



UNIVERSIDAD DE EL SALVADOR
Facultad Multidisciplinaria De Occidente
Departamento de Ingeniería y Arquitectura
Ingeniería en Desarrollo de Software



Asignatura:
ANÁLISIS DE DISEÑO DE SOFTWARE
Ciclo V - Tercer Año

Tema:
Proyecto Final

Coordinador de Cátedra:
Ing. Fernando Alexis Guardado Flores

Tutor/a GT01:
Ing. Fernando Alexis Guardado Flores

Alumnos/as:	
Walter Alexander Galicia Gertrudis	GG14054
Alexandra Quinteros Cárcamo	QC23006
Franklin Imanol Ramírez Gómez	RG22059
Gabriel Ernesto Díaz Galdámez	DG22023
Ricardo Adán Patiño Hernández	PH23011

Fecha:
15/06/2025

ÍNDICE

1. ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE (IEEE 830)	2
1.1 Introducción	2
1.1.1 Propósito	2
1.1.2 Alcance	2
1.1.3 Definiciones, Acrónimos y Abreviaturas	3
1.1.4 Referencias	3
1.1.5 Visión General	3
1.2 Descripción General	4
1.2.1 Perspectiva del Producto	4
1.2.2 Funciones del Producto	4
1.2.3 Características de los Usuarios	4
1.2.4 Restricciones	5
1.2.5 Suposiciones y Dependencias	5
1.3 Requerimientos Específicos	6
1.3.1 Requerimientos Funcionales (RF-001 a RF-016)	6
1.3.2 Requerimientos de Rendimiento (RNF-001 a RNF-003)	10
1.3.3 Requerimientos de Seguridad (RNF-004 a RNF-006)	11
1.3.4 Requerimientos de Usabilidad (RNF-007 a RNF-008)	11
1.3.5 Requerimientos de Compatibilidad (RNF-009 a RNF-010)	11
1.3.6 Requerimientos de Mantenimiento (RNF-011 a RNF-012)	12
1.4 Apéndices	12
Apéndice A: Glosario de Términos	12
Apéndice B: Casos de Uso Principales	14
Apéndice C: Matriz de Trazabilidad	14

2. DIAGRAMA DE CLASES Y DESCRIPCIÓN DEL SISTEMA	15
2.1 Diagrama de Clases UML	15
2.2 Descripción del Diagrama de Clases	17
2.2.1 Dominio Principal	17
2.2.2 Gestión Académica	19
2.2.3 Gestión de Asistencia	21
2.2.4 Integración de Hardware	23
2.2.5 Reportes y Análisis	25
2.2.6 Seguridad y Auditoría	26
3. METODOLOGÍA DE DESARROLLO Y DEFINICIÓN DE ROLES	28
3.1 Metodología Híbrida (SCRUM + WATERFALL)	28
3.1.1 Justificación de la Metodología	28
3.1.2 Descripción Detallada del Uso	29
3.2 Definición de Roles del Equipo	32
3.2.1 Tech Lead / Arquitecto de Software	32
3.2.2 Product Owner / Analista de Negocio	32
3.2.3 Frontend UI/UX Developer	32
3.2.4 Scrum Master	32
3.2.5 DevOps / QA Engineer	32
3.3 Calendario de Trabajo	33

Especificación de Requerimientos de Software

Sistema de Registro de Asistencia Universitaria (SIRAU)

Versión: 1.0

Fecha: 14 de junio de 2025

Preparado por: Grupo 10 - ADS135

Estándar aplicado: IEEE 830-1998

1. INTRODUCCIÓN

1.1 Propósito

Este documento especifica los requerimientos de software para el Sistema de Registro de Asistencia Universitaria (SIRAU). El documento está dirigido a desarrolladores, analistas, administradores universitarios, y personal involucrado en el desarrollo e implementación del sistema en cuestión.

1.2 Alcance

El SIRAU es una plataforma digital multiplataforma (web/móvil) que automatiza y centraliza el registro de asistencia de estudiantes y empleados universitarios. El sistema eliminará procesos manuales, proporcionará acceso en tiempo real a datos de asistencia, y se integrará con sistemas académicos existentes como plataformas moodle, google workspace, y otros.

Beneficios esperados:

- Reducción del 90% en tiempo de toma de lista de asistencia.
- Eliminación de fraudes en registros de asistencia al validar que el usuario esté en una localización válida para el sistema.
- Mejora en la retención estudiantil mediante identificación temprana de riesgos ante cualquier anomalía en la asistencia a clases o actividades.
- Cumplimiento de normativas institucionales de protección de datos personales, y así mantener integridad de datos y la confidencialidad.

1.3 Definiciones, Acrónimos y Abreviaturas

- **SIRAU:** Sistema de Registro de Asistencia Universitaria
- **SRS:** Software Requirements Specification
- **QR:** Quick Response (código de respuesta rápida)
- **RFID:** Radio Frequency Identification
- **GPS:** Global Positioning System
- **Geovallado:** Tecnología que define límites geográficos virtuales
- **Dashboard:** Panel de control interactivo

1.4 Referencias

- IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications
- Normativas institucionales de protección de datos de la Universidad.
- Estándares de integración con Moodle/Blackboard o sistemas existentes para ser integrados.

1.5 Visión General de este documento

Este documento está organizado en tres secciones principales: Introducción, Descripción General del sistema, y Requerimientos Específicos que detallan las funcionalidades, rendimiento y restricciones del sistema que se quiere implementar.

2. DESCRIPCIÓN GENERAL

2.1 Perspectiva del Producto

El SIRAU es un sistema independiente que se integrará con la infraestructura tecnológica universitaria existente. Operará como:

- Sistema web accesible desde navegadores web.
- Aplicación móvil para dispositivos iOS y Android por medio de una aplicación web responsive y adaptable.
- Interfaz con sistemas académicos (Moodle, Blackboard, Google Workspace).
- Conexión con sistemas de nómina institucional que existan (si aplica).

2.2 Funciones del Producto

Las principales funciones del sistema incluyen:

- Registro automático de asistencia mediante múltiples métodos de autenticación.
- Gestión de roles y permisos diferenciados por tipo de usuario.
- Generación automática de reportes y alertas tempranas.
- Análisis de datos mediante dashboards interactivos.
- Integración bidireccional con plataformas académicas y de nómina de empleados.

2.3 Características de los Usuarios

2.3.1 Estudiantes

- **Cantidad:** Miles de usuarios concurrentes.
- **Experiencia técnica:** Básica a intermedia.
- **Responsabilidades:** Registrar asistencia, consultar historial, solicitar justificaciones.

