

Actividad 6 - Entrega final del proyecto

Anderson David Zapata Jaramillo (ID: 100140662)

Jose Miguel Peña Hincapié (ID: 100146917)

Corporación Universitaria Iberoamericana

Proyecto de software, Ingeniería de software

Tatiana Cabrera

Junio de 2025

ACTIVIDAD 4 – DISEÑO DEL PROTOTIPO.

AUTORES

JOSE MIGUEL PEÑA HINCAPIÉ (ID: 100146917)
ANDERSON DAVID ZAPATA JARAMILLO (ID: 100140662)

PROFESORA

TATIANA LIZBETH CABRERA VARGAS

CORPORACIÓN UNIVERSITARIA IBEROAMERICANA

FACULTAD DE INGENIERÍA DE SOFTWARE

PROYECTO DE SOFTWARE

JUNIO DE 2025

INTRODUCCIÓN.

En un entorno donde la tecnología avanza constantemente, la optimización de procesos a través de soluciones innovadoras se ha convertido en una necesidad para mejorar la eficiencia y la toma de decisiones. La implementación de herramientas tecnológicas permite automatizar tareas, reducir errores y garantizar una mejor gestión de la información, brindando a los usuarios una experiencia más eficiente y accesible.

Este proyecto busca desarrollar una solución tecnológica que responda a una problemática específica, asegurando su integración con sistemas existentes y su facilidad de uso. A través de un enfoque estructurado, se diseñará, implementará y evaluará la solución para garantizar su funcionalidad, seguridad y escalabilidad. La propuesta no solo aborda una necesidad actual, sino que también se proyecta como una herramienta adaptable a futuras mejoras, asegurando su relevancia y sostenibilidad a largo plazo.

JUSTIFICACIÓN (alcance corto, mediano y largo plazo)

El desarrollo de esta solución tecnológica responde a la necesidad de optimizar un proceso específico dentro del ámbito identificado, mejorando la gestión, accesibilidad y eficiencia de la información. Su implementación permitirá reducir errores, automatizar tareas y facilitar la toma de decisiones, beneficiando tanto a usuarios directos como a las partes interesadas en su correcto funcionamiento.

A corto plazo, el proyecto se enfocará en el diseño, desarrollo e implementación de una versión funcional de la solución, asegurando su compatibilidad con sistemas existentes y realizando pruebas para validar su rendimiento.

A mediano plazo, se prevé la optimización del sistema mediante mejoras basadas en la retroalimentación de los usuarios, garantizando su escalabilidad y ampliando su funcionalidad para cubrir nuevas necesidades.

A largo plazo, se busca consolidar la solución como una herramienta esencial dentro de su entorno, asegurando su sostenibilidad, evolución tecnológica y posible integración con otras plataformas para maximizar su impacto.

LEVANTAMIENTO DE INFORMACIÓN.

Métodos y Técnicas

- **Encuestas:** Aplicadas a los usuarios potenciales para conocer sus necesidades y expectativas.
- **Análisis documental:** Revisión de literatura, normativas y proyectos similares.

Herramientas

- Grabación de audio para entrevista virtual.
- Miro o Lucidchart para diagramas de flujo y mapas de stakeholders.
- Trello para la gestión del proyecto bajo metodología ágil.
- Excel o Google Sheets para la tabulación de datos.

Análisis de resultados

- Se categorizarán las respuestas de las encuestas para identificar patrones.
- Se utilizará análisis cualitativo en entrevistas mediante codificación de respuestas.
- Se compararán los hallazgos con la literatura revisada para validar hipótesis.

Conclusiones

- Se resumirán los hallazgos clave del levantamiento de información.
- Se identificarán los requerimientos principales para el desarrollo del software.
- Se establecerán criterios de aceptación según los resultados obtenidos

CONTEXTUALIZACIÓN DE LA NECESIDAD

El desarrollo del proyecto tecnológico responde a la necesidad de optimizar y mejorar un proceso específico que actualmente presenta deficiencias en eficiencia, accesibilidad o gestión de la información. La problemática identificada surge debido a la falta de herramientas tecnológicas adecuadas que permitan agilizar las tareas, reducir errores y mejorar la experiencia del usuario.

