periodo académico. Riesgos <ul style="list-style-type: none">Falta de adopción por parte de los usuarios (estudiantes y docentes): Puede ocurrir si el sistema no resulta intuitivo o confiable.Problemas de seguridad y filtración de datos: Si no se implementan correctamente los mecanismos de autenticación y cifrado.Cambios frecuentes en los planes de estudio: Podrían requerir actualizaciones constantes en el sistema.Dependencia de proveedores externos (servicios de SMS o notificaciones): Podría haber problemas de disponibilidad o costos adicionales. Otros Elementos Relevantes <ul style="list-style-type: none">Normativas de Protección de Datos: Asegurar el cumplimiento de normativas locales e internacionales de privacidad y manejo de información (p. ej., GDPR u otras aplicables).Integración con Sistemas de la Facultad: Posible necesidad de integración con otros sistemas (por ejemplo, sistemas de matrícula o plataformas académicas).

III. ESPECIFICACION DE REQUERIMIENTOS

En la especificación de requerimientos se encuentran las historias de usuario, los requerimientos funcionales y no funcionales estipulados para el primer sprint:

- Historias de usuario:

Las historias de usuario representan los requerimientos desde la perspectiva de los diferentes actores del sistema. Cada historia describe una necesidad específica junto con los criterios de aceptación, asegurando que la funcionalidad se desarrolle de acuerdo con las expectativas del usuario final y cumpla con los objetivos del proyecto.

Código: HU-01	Usuario: Decano
Título: Validación de decisiones curriculares.	
Prioridad: Alta	Tipo: Funcionalidad
Descripción: Como decano, quiero poder consultar propuestas curriculares generadas por los directores de programa o el comité académico, evaluarlas en detalle y decidir su aprobación o rechazo, de manera que las decisiones se registren automáticamente en el sistema y los cambios curriculares aprobados se actualicen en el pensum y cursos correspondientes. Esto garantizará la trazabilidad y control académico de las decisiones tomadas.	
Criterios de aceptación: - Visualización de propuestas: <ul style="list-style-type: none"> El decano debe tener acceso a un módulo donde se listan todas las propuestas curriculares pendientes de aprobación. Cada propuesta debe incluir: nombre de la propuesta, fecha de creación, descripción, objetivos del cambio, impacto en el pensum, y usuario que la creó. - Decisión sobre propuestas: <ul style="list-style-type: none"> El decano debe poder aprobar o rechazar una propuesta mediante botones específicos ("Aprobar" o "Rechazar"). Al tomar una decisión, el sistema debe requerir un comentario obligatorio justificando la decisión. - Registro de decisiones: <ul style="list-style-type: none"> El sistema debe guardar: ID de la propuesta, decisión tomada (aprobada o rechazada), comentario del decano, fecha y hora de la acción, y usuario responsable. Cada registro debe ser accesible para auditoría posterior. - Actualización de cambios: <ul style="list-style-type: none"> Si la propuesta es aprobada, el sistema debe actualizar automáticamente los elementos curriculares afectados (por ejemplo, actualizar el pensum, añadir/eliminar cursos, modificar contenidos). - Notificaciones: <ul style="list-style-type: none"> Una vez que el decano toma una decisión, se debe enviar automáticamente una notificación a los usuarios involucrados (director de programa, comité académico, etc.) informando el resultado. 	

Código: HU-02	Usuario: Decano
Título: Designación de directores de programa.	
Prioridad: Alta	Tipo: Funcionalidad
Descripción: Como decano, quiero asignar, modificar o revocar el rol de director de programa a diferentes profesores registrados en el sistema, de modo que pueda asegurar una correcta gestión académica de cada programa. Cada designación o cambio debe quedar registrado con trazabilidad, y los usuarios afectados deben recibir notificaciones claras y oportunas.	
Criterios de aceptación: - Listado de posibles directores: <ul style="list-style-type: none"> El sistema debe permitir al decano ver una lista de todos los profesores activos disponibles para asignación (nombre completo, correo electrónico institucional, programa actual, rol actual). - Asignación y modificación de rol: <ul style="list-style-type: none"> El decano debe poder seleccionar un profesor y asignarle el rol de "Director de Programa" a un programa específico. Si ya existe un director asignado, el sistema debe solicitar confirmación explícita para reemplazar al actual. La acción de asignación o modificación debe requerir una justificación que quede registrada. - Registro de cambios: <ul style="list-style-type: none"> Cada acción de asignación, modificación o revocación debe ser registrada incluyendo: usuario afectado, rol anterior, rol nuevo, programa asociado, fecha y hora de cambio, usuario responsable de la acción. - Notificaciones automáticas: <ul style="list-style-type: none"> Una vez asignado o modificado un rol, el profesor debe recibir una notificación por correo electrónico informándole del cambio de rol, incluyendo: título del nuevo rol, programa asignado y observaciones si las hay. - Historial de roles: <ul style="list-style-type: none"> El sistema debe mantener un historial visible de todos los cambios de rol realizados para cada profesor, accesible solo para usuarios con permisos administrativos. 	

