



FACULTAD DE CIENCIAS E INGENIERÍA CARRERA DE INGENIERÍA DE SOFTWARE

TEMA:

Requerimientos del Sistema

AUTOR:

Edgar Adrián Badillo Pérez

ASIGNATURA:

Introducción a la Ingeniería de Software

DOCENTE:

Guevara Serrano Jorge Dumar

FECHA DE ENTREGA:

Fecha establecida

PERIODO:

Primer Nivel

MILAGRO-ECUADOR

Índice del Proyecto

I. Requerimientos del Sistema de Control de Asistencia

- A. Requerimientos Funcionales (RF)
- B. Requerimientos No Funcionales (RNF)

II. Documento de Requerimientos de Software (DRS)

- A. Propósito y Alcance del Sistema
- B. Descripción General del Sistema y de los Usuarios
- C. Lista de Requerimientos
- D. Criterios de Aceptación

III. Diseño y Correspondencia de Casos de Prueba

- A. Casos de Prueba Unitarios y de Validación

IV. Reflexión Final sobre la Validación

I. Requerimientos del Sistema de Control de Asistencia

El objetivo de esta sección es definir lo que el Sistema debe hacer (Funcionales) y las características que debe cumplir (No Funcionales).

A. Requerimientos Funcionales

Los requerimientos funcionales describen las acciones o funciones específicas que el sistema debe poder realizar para satisfacer las necesidades del usuario.

ID	Requerimiento Funcional
RF1	Registro de Asistencia: El Sistema debe permitir al profesor registrar la asistencia de un estudiante a una clase específica (marcar presente, ausente o con justificación).
RF2	Selección de Curso: El Sistema debe permitir al profesor seleccionar el curso o grupo específico para el cual desea tomar la asistencia.
RF3	Visualización de Historial Individual: El Sistema debe permitir al profesor o al estudiante (con credenciales) visualizar el historial completo de asistencia de ese estudiante.
RF4	Modificación de Registro: El Sistema debe permitir al profesor modificar una entrada de asistencia ya registrada (por ejemplo, corregir un error de marcado o cambiar "ausente" a "justificado").
RF5	Generación de Reportes: El Sistema debe generar un reporte simple que muestre el porcentaje total de asistencia de cada estudiante en un curso al final del semestre o periodo.

B. Requerimientos No Funcionales

Los requerimientos no funcionales describen las cualidades, restricciones o características de rendimiento que debe tener el sistema.

ID	Requerimiento No Funcional
RNF1	Seguridad de Acceso: Solo los profesores y administradores autorizados deben poder acceder a la función de registro y modificación de asistencia mediante un proceso de inicio de sesión con contraseña.
RNF2	Rendimiento de Registro: El proceso de guardar la asistencia de toda una clase (alrededor de 30 estudiantes) no debe tardar más de 5 segundos desde que el profesor presiona el botón de guardar.
RNF3	Usabilidad Móvil: La interfaz del Sistema para tomar la asistencia debe ser fácil de usar y compatible con dispositivos móviles, ya que los profesores podrían usar una tableta o celular en el aula

II. Documento de Requerimientos de Software

Aquí va la segunda parte: nuestro DRS. Esto es como el plano de nuestra aplicación de asistencia, donde decimos para qué sirve y quién la va a usar.

A. Propósito y Alcance del Sistema

El propósito principal de esta aplicación es que los profesores puedan registrar y revisar la asistencia de los estudiantes a sus clases de una forma sencilla y rápida, dejando de usar listas de papel.

El alcance de lo que haremos es específico para el control de asistencia (Caso 5). Incluye: registrar asistencias, ver el historial individual, cambiar un registro si hubo un error (RF4) y sacar reportes simples de porcentaje de asistencia (RF5).

No vamos a incluir en esta aplicación funciones extra como la gestión de notas académicas, el control de inventario de la institución o la programación de horarios complejos.

B. Descripción General del Sistema y de los Usuarios

La aplicación de asistencia será una herramienta digital que debe ser fácil de usar y funcionar bien en el celular o tableta (RNF3).

Cuando el profesor entre a la app con su usuario (RNF1), podrá ver sus cursos, seleccionar la clase del día y marcar si el estudiante está Presente, Ausente o Justificado (RF1). Toda esa información se guarda en la base de datos para que no se pierda.

Los principales que usarán este sistema son:

Profesores: Son los que usan la app en el aula para registrar y modificar la asistencia.