2.3.2 Docentes

- **Cantidad:** Cientos de usuarios.
- **Experiencia técnica:** Básica a intermedia.
- **Responsabilidades:** Monitorear asistencia por curso, gestionar ausencias justificadas.

2.3.3 Administradores

- **Cantidad:** Decenas de usuarios
- **Experiencia técnica:** Intermedia a avanzada
- **Responsabilidades:** Configuración del sistema, gestión de usuarios, análisis de datos, reportes de asistencia de estudiantes y/o empleados.

2.3.4 Coordinadores Académicos

- **Cantidad:** Decenas de usuarios
- **Experiencia técnica:** Básica a intermedia
- **Responsabilidades:** Revisar reportes, tomar decisiones basadas en análisis de asistencia de estudiantes.

2.4 Restricciones

- Cumplimiento con regulaciones de protección de datos personales para confidencialidad.
- Compatibilidad con infraestructura de red universitaria existente.
- Restricciones presupuestarias para hardware biométrico.
- Tiempo de implementación limitado a un período académico.

2.5 Suposiciones y Dependencias

- Disponibilidad de conectividad a internet estable en el campus.
- Dispositivos móviles de usuarios con capacidad para abrir aplicaciones web.
- Personal técnico disponible para mantenimiento del sistema.
- Cooperación de usuarios para adopción de nuevas tecnologías y una nueva plataforma.

3. REQUERIMIENTOS ESPECÍFICOS

3.1 Requerimientos Funcionales

3.1.1 Registro de Asistencia

RF-001: Autenticación por Huella Dactilar

- **Descripción:** El sistema debe permitir el registro de asistencia mediante lector de huella dactilar
- **Prioridad:** Alta
- **Entradas:** Huella dactilar del usuario
- **Salidas:** Confirmación de registro exitoso o mensaje de error
- **Criterios de aceptación:** El sistema debe procesar la huella en máximo 3 segundos con 99.5% de precisión

RF-002: Autenticación por Reconocimiento Facial

- **Descripción:** El sistema debe permitir el registro de asistencia mediante reconocimiento facial
- **Prioridad:** Media
- **Entradas:** Imagen facial del usuario
- **Salidas:** Confirmación de registro exitoso o mensaje de error
- **Criterios de aceptación:** El sistema debe procesar el reconocimiento en máximo 5 segundos con 98% de precisión

RF-003: Autenticación por Código QR

- **Descripción:** El sistema debe generar códigos QR personalizados para cada usuario que puedan ser escaneados desde la aplicación móvil
- **Prioridad:** Alta
- **Entradas:** Código QR escaneado
- **Salidas:** Confirmación de registro exitoso o mensaje de error
- **Criterios de aceptación:** Los códigos QR deben ser únicos, renovarse cada sesión y procesarse en máximo 2 segundos

RF-004: Autenticación por Tarjeta RFID

- **Descripción:** El sistema debe permitir el registro mediante tarjetas RFID para empleados y aulas equipadas
- **Prioridad:** Media
- **Entradas:** Señal RFID de tarjeta
- **Salidas:** Confirmación de registro exitoso o mensaje de error
- **Criterios de aceptación:** Lectura instantánea con rango efectivo de 10cm

RF-005: Validación por Geovallado

- **Descripción:** El sistema debe validar que el registro se realice dentro de los límites del campus universitario mediante GPS
- **Prioridad:** Alta
- **Entradas:** Coordenadas GPS del dispositivo
- **Salidas:** Validación de ubicación correcta o rechazo del registro
- **Criterios de aceptación:** Precisión de ± 10 metros, funcionamiento en condiciones meteorológicas normales

3.1.2 Panel de Control para Docentes

RF-006: Visualización de Asistencia en Tiempo Real

- **Descripción:** Los docentes deben poder visualizar la asistencia de estudiantes por curso/grupo en tiempo real
- **Prioridad:** Alta
- **Entradas:** Selección de curso/grupo
- **Salidas:** Lista de estudiantes con estado de asistencia
- **Criterios de aceptación:** Actualización automática cada 30 segundos, interfaz intuitiva

RF-007: Gestión de Ausencias Justificadas

- **Descripción:** Los docentes deben poder marcar ausencias como justificadas con soporte documental
- **Prioridad:** Alta
- **Entradas:** Selección de estudiante, documentos de respaldo
- **Salidas:** Confirmación de justificación registrada.

- **Criterios de aceptación:** Soporte para archivos PDF, JPG, PNG hasta 5MB, trazabilidad de cambios

3.1.3 Panel de Control para Estudiantes/Empleados

RF-008: Consulta de Historial Personal

- **Descripción:** Estudiantes y empleados deben poder consultar su historial de asistencia y porcentajes
- **Prioridad:** Alta
- **Entradas:** Credenciales de usuario
- **Salidas:** Historial detallado con estadísticas
- **Criterios de aceptación:** Datos históricos de al menos 2 años, cálculo automático de porcentajes

RF-009: Solicitud de Justificaciones

- **Descripción:** Los usuarios deben poder solicitar justificaciones por ausencias (ej. enfermedad)
- **Prioridad:** Media
- **Entradas:** Fecha de ausencia, motivo, documentos de respaldo
- **Salidas:** Confirmación de solicitud enviada
- **Criterios de aceptación:** Workflow de aprobación, notificaciones automáticas

3.1.4 Reportes y Análisis

RF-010: Generación de Reportes Automáticos

- **Descripción:** El sistema debe generar reportes semanales automáticos para coordinadores académicos
- **Prioridad:** Alta
- **Entradas:** Configuración de periodicidad y destinatarios
- **Salidas:** Reportes en formato PDF/Excel enviados por correo
- **Criterios de aceptación:** Envío programado los lunes antes de las 8:00 AM

RF-011: Alertas por Inasistencia Recurrente

- **Descripción:** El sistema debe generar alertas automáticas cuando un estudiante supere el 20% de inasistencias en un mes
- **Prioridad:** Alta
- **Entradas:** Datos de asistencia acumulados
- **Salidas:** Notificaciones a coordinadores y estudiante
- **Criterios de aceptación:** Verificación diaria, notificaciones en tiempo real

RF-012: Dashboards Interactivos

- **Descripción:** El sistema debe proporcionar dashboards con tendencias por facultad, carrera o departamento
- **Prioridad:** Media
- **Entradas:** Filtros de búsqueda y rango de fechas
- **Salidas:** Gráficos interactivos y estadísticas
- **Criterios de aceptación:** Actualización en tiempo real, exportación de datos