Código: HU-03	Usuario: Director de programa
Título: Generación y actualización del pensum	
Prioridad: Alta	Tipo: Funcionalidad
Descripción: Como director de programa, quiero poder generar nuevos pensums o actualizar versiones existentes en función de las decisiones curriculares aprobadas, asegurando que cada modificación sea registrada, versionada y visible para los usuarios autorizados, manteniendo un historial claro de los cambios realizados.	
Criterios de aceptación: - Generación de pensum: <ul style="list-style-type: none"> El sistema debe permitir crear un nuevo pensum desde cero o duplicar un pensum existente para modificaciones. El director debe ingresar: nombre del pensum, año de vigencia, descripción general, listado de cursos asociados. - Actualización de pensum: <ul style="list-style-type: none"> Al actualizar un pensum, el sistema debe requerir la selección de la versión a modificar y la descripción del cambio realizado. Cada actualización debe generar automáticamente una nueva versión numerada (v1, v2, etc.), preservando las versiones anteriores para consulta. - Control de cambios: <ul style="list-style-type: none"> Cada cambio (creación, edición, eliminación de un curso) debe quedar registrado con fecha, usuario responsable y tipo de cambio. - Acceso a versiones anteriores: <ul style="list-style-type: none"> Los usuarios autorizados deben poder consultar todas las versiones históricas de un pensum. - Notificaciones: <ul style="list-style-type: none"> El sistema debe notificar automáticamente a los miembros del comité académico y otros usuarios autorizados cuando un nuevo pensum o una actualización sea publicada. 	

- Requerimientos Funcionales:

En estas tablas se detallan las características, capacidades esenciales y funciones específicas que deben implementarse, asegurando que el sistema proporcione la información y las herramientas necesarias para la gestión del pensum académico y la administración de usuarios.

Identificación del requerimiento:	RF-06
Nombre del requerimiento:	Generación del Pensum
Descripción del requerimiento:	El sistema debe permitir al director de programa proponer y estructurar un nuevo pensum o actualizar uno existente. Esta propuesta será enviada al decano, quien tiene la función de aprobarla. Una vez aprobada, se debe registrar como una nueva versión del pensum.
Requerimiento no funcional:	RNF-01, RNF-04
Prioridad del requerimiento:	Alta

- Requerimientos No Funcionales:

En las siguientes tablas establecen los criterios de calidad, seguridad, rendimiento y escalabilidad (conocidos como requerimientos no funcionales), que debe cumplir el sistema. Estos aspectos garantizan que la plataforma sea eficiente, segura y fácil de usar, permitiendo su correcto funcionamiento y adaptación a futuras necesidades.

Identificación del requerimiento:	RNF-01
Nombre el requerimiento:	Seguridad
Características:	Autenticación de dos factores y auditoría de accesos
Descripción del requerimiento:	El sistema debe proteger los datos de los usuarios mediante autenticación robusta. Toda acción sensible (como modificar pensums o información de usuarios) debe ser registrada y accesible para fines de auditoría.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RNF-02
Nombre el requerimiento:	Usabilidad
Características:	Diseño responsivo y minimalista
Descripción del requerimiento:	La interfaz debe ser clara, moderna, y adaptable a diferentes dispositivos (PC, tablet, móvil), permitiendo que los usuarios comprendan y utilicen fácilmente las funcionalidades sin necesidad de formación previa.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RNF-03
Nombre el requerimiento:	Rendimiento
Características:	Respuesta rápida, optimización de consultas
Descripción del requerimiento:	El tiempo de respuesta del sistema no debe superar los 2 segundos en operaciones comunes como consulta de cursos o pensum. Las bases de datos deben estar optimizadas para garantizar eficiencia.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RNF-04
Nombre el requerimiento:	Escalabilidad
Características:	Arquitectura distribuida y balanceo de carga
Descripción del requerimiento:	La arquitectura del sistema debe permitir añadir nuevos servidores o recursos sin necesidad de rediseñar el sistema, soportando un crecimiento de hasta 10.000 usuarios concurrentes.
Prioridad del requerimiento:	Media

Identificación del requerimiento:	RNF-05
Nombre el requerimiento:	Mantenibilidad
Características:	Modularidad y buenas prácticas de desarrollo.
Descripción del requerimiento:	El sistema debe desarrollarse siguiendo principios de programación limpia, utilizando estructuras modulares, documentando el código y aplicando patrones que faciliten el mantenimiento y evolución del software.
Prioridad del requerimiento:	Alta

IV. CONCLUSIONES

La identificación y especificación de los requerimientos presentados en este documento constituyen un paso fundamental para el desarrollo del sistema de gestión curricular de la Facultad de Ciencias Básicas e Ingeniería. A través de los requerimientos funcionales, no funcionales e historias de usuario, se ha establecido una base clara y estructurada que refleja las necesidades de los distintos actores del sistema, garantizando que el software responda adecuadamente a los objetivos académicos y administrativos de la institución.

La documentación detallada permite al equipo de desarrollo comprender el alcance del sistema, facilita la planificación y ejecución del proyecto, y asegura que las funcionalidades esperadas estén alineadas con los procesos reales de la gestión curricular.