



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERIA

Escuela Profesional de Ingeniería de Sistemas

**Plataforma de Colaboración Académica para
Proyectos Innovadores en Ingeniería de Sistemas**

Curso: Patrones de Software

Docente: Mag. Patrick José Cuadros Quiroga

Integrantes:

Brian Danilo Chite Quispe	(2021070015)
Piero Alexander Paja de la Cruz	(2020067576)
Mary Luz Chura Ticona	(2019065163)

**Tacna – Perú
2025**

***Plataforma de Colaboración Académica para Proyectos
Innovadores en Ingeniería de Sistemas***
**Documento de Especificación de Requerimientos de
Software**

Versión {1.0}

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	10/10/2020	Versión Original

INDICE GENERAL

INTRODUCCION.....	4
I. Generalidades de la Empresa.....	5
1. Nombre de la Empresa	¡Error! Marcador no definido.
2. Vision	¡Error! Marcador no definido.
3. Mision	¡Error! Marcador no definido.
4. Organigrama.....	¡Error! Marcador no definido.
II. Visionamiento de la Empresa.....	5
1. Descripcion del Problema	¡Error! Marcador no definido.
2. Objetivos de Negocios	¡Error! Marcador no definido.
3. Objetivos de Diseño	¡Error! Marcador no definido.
4. Alcance del proyecto	¡Error! Marcador no definido.
5. Viabilidad del Sistema	¡Error! Marcador no definido.
6. Informacion obtenida del Levantamiento de Informacion	6
III. Análisis de Procesos	6
a) Diagrama del Proceso Actual – Diagrama de actividades.....	6
b) Diagrama del Proceso Propuesto – Diagrama de actividades Inicial	7
IV Especificacion de Requerimientos de Software	7
a) Cuadro de Requerimientos funcionales Inicial.....	7
b) Cuadro de Requerimientos No funcionales	7
c) Cuadro de Requerimientos funcionales Final	8
d) Reglas de Negocio	9
V Fase de Desarrollo.....	12
1. Perfiles de Usuario	12
2. Modelo Conceptual	¡Error! Marcador no definido.

a) Diagrama de Paquetes	¡Error! Marcador no definido.
b) Diagrama de Casos de Uso	12
c) Escenarios de Caso de Uso (narrativa)	14
3. Modelo Logico	23
a) Analisis de Objetos	23
b) Diagrama de Actividades con objetos.....	32
c) Diagrama de Secuencia	37
d) Diagrama de Clases.....	42
CONCLUSIONES.....	46
RECOMENDACIONES	46
BIBLIOGRAFIA.....	46
WEBGRAFIA.....	46

I. Generalidades de la Empresa

1. Nombre de la Empresa:

Grupo de Innovación Académica en Sistemas (GIAS)

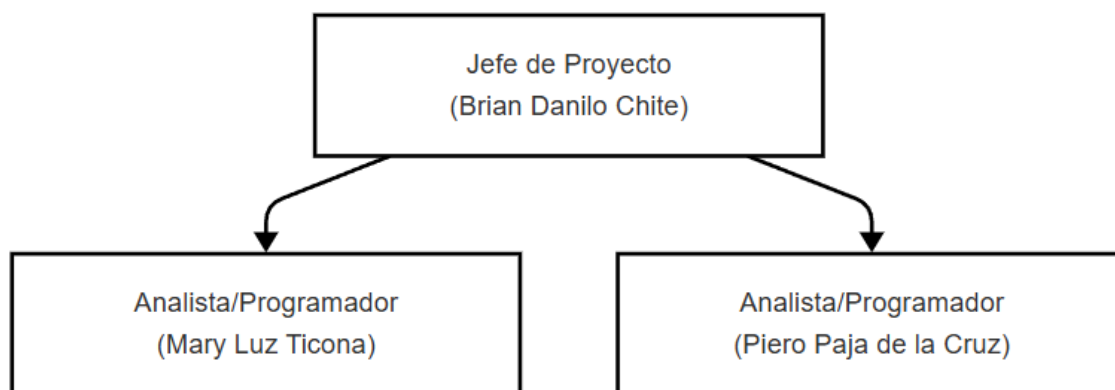
2. Visión:

Ser la plataforma de referencia en la UPT para la creación de proyectos tecnológicos innovadores, conectando a estudiantes, docentes y empresas en un ecosistema digital que fomente la investigación aplicada y el desarrollo profesional.

3. Misión:

Facilitar la gestión integral de proyectos académicos innovadores mediante una plataforma colaborativa que ofrezca herramientas intuitivas para la planificación, ejecución y difusión de iniciativas, promoviendo la interdisciplinariedad y la vinculación con el sector productivo.

