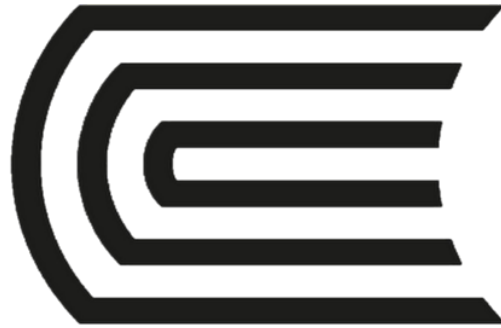


“Año de la recuperación y consolidación de la economía peruana”



ANÁLISIS Y DISEÑO DE SOFTWARE

SEMANA 16:

Docente: Rosario Delia Osorio Contreras

integrantes:

- ☐ Asto Condori Walter Hugo
- ☐ Vargas Flores Fabio Miguel
- ☐ Rojas Quincho Marco Antonio
- ☐ Machicao Huamán Sandro Luis

NRC:62152

Huancayo-Peru

2025-09

Índice

Unidad I – Fundamentos y Modelado Inicial

1. Presentación del Proyecto.....
2. Análisis de Necesidades y Requerimientos
3. Modelos Iniciales del Sistema

Unidad II – Modelos de Diseño y Metodología Ágil

4. Modelos de Diseño
5. Metodología de Trabajo (SCRUM)

Conclusiones y Recomendaciones

- Conclusiones del equipo
- Lecciones aprendidas
- Recomendaciones para futuras mejoras del sistema

Referencias bibliográficas

Anexos

Capítulo 1. Presentación del Proyecto

Este proyecto se encuentra con la ODS 6: Agua limpia y saneamiento, cuyo propósito es asegurar la disponibilidad y el uso sostenible del agua, además de garantizar el acceso equitativo a este recurso esencial. La incorporación de herramientas tecnológicas en los procesos educativos permite sensibilizar a la población sobre la importancia de este objetivo, incentivando cambios de hábitos desde edades tempranas.

Organización o institución beneficiaria

La aplicación está orientada principalmente a instituciones educativas de nivel secundario, tanto del sector público como privado. Dichas entidades desempeñan un papel fundamental en la formación de los adolescentes, quienes podrán adquirir, mediante el uso de la app, conocimientos y prácticas relacionadas con el consumo responsable del agua.

De manera indirecta, también se verán favorecidas las familias y la comunidad en general, ya que los aprendizajes obtenidos en el entorno escolar repercuten en los hábitos diarios del hogar.

Problema identificado

El uso inadecuado y el desperdicio de agua constituyen un problema significativo a nivel local y mundial. A pesar de las políticas y programas desarrollados, aún persiste una marcada falta de conciencia ambiental, en especial en los jóvenes, quienes son una generación clave para generar cambios.

Entre las principales dificultades encontradas se destacan:

- Deficiencia en la información sobre el impacto del consumo excesivo de agua.
- Escasez de materiales didácticos modernos y atractivos que motiven el aprendizaje.
- Desvinculación entre la enseñanza teórica y la aplicación práctica en la vida cotidiana.

Solución propuesta

La iniciativa consiste en el diseño de una Aplicación Educativa para dispositivos móviles que promueva el ahorro y uso responsable del agua, empleando estrategias de gamificación y aprendizaje interactivo.

Elementos centrales de la propuesta:

- **Espacio educativo interactivo:** con recursos multimedia como videos, infografías y módulos cortos de aprendizaje.
- **Retos y dinámicas gamificadas:** desafíos semanales relacionados con prácticas cotidianas de ahorro (por ejemplo, reducir el tiempo en la ducha o reutilizar agua en tareas domésticas).
- **Sistema de incentivos:** medallas y niveles que estimulen la participación y refuercen la motivación de los estudiantes.
- **Herramienta de registro y control:** que permita al usuario monitorear sus acciones vinculadas al ahorro de agua.
- **Reportes y estadísticas:** que presenten resultados individuales y colectivos dentro de cada institución educativa.

Valor innovador

La solución destaca por su enfoque innovador, ya que combina el uso de tecnologías móviles con la educación ambiental, favoreciendo el aprendizaje práctico.

Asimismo, aplica la gamificación como estrategia pedagógica para transformar hábitos de manera lúdica y sostenible. Finalmente, fomenta la colaboración comunitaria, al ofrecer la posibilidad de que las instituciones educativas comparen y compartan logros colectivos.

Capítulo 2. Análisis de Necesidades y Requerimientos

Descripción del problema:

El desperdicio de agua es un problema global agravado por la falta de información y herramientas para gestionar el consumo de forma responsable.

Necesidades de los usuarios:

- Medir y entender su consumo de agua.
- Recibir recomendaciones prácticas para ahorrar.
- Acceder a contenido educativo sobre el agua.
- Comparar su avance con otros usuarios.

Requerimientos funcionales (RF):

- RF1: Registro de consumo diario de agua.
- RF2: Visualización de estadísticas y progreso.
- RF3: Sistema de notificaciones con tips diarios.
- RF4: Retos comunitarios y logros.
- RF5: Perfil de usuario personalizable.

Requerimientos no funcionales (RNF):

- RNF1: Interfaz intuitiva y accesible.
- RNF2: Tiempo de respuesta menor a 2 segundos.
- RNF3: Compatibilidad con Android e iOS.
- RNF4: Seguridad de datos del usuario.

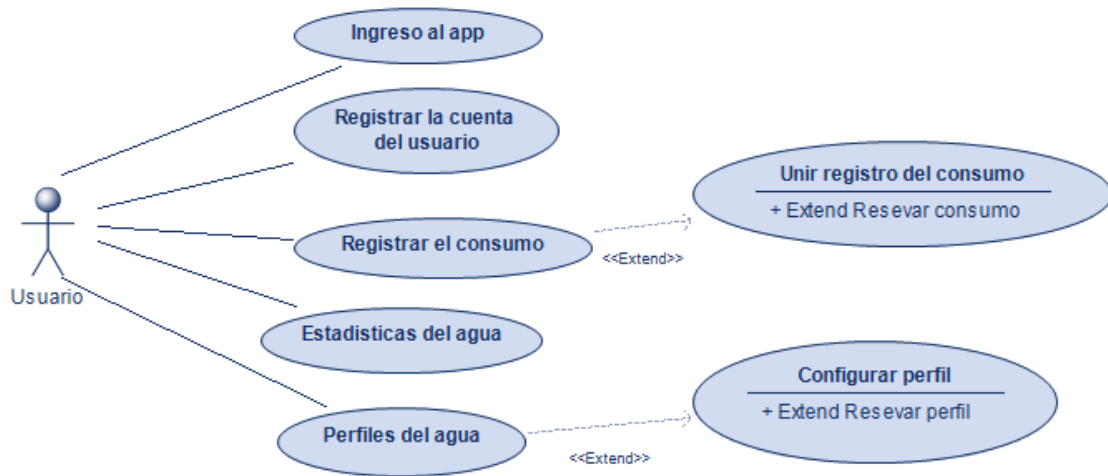
Requerimientos de dominio:

- Uso de estándares internacionales de consumo de agua (OMS).
- Integración con sensores IoT (futura versión).

Capítulo 3. Modelos Iniciales del Sistema

Modelo funcional:

- Diagrama de contexto: interacción entre usuario, app y base de datos.
- Casos de uso: Registrar consumo, ver estadísticas, configurar perfil, unirse a retos.



Modelo de procesos:

Flujo: Registro → Ingreso de datos → Análisis → Recomendación → Seguimiento.