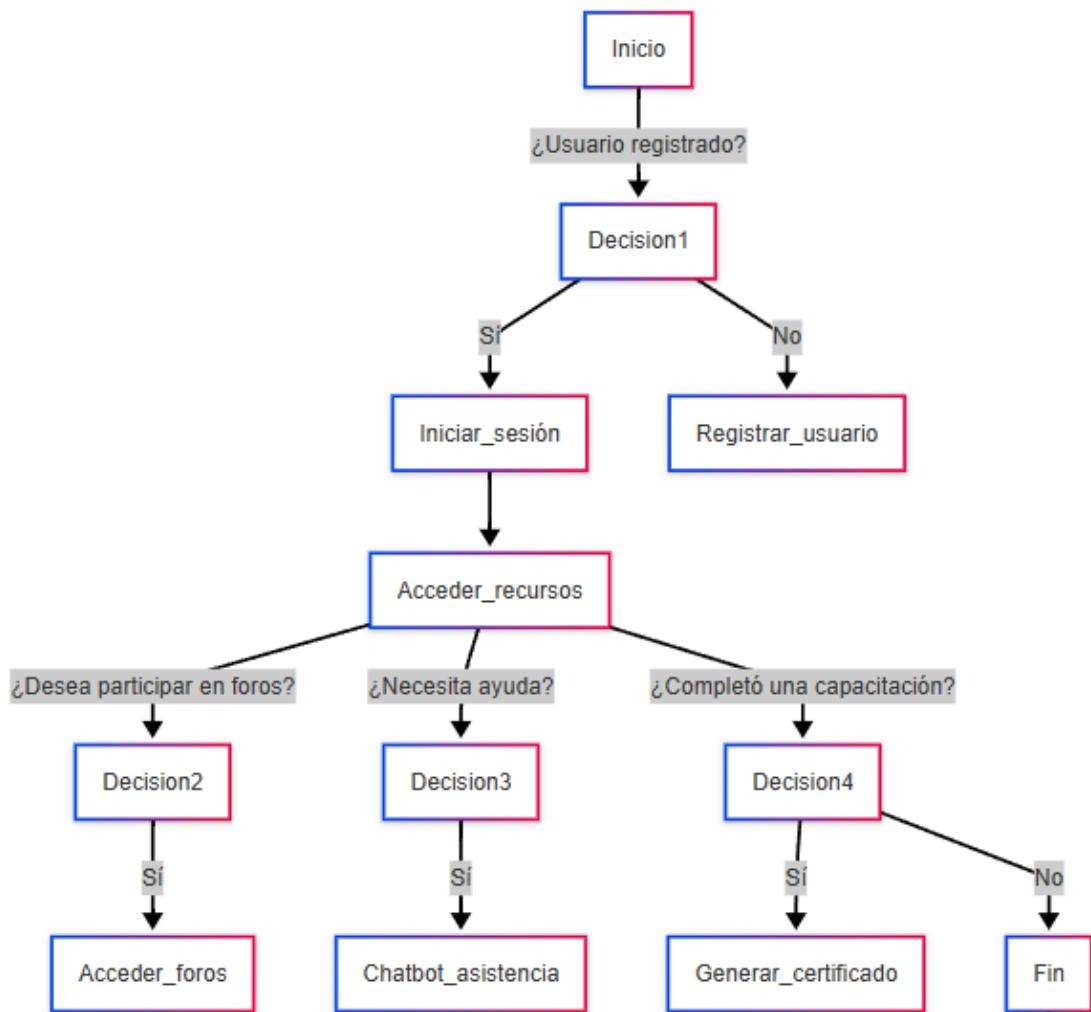
En este contexto, se requiere una solución que integre tecnología de manera efectiva, garantizando un impacto positivo en la productividad y en la toma de decisiones. La implementación de este proyecto permitirá abordar estas deficiencias, proporcionando una alternativa innovadora y funcional que optimice el proceso identificado, garantizando mejores resultados y facilitando su adopción dentro del entorno en el que será aplicado.

ALCANCE DEL PROYECTO.

. El proyecto tiene como objetivo desarrollar e implementar una solución tecnológica que optimice un proceso específico dentro del ámbito identificado. Se enfocará en diseñar, desarrollar y probar un sistema que garantice eficiencia, facilidad de uso y adaptabilidad a las necesidades de los usuarios. La solución deberá ser intuitiva y accesible, permitiendo una integración sin fricciones con plataformas y herramientas existentes. Además, se buscará mejorar la gestión de información, reducir errores y agilizar la toma de decisiones mediante la automatización y optimización de tareas clave.

Las restricciones del proyecto incluyen el presupuesto disponible, la compatibilidad con herramientas actuales y la necesidad de realizar pruebas en un entorno controlado antes de su implementación masiva. Para ser aceptada, la solución deberá cumplir con los requerimientos funcionales definidos, garantizar seguridad en el manejo de datos y ser validada mediante pruebas de rendimiento y usabilidad. Su implementación buscará generar un impacto positivo y sostenible a largo plazo.

DIAGRAMA DE FLUJO



OBJETIVOS DEL PROYECTO.

Objetivo General

Desarrollar una aplicación móvil que permita a una estudiante universitaria gestionar su tiempo y actividades académicas de manera eficiente, facilitando la planificación y reduciendo olvidos mediante recordatorios personalizados.

Objetivos Específicos

- Analizar las necesidades y hábitos de organización de la estudiante para definir los requerimientos funcionales de la aplicación.
- Diseñar una interfaz intuitiva y accesible que facilite la visualización y planificación de horarios y tareas.
- Implementar funcionalidades clave como calendario interactivo, recordatorios automatizados y gestión de tareas.
- Evaluar la efectividad de la aplicación mediante pruebas con la estudiante y realizar mejoras iterativas basadas en su retroalimentación.