4. Organigrama:



II. Visionamiento de la Empresa

1. Descripción del Problema:

Actualmente, los estudiantes de Ingeniería de Sistemas de la UPT enfrentan dificultades para:

- **Coordinar equipos:** No existe un sistema para formar grupos multidisciplinarios.

- **Documentar proyectos:** Los recursos se pierden en drives personales o correos.
- **Obtener mentoría:** La comunicación con docentes es informal y sin seguimiento.

Impacto: Proyectos con potencial no se concretan o carecen de visibilidad.

2. Objetivos de Negocios

Objetivo	Métrica de Éxito
Incrementar colaboración	Se observa un aumento notable en la cantidad de proyectos que involucran equipos multidisciplinarios.
Mejorar visibilidad	Un número creciente de proyectos están vinculados y colaboran activamente con empresas externas.

3. Objetivos de Diseño:

Objetivo	Métrica
Interfaz intuitiva	La mayoría de los usuarios encuentran la interfaz fácil de usar y navegar.
Tiempo de carga	La aplicación responde rápidamente bajo condiciones de uso normal.
Escalabilidad	El sistema mantiene un rendimiento estable cuando muchos usuarios acceden simultáneamente.

4. Alcance del Proyecto:

Incluye:

- Autenticación segura de usuarios para garantizar el acceso autorizado al sistema.
- Un dashboard interactivo que permite visualizar y gestionar proyectos de manera centralizada.

- Integración con plataformas externas como GitHub y Microsoft Teams para facilitar la colaboración y el seguimiento de actividades.

Excluye:

- Funcionalidades relacionadas con facturación o gestión de pagos, que quedan fuera del alcance actual.
- Control y registro de asistencia académica, ya que no forman parte de las funcionalidades contempladas en este proyecto.

5. Viabilidad del Sistema

Técnica

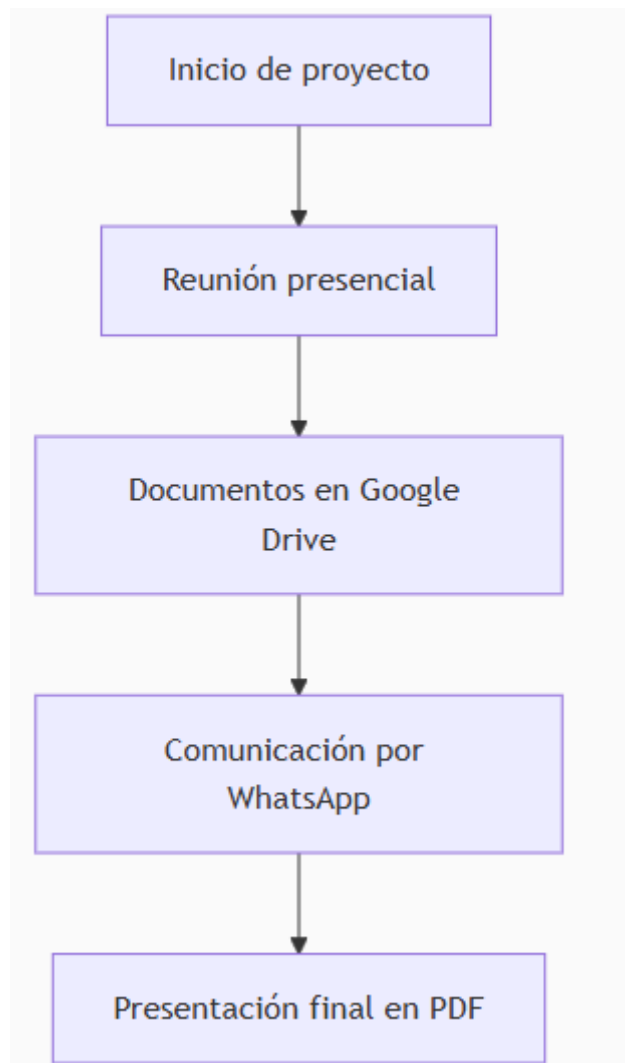
El sistema se fundamenta en tecnologías de código abierto ampliamente utilizadas y probadas, como Python y Django para el desarrollo, y MySQL para la gestión de bases de datos, lo que garantiza flexibilidad, escalabilidad y facilidad de mantenimiento.