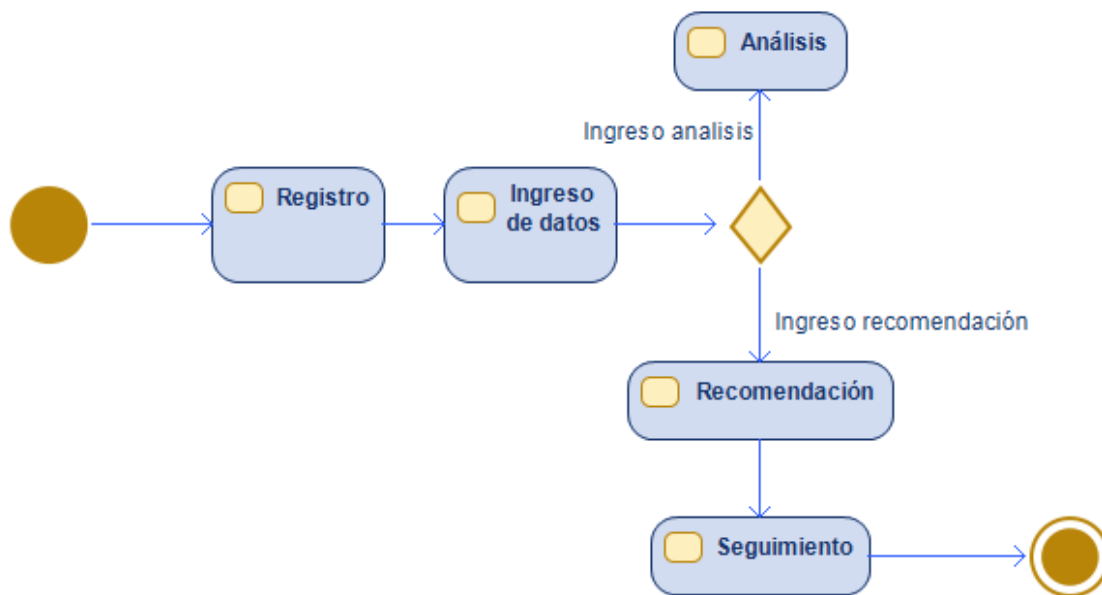
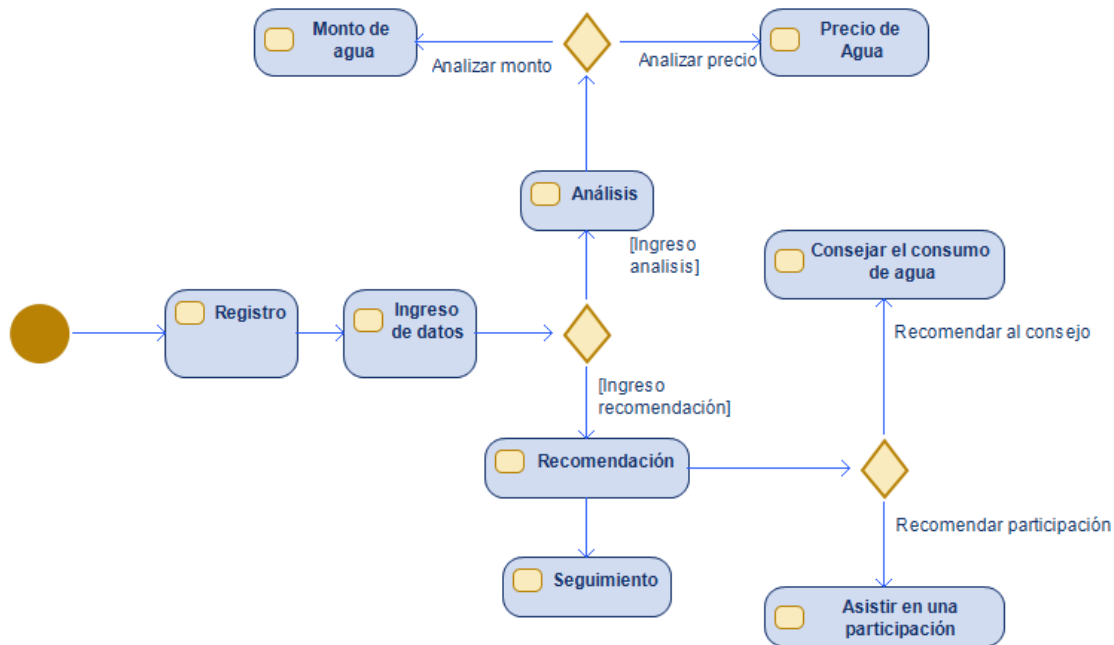


Diagrama de actividad UML:

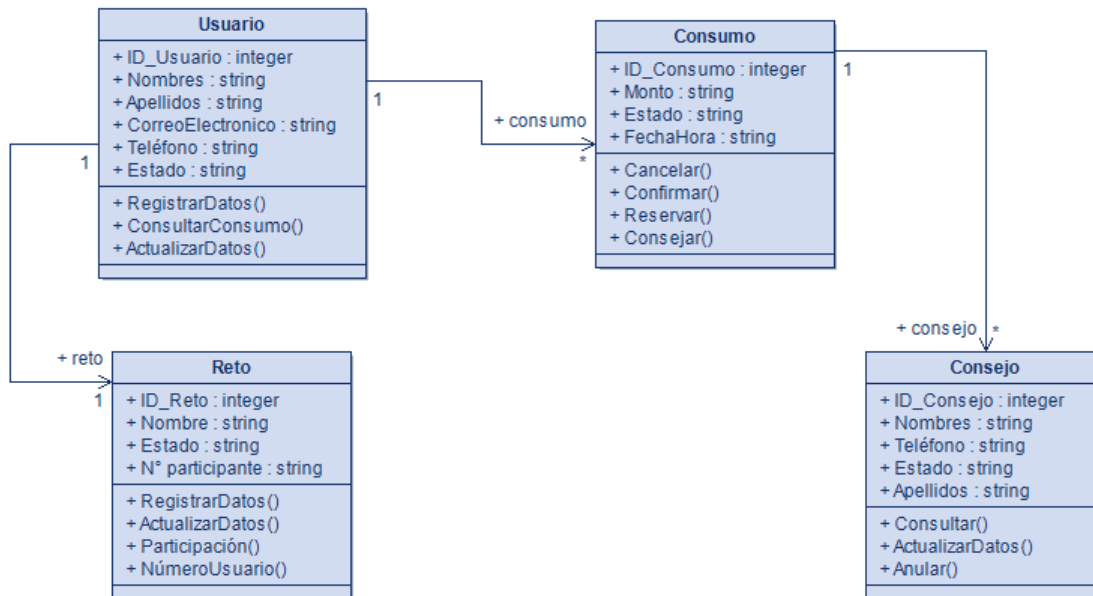
(Incluir diagrama con flujo de registro y visualización de datos)



Modelo de datos (Modelo E-R):

Entidades: Usuario, Consumo, Reto, Consejo.

Relaciones: Usuario registra Consumo, Usuario participa en Reto, etc.

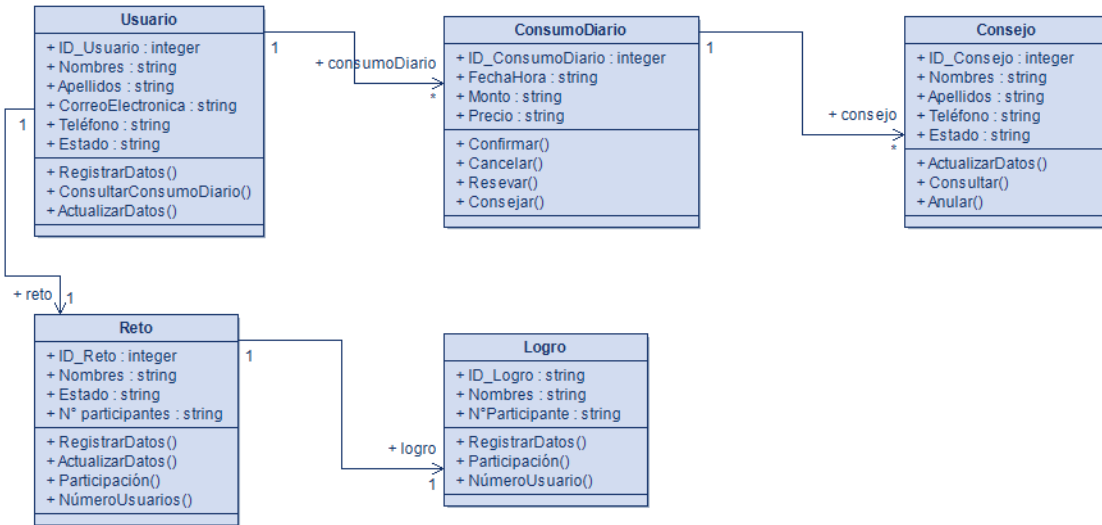


Unidad II – Modelos de Diseño y Metodología Ágil (Semanas 5–7)