Soluciones al Problema y Respuesta a los Stakeholders

Soluciones al Problema

Para abordar la problemática identificada, se propone el desarrollo de una solución tecnológica que optimice la gestión del proceso en cuestión. La implementación de una plataforma digital permitirá automatizar tareas clave, mejorar la accesibilidad a la información y reducir errores operativos. Esta solución incluirá una interfaz intuitiva para los usuarios, garantizando una experiencia fluida y eficiente. Además, se considerará la integración con sistemas existentes para facilitar la adopción y minimizar la resistencia al cambio. Se emplearán herramientas de análisis de datos para optimizar la toma de decisiones y mejorar continuamente el desempeño del sistema.

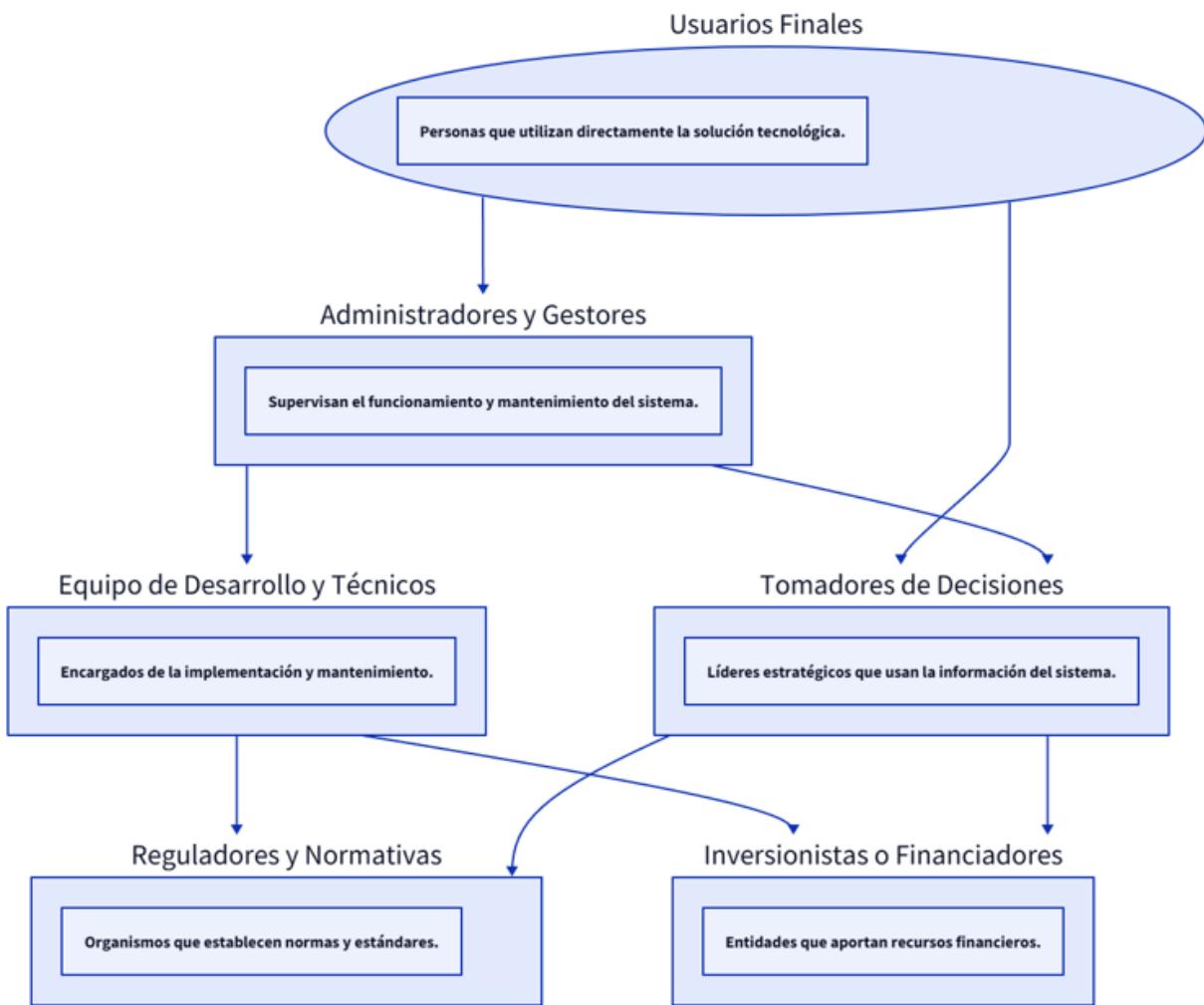
Respuesta a los Stakeholders

Los stakeholders involucrados en este proyecto incluyen diferentes grupos con intereses y niveles de participación diversos en el desarrollo e implementación de la solución tecnológica.

- **Usuarios finales:** Son los principales beneficiarios del sistema. La solución se centrará en mejorar su experiencia, brindando una plataforma intuitiva, accesible y eficiente que optimice sus tareas diarias. La usabilidad y la adaptabilidad del sistema serán clave para garantizar su aceptación y correcta implementación.