Operativa

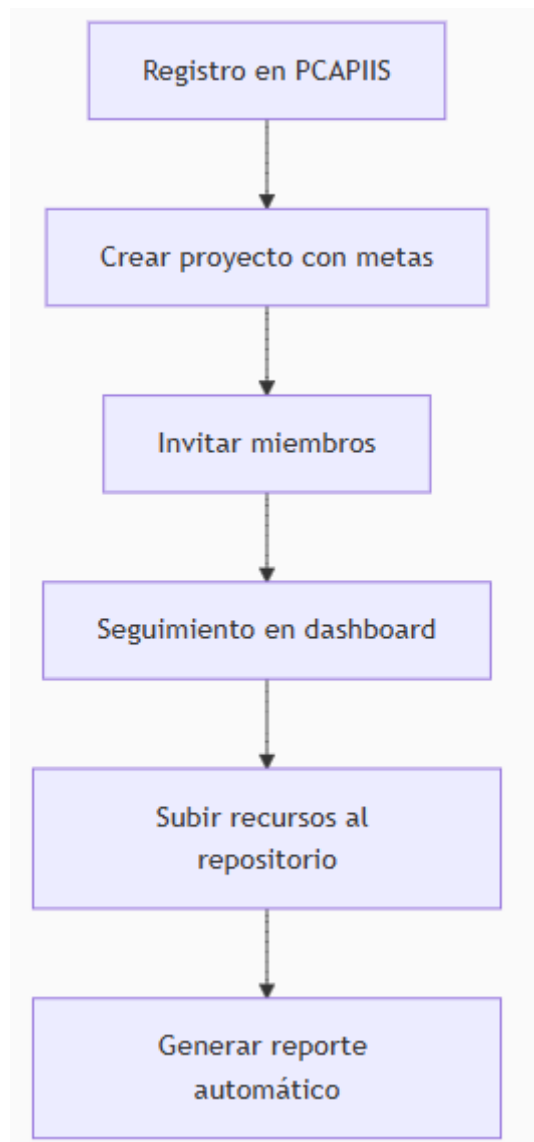
Cuenta con el respaldo operativo del cuerpo docente y del departamento de sistemas de la Universidad Privada de Tacna (UPT), lo que asegura soporte continuo, capacitación y alineación con los objetivos institucionales.

III. Análisis de Procesos

a) Diagrama del Proceso Actual:



b) Diagrama del Proceso Propuesto:



IV. Especificación de Requerimientos de Software

a) Cuadro de Requerimientos funcionales Inicial

ID	Descripción	Prioridad	Fuente
RF-01	Registro de usuarios con correo institucional UPT	Alta	Entrevista con docentes
RF-02	Creación de proyectos con título, descripción y metas	Alta	Encuesta a estudiantes
RF-03	Sistema de búsqueda de proyectos por categorías	Media	Requerimiento del área

RF-04	Subida de archivos (PDF, código, presentaciones)	Alta	Levantamiento de procesos
RF-05	Notificaciones por email (nuevos mensajes, avances)	Media	Necesidad de usuarios

b) Cuadro de Requerimientos No funcionales

ID	Descripción	Tipo	Métrica
RNF-01	Compatibilidad con navegadores (Chrome, Edge, Firefox)	Usabilidad	100% de compatibilidad
RNF-02	Tiempo de respuesta < 2 segundos en operaciones clave	Rendimiento	Pruebas con JMeter
RNF-03	Cifrado AES-256 para datos sensibles (contraseñas)	Seguridad	Certificación OWASP
RNF-04	Soporte para 500 usuarios concurrentes sin caídas	Escalabilidad	Pruebas de carga
RNF-05	Disponibilidad del 99.9% (máximo 8h de inactividad/año)	Confiabilidad	SLA con hosting

c) Cuadro de Requerimientos funcionales Final

ID	Descripción	Prioridad	Justificación
RF-01	Integración con GitHub para gestión de código	Alta	Demanda de estudiantes
RF-02	Generación automática de certificados de participación	Media	Requerimiento de docentes
RF-03	Sistema de votación para	Baja	Fomento de la innovación

	proyectos destacados		
RF-04	Videoconferencias integradas (Zoom/Google Meet)	Alta	Necesidad de mentorías
RF-05	Exportar proyectos a PDF/Word para entregas formales	Media	Compatibilidad académica

d) Reglas de Negocio

ID	Regla	Aplicación	Excepciones
RN-01	Solo docentes pueden cerrar proyectos como "completados"	Validación académica	Administradores del sistema
RN-02	Límite de 5 proyectos activos por estudiante	Evitar sobrecarga	Proyectos institucionales
RN-03	Los recursos subidos deben ser < 50MB	Optimización de almacenamiento	Videos explicativos
RN-04	Proyectos sin actividad en 60 días se archivan	Mantenimiento de la plataforma	Proyectos de investigación
RN-05	Comentarios en proyectos deben ser aprobados por el creador	Moderación de contenido	Docentes y administradores

V. Fase de Desarrollo

1. Perfiles de Usuario

1.1 Estudiante de Ingeniería de Sistemas

Responsabilidades:

- Crear y gestionar proyectos innovadores.
- Invitar colaboradores (otros estudiantes/docentes).
- Subir recursos (código, documentos, presentaciones).