Estudiantes: Solo podrán entrar a consultar su propio historial de asistencia (RF3).

Administradores: Son los encargados de configurar los cursos y subir la lista inicial de todos los estudiantes.

C. Lista de Requerimientos

Todos los requerimientos importantes funcionales (RF1 al RF5) y los no funcionales (RNF1 al RNF3) están listados en la Sección I.

D. Criterios de Aceptación

Estos son los puntos que hay que revisar para que estemos seguros de que la aplicación funciona bien y hace lo que nos pidieron:

Registro Confiable: Un profesor debe poder registrar la asistencia de una clase completa y, si necesita cambiar un estado (por ejemplo, cambiar "ausente" a "justificado"), el sistema debe permitirlo sin perder los datos originales (basado en RF1 y RF4).

Velocidad de Guardado: Cuando el profesor finalice el registro de una clase (digamos, 35 estudiantes) y presione 'Guardar', la aplicación no puede tardar más de 5 segundos en confirmar que el registro se ha guardado (basado en RNF2).

Cálculo Correcto: Si pedimos el reporte final (RF5), este tiene que calcular correctamente el porcentaje. Si un estudiante vino a 10 de 20 clases, el reporte debe decir 50%.

Bloqueo de Seguridad: Si alguien intenta acceder a la toma de asistencia sin haber puesto su usuario y contraseña correctamente, la aplicación tiene que bloquear el acceso sí o sí (basado en RNF1).

III. Diseño y Correspondencia de Casos de Prueba

Antes de que un sistema se entregue, hay que asegurarse de que realmente hace lo que dijimos que haría, por eso, esta tabla simula los pasos que seguiríamos para verificar los requerimientos funcionales y no funcionales.

A. Casos de Prueba Unitarios y de Validación

ID Caso	Tipo de Prueba	Requerimiento Asociado	Datos de Entrada	Resultado Esperado	Resultado Obtenido
CP01	Unitario	RF1	Un Profesor selecciona el curso "Cálculo". Selecciona al Estudiante "Ana Soto". Marca el estado: Presente	El sistema debe mostrar un mensaje de confirmación y el estado del estudiante debe cambiar a "Presente".	Éxito.
CP02	Unitario	RF4	El profesor revisa la asistencia de ayer. La asistencia de "Carlos Ruiz" está marcada como "Ausente". El profesor cambia el estado a "Justificado".	El sistema debe permitir el cambio y actualizar el registro, manteniendo un registro de quién y cuándo se hizo la modificación	Éxito.
CP03	Unitario	RNF1	Intento de iniciar sesión con el nombre de usuario "ProfesorX" y la contraseña: "Incorrecta123".	El sistema debe mostrar un mensaje de error ("Credenciales incorrectas") y denegar el acceso al sistema.	Éxito.
CP04	Validación	RNF2	Profesor registra la asistencia de una clase de 35 estudiantes y pulsa el botón 'Guardar'. Se mide el tiempo de respuesta.	El proceso de guardado y confirmación en la pantalla debe completarse en un tiempo máximo de 5 segundos.	Éxito.
CP05	Validación	RF5	Se genera el reporte final del curso "Programación". El Estudiante "María Gómez" tiene 10 asistencias registradas de un total de 20 clases.	El reporte generado debe mostrar el porcentaje de asistencia de María Gómez como 50%.	Éxito.

IV. Reflexión Final sobre la Validación

Entiendo que el trabajo no termina cuando defino los requerimientos. La parte más importante es validar que el sistema que propongo realmente hará lo que dije.

La razón principal por la que se incluye los casos de prueba es precisamente para demostrar cómo las pruebas ayudan a validar los requerimientos definidos. Validar significa confirmar que el sistema de Control de Asistencia es el correcto y cumple con las necesidades del profesor.

Por ejemplo, al realizar la prueba de rendimiento (CP04), no solo se verifica que la función de guardar asistencia exista, sino que se comprueba que se cumple el RNF2, asegurando que el proceso no tarde más de 5 segundos. Si esa prueba falla, sé exactamente que ese requerimiento no funcional no se está cumpliendo, y que el sistema necesita una corrección.

Las pruebas son un mecanismo para demostrar la consistencia entre lo que escribí en la Sección I y el funcionamiento real esperado del sistema. Esto me da la certeza de que la aplicación será útil y confiable para la institución.