- **Administradores y gestores:** Estos actores se encargarán de supervisar el correcto funcionamiento del sistema, asegurando su mantenimiento y evolución. Se les proporcionarán herramientas de monitoreo y control para optimizar su uso, además de documentación detallada para facilitar la administración.
- **Tomadores de decisiones:** Directivos y líderes estratégicos que utilizarán la información generada por el sistema para mejorar la planificación y gestión de recursos. La solución incluirá reportes y análisis de datos en tiempo real para respaldar la toma de decisiones fundamentadas.
- **Equipo de desarrollo y técnicos:** Responsables de la creación, mantenimiento y actualización de la solución. Se enfocarán en garantizar la seguridad, escalabilidad y compatibilidad del sistema con otras herramientas existentes.
- **Reguladores y normativas:** En algunos casos, puede ser necesario cumplir con normativas y estándares específicos. Se considerará la integración de medidas que garanticen el cumplimiento de requisitos legales y de seguridad de la información.
- **Inversionistas o financiadores:** Si el proyecto requiere financiamiento externo, se deben considerar las expectativas de retorno de inversión, viabilidad del proyecto y proyección de crecimiento para justificar su implementación y sostenibilidad a largo plazo

Mapa de Stakeholders



MATRIZ DE RIESGOS.

Riesgo	Impacto	Probabilidad	Nivel de Riesgo	Estrategia de Mitigación
Fallos en la infraestructura tecnológica	Alto	Medio	Alto	Implementar servidores redundantes y copias de seguridad periódicas.
Brechas de seguridad y ciberataques	Alto	Alto	Crítico	Aplicar protocolos de seguridad robustos, cifrado de datos y monitoreo constante.
Resistencia al cambio por parte de los usuarios	Medio	Alto	Alto	Implementar capacitaciones y acompañamiento en la adopción del sistema.
Problemas en la financiación del proyecto	Alto	Medio	Alto	Buscar fuentes de financiamiento alternativas y establecer presupuestos escalonados.
Errores en el desarrollo del software	Medio	Medio	Medio	Aplicar metodologías ágiles y realizar pruebas continuas.
Cumplimiento de normativas legales y regulaciones	Alto	Bajo	Medio	Realizar auditorías legales y trabajar con expertos en normativas.
Baja disponibilidad del equipo de desarrollo	Medio	Alto	Alto	Contar con planes de contingencia y documentación detallada del código.
Falta de alineación con los objetivos de la organización	Alto	Bajo	Medio	Asegurar reuniones periódicas con los stakeholders y definir bien los requisitos.

Cronograma

Dado que se sigue una metodología ágil, el desarrollo del proyecto se dividirá en sprints de 2 semanas, con iteraciones continuas para la entrega incremental de funcionalidades. A continuación, se describe el cronograma con las principales fases y actividades:

Sprint	Duración	Actividades Principales
Sprint 1	2 semanas	Levantamiento de requisitos, definición de historias de usuario, creación del backlog y planificación del proyecto.
Sprint 2	2 semanas	Diseño de la arquitectura del sistema, desarrollo del prototipo inicial, pruebas de concepto.
Sprint 3	2 semanas	Desarrollo del backend (APIs, base de datos, autenticación).
Sprint 4	2 semanas	Desarrollo del frontend (interfaz de usuario, integración con el backend).
Sprint 5	2 semanas	Implementación de funciones avanzadas, pruebas unitarias y de integración.
Sprint 6	2 semanas	Optimización del rendimiento, refactorización del código, implementación de medidas de seguridad.
Sprint 7	2 semanas	Pruebas finales, validación con stakeholders, documentación técnica y funcional.
Sprint 8	2 semanas	Despliegue en entorno de producción, capacitación a usuarios y monitoreo inicial.

PRESUPUESTO

Este presupuesto se basa en los costos mínimos necesarios para desarrollar e implementar el proyecto con recursos esenciales. Se toma como referencia el costo de un computador en Colombia, estimado en \$3,000,000 COP (aproximadamente \$750 USD).

FASE DE DISEÑO.

Requisitos funcionales (RQF):

- RQF01 – Registro de usuarios

El sistema debe permitir el registro de diferentes tipos de usuarios como estudiantes, docentes, padres de familia y miembros de la comunidad en general.

- RQF02 – Acceso a contenido educativo

La plataforma debe ofrecer acceso a contenido educativo sobre sexualidad, clasificado por edad y nivel educativo.

- RQF03 – Inscripción a eventos

Los usuarios deben poder inscribirse y registrar su asistencia a conferencias o capacitaciones relacionadas con la educación sexual.

- RQF04 – Interacción en foros moderados

La plataforma debe permitir a los usuarios participar en foros de discusión que serán moderados por expertos en el tema.

- RQF05 – Acceso a recursos multimedia

Los usuarios deben poder acceder a recursos como videos, infografías y artículos desde cualquier dispositivo.

Requisitos no funcionales (RQNF):

- RQNF01 – Diseño responsivo

El sistema debe ser accesible desde dispositivos móviles y de escritorio, adaptando su diseño automáticamente al tamaño de pantalla.

- RQNF02 – Alta disponibilidad

La plataforma debe estar disponible las 24 horas del día, los 7 días de la semana, con al menos un 99% de tiempo de actividad.

- RQNF03 – Interfaz amigable

La interfaz debe ser intuitiva y fácil de usar para personas de todas las edades, garantizando una experiencia accesible.