Capítulo 4. Modelos de Diseño

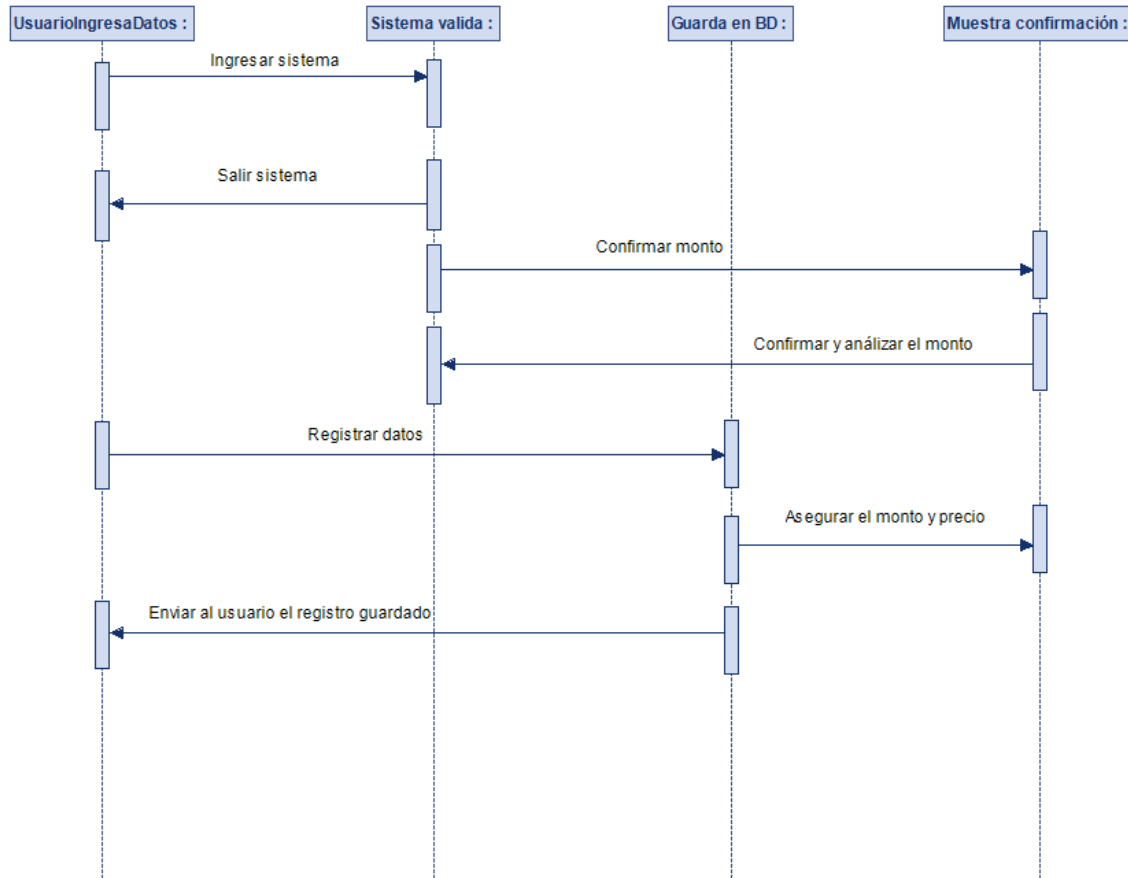
Modelo estructural (diagrama de clases inicial):

Clases: Usuario, Consumo Diario, Reto, Logro, Consejo.



Modelo de interacción (diagrama de secuencial):

Secuencia: Usuario ingresa datos → Sistema valida → Guarda en BD → Muestra confirmación.



Capítulo 5. Metodología de Trabajo (SCRUM)

Definición de la metodología ágil usada:

Scrum, con sprints de 2 semanas, reuniones diarias y revisión al final de cada iteración.

Backlog del producto (epics e historias de usuario):

- Epic 1: Gestión de Perfil y Consumo.
 - HU1: Registrar consumo diario.
 - HU2: Visualizar estadísticas.
 - HU3: Editar o eliminar registros.
- Epic 2: Educación y Participación.
 - HU4: Recibir consejos diarios.
 - HU5: Unirse a retos de ahorro.
 - HU6: Recibir recompensas.
- Epic 3: Sistema de Notificación.

- HU7: Recordatorios de consumo.
- HU8: Configuración de alertas
- **Epic 4: Seguridad y Gestión de Usuarios**
 - HU9: Registro y autenticación.
 - HU10: Personalizar perfil.
 - HU11: Protección de datos.
- **Epic 5: Integración con IoT.**
 - HU12: Conectar sensores IoT
 - HU13: Compatibilidad con estándares.
- **Epic 6: Interfaz y Experiencia de Usuario (UI/UX)**
 - HU14: Interfaz intuitiva.
 - HU15: Acceso multiplataforma.
 - HU16: Carga rápida.

Planificación de sprints (Sprint 1 y Sprint 2):

- Sprint 1: Núcleo del sistema
- Sprint 2: Gestión y experiencia
- Sprint 3: Innovación futura

Herramientas utilizadas:

Jira para gestión de tareas, [Draw.io](#) para diagramas, Dbdiagram.io para modelo de datos, GitHub para control de versiones.

Unidad III – Diseño de Software

Capítulo 6. Diseño de Arquitectura y Patrones

Estrategia de diseño del software

La estrategia de diseño seleccionada es **orientada a objetos**, aplicando un enfoque **modular y top-down** que permite dividir el sistema en componentes lógicos bien definidos.

Se emplea el principio de **bajo acoplamiento y alta cohesión**, lo que facilita la escalabilidad y el mantenimiento del sistema.

Esta estrategia se justifica porque la aplicación móvil debe integrar diversos módulos —registro de consumo, retos gamificados, estadísticas y consejos— que funcionan de manera independiente pero comparten datos en una base central.

Tipo de arquitectura del sistema

El sistema adopta una **arquitectura multicapa (Modelo MVC)** con separación clara entre las capas de presentación, lógica de negocio y persistencia de datos:

- **Capa de presentación:** interfaz móvil desarrollada con Flutter o React Native, optimizada para Android e iOS.
- **Capa de lógica de negocio:** gestión de usuarios, cálculo de consumo y manejo de logros gamificados.
- **Capa de datos:** base de datos relacional en MySQL o PostgreSQL, responsable del almacenamiento seguro y eficiente de la información.

Diagrama general de arquitectura (descripción):

El cliente (app móvil) se comunica con una API REST alojada en un servidor intermedio, el cual interactúa con la base de datos. Las solicitudes se gestionan a través de controladores que procesan la lógica y devuelven respuestas en formato JSON.

Patrones de diseño aplicados

1. **Patrón Modelo-Vista-Controlador (MVC):** organiza la aplicación en tres componentes para separar la interfaz, la lógica y los datos.
 - *Aporte:* facilita las pruebas y el mantenimiento del código.
2. **Patrón Singleton:** utilizado para la gestión de la conexión a la base de datos, asegurando que exista una única instancia activa.
 - *Aporte:* reduce el consumo de recursos y mejora la eficiencia del sistema.

Diseño estructural (UML)

Clase	Atributos principales	Métodos
Usuario	idUsuario, nombre, correo, contraseña	registrar(), autenticar(), actualizarPerfil()
ConsumoDiario	idConsumo, fecha, litrosUsados, idUsuario	registrarConsumo(), calcularPromedio()
Reto	idReto, descripcion, puntos, fechaInicio	completarReto(), obtenerRecompensa()
Consejo	idConsejo, texto, categoría	mostrarConsejo()
Logro	idLogro, nombre, puntos	asignarLogro(), mostrarLogro()

Se

desarrolló un **diagrama de clases UML** con las principales entidades:

Capítulo 7. Diseño Detallado de la Base de Datos

Modelo lógico y físico

Se diseñó un **modelo entidad-relación (E-R)** normalizado (3FN) con las siguientes entidades principales:

- **Usuario (1-N)** → registra varios **ConsumoDiario**
- **Usuario (N-M)** ↔ participa en varios **Reto** mediante una tabla intermedia **Usuario_Reto**
- **Consejo (1-N)** → puede estar asociado a varios **Reto**
- **Logro (1-N)** → asignado a **Usuario**

Modelo físico:

La base de datos se implementará en **MySQL**, con claves primarias, foráneas e índices optimizados (clustered y non-clustered).