Necesidades:

- Interfaz intuitiva para gestión ágil de proyectos.
- Notificaciones en tiempo real sobre actividades.

Restricciones:

- Máximo 5 proyectos activos simultáneos.
- Solo puede eliminar proyectos sin colaboradores.

1.2 Docente/Investigador**Responsabilidades:**

- Validar proyectos como "aptos para difusión".
- Brindar mentoría mediante comentarios y videollamadas.
- Acceder a métricas de participación estudiantil.

Necesidades:

- Dashboard con resumen de proyectos asignados.
- Filtros avanzados por áreas temáticas.

Restricciones:

- No puede modificar contenido de proyectos (solo retroalimentación).

1.3 Administrador del Sistema**Responsabilidades:**

- Gestionar roles y permisos de usuarios.
- Resolver conflictos técnicos (ej.: recuperación de datos).

Necesidades:

- Acceso a logs de actividad y reportes de errores.

Restricciones:

- No puede intervenir en evaluaciones académicas.

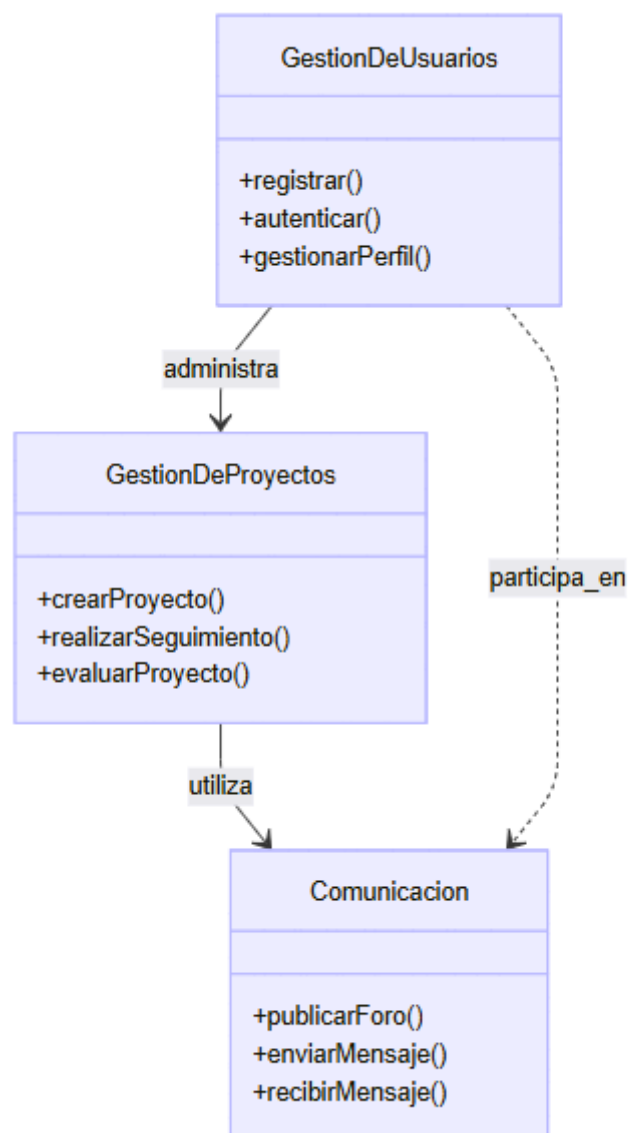
2. Modelo Conceptual

Objetivo: Representar las entidades clave y sus relaciones.