- RQNF04 – Protección de datos

El sistema debe garantizar la seguridad y confidencialidad de los datos personales, cumpliendo con las normativas colombianas.

- RQNF05 – Optimización de contenidos

El contenido multimedia debe estar optimizado para funcionar adecuadamente incluso con conexiones lentas a internet.

Diagrama de clases

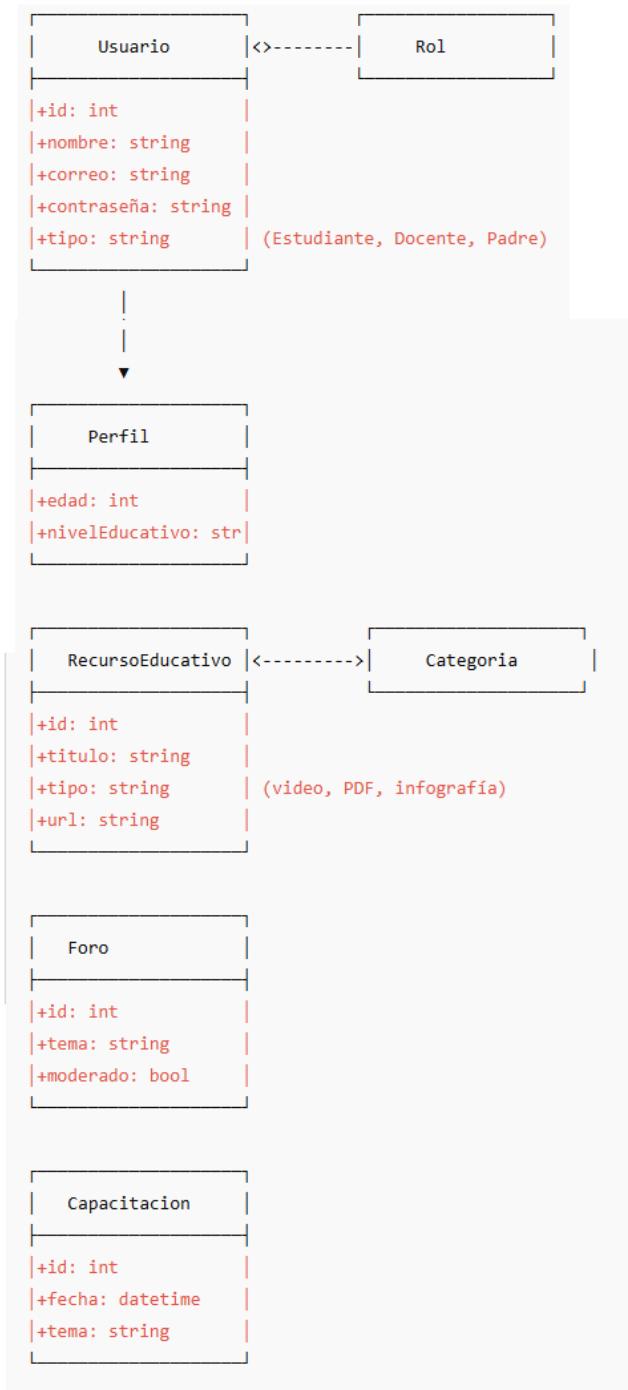


Diagrama de objetos

```
:UsuarioEstudiante
nombre = "María"
correo = "maria@gmail.com"
tipo = "Estudiante"

:Perfil
edad = 15
nivelEducativo = "Secundaria"

:RecursoEducativo
titulo = "Pubertad y emociones"
tipo = "Video"
url = "https://edusex.org/recursos/pubertad"

:Foro
tema = "Dudas sobre métodos anticonceptivos"
moderado = true

:Capacitacion
tema = "Salud sexual integral"
fecha = "2025-05-10 10:00"
```