3.1.5 Integraciones

RF-013: Integración con Plataformas Académicas

- **Descripción:** El sistema debe sincronizar datos con Moodle o Blackboard
- **Prioridad:** Alta
- **Entradas:** APIs de plataformas académicas
- **Salidas:** Datos de asistencia sincronizados
- **Criterios de aceptación:** Sincronización bidireccional cada 15 minutos, manejo de errores

RF-014: Integración con Sistema de Nómina

- **Descripción:** El sistema debe impactar inasistencias injustificadas de empleados en el sistema de nómina
- **Prioridad:** Media
- **Entradas:** Datos de asistencia de empleados
- **Salidas:** Descuentos calculados automáticamente
- **Criterios de aceptación:** Integración mensual, validación manual previa a aplicación

3.1.6 Seguridad y Roles

RF-015: Gestión de Roles y Permisos

- **Descripción:** El sistema debe implementar un sistema de roles con permisos diferenciados
- **Prioridad:** Alta
- **Entradas:** Asignación de roles por usuario
- **Salidas:** Acceso controlado según permisos
- **Criterios de aceptación:** Mínimo 4 roles definidos, herencia de permisos

RF-016: Trazabilidad de Cambios

- **Descripción:** El sistema debe registrar todos los cambios realizados por administradores en los registros
- **Prioridad:** Alta
- **Entradas:** Acciones de modificación
- **Salidas:** Log detallado de cambios
- **Criterios de aceptación:** Registro de usuario, fecha, hora y tipo de cambio.

3.2 Requerimientos de Rendimiento

RNF-001: Tiempo de Respuesta

- El sistema debe responder a las solicitudes de registro en máximo 3 segundos en condiciones normales
- Las consultas de historial deben completarse en máximo 5 segundos
- Los reportes deben generarse en máximo 30 segundos

RNF-002: Capacidad de Usuarios Concurrentes

- El sistema debe soportar al menos 2,000 usuarios concurrentes durante horarios pico
- Debe mantener rendimiento estable con hasta 10,000 registros de asistencia por hora

RNF-003: Disponibilidad

- El sistema debe tener una disponibilidad mínima del 99.5% durante horario académico
- Tiempo máximo de inactividad programada: 4 horas mensuales fuera de horario académico

3.3 Requerimientos de Seguridad

RNF-004: Cifrado de Datos

- Todos los datos biométricos deben almacenarse con cifrado AES-256
- Las comunicaciones deben utilizar protocolos HTTPS/TLS 1.3

RNF-005: Autenticación y Autorización

- Implementación de autenticación de doble factor para administradores
- Sesiones de usuario con expiración automática después de 2 horas de inactividad

RNF-006: Protección de Datos Personales

- Cumplimiento con regulaciones locales de protección de datos
- Políticas de retención de datos: máximo 7 años para datos académicos

3.4 Requerimientos de Usabilidad

RNF-007: Interfaz de Usuario

- Interfaz intuitiva que requiera máximo 30 minutos de capacitación para usuarios básicos
- Compatibilidad con dispositivos móviles (responsive design)
- Soporte para al menos español e inglés

RNF-008: Accesibilidad

- Cumplimiento con estándares WCAG 2.1 nivel AA
- Soporte para lectores de pantalla y navegación por teclado

3.5 Requerimientos de Compatibilidad

RNF-009: Plataformas Soportadas

- Aplicación web compatible con Chrome 90+, Firefox 88+, Safari 14+, Edge 90+
- Aplicación móvil para iOS 12+ y Android 8.0+

RNF-010: Hardware

- Funcionamiento en dispositivos con mínimo 2GB RAM y 16GB almacenamiento

- Compatibilidad con lectores biométricos estándar del mercado

3.6 Requerimientos de Mantenimiento

RNF-011: Backup y Recuperación

- Respaldos automáticos diarios de la base de datos
- Tiempo de recuperación objetivo (RTO): máximo 4 horas
- Punto de recuperación objetivo (RPO): máximo 1 hora

RNF-012: Monitoreo

- Logs detallados de todas las transacciones del sistema
- Alertas automáticas por fallos del sistema o rendimiento degradado

4. APÉNDICES

Apéndice A: Glosario de Términos

Términos Técnicos

- **API (Application Programming Interface):** Conjunto de protocolos y herramientas que permiten la comunicación entre diferentes sistemas de software
- **Biometría:** Tecnología de identificación basada en características físicas únicas como huellas dactilares o rasgos faciales
- **Dashboard:** Panel de control visual que muestra métricas e indicadores clave en tiempo real
- **ETL (Extract, Transform, Load):** Proceso de extracción, transformación y carga de datos entre sistemas
- **Geovallado:** Tecnología que utiliza GPS para crear límites virtuales geográficos y validar ubicaciones
- **Trazabilidad:** Capacidad de seguir el historial completo de cambios en los datos con registro de usuario, fecha y tipo de modificación.

Términos de Negocio

- **Ausencia Justificada:** Inasistencia validada por documentación oficial (médica, personal, académica) que no afecta el porcentaje de asistencia

- **Coordinador Académico:** Personal administrativo responsable de supervisar el rendimiento académico y asistencia por facultad o departamento
- **Inasistencia Recurrente:** Patrón de ausencias que supera el 20% en un período de un mes, activando alertas automáticas
- **Porcentaje de Asistencia:** Cálculo automático del ratio entre clases asistidas y clases programadas en un período determinado
- **Registro de Asistencia:** Acto de confirmar la presencia física de una persona en una clase o jornada laboral específica
- **Stakeholder:** Cualquier persona o grupo con interés en el proyecto, incluyendo estudiantes, docentes, administradores y directivos

Términos de Hardware y Dispositivos

- **Lector Biométrico:** Dispositivo especializado que captura y procesa características físicas únicas para identificación
- **Código QR:** Código de barras bidimensional que almacena información y puede ser escaneado por dispositivos móviles
- **GPS (Global Positioning System):** Sistema de navegación satelital que proporciona coordenadas geográficas precisas
- **RFID (Radio Frequency Identification):** Tecnología de identificación que utiliza campos electromagnéticos para identificar tarjetas o chips
- **Tablet Institucional:** Dispositivo móvil proporcionado por la universidad para registro de asistencia en aulas.