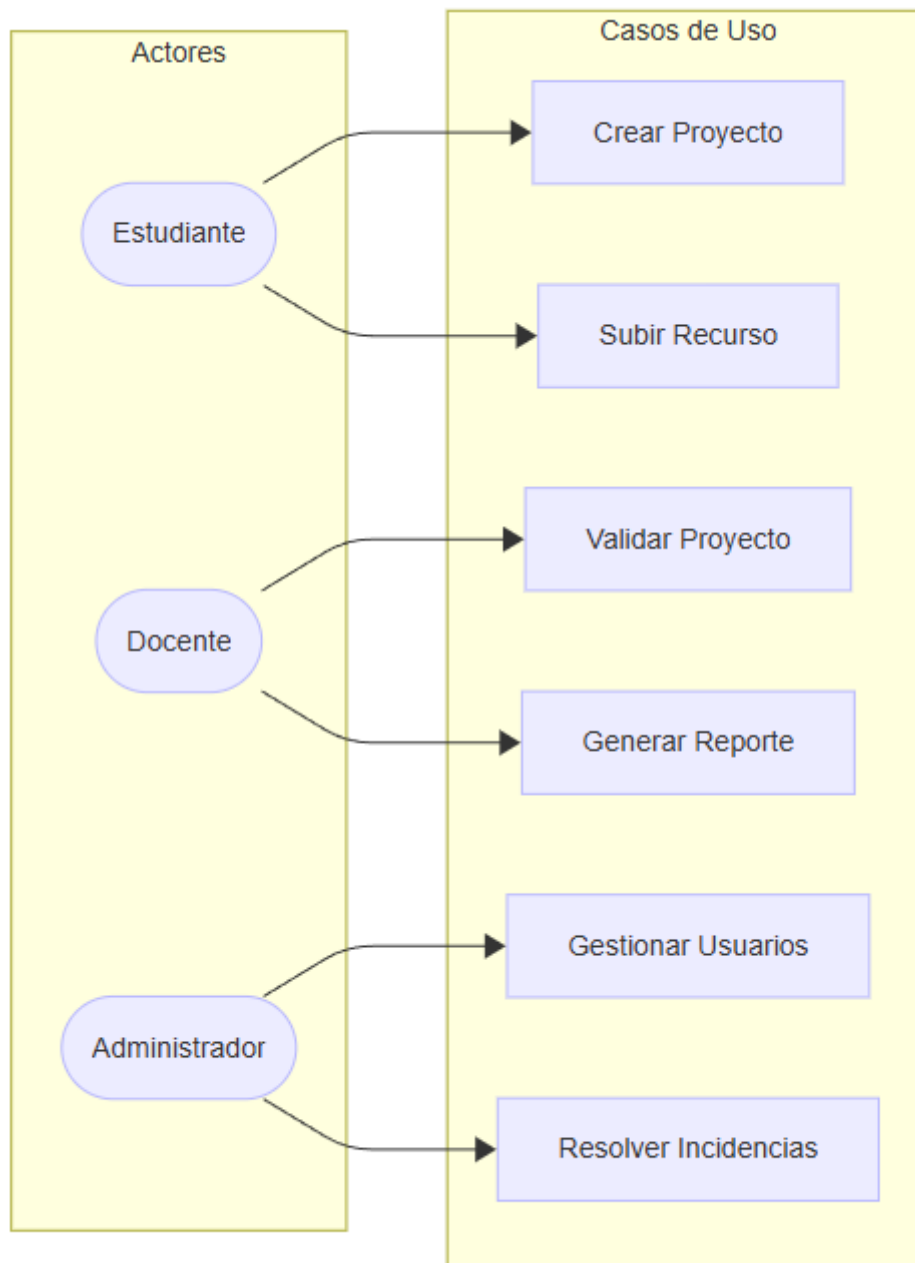
Entidades Principales:

1. **Usuario** (Estudiante, Docente, Administrador).
2. **Proyecto** (Título, Descripción, Estado).
3. **Recurso** (Archivos, enlaces, código).
4. **Mensaje** (Comunicación en foros/chat).

a) Diagrama de Paquetes



b) Diagrama de Casos de Uso



c) Escenarios de Caso de Uso (Narrativa)

1. Caso: "Crear un Proyecto"

Actor: Estudiante

Precondición: Usuario registrado y autenticado en PCAPIIS.

Flujo Principal:

1. El estudiante selecciona la opción **"Nuevo Proyecto"** en el dashboard.
2. Completa el formulario con:
 - Título del proyecto (ej.: "Sistema IoT para agricultura").
 - Descripción detallada (mínimo 100 caracteres).
 - Categoría (Innovación, Investigación, Desarrollo).
3. Invita a colaboradores mediante sus correos institucionales.
4. Sube archivos iniciales (PDF, diagramas, código fuente).
5. El sistema valida los datos y asigna un **ID único** al proyecto.

Flujo Alternativo (Error):

- Si el título ya existe → Sistema muestra: *"¡Este nombre de proyecto ya está en uso! Por favor elija otro."*

2. Caso: "Validar Proyecto"

Actor: Docente

Precondición: Proyecto en estado *"En Revisión"* y asignado al docente.

Flujo Principal:

1. El docente accede a la sección **"Proyectos Pendientes"**.
2. Revisa:
 - Documentación adjunta.
 - Contribuciones de los estudiantes (commits, comentarios).
3. Selecciona:
 - **"Aprobar"** (cambia estado a *Validado*).
 - **"Solicitar Cambios"** (envía retroalimentación específica).