Script SQL

```
-- =====  
  
-- TABLA: INSTITUCION_EDUCATIVA  
  
-- =====  
  
CREATE TABLE INSTITUCION_EDUCATIVA (  
    idInstitucion INT AUTO_INCREMENT PRIMARY KEY,  
    nombreInstitucion VARCHAR(200) NOT NULL,  
    tipo ENUM('pública', 'privada') NOT NULL,  
    direccion VARCHAR(250),  
    distrito VARCHAR(100),  
    provincia VARCHAR(100),  
    region VARCHAR(100),  
    codigoModular VARCHAR(10) UNIQUE,
```

```

        contacto VARCHAR(100),

        INDEX idx_nombre (nombreInstitucion),

        INDEX idx_region (region)

) ENGINE=InnoDB;

-- =====
-- TABLA: USUARIO
-- =====

CREATE TABLE USUARIO (

    idUsuario INT AUTO_INCREMENT PRIMARY KEY,

    nombre VARCHAR(100) NOT NULL,

    apellido VARCHAR(100) NOT NULL,

    correo VARCHAR(150) UNIQUE NOT NULL,

    contraseña VARCHAR(255) NOT NULL,

    fechaNacimiento DATE,

    idInstitucion INT,

    grado TINYINT CHECK (grado BETWEEN 1 AND 5),

    seccion VARCHAR(10),

    fechaRegistro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    ultimoAcceso TIMESTAMP NULL,

    estado ENUM('activo', 'inactivo') DEFAULT 'activo',

    FOREIGN KEY (idInstitucion) REFERENCES
INSTITUCION_EDUCATIVA(idInstitucion)

    ON DELETE SET NULL

    ON UPDATE CASCADE,

    INDEX idx_correo (correo),

    INDEX idx_institucion (idInstitucion),

    INDEX idx_estado (estado)

```

```
) ENGINE=InnoDB;
```

```
-- =====
```

```
-- TABLA: CONSUMO_DIARIO
```

```
-- =====
```

```
CREATE TABLE CONSUMO_DIARIO (  
    idConsumo INT AUTO_INCREMENT PRIMARY KEY,  
    idUsuario INT NOT NULL,  
    fecha DATE NOT NULL,  
    litrosUsados DECIMAL(8,2) NOT NULL CHECK (litrosUsados >= 0),  
    tipoActividad ENUM('ducha', 'lavado_manos', 'lavado_ropa',  
                        'riego', 'cocina', 'otros') DEFAULT 'otros',  
    duracionMinutos INT CHECK (duracionMinutos >= 0),  
    horaRegistro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    observaciones TEXT,  
    FOREIGN KEY (idUsuario) REFERENCES USUARIO(idUsuario)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE,  
    INDEX idx_usuario_fecha (idUsuario, fecha),  
    INDEX idx_fecha (fecha),  
    UNIQUE KEY uk_usuario_fecha_actividad (idUsuario, fecha,  
    tipoActividad)  
) ENGINE=InnoDB;
```

```
-- =====
```

```
-- TABLA: RETO
```

```
-- =====
```

```
CREATE TABLE RETO (  

```

```

    idReto INT AUTO_INCREMENT PRIMARY KEY,

    nombreReto VARCHAR(150) NOT NULL,

    descripcion TEXT NOT NULL,

    objetivo VARCHAR(200),

    puntosRecompensa INT DEFAULT 0 CHECK (puntosRecompensa >= 0),

    categoria ENUM('individual', 'grupál') DEFAULT 'individual',

    fechaInicio DATE NOT NULL,

    fechaFin DATE NOT NULL,

    nivelDificultad ENUM('fácil', 'medio', 'difícil') DEFAULT 'medio',

    estado ENUM('activo', 'finalizado', 'cancelado') DEFAULT 'activo',

    CHECK (fechaFin >= fechaInicio),

    INDEX idx_fechas (fechaInicio, fechaFin),

    INDEX idx_estado (estado)

) ENGINE=InnoDB;

```

```

-- =====

```

```

-- TABLA: USUARIO_RETO

```

```

-- =====

```

```

CREATE TABLE USUARIO_RETO (

    idUsuarioReto INT AUTO_INCREMENT PRIMARY KEY,

    idUsuario INT NOT NULL,

    idReto INT NOT NULL,

    fechaInscripcion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    progreso DECIMAL(5,2) DEFAULT 0.00 CHECK (progreso BETWEEN 0 AND
100),

    completado BOOLEAN DEFAULT FALSE,

    fechaCompletado TIMESTAMP NULL,

    puntosObtenidos INT DEFAULT 0,

```



```

FOREIGN KEY (idUserario) REFERENCES USUARIO(idUsuario)

    ON DELETE CASCADE

    ON UPDATE CASCADE,

FOREIGN KEY (idReto) REFERENCES RETO(idReto)

    ON DELETE CASCADE

    ON UPDATE CASCADE,

UNIQUE KEY uk_usuario_reto (idUserario, idReto),

INDEX idx_completado (completado),

INDEX idx_usuario (idUserario)

) ENGINE=InnoDB;

-- =====

-- TABLA: CONSEJO

-- =====

CREATE TABLE CONSEJO (

    idConsejo INT AUTO_INCREMENT PRIMARY KEY,

    titulo VARCHAR(200) NOT NULL,

    contenido TEXT NOT NULL,

    categoria ENUM('ahorro', 'reciclaje', 'educativo', 'general')
DEFAULT 'general',

    nivelEducativo ENUM('primaria', 'secundaria', 'general') DEFAULT
'secundaria',

    fechaPublicacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    autor VARCHAR(100),

    imagenURL VARCHAR(300),

    vecesVisto INT DEFAULT 0,

    INDEX idx_categoria (categoria),

    INDEX idx_nivel (nivelEducativo)

```

```
) ENGINE=InnoDB;
```

```
-- =====
```

```
-- TABLA: LOGRO
```

```
-- =====
```

```
CREATE TABLE LOGRO (  
    idLogro INT AUTO_INCREMENT PRIMARY KEY,  
    nombreLogro VARCHAR(150) NOT NULL,  
    descripcion TEXT,  
    iconoURL VARCHAR(300),  
    puntosRequeridos INT DEFAULT 0,  
    tipoLogro ENUM('bronce', 'plata', 'oro', 'especial') DEFAULT  
'bronce',  
    requisito VARCHAR(200),  
    INDEX idx_tipo (tipoLogro)  
) ENGINE=InnoDB;
```

```
-- =====
```

```
-- TABLA: USUARIO_LOGRO
```

```
-- =====
```

```
CREATE TABLE USUARIO_LOGRO (  
    idUsuarioLogro INT AUTO_INCREMENT PRIMARY KEY,  
    idUsuario INT NOT NULL,  
    idLogro INT NOT NULL,  
    fechaObtencion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    visible BOOLEAN DEFAULT TRUE,  
    FOREIGN KEY (idUsuario) REFERENCES USUARIO(idUsuario)  
    ON DELETE CASCADE
```

```

        ON UPDATE CASCADE,

FOREIGN KEY (idLogro) REFERENCES LOGRO(idLogro)

        ON DELETE CASCADE

        ON UPDATE CASCADE,

UNIQUE KEY uk_usuario_logro (idUserario, idLogro),

INDEX idx_fecha (fechaObtencion)

) ENGINE=InnoDB;

```

```
-- =====
```

```
-- TABLA: NOTIFICACION
```

```
-- =====
```

```

CREATE TABLE NOTIFICACION (

    idNotificacion INT AUTO_INCREMENT PRIMARY KEY,

    idUsuario INT NOT NULL,

    tipoNotificacion ENUM('recordatorio', 'logro', 'consejo', 'reto',
'sistema')

                                DEFAULT 'sistema',

    titulo VARCHAR(150) NOT NULL,

    mensaje TEXT NOT NULL,

    fechaEnvio TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    leida BOOLEAN DEFAULT FALSE,

    fechaLectura TIMESTAMP NULL,

    FOREIGN KEY (idUserario) REFERENCES USUARIO(idUsuario)

        ON DELETE CASCADE

        ON UPDATE CASCADE,

    INDEX idx_usuario_leida (idUserario, leida),

    INDEX idx_fecha (fechaEnvio)

) ENGINE=InnoDB;

