



Programación Orientada a Objetos

Proyecto final

Nombre:

- Johaneris Sayrín Avalos Fernandez
- Gabriel Antonio Rojas
- Steven José Mejía
- Manuel Joaquín Chamorro
- Valeria Carolina Grijalva

Docente: Jose Duran Garcia

Managua 4 de noviembre del 2025.

Contenido

1.	Formato de Propuesta de Proyecto Final	1
1.1.	Integrantes del equipo	1
1.2.	Información general del proyecto	1
1.3.	Definición del problema	2
1.4.	Alcance del sistema.....	4
1.5.	Actores y usuarios del sistema.....	5
1.6.	Requerimientos funcionales y no funcionales	5
1.6.1.	Requerimientos funcionales:.....	5
1.6.2.	Requerimientos no funcionales:.....	6
1.7.	Modelo inicial del dominio	7
1.8.	URL del repositorio	7
1.9.	Cronograma estimado de desarrollo	8
1.10.	Recursos necesarios	8

1. FORMATO DE PROPUESTA DE PROYECTO FINAL

1.1. Integrantes del equipo

Cif	Nombre Completo	Rol
23021723	Johaneris Sayrín Ávalos Fernández	Líder
23021288	Valeria Carolina Grijalva Arévalo	Analista
24011094	Steven José Mejía Torrez	Diseñador
24010226	Gabriel Antonio Rojas Uriarte	Desarrollador
23020867	Manuel Joaquín Chamorro	Desarrollador

1.2. Información general del proyecto

Título del proyecto	Sistema web para la gestión de clubes de Vida Estudiantil (UAM)
Tipo de Sistema	Aplicación web desarrollada en Java, usando el framework OpenXava e integrada con una base de datos relacional.
Breve descripción (máximo 150 palabras)	Actualmente la Dirección de Vida Estudiantil de la Universidad Americana (UAM) ofrece diversos clubes y voluntarios organizados en áreas ambientales, sociales, culturales y deportivas. La inscripción y seguimiento de actividades se realiza de forma manual (mediante llamadas, correo o mensajes al coordinador de cada club), lo que dificulta a los estudiantes conocer los eventos y mantener actualizada la lista de miembros. El proyecto propone un sistema web de gestión de clubes que permita a la dirección de Vida Estudiantil crear y administrar clubes, registrar socios, gestionar eventos y difundir anuncios de manera centralizada. El sistema busca facilitar la participación estudiantil, mejorar la comunicación y brindar reportes de asistencia y cuotas.

1.3. Definición del problema

Situación problemática

La UAM promueve la formación integral mediante clubes y voluntariados. En 2025 la Dirección de Vida Estudiantil impulsa diversos clubes y voluntariados creados desde 2021 para conectar los intereses de los estudiantes. Cada club cuenta con un coordinador, líderes estudiantiles y decenas de participantes. Sin embargo, **no existe una plataforma digital** unificada para:

- Registrar y controlar la membresía de estudiantes y voluntarios.
- Difundir los calendarios de eventos y confirmar la asistencia.
- Generar reportes de participación.

Para inscribirse a un club, los alumnos deben contactar de manera informal al responsable o a la Dirección de Vida Estudiantil. La información sobre reuniones o eventos se comparte por WhatsApp o redes sociales, lo que ocasiona demoras, listas desactualizadas y baja visibilidad de las actividades. Esta situación limita la capacidad de la UAM para dar seguimiento al liderazgo estudiantil y a los beneficios sociales de los clubes.

Justificación

La digitalización de la gestión de clubes es una necesidad inaplazable. Según estudios sobre clubs sociales, incorporar herramientas tecnológicas agiliza y optimiza los procesos internos, como el registro de miembros, control de pagos y organización de eventos. Las plataformas web permiten una **comunicación más ágil y personalizada** con los

socios mediante notificaciones y recordatorios. Además, ofrecen servicios en línea (inscripciones y reservas) y facilitan el análisis de datos para tomar decisiones estratégicas. En el contexto de la UAM, un sistema web facilitará la inscripción a los clubes, permitirá a los responsables programar eventos y llevar un control de asistencia y cuotas, y fortalecerá el sentido de comunidad fomentado por la Dirección de Vida Estudiantil. También reducirá el trabajo manual del personal y mejorará la transparencia al contar con reportes automatizados.