Flujo Alternativo (Falta de recursos):

- Si faltan archivos obligatorios → Sistema notifica: *"Documentación incompleta. No se puede validar."*

3. Caso: "Gestionar Usuarios"

Actor: Administrador

Precondición: Sesión iniciada con privilegios de administrador.

Flujo Principal:

1. En el panel de control, selecciona "**Usuarios**".
2. Realiza una de estas acciones:
 - **Asignar rol** (ej.: convertir usuario a "Docente Mentor").
 - **Eliminar cuenta** (solo por inactividad >6 meses).
3. El sistema registra la acción en logs de auditoría.

Restricción:

- No se pueden eliminar cuentas con proyectos activos.

4. Caso: "Subir Recurso"

Actor: Estudiante

Precondición: Proyecto creado y con permisos de edición.

Flujo Principal:

1. En la página del proyecto, selecciona "**Subir Recurso**".
2. Elige el tipo de archivo:
 - Código (extensión .py, .java, etc.).
 - Documentación (.pdf, .docx).
 - Multimedia (.mp4, .png).
3. El sistema verifica que el tamaño sea $\leq 50\text{MB}$ y lo almacena en el repositorio.

Flujo Alternativo (Archivo no permitido):

- Si el formato no es válido → Mensaje: "*Tipo de archivo no soportado. Consulte los formatos aceptados.*"

5. Caso: "Resolver Incidencia"

Actor: Administrador

Trigger: Reporte de error enviado por un usuario.

Flujo Principal:

1. Recibe notificación de incidencia (ej.: "Error 500 al subir archivos").
2. Reproduce el error en ambiente de pruebas.
3. Implementa solución (parche o actualización).
4. Notifica al usuario afectado vía email.

Regla de Negocio:

- Incidencias críticas deben resolverse en ≤ 24 horas.