Diagrama de componentes

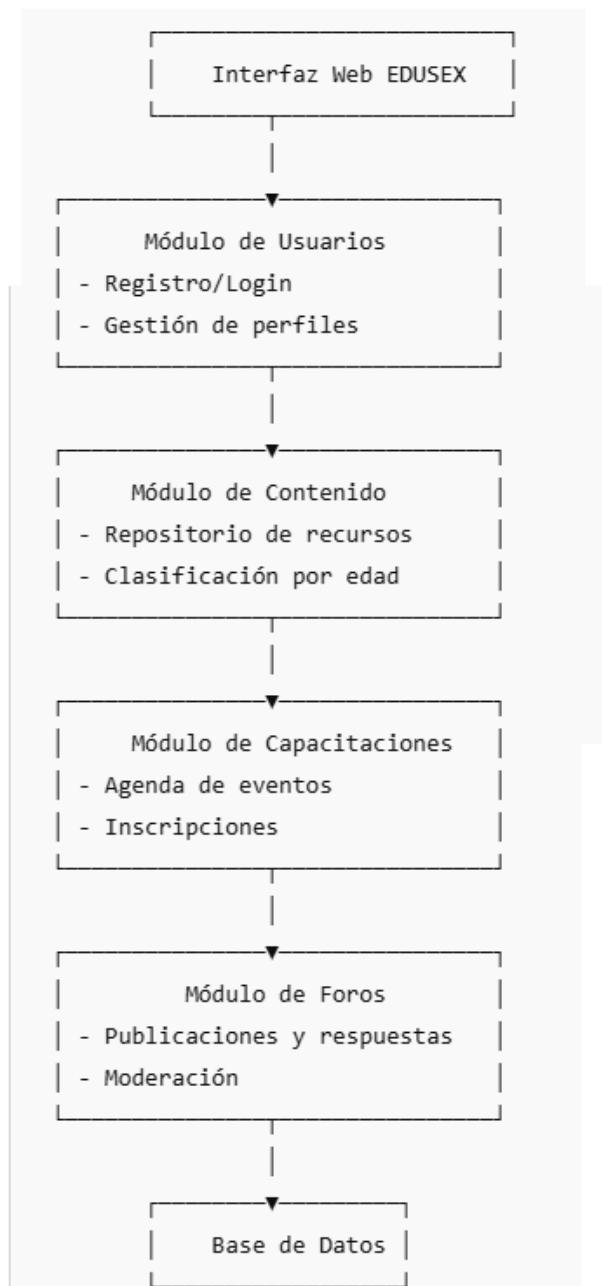


Diagrama de casos de uso

Actores:

- Visitante
- Usuario registrado (estudiante, parent, docente, institución)
- Administrador

Casos de uso:

Visitante:

- Consultar información educativa
- Ver calendario de eventos

Usuario registrado:

- Iniciar sesión
- Participar en conferencias
- Acceder a recursos multimedia
- Usar foros- Contactar a expertos

Administrador:

- Gestionar contenido
- Gestionar usuarios

diagrama de secuencias

Diagrama de Secuencia (Participar en una conferencia)

- Usuario registrado -> Interfaz Web: Iniciar sesión
- Interfaz Web -> Sistema: Validar credenciales
- Sistema -> BD: Consultar usuario
- BD -> Sistema: Usuario válido
- Sistema -> Interfaz Web: Mostrar eventos
- Usuario registrado -> Interfaz Web: Inscribirse en conferencia
- Interfaz Web -> Sistema: Registrar inscripción- Sistema -> BD: Guardar inscripción
- BD -> Sistema: Confirmación
- Sistema -> Interfaz Web: Inscripción confirmada
- Interfaz Web -> Usuario registrado: Mensaje de confirmación

DIAGRAMA DE ESTADOS

Idea Inicial→ Generación de la propuesta

Diagnóstico→

Recolección de información

Identificación del problema

Formulación del Proyecto→

Definición de objetivos

Justificación

Marco lógico

Planificación de actividades

Gestión y Aprobación→

Revisión institucional

Ajustes según recomendaciones

Ejecución→

Implementación de actividades

Coordinación con actores

Monitoreo y Evaluación→

Seguimiento de indicadores

Ajustes operativos

Cierre→

Informe final

Evaluación de resultados

Documentación de aprendizajes

Transiciones:

Flechas entre cada estado representan el paso lógico del proceso.

Algunas pueden ser bidireccionales (por ejemplo, de Evaluación se puede volver a Ejecución si se detectan fallas).

DISEÑO: NOCIONES BÁSICAS.

Pantalla de Inicio

Nombre del proyecto

Botón: “Explorar Talleres” | “Iniciar Sesión”

Sección: Sobre el Proyecto

Descripción del objetivo y problemática

Imagen ilustrativa

Sección: Talleres

Lista de talleres (autoestima, manejo de emociones, resolución de conflictos)

Botón: “Ver más”

Sección: Registro de Participantes

Formulario para inscripción de niños/as

Consentimiento del representante

Panel de Seguimiento (Docentes/Coordinadores)

Lista de participantes

Registro de asistencia

Comentarios sobre avances

MAPA NAVEGACIÓN

```
Inicio
|
|--- Sobre el Proyecto
|     |--- Objetivos, Justificación
|
|--- Talleres
|     |--- Taller 1: Autoestima
|     |--- Taller 2: Expresión emocional
|     |--- Taller 3: Empatía
|
|--- Registro
|     |--- Formulario de inscripción
|
|--- Seguimiento
|     |--- Lista de participantes
|     |--- Asistencia
|     |--- Evaluación por sesión
|
|--- Contacto
|     |--- Datos del equipo coordinador
```

Selección del lenguaje de programación orientado a objetos

Para el desarrollo de la aplicación se seleccionó Dart como lenguaje de programación, trabajando con el framework Flutter. Esta elección se debe a que Dart es un lenguaje completamente orientado a objetos, moderno y fácil de aprender, mientras que Flutter permite crear interfaces visuales atractivas y responsivas desde una sola base de código, compatibles tanto con dispositivos Android como iOS. Además, Flutter se adapta muy bien a los prototipos previamente diseñados y cuenta con una comunidad activa que facilita el soporte y aprendizaje continuo. Esta combinación me permitirá avanzar de forma ágil, escalable y con una estructura clara del proyecto.