Términos de Seguridad

- **Autenticación:** Proceso de verificar la identidad de un usuario antes de otorgar acceso al sistema
- **Cifrado AES-256:** Estándar de cifrado avanzado que utiliza claves de 256 bits para proteger datos sensibles
- **HTTPS/TLS:** Protocolos de comunicación segura que cifran datos transmitidos entre cliente y servidor
- **Rol de Usuario:** Conjunto predefinido de permisos y funcionalidades asignadas a diferentes tipos de usuarios del sistema

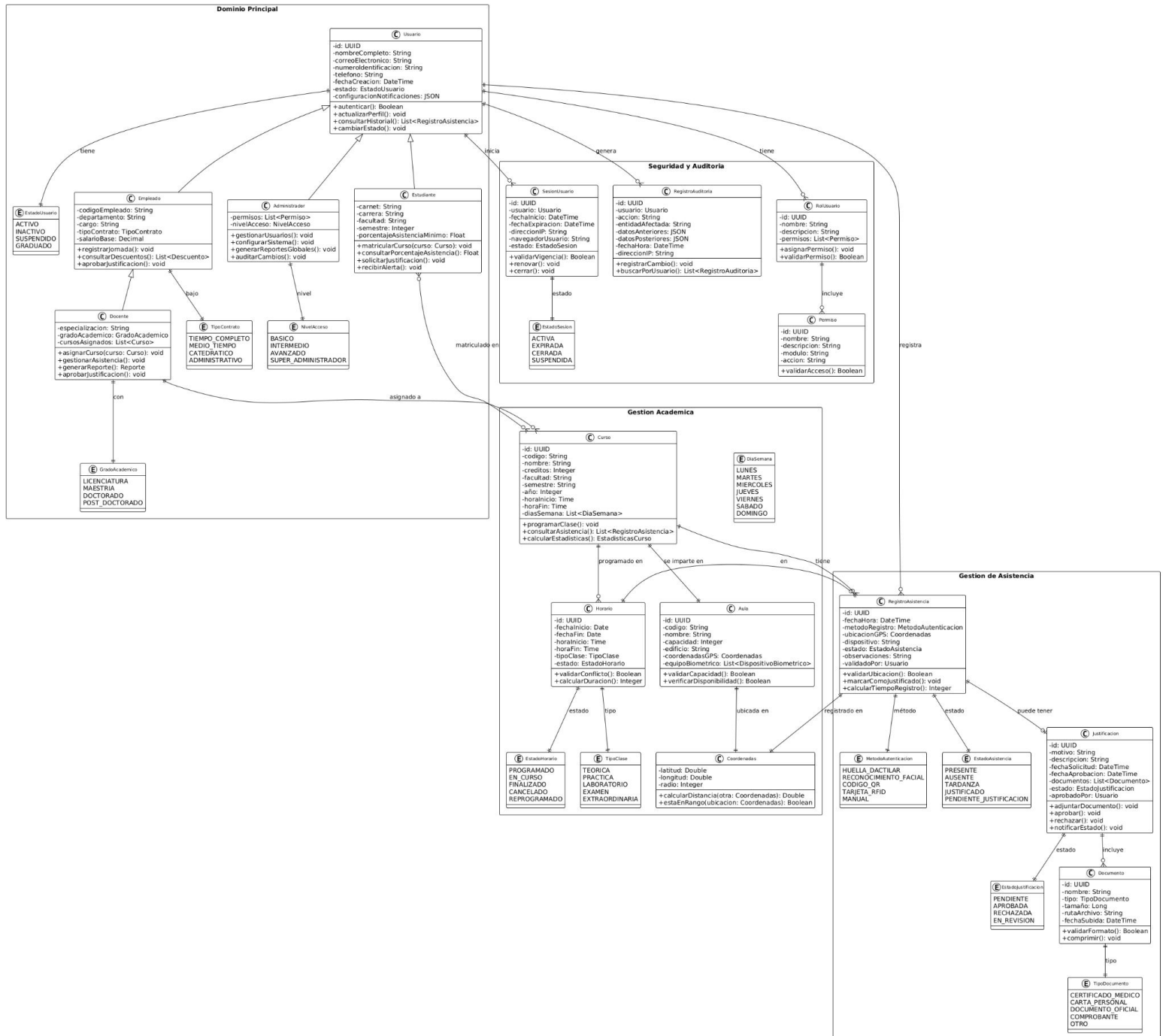
Apéndice B: Casos de Uso Principales

1. **Registro de Asistencia por QR:** Estudiante escanea código QR, sistema valida ubicación y registra asistencia.
2. **Gestión de Justificación:** Estudiante solicita justificación, docente revisa y aprueba con documentación.
3. **Generación de Reporte:** Sistema genera automáticamente reporte semanal y lo envía a coordinadores.

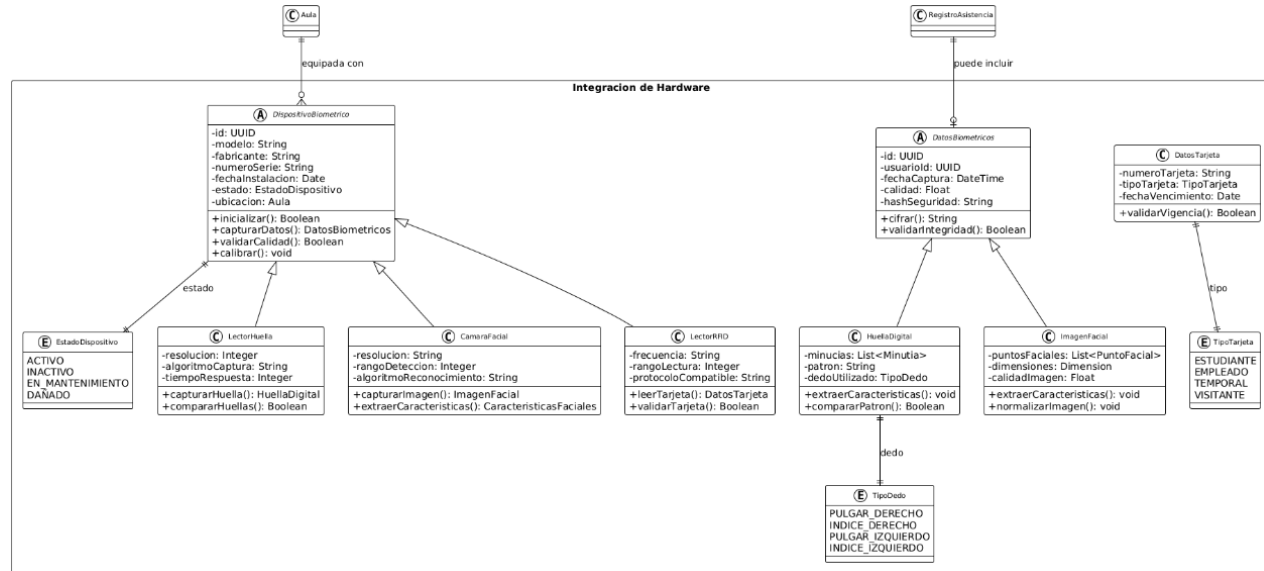
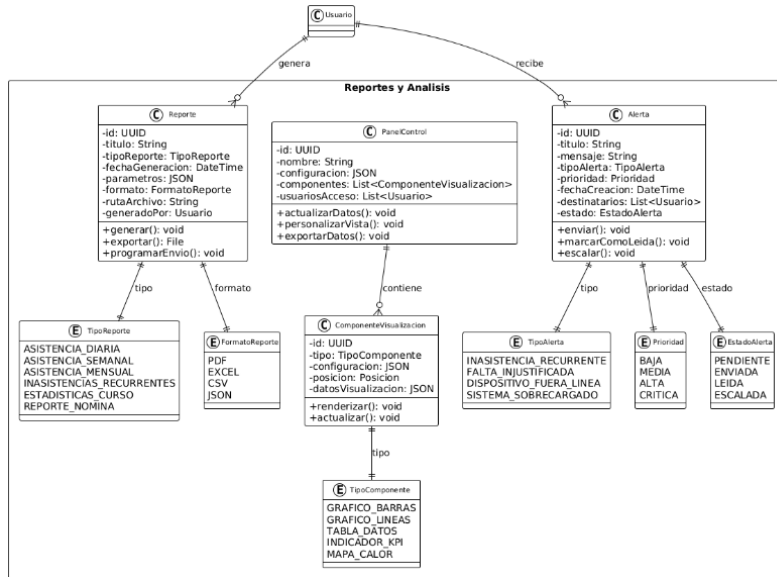
Apéndice C: Matriz de Trazabilidad de requerimientos