```

Procedimientos almacenados, vistas y triggers}

Procedimiento: Registrar consumo

```
CREATE PROCEDURE sp_registrarConsumo(IN p_idUsuario INT, IN p_litros  
DECIMAL(6,2))
```

```
BEGIN
```

```
    INSERT INTO ConsumoDiario(fecha, litrosUsados, idUsuario)
```

```
    VALUES (CURDATE(), p_litros, p_idUsuario);
```

```
END;
```

Vista: Ranking de usuarios

```
CREATE VIEW vw_ranking AS
```

```
SELECT U.nombre, SUM(C.litrosUsados) AS total
```

```
FROM Usuario U
```

```
JOIN ConsumoDiario C ON U.idUsuario = C.idUsuario
```

```
GROUP BY U.idUsuario
```

```
ORDER BY total ASC;
```

Trigger: Alerta de consumo

```
CREATE TRIGGER trg_consumoAlto
```

```
AFTER INSERT ON ConsumoDiario
```

```
FOR EACH ROW
```

```
BEGIN
```

```
IF NEW.litrosUsados > 50 THEN
```

```
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT='Consumo excesivo';
```

```
END IF;
```

```
END;
```

Capítulo 8. Diseño Detallado de Sistemas en Red y Móviles

Modelo de comunicación

El sistema sigue un modelo **cliente-servidor** basado en **API REST**.

Los dispositivos móviles se comunican con el backend mediante peticiones **HTTPS**.

Los protocolos de comunicación principales son:

- **HTTP/HTTPS** para transferencia de datos.
- **JSON** como formato de intercambio.
- **Firebase Cloud Messaging (FCM)** para notificaciones push.

Diseño de sistema móvil

La interfaz móvil presenta:

- Pantalla principal con resumen de consumo.
- Módulo de retos gamificados.
- Sección de consejos educativos.
- Gráficos interactivos con estadísticas.

Se usará **Flutter** para permitir compatibilidad multiplataforma y rendimiento nativo.

Gestión de datos en red

Los datos se gestionan en tiempo real mediante la sincronización con la base de datos del servidor. Se implementa un mecanismo de **caché local** para uso offline temporal.

Seguridad en red y móviles

- Autenticación mediante **tokens JWT**.
- Validación de sesiones activas.
- Uso de certificados SSL/TLS para cifrar el tráfico.
- Políticas de actualización y bloqueo de cuenta ante intentos fallidos.

Justificación técnica

El diseño propuesto garantiza **escalabilidad, seguridad y disponibilidad**, alineado con los requerimientos funcionales y no funcionales establecidos.

El uso de **MVC + REST + Flutter + MySQL** ofrece una solución moderna, flexible y de fácil mantenimiento, ideal para proyectos educativos de bajo costo e impacto social.

Capítulo 9. Diseño de Interfaz y Experiencia de Usuario (UX/UI)

9.1. Perfil del usuario / usuario meta

Definición del usuario final según la ODS 6

El usuario principal del sistema son estudiantes de nivel secundario pertenecientes a instituciones educativas públicas y privadas. Este perfil se alinea directamente con la ODS 6 (Agua limpia y saneamiento), ya que la aplicación tiene como objetivo promover hábitos responsables de consumo de agua desde edades tempranas.

Los usuarios requieren herramientas educativas, dinámicas y gamificadas que les permitan comprender su impacto individual y colectivo en el uso del recurso.

Necesidades reales del usuario

- Acceder a información simple y visual sobre el ahorro de agua.
- Registrar su consumo de manera rápida.
- Participar en retos y actividades dinámicas.
- Visualizar estadísticas que los motiven a mejorar sus hábitos.
- Utilizar una interfaz intuitiva que no requiera conocimientos técnicos.

Contexto de uso

La aplicación se utilizará en:

- **Dispositivos móviles personales** (Android/iOS).
- **Ambientes escolares**, como aulas o laboratorios.
- **Entornos cotidianos del hogar**, donde el estudiante registra y evalúa su consumo.

El sistema debe funcionar correctamente en situaciones de conectividad limitada, con diseño ligero y navegación rápida.

9.2. Principios de diseño aplicados (HCI) y su aplicación en el prototipo

✓ Consistencia

Se mantiene un estilo uniforme en colores, íconos, botones y tipografías para que el usuario identifique fácilmente patrones de acción. Todas las pantallas usan el mismo menú inferior y la misma estructura de encabezados.

✓ Visibilidad

Los elementos principales (botón de registro de consumo, acceso a retos y estadísticas) se ubican en zonas visibles y accesibles desde la pantalla inicial. Los indicadores del progreso del ahorro emplean gráficos simples que permiten una comprensión inmediata.

✓ Accesibilidad

El prototipo utiliza:

- Tipografías legibles

- Contraste de colores adecuado
- Botones grandes para facilitar interacción
Esto favorece a usuarios con dificultades visuales o motrices.

✓ **Control del usuario**

El usuario puede editar registros, retroceder, cancelar acciones y navegar libremente por los módulos. Cada flujo muestra rutas reversibles y claras.

✓ **Retroalimentación**

Cada acción genera una respuesta inmediata:

- Notificaciones cuando se completa un reto
- Mensajes al registrar consumo
- Cambios visuales en los indicadores
La app informa sobre el estado de las tareas realizadas.

✓ **Simplicidad**

Se utiliza el principio “menos es más”:

- Navegación mínima
- Distribución limpia
- Íconos universales
- Textos breves
Esto permite que el usuario aprenda a usar la aplicación sin capacitación.

9.3. Diseño del prototipo (baja y alta fidelidad)

Prototipos realizados

Los prototipos se elaboraron en herramientas como **Figma**, **Draw.io** y/o **Balsamiq**.

link de Figma:

<https://www.figma.com/design/9WjtxRO8HZWIsvtZam2JMA/Untitled?node-id=0-1&t=muUOJjSRnuR0dGrz-1>

Pantallas principales :



**Agua limpia y
saneamiento**



Iniciar sesión

Registrar

[Olvidar contraseña](#)





AguaSmart

Registrar Consumo Diario

Fecha



Litros Usados



Actividad



Notas (opcional)



Registrar Consumo



AguaSmart

CREAR NUEVO RETO

Nombre del Reto

Descripción

Meta (Litros)

Duración (Días)

Tipo de Reto

- ☐ Individual
- ☐ Grupal
- ☐ Institucional

Puntos de Recompensa

+ Crear Reto



AguaSmart

CONFIGURATION PROFILE

NOMBRE COMPLETO

INSTITUCIÓN EDUCATIVA

GRADO

SECCIÓN

CORREO ELECTRONICO

NÚMERO DE CELULAR (OPCIONAL)