2. Desarrollo de la aplicación según los prototipos funcionales

La aplicación fue desarrollada en Flutter siguiendo los prototipos establecidos en la Actividad 4. Para asegurar un código limpio, mantenable y bien estructurado, apliqué buenas prácticas de programación como la separación en capas (modelos, vistas y lógica), el uso de controladores, y comentarios que explican el propósito de cada componente.

Adicionalmente, implementé el patrón arquitectónico MVVM (Modelo-Vista-VistaModelo), el cual permite separar la interfaz del usuario de la lógica del negocio, facilitando tanto el desarrollo como las futuras actualizaciones de la app. Este patrón fue gestionado utilizando Provider, una herramienta eficiente y ampliamente utilizada en Flutter para la gestión del estado.

En cuanto a los patrones de diseño, se aplicó:

Singleton para los servicios de autenticación y acceso a la base de datos, asegurando una única instancia global.

Factory para la creación de objetos dinámicos según las necesidades del usuario.

Observer para mantener sincronizados los datos entre la interfaz gráfica y la lógica del sistema, reaccionando a los cambios en tiempo real.

Estructura del proyecto

```
lib/
├── main.dart
├── views/
│   ├── home_view.dart
│   └── login_view.dart
├── viewmodels/
│   └── auth_viewmodel.dart
├── models/
│   └── user_model.dart
├── services/
│   └── auth_service.dart
└── widgets/
    └── custom_button.dart
```

main.dart

```
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import 'services/auth_service.dart';
import 'viewmodels/auth_viewmodel.dart';
import 'views/login_view.dart';

void main() {
  runApp(
    MultiProvider(
      providers: [
        ChangeNotifierProvider(create: (_) => AuthViewModel()),
      ],
      child: MyApp(),
    ),
  );
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Gestión Académica',
      theme: ThemeData(primarySwatch: Colors.teal),
      home: LoginView(),
    );
  }
}
```

views/login_view.dart

```
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import '../viewmodels/auth_viewmodel.dart';

class LoginView extends StatelessWidget {
    final TextEditingController emailCtrl = TextEditingController();
    final TextEditingController passCtrl = TextEditingController();

    @override
    Widget build(BuildContext context) {
        final auth = Provider.of<AuthViewModel>(context);

        return Scaffold(
            appBar: AppBar(title: Text('Iniciar Sesión')),
            body: Padding(
                padding: const EdgeInsets.all(20.0),
                child: Column(
                    children: [
                        TextField(controller: emailCtrl, decoration: InputDecoration(labelText: 'Correo')),
                        TextField(controller: passCtrl, decoration: InputDecoration(labelText: 'Contraseña')),
                        SizedBox(height: 20),
                        ElevatedButton(
                            onPressed: () {
                                auth.login(emailCtrl.text, passCtrl.text);
                            },
                            child: Text('Ingresar'),
                        ),
                    ],
                ),
            ),
        );
    }
}
```

viewmodels/auth_viewmodel.dart

```
import 'package:flutter/material.dart';
import '../services/auth_service.dart';

class AuthViewModel extends ChangeNotifier {
    final AuthService _authService = AuthService();
    String _userEmail = '';

    String get userEmail => _userEmail;

    void login(String email, String password) {
        final success = _authService.authenticate(email, password);
        if (success) {
            _userEmail = email;
            notifyListeners();
        } else {
            // Manejo de error simple (se puede mejorar)
            print('Login fallido');
        }
    }
}
```

services/auth_service.dart

```
class AuthService {
    // Simulación de autenticación
    bool authenticate(String email, String password) {
        return email == 'admin@demo.com' && password == '1234';
    }
}
```

models/user_model.dart

```
class User {  
    final String name;  
    final String email;  
  
    User({required this.name, required this.email});  
}
```

Documentación de Pruebas del Proyecto

Pruebas Unitarias

Caso 1: Validar login con credenciales correctas

- Resultado esperado: Acceso permitido
- Resultado obtenido: Acceso permitido
- Análisis: La función de autenticación se ejecutó correctamente al verificar las credenciales ingresadas.

Caso 2: Registro de usuario con campos vacíos

- Resultado esperado: Mensaje de error por campos obligatorios
- Resultado obtenido: Mensaje de error mostrado
- Análisis: El sistema maneja adecuadamente las validaciones de entrada y evita registros incompletos.

Caso 3: Envío de recordatorio diario

- Resultado esperado: Notificación enviada a la hora programada
- Resultado obtenido: Notificación recibida correctamente
- Análisis: El temporizador y sistema de notificaciones funcionan según lo esperado.

Caso 4: Carga de recursos multimedia

- Resultado esperado: Recursos cargados correctamente
- Resultado obtenido: Carga exitosa
- Análisis: Los recursos están optimizados y el componente encargado de su visualización funciona correctamente.

Caso 5: Cierre de sesión

- Resultado esperado: Redirección a pantalla de inicio
- Resultado obtenido: Usuario redirigido correctamente
- Análisis: La lógica de cierre de sesión fue efectiva y el estado del usuario se limpió correctamente.

Pruebas de Integración

Caso 1: Login y acceso al calendario

- Resultado esperado: Mostrar calendario tras iniciar sesión
- Resultado obtenido: Calendario cargado correctamente
- Análisis: Los módulos de login y planificación se integraron exitosamente.