Descripción del objetivo	Requerimiento relacionado
Digitalizar registro	RF-001, RF-002, RF-003, RF-004
Facilitar acceso a datos	RF-006, RF-008, RF-012
Generar alertas tempranas	RF-011
Integración con sistemas	RF-013, RF-014
Garantizar seguridad	RF-015, RF-016, RNF-004, RNF-005

Diagrama de Clases y Descripción del Sistema SIRAU



Continuación



DESCRIPCIÓN DEL DIAGRAMA DE CLASES

1. DOMINIO PRINCIPAL

Descripción General

Este dominio contiene las entidades fundamentales del sistema que representan a los actores principales y sus características básicas. Implementa un patrón de herencia donde Usuario actúa como clase base para todas las especializaciones.

Clases Principales

Usuario (Clase Base)

Propósito: Clase abstracta que define las características comunes de todos los usuarios del sistema.

Atributos clave:

- id: Identificador único universal (UUID) para cada usuario
- nombreCompleto: Información personal completa del usuario
- correoElectronico: Canal principal de comunicación
- numeroIdentificacion: Documento de identidad oficial
- estado: Control del ciclo de vida del usuario mediante enum

Métodos principales:

- autenticar(): Validación de credenciales de acceso
- consultarHistorial(): Acceso a registros históricos de asistencia
- cambiarEstado(): Gestión del ciclo de vida del usuario

Estudiante (Hereda de Usuario)

Propósito: Representa a los estudiantes universitarios con sus características académicas específicas.

Atributos específicos:

- carnet: Identificador único estudiantil
- carrera y facultad: Ubicación académica del estudiante
- porcentajeAsistenciaMinimo: Requisito mínimo para aprobar materias

Funcionalidades especiales:

- Matriculación en cursos

- Consulta de porcentajes de asistencia
- Solicitud de justificaciones por ausencias
- Recepción de alertas tempranas

Empleado (Hereda de Usuario)

Propósito: Modelar al personal universitario con características laborales.

Atributos específicos:

- codigoEmpleado: Identificación laboral única
- departamento y cargo: Ubicación organizacional
- tipoContrato: Clasificación laboral (enum)
- salarioBase: Base para cálculos de descuentos

Funcionalidades laborales:

- Registro de jornadas laborales
- Consulta de descuentos por inasistencias
- Aprobación de justificaciones (según cargo)

Docente (Hereda de Empleado)

Propósito: Especialización de empleado con responsabilidades académicas adicionales.

Atributos académicos:

- especializacion: Área de expertise del docente
- gradoAcademico: Nivel de formación académica
- cursosAsignados: Lista de materias bajo su responsabilidad

Responsabilidades específicas:

- Gestión directa de asistencia estudiantil
- Generación de reportes académicos
- Aprobación de justificaciones estudiantiles

Administrador (Hereda de Usuario)

Propósito: Usuario con privilegios administrativos y de configuración del sistema.

Atributos de control:

- permisos: Lista específica de acciones permitidas
- nivelAcceso: Categorización de privilegios administrativos

Capacidades administrativas:

- Gestión completa de usuarios del sistema
- Configuración de parámetros del sistema
- Generación de reportes globales
- Auditoría de cambios del sistema

Enumeraciones del Dominio

EstadoUsuario

Define el ciclo de vida de los usuarios:

- ACTIVO: Usuario operativo normal
- INACTIVO: Usuario temporalmente deshabilitado
- SUSPENDIDO: Usuario con restricciones disciplinarias
- GRADUADO: Estudiante que completó sus estudios

TipoContrato

Clasifica las relaciones laborales:

- TIEMPO_COMPLETO: Empleado de jornada completa
- MEDIO_TIEMPO: Empleado de jornada parcial
- CATEDRATICO: Docente por horas
- ADMINISTRATIVO: Personal no académico

2. GESTIÓN ACADÉMICA

Descripción General

Este módulo modela la estructura académica universitaria, incluyendo cursos, aulas, horarios y elementos de geolocalización necesarios para el registro de asistencia.

Clases Principales

Curso

Propósito: Representa una materia específica en un período académico determinado.

Atributos académicos:

- codigo y nombre: Identificación de la materia
- creditos: Valor académico del curso
- facultad: Unidad académica responsable
- semestre y año: Período de impartición

Atributos temporales:

- horaInicio y horaFin: Franja horaria de clases
- diasSemana: Días de la semana programados

Relaciones importantes:

- Asignación a un docente específico
- Lista de estudiantes matriculados
- Ubicación en aula específica
- Múltiples horarios de clase

Aula

Propósito: Representa los espacios físicos donde se imparten clases y se registra asistencia.