Objetivo General	<ul style="list-style-type: none"> • Diseñar e implementar un sistema web para la UAM que permita a la Dirección de Vida Estudiantil gestionar clubes de manera eficiente
Objetivos Específicos (mínimo 3)	<ul style="list-style-type: none"> • Desarrollar un módulo de gestión de clubes y miembros que permita crear clubes, registrar estudiantes y voluntarios, actualizar datos de contacto y roles, y generar listados de socios. • Implementar la gestión de eventos y asistencias: crear calendarios de actividades por club, enviar invitaciones, registrar asistencia, y emitir reportes de participación. • Desarrollar un sistema de comunicación y notificaciones que publique anuncios de eventos en un portal web y envíe recordatorios a los participantes (correo o notificaciones internas) para mejorar la difusión.

1.4. Alcance del sistema

Usuarios: Dirección de Vida Estudiantil (administradores), coordinadores de clubes y estudiantes/voluntarios.

Módulos principales:

1. **Administración de clubes y usuarios:** altas y bajas de clubes, asignación de coordinadores, registro de estudiantes y roles (miembro, líder, voluntario).
2. **Gestión de eventos:** creación de actividades (nombre, descripción, fecha, hora, lugar), registro de asistentes, control de aforo y seguimiento de asistencia.
3. **Comunicación y anuncios:** publicación de noticias y eventos en el portal, envío de notificaciones y recordatorios a los miembros del club.
4. **Reportes:** generación de reportes de membresía y asistencia para la Dirección de Vida Estudiantil.

Limitaciones:

- El sistema se centrará en la administración de clubes de la UAM y no incluirá gestión académica ni procesos de matrícula universitaria.
- Integraciones con pasarelas de pago externas se dejarán para futuras fases.

1.5. Actores y usuarios del sistema

Actor/Usuario	Descripción/Rol en el sistema
Administrador	Usuario de la Dirección de Vida Estudiantil con privilegios totales. Gestiona clubes, usuarios y reportes.
Coordinador de club	Responsable de un club específico. Crea actividades, administra inscripciones y confirma asistencias.
Estudiante / Miembro	Alumno que se inscribe en uno o varios clubes. Puede consultar eventos, inscribirse y ver su historial.
Líder o voluntario	Miembro con funciones de apoyo en la organización de actividades. Ayuda al coordinador en el registro de asistencias.
Invitado o externo	Persona externa (p. ej. egresados o participantes invitados) que puede registrarse en eventos públicos del club.

1.6. Requerimientos funcionales y no funcionales

1.6.1. Requerimientos funcionales:

- **RF-01 – Gestión de clubes:** el sistema permitirá a los administradores crear, editar y eliminar clubes, definir categorías (cultural, social, deportivo, ambiental) y asignar coordinadores.
- **RF-02 – Registro de miembros:** los estudiantes podrán registrarse en clubes mediante un formulario web; el sistema validará duplicados.
- **RF-03 – Gestión de eventos:** los coordinadores podrán crear eventos indicando nombre, descripción, fecha, hora, lugar y cupo máximo; los miembros se podrán inscribir.
- **RF-04 – Comunicación y anuncios:** el administrador y los coordinadores podrán publicar anuncios y noticias en un portal web y enviar notificaciones a los miembros mediante correo o mensajes internos.