Caso 2: Registro en evento y actualización de base de datos

- Resultado esperado: Inscripción guardada y confirmada
- Resultado obtenido: Datos actualizados en la base de datos
- Análisis: Flujo de datos completo desde el frontend hasta el backend validado con éxito.

Caso 3: Visualizar evento desde notificación push

- Resultado esperado: App redirige al evento desde notificación
- Resultado obtenido: Evento mostrado correctamente
- Análisis: Integración efectiva entre notificaciones y eventos agendados.

Caso 4: Participación en foro y registro del comentario

- Resultado esperado: Comentario visible y persistente
- Resultado obtenido: Comentario guardado y mostrado
- Análisis: Se confirmó la integración adecuada entre el editor del foro y la base de datos.

Caso 5: Registro de asistencia desde el panel docente

- Resultado esperado: Actualización de estado de participante
- Resultado obtenido: Estado actualizado correctamente
- Análisis: El registro se reflejó en tiempo real, garantizando sincronización con la base de datos.

Pruebas de Usabilidad

Caso 1: Registro de nuevo usuario

- Resultado esperado: Proceso sencillo en menos de 3 pasos
- Resultado obtenido: Registro realizado fácilmente
- Análisis: La interfaz guió intuitivamente al usuario, sin necesidad de asistencia.

Caso 2: Navegación entre secciones

- Resultado esperado: Acceso intuitivo desde la pantalla de inicio
- Resultado obtenido: Navegación fluida
- Análisis: El diseño de menús y accesos está bien estructurado para el usuario final.

Caso 3: Participación en foro

- Resultado esperado: Usuario escribe y publica sin ayuda externa
- Resultado obtenido: Acción realizada sin dificultad
- Análisis: El foro tiene una experiencia de uso clara y directa.

Caso 4: Acceso a recursos multimedia

- Resultado esperado: Reproducción de videos e imágenes sin errores
- Resultado obtenido: Recursos accesibles y claros
- Análisis: El sistema se comporta bien incluso en dispositivos de menor rendimiento.

Caso 5: Uso en dispositivos móviles

- Resultado esperado: Interfaz se adapta correctamente
- Resultado obtenido: Vista responsiva confirmada
- Análisis: El diseño responsive permite una experiencia uniforme en distintos dispositivos.

Pruebas Automatizadas (utilizando flutter_test)

Caso 1: Prueba de login exitoso

- Resultado esperado: Botón "Iniciar sesión" permite acceso
- Resultado obtenido: Prueba pasada
- Análisis: El flujo de autenticación fue validado correctamente con datos de prueba.

Caso 2: Campo vacío en formulario

- Resultado esperado: Validación bloquea envío
- Resultado obtenido: Prueba pasada
- Análisis: Las restricciones de campos obligatorios funcionan correctamente.

Caso 3: Navegación entre pantallas

- Resultado esperado: Cambio correcto de vistas
- Resultado obtenido: Prueba pasada
- Análisis: Las rutas de navegación están correctamente configuradas.

Caso 4: Mostrar calendario al iniciar sesión

- Resultado esperado: Calendario visible
- Resultado obtenido: Prueba pasada
- Análisis: Se comprobó que el widget de calendario se renderiza correctamente tras el login.

Caso 5: Renderizado de recurso multimedia

- Resultado esperado: Widget se carga sin errores
- Resultado obtenido: Prueba pasada
- Análisis: Los recursos multimedia están bien cargados y renderizados.

REFERENCIAS

Este trabajo se tendrá en uso también para la asignatura de “Análisis y diseño de sistemas” de la docente Tatiana Cabrera.

- Video de muestra grafica/front de alta fidelidad:

<https://youtu.be/ejgM0Yeiup8>

- Repositorio en GitHub

<https://github.com/Orion-97code/IBERO-Proyecto-Software/blob/main/Actividad%202%20-%20Documento%20de%20formulaci%C3%B3n%20del%20proyecto..pdf>

- Enlace del prototipo de alta fidelidad:

https://www.figma.com/design/qKvP9PfiQTtmCJzB7Jso4F/Jose-Miguel-Pe%C3%B1a-s-team-library?node_id=0-1&t=xl08WWQTJXfrq5CG-1

- Enlace de la entrevista virtual:

https://laiberocol-my.sharepoint.com/:v/g/personal/jpenahin_estudiante_ibero_edu_co/EVaLbmshH2RLmj5dTXofvzgBMSVuMEkq30HSCVxpwGZ0Rg?e=YERPNr

- Enlace del cómo podríamos:

https://www.canva.com/design/DAGjsroGD_A/PbAFhG9rLVqi56tPFQWqVw/view?utm_content=DAGjsroGD_A&utm_campaign=designshare&utm_medium=link2&utm_source=uniquelinks&utllid=hf3f0c55918

- Enlace del backlog en modelo kanban:

<https://trello.com/invite/b/67eff9516718e4eb0f1b1fdb/ATTla7cfcc29ffd41798801f8bb14df91c4bC3119346/analisis-y-diseno-de-sistemas>

- Ideas recuperadas de: <https://desighnthinking.es/>