Atributos físicos:

- codigo y nombre: Identificación del espacio
- capacidad: Límite de personas
- edificio: Ubicación dentro del campus

Elementos tecnológicos:

- coordenadasGPS: Ubicación exacta para geovallado
- equipoBiometrico: Dispositivos instalados para registro

Funcionalidades:

- Validación de capacidad disponible
- Verificación de disponibilidad temporal
- Soporte para múltiples dispositivos biométricos

Coordenadas

Propósito: Clase utilitaria para gestión de geolocalización y validación de ubicación.

Atributos geográficos:

- latitud y longitud: Posición GPS exacta
- radio: Área de tolerancia en metros

Métodos geográficos:

- calcularDistancia(): Distancia entre dos puntos
- estaEnRango(): Validación de proximidad

Horario

Propósito: Modelar las sesiones específicas de clase con control temporal.

Atributos temporales:

- fechaInicio y fechaFin: Período de vigencia
- horaInicio y horaFin: Franja horaria específica
- tipoClase: Categorización de la sesión
- estado: Control del ciclo de vida del horario

Funcionalidades:

- Detección de conflictos de horario
- Cálculo automático de duración
- Control de estados (programado, en curso, finalizado)

3. GESTIÓN DE ASISTENCIA

Descripción General

Este módulo contiene el núcleo funcional del sistema SIRAU, modelando el registro de asistencia, las justificaciones y la documentación de soporte.

Clases Principales

RegistroAsistencia

Propósito: Entidad central que almacena cada evento de registro de asistencia con todos sus metadatos.

Atributos de registro:

- fechaHora: Timestamp exacto del registro
- metodoRegistro: Tecnología utilizada (enum)
- ubicacionGPS: Coordenadas del registro
- dispositivo: Identificación del hardware usado

Atributos de control:

- estado: Estado actual del registro (enum)
- observaciones: Notas adicionales
- validadoPor: Usuario que validó el registro

Funcionalidades críticas:

- validarUbicacion(): Verificación de geovallado
- marcarComoJustificado(): Cambio de estado por justificación
- calcularTiempoRegistro(): Métricas de performance

Justificacion

Propósito: Modela el proceso de justificación de ausencias con workflow de aprobación.

Atributos de solicitud:

- motivo y descripcion: Razón de la ausencia
- fechaSolicitud: Timestamp de solicitud
- documentos: Evidencia de soporte

Atributos de aprobación:

- fechaAprobacion: Timestamp de decisión
- estado: Estado del proceso (enum)
- aprobadoPor: Usuario autorizado que decide

Workflow de aprobación:

- adjuntarDocumento(): Carga de evidencia
- aprobar() y rechazar(): Decisiones de autorización
- notificarEstado(): Comunicación automática

Documento

Propósito: Gestiona los archivos de soporte para justificaciones.

Atributos técnicos:

- nombre y tipo: Identificación del archivo
- tamaño: Control de límites de almacenamiento
- rutaArchivo: Ubicación en el sistema de archivos
- fechaSubida: Auditoría temporal

Funcionalidades:

- validarFormato(): Verificación de tipos permitidos
- comprimir(): Optimización de almacenamiento

Enumeraciones Críticas

MetodoAutenticacion

Define las tecnologías de registro disponibles:

- HUELLA_DACTILAR: Biometría dactilar
- RECONOCIMIENTO_FACIAL: Biometría facial
- CODIGO_QR: Código de respuesta rápida
- TARJETA_RFID: Radio frecuencia
- MANUAL: Registro administrativo

EstadoAsistencia

Clasifica los tipos de presencia:

- PRESENTE: Asistencia confirmada
- AUSENTE: No asistencia registrada
- TARDANZA: Llegada fuera de horario
- JUSTIFICADO: Ausencia con justificación aprobada
- PENDIENTE_JUSTIFICACION: Ausencia con solicitud en proceso.

4. INTEGRACIÓN DE HARDWARE

Descripción General

Este módulo modela la integración con dispositivos biométricos y de identificación, implementando un patrón de herencia para diferentes tipos de hardware.

Jerarquía de Clases

DispositivoBiometrico (Clase Abstracta)

Propósito: Clase base que define la interfaz común para todos los dispositivos de captura biométrica.

Atributos comunes:

- modelo, fabricante, numeroSerie: Identificación del hardware
- fechaInstalacion: Control de ciclo de vida
- estado: Monitoreo operacional
- ubicacion: Referencia al aula donde está instalado

Métodos abstractos:

- inicializar(): Configuración inicial del dispositivo
- capturarDatos(): Obtención de datos biométricos
- validarCalidad(): Verificación de calidad de captura
- calibrar(): Ajuste de parámetros operacionales

LectorHuella (Hereda de DispositivoBiometrico)

Propósito: Especialización para dispositivos de captura dactilar.

Atributos específicos:

- `resolucion`: Calidad de captura en DPI
- `algoritmoCaptura`: Tecnología de procesamiento
- `tiempoRespuesta`: Performance del dispositivo

Métodos especializados:

- `capturarHuella()`: Obtención de huella dactilar
- `compararHuellas()`: Validación biométrica

CamaraFacial (Hereda de DispositivoBiometrico)

Propósito: Especialización para reconocimiento facial.

Atributos específicos:

- `resolucion`: Calidad de imagen
- `rangoDeteccion`: Distancia efectiva
- `algoritmoReconocimiento`: Motor de reconocimiento

Métodos especializados:

- `capturarImagen()`: Obtención de imagen facial
- `extraerCaracteristicas()`: Procesamiento de rasgos

LectorRFID (Hereda de DispositivoBiometrico)

Propósito: Especialización para tarjetas de radio frecuencia.

Atributos específicos:

- `frecuencia`: Banda de operación
- `rangoLectura`: Distancia efectiva
- `protocoloCompatible`: Estándares soportados

Modelo de Datos Biométricos

DatosBiometricos (Clase Abstracta)

Propósito: Clase base para almacenamiento seguro de información biométrica.

Atributos de seguridad:

- `usuarioid`: Referencia al propietario
- `fechaCaptura`: Timestamp de obtención

- calidad: Métrica de calidad de captura
- hashSeguridad: Huella digital para integridad

Métodos de seguridad:

- cifrar(): Protección de datos sensibles
- validarIntegridad(): Verificación de alteraciones

HuellaDigital e ImagenFacial

- Especializaciones que contienen los datos específicos de cada tipo biométrico con sus algoritmos de procesamiento correspondientes.

5. REPORTES Y ANÁLISIS

Descripción General

Este módulo implementa el sistema de business intelligence del SIRAUI, proporcionando capacidades de reporte, visualización y alertas automáticas.

Clases Principales

Reporte

Propósito: Generación automatizada de documentos de análisis con múltiples formatos de salida.