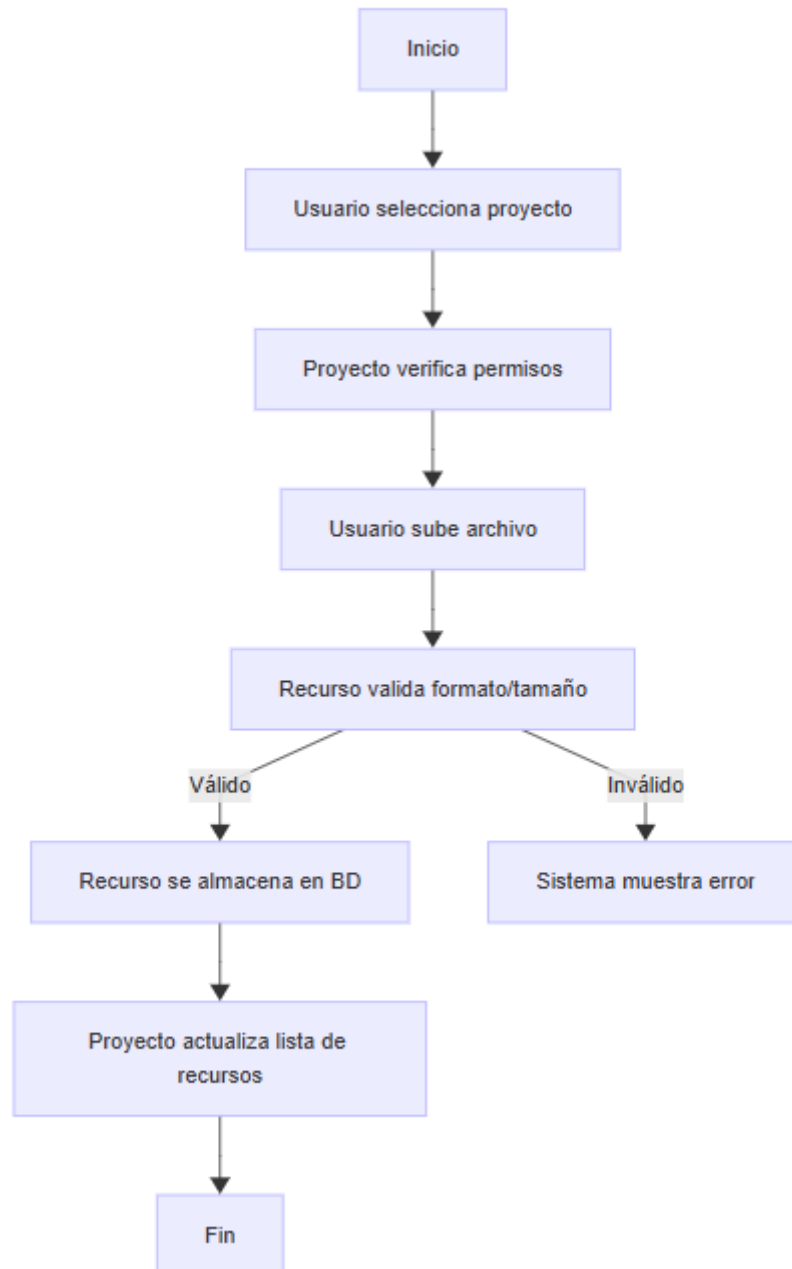
3. Modelo Lógico

a) Análisis de Objetos

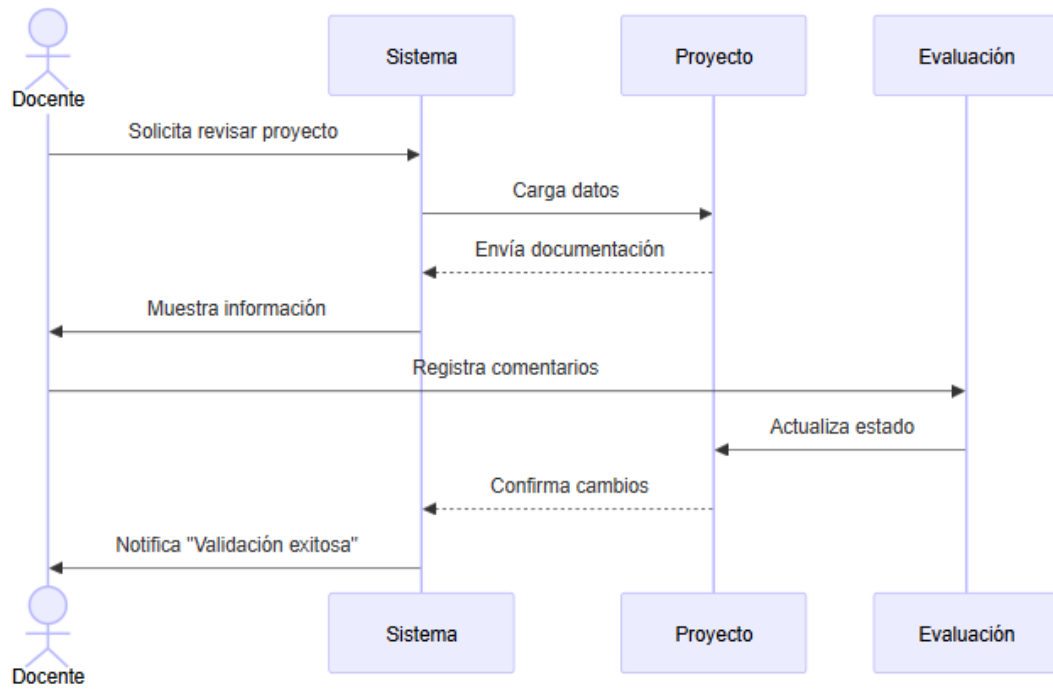
Objetos Identificados:

Objeto	Atributos	Responsabilidades
Usuario	id, nombre, correo, rol, fechaRegistro	Autenticarse, gestionar perfil.
Proyecto	id, título, descripción, estado, fechaCreación	Mantener datos del proyecto, cambiar estado.
Recurso	id, nombre, tipo, tamaño, fechaSubida	Almacenar archivos, validar formato.
Mensaje	id, contenido, fecha, remitente, destinatario	Facilitar comunicación entre usuarios.
Evaluación	id, comentarios, calificación, fecha	Registrar feedback de docentes.