GUADRA PERFIL



AguaSmart

Generar Reportes

generar reporte de consumo

TIPO DE REPORTE

- ☐ INDIVIDUAL
- ☐ POR SECCIÓN
- ☐ INSTITUCIONAL

FECHA INICIO

FECHA INICIO

FORMATO DE EXPORTACIÓN

- ☐ PDF
- ☐ EXCEL
- ☐ OTROS

INCLUIR EN EL REPORTE

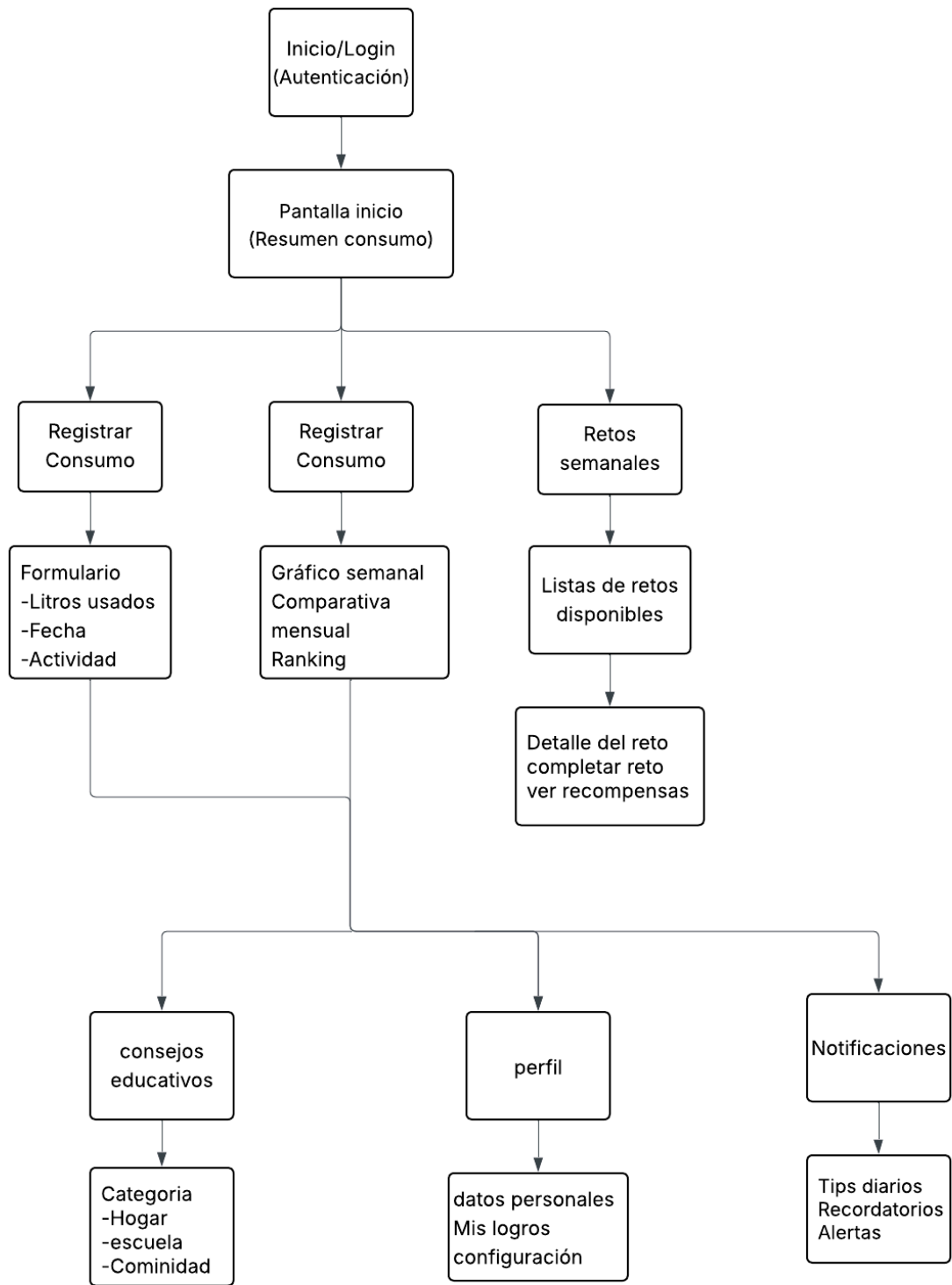
- ☐ CONSUMO DIARIO DETALLADO
- ☐ GRÁFICOS ESTADÍSTICOS
- ☐ COMPARATIVA CON PROMEDIO
- ☐ RETOS COMPLETADOS
- ☐ LOGROS OBTENIDOS

GENERAR REPORTE

Elemento de Diseño	Justificación y Aplicación	Principio HCI Relacionado
Colores (Azules/Verdes)	Predominio de azules y celestes (agua, calma) y verdes brillantes (sostenibilidad, progreso). Se utiliza el color para asociar la acción positiva (registro, creación) con el recurso hídrico.	Consistencia, Visibilidad
Tipografía	Se utiliza texto legible y en algunos casos tipografías gruesas para captar la atención. Se recomienda Sans-serif moderna (Inter/Roboto) en la implementación final para garantizar alta legibilidad en móviles.	Accesibilidad, Simplicidad
Organización de la Información	Uso de formularios cortos y verticales ("Registrar Consumo") y agrupaciones lógicas de opciones ("Generar Reportes"). Uso de íconos universales.	Simplicidad, Control del Usuario

9.4. Flujo de navegación del sistema

Mapa general de navegación



Descripción del flujo

1. El usuario ingresa al sistema y visualiza el resumen de su consumo.
2. Puede registrar nuevos datos rápidamente.
3. Revisa su progreso en la sección de estadísticas.
4. Participa en retos y recibe recompensas.
5. Consulta consejos educativos.
6. Gestiona su información personal desde el módulo de perfil.

El sistema se organiza en seis módulos principales accesibles desde la pantalla de inicio. Cada módulo representa un flujo independiente que guía al usuario según la funcionalidad seleccionada.

Estructura del módulo-navegación:

1. Inicio
 - Acceso directo a los módulos:
 - Registrar Consumo
 - Estadísticas
 - Retos Semanales
 - Consejos Educativos
 - Perfil de Usuario
 - Notificaciones (automáticas)
2. Registrar Consumo
 - Formulario de ingreso → Validación → Confirmación → Retorno a Inicio
3. Estadísticas
 - Panel de gráficos → "Generar Reporte" → Descargar o compartir
4. Retos Semanales
 - Lista de retos → Detalle del reto → Unirse o completar → Registro de progreso → Asignación de puntos o medallas
5. Consejos Educativos
 - Biblioteca de consejos → Filtro por categoría → Vista detallada → Marcar como favorito
6. Perfil de Usuario

- Datos del usuario → Editar información → Ver logros → Configuración → Cerrar sesión

7. Notificaciones

- Sistema automático de avisos: recordatorios diarios, consejos del día, nuevos retos y logros obtenidos.

9.5. Relación entre prototipo y requerimientos

Pantalla 1: Inicio / Dashboard

Incluye accesos a: Registrar Consumo, Estadísticas, Retos, Consejos, Perfil y Notificaciones.

Requerimientos que cumple:

- RF2: Visualización de estadísticas y progreso
→ El inicio muestra el resumen del consumo.
- RF3: Notificaciones (indirecto)
→ Se muestra acceso a avisos automáticos.
- RNF1: Interfaz intuitiva y accesible
→ Menú inferior visible y navegación clara.

Justificación

Esta pantalla centraliza el estado del usuario y permite llegar rápido a todos los módulos, cumpliendo visibilidad, consistencia y simplicidad.

✓ Pantalla 2: Registrar Consumo

Flujo: Formulario → Validación → Confirmación → Regreso a inicio.

Requerimientos que cumple:

- RF1: Registro de consumo diario
→ Es la pantalla donde el usuario ingresa litros usados.
- RNF1: Interfaz accesible
→ Usa formularios cortos, botones grandes y legibles.

Justificación

Cumple el requerimiento principal del sistema: registrar el consumo y actualizar indicadores.

Pantalla 3: Estadísticas

Panel de gráficos + opción de generar reportes.

Requerimientos que cumple:

- RF2: Visualización de estadísticas y progreso
→ Muestra gráficos y evolución del consumo.
- RNF3: Compatibilidad con Android/iOS
→ El diseño está pensado para Flutter.

Justificación

Transforma los registros en gráficos comprensibles que ayudan al aprendizaje del usuario.

Pantalla 4: Retos Semanales

Lista → Detalle → Unirse → Completar → Asignación de medallas.

Requerimientos que cumple:

- RF4: Retos comunitarios y logros
→ El usuario participa en desafíos de ahorro de agua.
- RNF1: Interfaz simple
→ Íconos y flujos cortos facilitan interacción.

Justificación

Implementa la parte de gamificación del sistema, un elemento esencial de tu propuesta educativa.

Pantalla 5: Consejos Educativos

Biblioteca → filtro → vista detallada → favoritos.

Requerimientos que cumple:

- RF3: Sistema de notificaciones/tips diarios
→ Los consejos del día salen de esta sección.
- RNF1: Accesibilidad visual
→ Uso de tipografías legibles y buen contraste.

Justificación

Brinda contenido educativo alineado a ODS 6 y a los hábitos de ahorro.

Pantalla 6: Perfil de Usuario

Datos → editar → logros → configuración → cerrar sesión.

Requerimientos que cumple:

- RF5: Perfil personalizable
→ El usuario edita su información y ve sus logros.
- RNF4: Seguridad de datos
→ Soporta autenticación y protección de datos.

Justificación

Gestiona todo lo relacionado con la identidad del usuario en la aplicación.

Pantalla 7: Notificaciones

Avisos automáticos: consumo, consejos, retos y logros.

Requerimientos que cumple:

- RF3: Notificaciones
→ Envío de recordatorios diarios y alertas.
- RNF3: Compatibilidad
→ Usa Firebase Cloud Messaging (FCM).