Atributos de configuración:

- titulo y tipoReporte: Identificación y categorización
- parametros: Configuración dinámica en formato JSON
- formato: Tipo de archivo de salida
- generadoPor: Auditoría de autoría

Capacidades:

- Generación bajo demanda o programada
- Múltiples formatos de exportación (PDF, Excel, CSV)
- Programación de envíos automáticos

PanelControl

Propósito: Dashboard interactivo para visualización en tiempo real de métricas del sistema.

Atributos de personalización:

- configuración: Layout y preferencias en JSON
- componentes: Lista de widgets de visualización
- usuariosAcceso: Control de permisos de visualización

Funcionalidades:

- Actualización automática de datos
- Personalización por usuario
- Exportación de datos visualizados

ComponenteVisualizacion

Propósito: Widgets individuales para diferentes tipos de representación gráfica.

Tipos soportados:

- Gráficos de barras y líneas
- Tablas de datos
- Indicadores KPI
- Mapas de calor

Alerta

Propósito: Sistema proactivo de notificaciones automáticas basadas en reglas de negocio.

Atributos de clasificación:

- tipoAlerta: Categorización del evento
- prioridad: Nivel de urgencia
- destinatarios: Lista de usuarios a notificar

Funcionalidades:

- Envío multicanal (email, SMS, push)
- Escalación automática
- Tracking de estado de lectura

6. SEGURIDAD Y AUDITORÍA

Descripción General

Este módulo implementa los mecanismos de seguridad y control de acceso necesarios para cumplir con regulaciones de protección de datos.

Clases Principales

SesionUsuario

Propósito: Gestión del ciclo de vida de sesiones de usuario con control de seguridad.

Atributos de sesión:

- fechaInicio y fechaExpiracion: Control temporal
- direccionIP y navegadorUsuario: Información de contexto
- estado: Monitoreo de actividad

Funcionalidades de seguridad:

- Validación de vigencia automática
- Renovación de sesiones
- Cierre seguro y limpieza

RoUsuario y Permiso

Propósito: Sistema granular de control de acceso basado en roles.

METODOLOGÍA DE DESARROLLO DE DEFINICIÓN DE ROLES

Metodología seleccionada: Metodología Híbrida (SCRUM + WATERFALL)

¿Por qué?

Para el desarrollo del sistema de registro de asistencia universitaria (SIRAU), se optó por una metodología híbrida, la cual combina lo mejor del enfoque (cascada) con prácticas ágiles (SCRUM), para las fases críticas de planificación e infraestructura, con scrum para el desarrollo iterativo de funcionalidades. Esta decisión se fundamenta en las características específicas del proyecto y el contexto universitario donde será implementado.

El SIRAU presenta un nivel de complejidad técnica moderado-alto, por lo que requiere una planificación arquitectónica rigurosa inicial, pero también flexibilidad para ajustar implementaciones según resultados de pruebas o retroalimentación de parte de los usuarios.

El proyecto involucra múltiples tipos de usuarios dentro del sistema:

- **Estudiantes:** Requieren interfaces intuitivas y procesos rápidos
- **Docentes:** Necesitan herramientas de gestión eficientes y confiables
- **Administradores:** Demandan reportes detallados y controles de seguridad
- **Coordinadores académicos:** Necesitan análisis de datos para toma de decisiones

Esta diversidad de usuarios hace necesario un enfoque iterativo que permita validación temprana y ajustes basados en retroalimentación de parte de ellos cuando se realicen pruebas.

La metodología híbrida seleccionada para SIRAU optimiza las fortalezas de ambos enfoques metodológicos, proporcionando la estabilidad y planificación requerida por el contexto universitario, mientras mantiene la flexibilidad necesaria para desarrollar un sistema centrado en el usuario. Esta decisión se alinea con las mejores prácticas de la industria para proyectos de complejidad similar y maximiza las probabilidades de éxito tanto técnico como de adopción del sistema.

La combinación de una base sólida establecida mediante Waterfall y un desarrollo iterativo guiado por Scrum permitirá entregar un sistema robusto, seguro y altamente usable que cumpla con los objetivos institucionales y las expectativas de todos los stakeholders involucrados.

Descripción detallada del uso de la metodología

La metodología híbrida implementada para SIRAU combina Waterfall (cascada) para fases críticas de planificación e infraestructura, seguido de Scrum para desarrollo iterativo. Esta aproximación se estructura en dos fases principales con transición controlada entre ambas.

Duración Total del Proyecto: 9 meses

Fase 1 (Waterfall): 3 meses - Fundación y Arquitectura

Objetivos de la fase

- Establecer arquitectura técnica sólida y escalable
- Validar viabilidad técnica de componentes críticos
- Definir especificaciones completas de integración
- Configurar infraestructura base y protocolos de seguridad
- Obtener aprobaciones institucionales necesarias

Estructura de Trabajo

Mes 1: Análisis y Diseño

Semanas 1-2: Análisis detallado de requerimientos y refinamiento de especificaciones

Semanas 3-4: Diseño arquitectónico del sistema y selección de tecnologías

Mes 2: Investigación e Infraestructura

Semanas 5-6: Investigación y pruebas de hardware biométrico

Semanas 7-8: Configuración de infraestructura y análisis de integraciones

Mes 3: Validación y Preparación

Semanas 9-10: Prototipos técnicos y validación de conceptos

Semanas 11-12: Documentación final y preparación para fase ágil

Ceremonias y Reuniones Waterfall

Weekly Architecture Reviews (Viernes 2:00-4:00 PM)

Participantes: Todo el equipo

Objetivo: Revisar avances arquitectónicos y tomar decisiones técnicas

Entregables: Actas de decisiones técnicas, actualización de diagramas

Stakeholder Meetings (Bi-semanales, Martes 10:00-11:00 AM)

Participantes: Product Owner, Tech Lead + Stakeholders universitarios

Objetivo: Validar especificaciones y obtener aprobaciones

Entregables: Documentos de aprobación, cambios validados

Research & Spike Sessions (Diarias 1:00-2:00 PM)

Participantes: Desarrolladores y DevOps según necesidad

Objetivo: Investigación técnica profunda y pruebas de concepto

Entregables: Reportes de investigación, prototipos técnicos

Fase 2 (Agile/Scrum): 6 meses - Desarrollo Iterativo

Desarrollar funcionalidades de forma iterativa e incremental

Validar continuamente con usuarios reales

Mantener alta calidad de código y testing

Entregar valor de negocio temprano y frecuente

Facilitar adaptación a cambios y nuevos requerimientos

Estructura de Sprints

Configuración de Sprints:

Duración: 2 semanas por sprint

Total, de Sprints: 12 sprints

Capacity Planning: 40 horas/persona por sprint

Velocidad objetivo: 80-100 story points por sprint (a partir del Sprint 3)

Sprint Planning Detallado

Sprint 1-2: Fundación de Desarrollo (Mes 4)

Configuración de pipelines CI/CD

Implementación de autenticación básica

Desarrollo de registro por código QR

Setup de testing automatizado

Sprint 3-4: Funcionalidades Core (Mes 4-5)

Validación por geolocalización (GPS)

Panel básico para estudiantes

Panel básico para docentes

Base de datos de usuarios y registros

Sprint 5-6: Integraciones Biométricas (Mes 5-6)

- Integración con lectores de huella dactilar
- Implementación de reconocimiento facial
- Soporte para tarjetas RFID
- Optimización de performance de autenticación

Sprint 7-8: Gestión y Reportes (Mes 6-7)

- Sistema de justificaciones
- Reportes automáticos semanales
- Dashboards básicos para coordinadores
- Alertas por inasistencia recurrente

Sprint 9-10: Integraciones Externas (Mes 7-8)

- Sincronización con Moodle/Blackboard
- Integración con sistema de nómina
- APIs para sistemas terceros
- Mejoras de seguridad avanzadas

Sprint 11-12: Refinamiento y Despliegue (Mes 8-9)

- Testing integral y corrección de bugs
- Optimización de performance
- Capacitación de usuarios
- Despliegue en producción

DEFINICIÓN Y ROLES DENTRO DEL EQUIPO DE TRABAJO

Integrantes	Rol principal	Integrante del equipo	Perfil y responsabilidad generales
Tech Lead / Arquitecto de Software	Liderazgo técnico y diseño arquitectónico	Walter Alexander Galicia Gertrudis	<ul style="list-style-type: none"> - Diseña la arquitectura general del sistema (backend, frontend, integración). - Toma decisiones tecnológicas clave. - Supervisa estándares de codificación. - Actúa como mentor técnico.
Product Owner / Analista de Negocio	Gestión de producto y requerimientos	Gabriel Ernesto Días Galdámez	<ul style="list-style-type: none"> - Recoge necesidades del usuario. - Prioriza funcionalidades con base en valor de negocio. - Interactúa con stakeholders institucionales.
Frontend UI/UX Developer	Desarrollo de interfaz y experiencia de usuario	Franklin Imanol Ramírez Gómez	<ul style="list-style-type: none"> - Desarrolla aplicaciones cliente responsivas. - Trabaja en React, HTML/CSS, y frameworks UI. - Implementa la lógica de interacción del usuario. - Garantiza usabilidad y accesibilidad.
Scrum master	Liderar Daily Standups, Sprint Planning, Sprint Reviews y Retrospectivas	Ricardo Adan Patino Hernández	<p>Moderar reuniones diarias (15 min máximo)</p> <p>Facilitar la planificación de sprints y estimación de tareas</p> <p>Organizar retrospectivas y implementar acciones de mejora</p> <p>Coordinar con stakeholders externos para resolver impedimentos</p>
DevOps / QA Engineer	Automatización, despliegue y pruebas	Alexandra Quinteros Carcamo	<p>Configura la infraestructura CI/CD.</p> <ul style="list-style-type: none"> - Automatiza pruebas funcionales y de seguridad. - Supervisa la calidad del sistema. - Monitorea versiones y despliegues.

CALENDARIO DE TRABAJO INICIAL

Fase	Actividad	Responsable(s)	Comienzo	Fin	Dur. (semanas)
Análisis y Diseño	Levantar requisitos, análisis de procesos	Product Owner, Tech Lead	10/06/25	24/06/25	2
	Priorizar requisitos junto a stakeholders	Product Owner	10/06/25	24/06/25	2
	Diseño de arquitectura, base de datos, UI	Tech Lead, Product Owner, Full-Stack, Backend	25/06/25	05/07/25	2
	Investigación de métodos de autenticado (QR, huella, facial)	Tech Lead, Backend	06/07/25	20/07/25	2
	Aprobaciones institucionalizadas	Product Owner, Tech Lead	21/07/25	04/08/25	2
	Validar diseño de seguridad, roles y permisos	Tech Lead, Product Owner	21/07/25	04/08/25	2
Desarrollo	Implementar Registro de Asistencia (QR, Huella, Tarjeta)	Tech Lead, Backend, DevOps / QA Engineer	05/08/25	20/09/25	6
	Implementar Panel de Control (Visualización, Justificar ausencias)	Tech Lead, Full-Stack, Product Owner	21/09/25	31/10/25	6
	Implementar Reportes y Análisis (Dashboards, Alertas)	Tech Lead, Full-Stack, Backend	01/11/25	15/12/25	6

	Integrar nómina, Seguridad, Permisos	Tech Lead, DevOps / QA Engineer, Backend	16/12/25	31/01/26	6
	Implementar geovallado, autenticaciones avanzadas, roles de administrador	Tech Lead, Backend, DevOps / QA Engineer	16/12/25	31/01/26	6
Pruebas y Ajustes	Pruebas Unitarias, Integrales, Aceptación	DevOps / QA Engineer, Tech Lead, Product Owner	01/02/26	28/02/26	4
	Corrección de Errores y Ajustes	Tech Lead, DevOps / QA Engineer	01/03/26	15/03/26	2
	Pruebas de seguridad, carga y estrés	Tech Lead, DevOps / QA Engineer	01/03/26	15/03/26	2
Despliegue y Capacitación	Despliegue en Servidores de Producción	DevOps / QA Engineer, Tech Lead	16/03/26	31/03/26	2
	Capacitación a Usuarios (estudiantes, maestros, administrativos)	Product Owner, Tech Lead	01/04/26	15/04/26	2
	Generar Manual de Usuario, Guía de Administración	Product Owner, Tech Lead	01/04/26	15/04/26	2
Mantenimiento y Soporte	Soporte técnico, actualización, seguimiento de métricas	DevOps / QA Engineer, Tech Lead	16/04/26	en adelante	continuo

	Aplicar parches de seguridad, nuevas versiones	Tech Lead, DevOps / QA Engineer	16/04/26	en adelante	continuo
	Monitorear rendimiento, implementar optimizaciones	Tech Lead, DevOps / QA Engineer	16/04/26	en adelante	continuo
	Atender solicitudes de mejora o problemas reportados	Tech Lead, Product Owner, DevOps / QA Engineer	16/04/26	en adelante	continuo