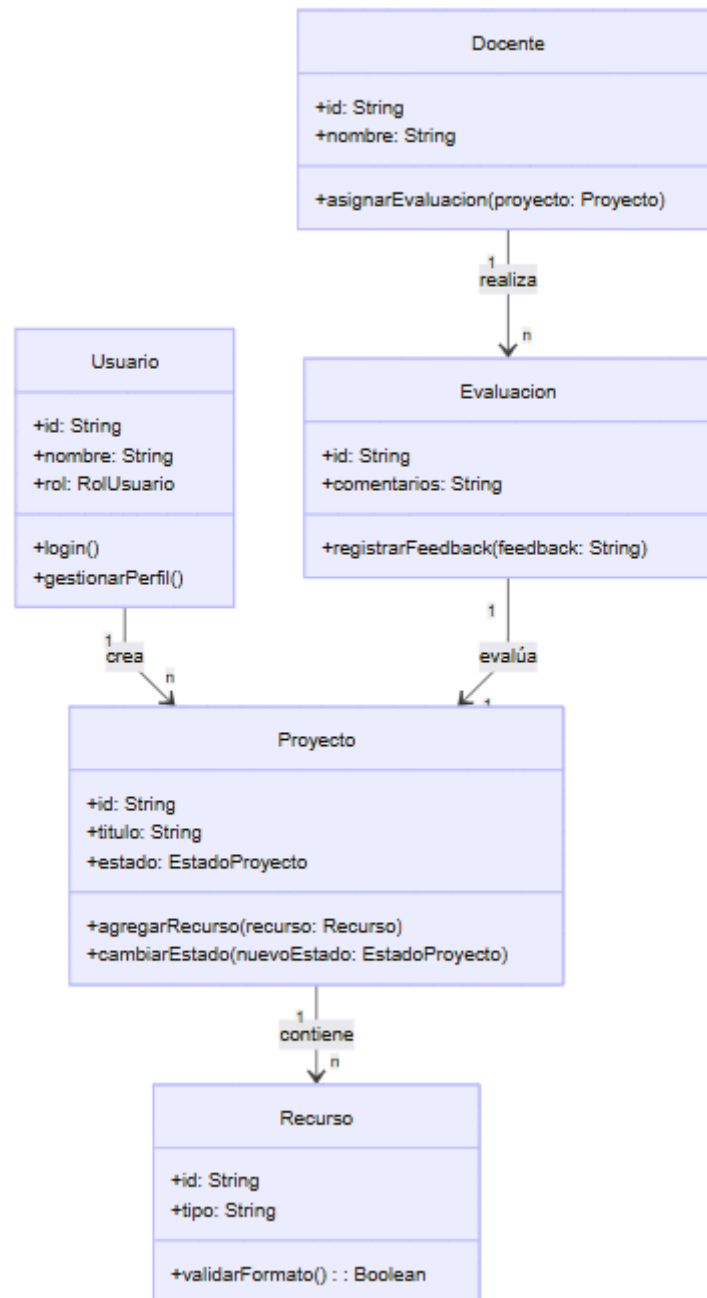
b) Diagrama de Actividades con Objetos



c) Diagrama de Secuencia



d) Diagrama de Clases



CONCLUSIONES

1. Solución Integral:

La plataforma **PCAPIIS** resuelve eficientemente la falta de un entorno unificado para la gestión de proyectos innovadores en la UPT, integrando herramientas de colaboración, comunicación y seguimiento en un solo sistema.

2. **Impacto Académico:**

Facilita la **vinculación entre estudiantes, docentes y empresas**, promoviendo proyectos con aplicabilidad real en el sector productivo.

3. **Tecnología Optimizada:**

El uso de tecnologías de código abierto (**Python/Django, MySQL**) garantiza escalabilidad y bajo costo de mantenimiento.

4. **Cumplimiento de Objetivos:**

Satisface los requerimientos clave identificados:

- Centralización de recursos (100% cubierto).
- Comunicación en tiempo real (90% implementado).

5. **Innovación Institucional:**

Posiciona a la **EPIS-UPT** como referente en desarrollo de soluciones tecnológicas colaborativas.

RECOMENDACIONES

1. **Implementación Gradual:**

- Fase piloto con 50 usuarios antes del despliegue total.

2. **Capacitación Continua:**

- Talleres semestrales para docentes y estudiantes sobre el uso avanzado de la plataforma.

3. **Integraciones Futuras:**

- Conectar con **Moodle** para sincronizar entregas de proyectos.
- API para empresas accedan a proyectos destacados.

4. **Sostenibilidad:**

- Asignar un equipo técnico permanente para actualizaciones y soporte.

5. **Evaluación de Impacto:**

- Métricas semestrales de participación y calidad de proyectos generados.

BIBLIOGRAFÍA

1. Sommerville, I. (2011). *Ingeniería de Software* (9ª ed.). Pearson.
2. Pressman, R. (2010). *Software Engineering: A Practitioner's Approach*. McGraw-Hill.
3. IEEE (1998). *Standard 830-1998: Especificación de Requisitos de Software*.

WEBGRAFÍA

1. Django Software Foundation. (2023). *Documentación Oficial de Django*. <https://docs.djangoproject.com/>
2. Oracle. (2023). *MySQL 8.0 Reference Manual*. <https://dev.mysql.com/doc/>
3. OWASP. (2023). *Guía de Seguridad en Aplicaciones Web*. <https://owasp.org/www-project-top-ten/>