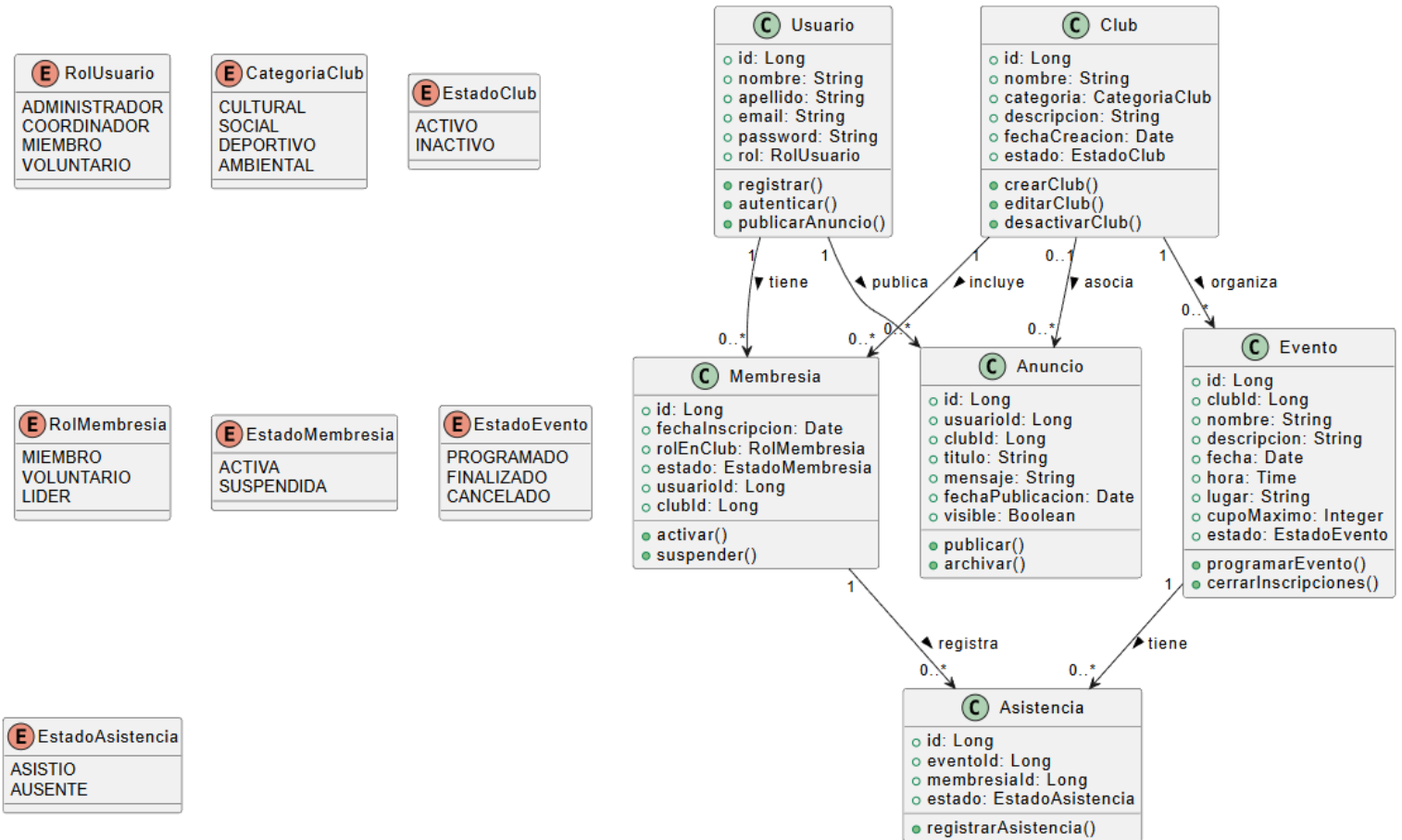
- **RF-05 – Reportes:** se generarán reportes de membresía, eventos y asistencias para uso de la Dirección de Vida Estudiantil.

1.6.2. Requerimientos no funcionales:

- **RNF-01 – Plataforma tecnológica:** la aplicación se desarrollará en Java utilizando **OpenXava** y **JPA**, con base de datos **PostgreSQL** para persistencia.
- **RNF-02 – Seguridad:** se implementará autenticación (usuario/contraseña) y autorización basada en roles; las contraseñas se almacenarán cifradas.
- **RNF-03 – Usabilidad:** la interfaz será intuitiva y responsive, diseñada en español, accesible desde navegadores modernos.
- **RNF-04 – Mantenibilidad:** se usará arquitectura MVC y buenas prácticas de desarrollo para facilitar futuras ampliaciones y mantenimiento.
- **RNF-05 – Integridad de datos:** se aplicarán restricciones y validaciones en la base de datos para evitar inconsistencias (p. ej. claves foráneas y restricciones de unicidad).

1.7. Modelo inicial del dominio

Adjunte o dibuje el diagrama preliminar de clases.



1.8. URL del repositorio

El código fuente se alojará en GitHub en el siguiente enlace:

https://github.com/johaneris/Proyecto_Final_POO.git

1.9. Cronograma estimado de desarrollo

Etapa	Actividad	Semana #
Análisis	Definición de requerimientos y modelo de clases	11
Diseño	Estructura de entidades y vistas en OpenXava	12
Desarrollo	Implementación de módulos CRUD	13, 14
Pruebas	Validación y corrección de errores	15
Presentación	Exposición final y documentación	16

1.10. Recursos necesarios

Indique herramientas, frameworks o librerías adicionales.

- **Lenguaje de programación:** Java
- **Framework:** OpenXava con JPA y mapeo a PostgreSQL.
- **Gestor de base de datos:** PostgreSQL (requiere instalación local o en servidor).
- **IDE:** IntelliJ IDEA Community.
- **Herramientas de control de versiones:** Git y acceso al repositorio en GitHub.

Bibliotecas adicionales:

- Spring Security (opcional) para mejorar la autenticación y autorización.
- JavaMail u otra librería para el envío de notificaciones por correo electrónico.