Capítulo 10. Evaluación del Diseño y Matriz de Trazabilidad

10.1. Matriz de Trazabilidad

La siguiente matriz relaciona los requerimientos del sistema con los casos de uso, módulos, pantallas del prototipo y si el requerimiento se cumple total, parcialmente o no cumple.

Req.	Descripción	Caso de uso	Clase / módulo	Pantalla del prototipo	Cumple / Parcial / No cumple
RF1	Registro de consumo diario	CU1: Registrar consumo	ConsumoDiario	Registrar Consumo	Cumple
RF2	Visualización de estadísticas	CU2: Ver estadísticas	Estadísticas	Estadísticas (gráficos)	Cumple
RF3	Sistema de tips y notificaciones	CU3: Recibir consejos y alertas	Notificación / Consejo	Consejos Educativos / Notificaciones	Cumple
RF4	Retos y logros gamificados	CU4: Participar en retos	Reto / Usuario_Reto	Retos Semanales	Cumple
RF5	Perfil personalizable	CU5: Gestionar perfil	Usuario	Perfil de Usuario	Cumple
RNF1	Interfaz intuitiva y accesible	Todos	UI/UX	Todas las pantallas	Cumple
RNF2	Tiempo de respuesta < 2 s	Backend/API	Lógica de negocio	No aplica (diseño)	Parcial
RNF3	Compatibilidad Android/iOS	Sistema móvil	Flutter	Todas	Cumple
RNF4	Seguridad de datos	Autenticación / BD	Usuario / Seguridad	Perfil / Inicio	Cumple

10.2. Evaluación del Diseño del Sistema

La evaluación del diseño se realizó considerando la coherencia entre la arquitectura propuesta, el prototipo, las clases, casos de uso y la base de datos.

✓ Coherencia arquitectura–prototipo–BD

- La arquitectura multicapa (MVC) está alineada con el flujo del prototipo.
- Las clases del diseño estructural coinciden con las tablas del modelo lógico (Usuario, ConsumoDiario, Reto, Logro, Consejo).
- El flujo de navegación del prototipo refleja correctamente los casos de uso definidos.

✓ Identificación de mejoras

- Se recomienda optimizar la estructura de notificaciones (filtros, historial, prioridad).
- Sugerencia de incorporar accesibilidad avanzada (modo oscuro, lectura de textos).

✓ Evaluación de usabilidad

Basada en los principios de Nielsen:

- Visibilidad del estado del sistema: correcto (mensajes, alertas).
- Control del usuario: formularios editables y navegación reversible.
- Prevención de errores: formularios simples y validaciones básicas.
- Consistencia: colores, iconos y estilos uniformes.

Resultado general: Diseño usable y coherente con los objetivos educativos.

10.3. Retroalimentación Obtenida

La retroalimentación provino de:

- ✓ Evaluación entre pares
- ✓ Revisión heurística simple
- ✓ Prueba exploratoria del prototipo

Principales hallazgos

- Errores detectados: botones poco visibles en la pantalla de Retos y tipografía pequeña en Consejos.
- Cambios realizados: se aumentó el contraste, el tamaño de fuente y se reorganizaron elementos.
- Mejoras propuestas:
 - Incluir un buscador en Consejos.
 - Agregar historial de retos completados.
 - Mostrar comparaciones más visuales en estadísticas.

10.4. Reflexión sobre el aporte al ODS

Este proyecto contribuye directamente al ODS 6: Agua limpia y saneamiento, ya que:

- El diseño de la interfaz: usa elementos visuales relacionados al cuidado del agua (colores verdes y azules, iconografía educativa).
- La navegación: facilita que los usuarios aprendan y apliquen hábitos sostenibles.
- Las funcionalidades: como el registro de consumo, los retos semanales y los consejos educativos, promueven cambios reales en el comportamiento de los estudiantes.

El sistema está diseñado para generar conciencia ambiental, fomentar hábitos sostenibles y mejorar la comprensión sobre la importancia del ahorro del agua.

Justificación

Garantiza que el usuario reciba retroalimentación continua.

Conclusiones y Recomendaciones

Conclusiones del equipo:

La app educativa es una herramienta viable para fomentar el ahorro de agua y la concienciación ambiental.

Lecciones aprendidas:

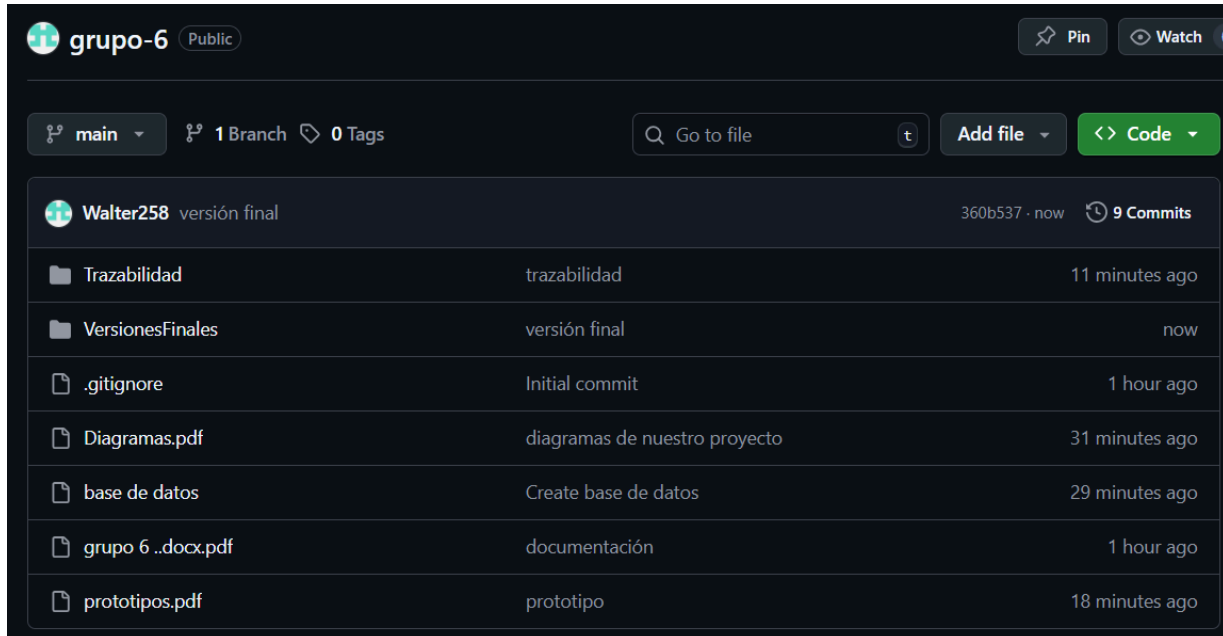
- La retroalimentación temprana con usuarios es clave.
- La simplicidad en la UI mejora la adopción.

Recomendaciones para futuras mejoras del sistema:

- Integrar IoT para medición automática.
- Ampliar a más idiomas y regiones.
- Incluir gamificación avanzada.

Capítulo 11. Consolidación Final y Evidencia de Trabajo Colaborativo (GitHub)

11.1. Repositorio GitHub del proyecto



11.2. Evidencias de trabajo colaborativo

This screenshot shows the main page of a GitHub repository named 'grupo-6' owned by 'Walter258'. The repository is public and has 1 branch and 0 tags. The file list includes:

File	Commit Message	Time Ago
Walter258 trazabilidad	bd6528e	4 minutes ago
Trazabilidad	trazabilidad	4 minutes ago
.gitignore	Initial commit	1 hour ago
Diagramas.pdf	diagramas de nuestro proyecto	24 minutes ago
base de datos	Create base de datos	22 minutes ago
grupo 6 .docx.pdf	documentación	1 hour ago
prototipos.pdf	prototipo	11 minutes ago

The right sidebar contains sections for 'About' (No description, website, or topics provided), 'Activity' (0 stars, 0 watching, 0 forks), 'Releases' (No releases published), and 'Packages' (No packages published).

This screenshot shows the commit history for the 'Trazabilidad' file. The table lists the following commits:

Name	Last commit message	Last commit date
...		
avance1.pdf	trazabilidad	5 minutes ago
avance2.pdf	trazabilidad	5 minutes ago
avance3.pdf	trazabilidad	5 minutes ago

This screenshot shows the content of the 'Diagramas.pdf' file. The document features a large title 'DIAGRAMAS' and a section titled 'DIAGRAMA DE CONTEXTO:'. Below this, a flowchart illustrates a process flow:

```
graph LR; A[Registra consumo / participa en retos / consulta estadísticas y consejos] --> B[Consulta reportes]; B --> C[Institución Educativa]; C --> D[Envía peticiones];
```

11.3. Control de versiones de documentos y modelos

El equipo utilizó GitHub como herramienta de control de versiones para gestionar la evolución del proyecto.

Se aplicó versionado semántico simple:

Versiones etiquetadas

- **v1.0:**
Primera versión del informe con capítulos 1–4 completos.
- **v2.0:**
Inclusión de diagramas UML, prototipos y modelo de base de datos.
- **vFinal:** Documento consolidado con correcciones finales, prototipo, trazabilidad y evidencias.

11.4 – Reflexión del equipo

El uso de GitHub mejoró la organización del proyecto, ya que permitió trabajar de forma ordenada, guardar versiones de los archivos y colaborar sin perder información. Aunque al inicio hubo retos como aprender a manejar ramas y resolver conflictos de merge, el equipo logró adaptarse y fortalecer su trabajo colaborativo. Entre los principales aprendizajes destacan la importancia del control de versiones, la comunicación técnica y la organización del flujo de trabajo. Como mejoras futuras, se propone usar GitHub Actions, crear una Wiki para documentar decisiones y agregar pruebas automatizadas para optimizar el desarrollo.

link del github: <https://github.com/Walter258/grupo-6.git>

Referencias

(Formato ISO 690 numérico)

1. Naciones Unidas. (2018). *Objetivo de Desarrollo Sostenible 6: Agua limpia y saneamiento*. [en línea] Disponible en:
<https://www.un.org/sustainabledevelopment/es/water-and-sanitation/>
2. Sommerville, I. (2011). *Ingeniería de Software*. 9ª ed. Pearson.
3. Martin, R. C. (2008). *Clean Code: A Handbook of Agile Software Craftsmanship*. Prentice Hall.

The screenshot shows a Google Meet interface. The main window displays a Google Docs document titled "grupo 6" with the following content:

Herramientas utilizadas:
 Jira para gestión de tareas, Draw.io para diagramas, Dbdiagram.io para modelo de datos, GitHub para control de versiones.

Conclusiones y Recomendaciones

Conclusiones del equipo:
 La app educativa es una herramienta viable para fomentar el ahorro de agua y la concienciación ambiental.

Lecciones aprendidas:

- La retroalimentación temprana con usuarios es clave.
- La simplicidad en la UI mejora la adopción.

Recomendaciones para futuras mejoras del sistema:

- Integrar IoT para medición automática.
- Ampliar a más idiomas y regiones.
- Incluir gamificación avanzada.

Referencias

On the right side of the screen, there are four participant avatars in a 2x2 grid:

- Top-left: MARCO ANTONIO ROJA... (purple background, white letter M)
- Top-right: Walter hugo Asto condori (teal background, white letter W)
- Bottom-left: SANDRO LUIS MACHICA... (teal background, white letter S)
- Bottom-right: FABIO MIGUEL VARGAS ... (dark teal background, profile picture)

The top of the screen shows the browser address bar with the URL "meet.google.com/esg-vdtr-nhs?authuser=0" and a share bar with the text "Compartiendo doc.google.com en esta pestaña" and a "Dejar de compartir" button.

Anexos

🏠

🔍

👤 Uira

🔍

Buscar

+ Crear

Versión de prueba de Premium

🗨 Chat de Rovo

🔔

?

⚙

SH

👤 Para ti

🕒 Recientes

⭐ Marcados como fav...

🔧 Aplicaciones

📅 Planes PREMIUM

📁 Proyectos + ...

Recientes

🔖 agua

☰ Más proyectos

🔍 Filtros

📄 Paneles

🎯 Metas

👥 Equipos

⚙ Personalizar barra lateral

Proyectos

🔖 agua

...

📊 Resumen

📅 Cronograma

📋 Backlog

📊 Tablero

📅 Calendario

📄 Lista

📄 Formularios

🎯 Metas

More 5 +

🔍 Buscar en...

👤 SH

👤 FF

👤 MQ

👤 WC

Filter

📈

🔍

...

📌

▼

Sprint 1 - Núcleo del sistema

30 sep - 14 oct (6 actividades)

25 0 0

Iniciar sprint

...

Implementar las funciones básicas: registro de consumo, visualización de estadísticas y primeras funciones educativas.

🔖

AGUA-7

Registrar consumo diario

GESTIÓN DE PERFIL ...

TAREAS POR HACER

📅 15 oct

5

MQ

🔖

AGUA-8

Visualizar estadísticas

GESTIÓN DE PERFIL ...

TAREAS POR HACER

📅 15 oct

5

MQ

🔖

AGUA-9

Editar o eliminar registros

GESTIÓN DE PERFIL ...

TAREAS POR HACER

📅 15 oct

3

FF

🔖

AGUA-10

Recibir consejos diarios

EDUCACIÓN Y PARTI...

TAREAS POR HACER

📅 15 oct

2

MQ

🔖

AGUA-11

Unirse a retos de ahorro

EDUCACIÓN Y PARTI...

TAREAS POR HACER

📅 15 oct

5

WC

🔖

AGUA-12

Recibir recompensas

EDUCACIÓN Y PARTI...

TAREAS POR HACER

📅 15 oct

5

MQ

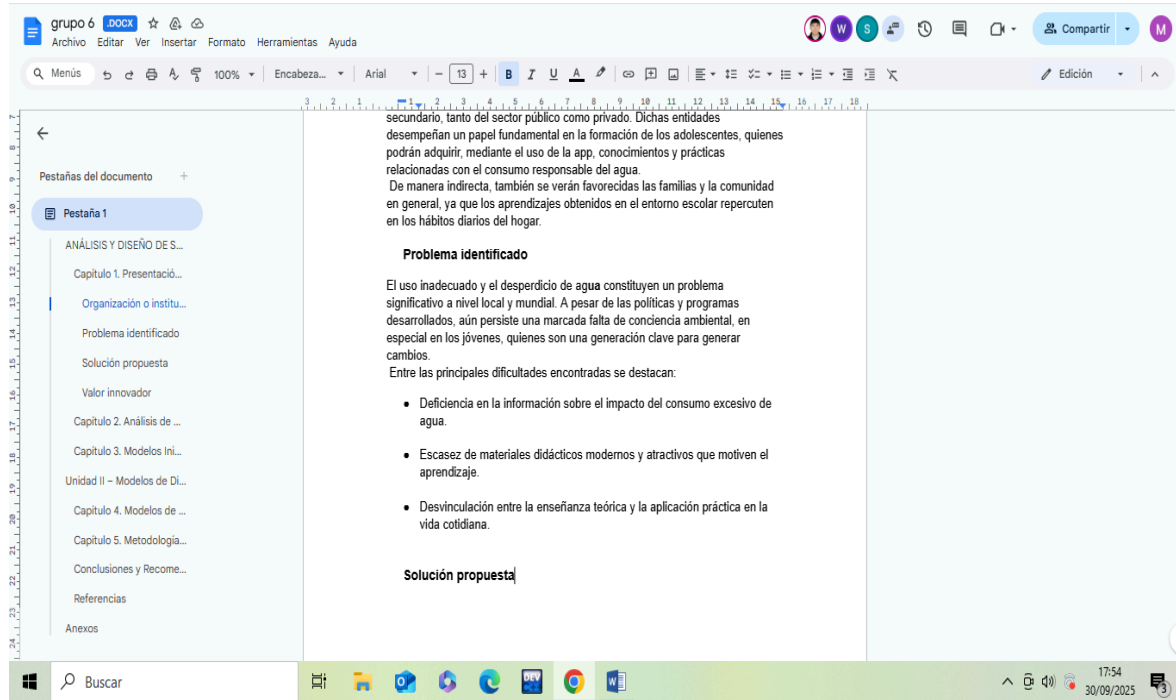
+ Crear

⌵

6 de 6 actividades visibles

Estimación: 25 de 25

🗨 Enviar comentarios so...



link de canva:

https://www.canva.com/design/DAG6I5_gKfA/Ebb5juNcsOFdCDfew7xxAA/edit?utm_content=DAG6I5_gKfA&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton