

# Sistema de Control de Asistencia

APLICACIONES DISTRIBUIDAS

El presente documento es aprobado y constatado por las personas directamente involucradas, mostradas a continuación:

Estudiante de Ingeniería de Sistemas:

**Luis Chitacapa**

---

Estudiante de Ingeniería de Sistemas:

**Luis Tito**

---

Estudiante de Ingeniería de Sistemas:

**Christian Espinoza**

---

Estudiante de Ingeniería de Sistemas:

**Rodrigo Nacipucha**

---

Estudiante de Ingeniería de Sistemas:

**Cristian Cabrera**

---

## **1. INTRODUCCIÓN**

La falta de control de asistencia al cuerpo de docentes como del alumando es uno de los problemas con los que se enfrentan las instituciones educativas, esto se da mayormente en la educación superior, por ello el objetivo de este proyecto es automatizar el control de asistencia docente/alumnos mediante la creación de un sistema informático para poder manejar la información de una manera más óptima y así no se vea afectada las instituciones educativas.

Al referirnos de la automatización de control de asistencia docente/alumno, las instituciones educativas trata de mejorar la calidad de enseñanza mediante la verificación del cumplimiento de actividades, objetivos, entre otros, además de la puntualidad y responsabilidad en cada uno de ellos, para de esta manera fomentar tanto a los docentes como al alumnado valores.

En el proceso de control de asistencia del docente/alumno se asegura la disponibilidad y confiabilidad de la información que requiera las instituciones educativas en un momento determinado y así poder obtener reportes o tomar decisiones oportunas para que estas ausencias no se vuelvan a repetir, y por tanto el docente/alumno, al salir exento siga repitiendo estas inasistencias cada vez más a menudo.

## **2. OBJETIVOS**

### **2.1. Objetivo General:**

Diseñar y desarrollar un software el cual permita registrar la asistencia docentes/alumnos por cercanía geográfica, obtener reportes y realizar encuestas o consultas hechas por el Jefe de Área.

### **2.2. Objetivos Específicos:**

- Realizar un registro de asistencia docente/alumno que sea asistido por cercanía geográfica.
- Emitir reportes de asistencia docente/alumno para verificar las inasistencias continuas.
- Verificar el control de cumplimiento de actividades, objetivos y resultado de aprendizaje.
- Diseñar la interfaz de la aplicación de escritorio y de la página web.
- Codificar el sistema de control de asistencia.
- Realizar las pruebas y comprobar que el sistema esté funcionando correctamente.

## **3. ALCANCE**

Este sistema de automatización tiene como propósito realizar un seguimiento de la asistencia de docentes/alumnos para comprobar si están cumpliendo con los horarios

establecidos por las autoridades de la institución educativa, por ende se realizará un registro académico de la institución educativa, también contará de los siguientes módulos:

✓ **Módulo Docente**

En este módulo se registrará la información general de los docentes, se asignará materias con su respectivo horario.

✓ **Módulo Alumno**

Controlará la información de alumnos, así también nos permitirá ingresar datos de nuevos alumnos con sus datos personales.

✓ **Módulo Consultas**

Se encargará de generar los reportes o consultas que sean solicitadas por el Jefe de Área de la institución educativa.

#### **4. Características de los usuarios**

<b>Tipo de usuario</b>	Estudiante
<b>Formación</b>	Primaria
<b>Actividades</b>	Leer y Escribir

<b>Tipo de usuario</b>	Docente
<b>Formación</b>	Tercer Nivel
<b>Actividades</b>	Facilitar el control de asistencia

#### **Restricciones**

- Interfaz para ser usada vía web y aplicativo móvil.
- Lenguajes y tecnologías en uso: HTML, JAVA.PHP, JEE.
- El sistema debe ser capaces de realizar un estricto control de asistencia.
- El sistema se diseñará según un modelo de n-capas
- El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma o del lenguaje de programación.

#### **Suposiciones y dependencias**

- Se asume que los requisitos aquí descritos son estables
- Los equipos en los que se vaya a ejecutar el sistema deben cumplir los requisitos antes indicados para garantizar una ejecución correcta de la misma.

#### **5. Requerimientos**

##### **5.1. Requerimientos funcionales:**

- ✓ Autenticación de los docentes.

- ✓ Realizar administración de asistencia
- ✓ Elaborar un reporte.
- ✓ Registrar a los estudiantes y a docentes.
- ✓ Realizar la nómina de docentes y estudiantes.
- ✓ Realizar el control de asistencia geológica.
- ✓ Crear perfil de docentes.
- ✓ Crear perfil de estudiantes.
- ✓ Hacer informes de asistencias de estudiantes.
- ✓ Hacer informe de asistencia de docentes.

## **5.2. Requerimientos no funcionales:**

- ✓ Facilidad de mantenimiento.
- ✓ Fiabilidad.

## **6. MARCO TEÓRICO**

### **Sistema de Bases de datos**

Es un conjunto de recursos que nos ayudan a manejar grandes cantidades de datos, los sistemas de bases de datos deben tener plantearse las siguientes características:

- Independencia de datos. – esto se refiere a que los datos no deben dependa del programa, sino cualquier aplicación pueda hacer uso de dichos datos.
- Reducción de redundancia. – la redundancia no es más que la duplicación de los datos, reduciendo esto tenemos la posibilidad de tener más cantidad de almacenamiento y se evita inconsistencias entre datos. La inconsistencia se da cuando tenemos datos contradictorios.
- Seguridad. – Un sistema de base de datos debe permitirnos tener un control de seguridad de los datos.

### **MySQL**

MySQL es un sistema para la gestión de base de datos relacional (RDBMS) que tiene código abierto, y está basado en un lenguaje de consultas SQL. En otras palabras, MySQL no es más que una aplicación que permite gestionar archivos llamados desde la base de datos.

Como MySQL es una base de datos relacional almacena y organiza la información en múltiples tablas. MySQL fue escrito en c y c++, y se puede adaptar a diferentes entornos de desarrollo, lo cual le permite comunicarse con lenguajes de programación como son: java, PHP, perl entre otros. Y puede correr en cualquier sistema operativo.

Para las empresas es necesario utilizar esta aplicación para poder administrar de una forma ordenada sus datos que pueden ser: datos de clientes, datos de empleados, datos de proveedores, etc.

### **Eclipse**

No es más que un entorno que posee un conjunto de herramientas de programación de código abierto y es multiplataforma. Se la utiliza para el desarrollo de aplicaciones de cliente enriquecido.

Esta aplicación es una potente y completa plataforma para programación, para el desarrollo y compilación de elementos tan variados como sitios web. Programas hechos en c++ o aplicaciones javas. Eclipse también es una comunidad de usuarios , extendiendo constantemente las áreas cubiertas.

### **Balsamiq**

Es una herramienta que nos va a permitir crear prototipos de nuestros proyectos, con esto logramos hacer un boceto, dibujo rápido o sketches para tener una imagen visual de cómo va a quedar nuestra interfaz. Esto nos va a permitir tener una idea de cómo va a quedar nuestro proyecto ya cuando lo estemos programando, donde podemos crear estructuras de una interfaz a lo que denominamos wireframing.

### **JBoss**

JBoss es un servidor de aplicaciones J2EE de código abierto implementado en Java puro. Al estar basado en Java, se puede utilizar en cualquier sistema operativo para el que esté disponible Java. Los principales desarrolladores trabajan para una empresa de servicios.

### **Servidor de aplicaciones Java EE**

Una característica fundamental de la plataforma Java EE es la existencia y la necesidad de utilizar servidores Java EE.

Este es un servidor de aplicaciones que implementa las APIs de la plataforma Java EE y provee de los servicios estándar de Java EE. Los servidores Java EE son llamados, a veces, servidores de aplicaciones, porque permiten obtener datos de las aplicaciones cliente, del mismo modo que un servidor web sirve páginas web a un navegador. Un servidor Java EE contiene diferentes tipos de componentes, los cuales se corresponden con cada una de las capas de una aplicación multicapa.

### **EJB (ENTERPRISE JAVA BEANS)**

La especificación EJB 3.0 introduce un modelo de desarrollo simplificado que reduce la complejidad de los enterprise beans al mismo tiempo que incorpora muchas de las ideas encontradas en tecnologías afines. Las características más relevantes del modelo son:

- ✓ XML y Anotaciones
- ✓ Inyección de Dependencia
- ✓ Interceptores
- ✓ Implementación de POJOS
- ✓ Uso inteligente de Defaults
- ✓ Lenguaje de Consulta Simplificado

## **Java Server Faces**

JavaServer Faces (JSF) es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF es un framework de interfaz gráfica de usuario para aplicaciones J2EE, está basado en el patrón MVC (Modelo Vista Controlador) el cual ha sido diseñado para mejorar la productividad de los desarrolladores, al utilizar un conjunto de componentes reutilizables en la construcción de un interfaz de usuario.

## **XHTML**

XHTML (eXtensible HyperText Markup Language) XHTML es la mezcla de HTML con un XML valido. Es un lenguaje de programación más técnico, es decir este no soporta errores, se puede considerar una versión más estricta y limpia de HTML, pero la ventaja del mismo es que se hace más fácil su mantenimiento debido a que tiene un estándar de programación.

## **7. PLANTEAMIENTO**

Se realizará la adaptación de la metodología Scrum en el proyecto q se presentará en donde se realizará un marco de trabajo mediante el cual se aplicará procesos y técnicas de la Ingeniería de Software.

## **8. FASES METODOLOGICAS DE DESARROLLO**

- Se realizará el análisis de requisitos con el fin de obtener las necesidades puntuales. Actividades:
  - Identificación de Usuario
  - Planeación Y ejecución de reuniones para especificación de requerimientos.
- Se especificará los casos de uso del desarrollo de las funcionalidades que conforman la aplicación. Actividades:
  - El diseño de los casos de uso se realizará mediante la especificación de cada uno de los requisitos previamente establecidos.
- Se creará los prototipos de desarrollo de la aplicación planteada.
  - Validación de los posibles componentes externos que se relacionaran con el proyecto.
  - Definición de los campos, tipos de datos, llaves primarias, y descripción de los mismos con el fin de diseñar el modelo de base de datos.
- Se realizará pruebas a los procesos o actividades desarrolladas utilizando metodología de pruebas ágil.
  - Empezaremos con las pruebas de exploración a medida q se vaya completando el desarrollo.
  - Planeación y ejecución de los casos pruebas funcionales.

## 9. CRONOGRAMA

A continuación, se presenta el cronograma empleado para la realización del proyecto. En donde se observa los días de inicio, duración de cada sprint, tipos de tareas, conformación del equipo de trabajo y ejecución de tareas.

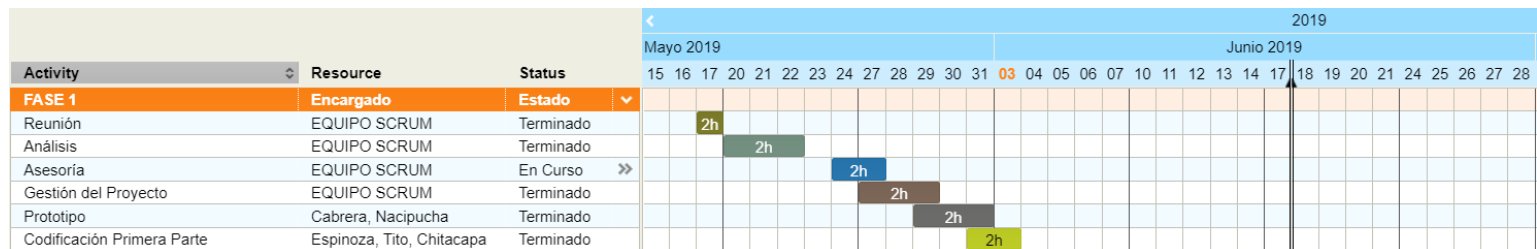


Figura 1. Cronograma de Actividades - Fase 1

PROYECTO				
# SPRINT	Inicio	Hora	Días	Jornada
1	17/05/2019	18:00 a 20:00	--	Vespertina
2	22/05/2019	16:00 a 18:00	5	Vespertina
3	24/05/2019	15:00 a 17:00	2	Vespertina
4	27/05/2019	14:00 a 16:00	3	Vespertina
5	29/05/2019	16:00 a 18:00	2	Vespertina
6	31/05/2019	14:00 a 16:00	2	Vespertina
7	31/05/2019	18:00 a 20:00	3	Vespertina
8	03/06/2019	15:00 a 17:00	--	
TAREAS				
TIPOS	ESTADOS	EQUIPO		
Reunión	Terminada	EQUIPO SCRUM		
Análisis	Terminada	EQUIPO SCRUM		
Asesoría	En curso	EQUIPO SCRUM		
Gestión de proyecto	Terminada	EQUIPO SCRUM		
Prototipo	Terminada	Rodrigo Nacipucha		
Codificación Primera parte	Terminada	Cristian Espinoza Luis Tito Luis Chitacapa		
Codificación Intermedia	En curso	Cristian Espinoza Luis Tito Luis Chitacapa		
Pruebas	En progreso	EQUIPO SCRUM		

## 10. APLICACIÓN

Para el desarrollo del proyecto se aplicará la metodología ágiles que facilita la forma de trabajar de las personas implicadas en el proyecto.

- Los valores que se mantendrá dentro del SCRUM TEAM son: Colaboración, Respeto en el equipo, Responsabilidad y auto disciplina, Centrarse en la Tarea, Calidad de trabajo.



## 11. FASES EJECUTABLES: ROLES

- Product Owner: Todos los involucrados o representantes en el proyecto (EQUIPO SCRUM).
- SCRUM MASTER: Encargado de liderar el equipo con el objetivo de llevarlos a todos los integrantes a un solo objetivo siguiendo las reglas y los procesos de la metodología.
- Equipo de Trabajo: Conformado por Luis Chitacapa, Cristian Espinoza, Luis Tito, Rodrigo Nacipucha, Cristian Cabrera

## 12. PERSONAS Y ROLES DEL PROYECTO

Perona	Contacto	Rol - Desempeño
Luis Chitacapa	lchitacpac@est.ups.edu.ec	PRODUCT OWNER
Cristian Cabrera	ccabrerag@est.ups.edu.ec	PRODUCT OWNER
Christian Espinoza	cespinozaz@est.ups.edu.ec	PRODUCT OWNER
Luis Tito	lsaavedrat@est.ups.edu.ec	PRODUCT OWNER
Rodrigo Nacipucha	rnacipucha@est.ups.edu.ec	PRODUCT OWNER

## 13. DIAGRAMA DE CALASES

14.ARQUITECTURA

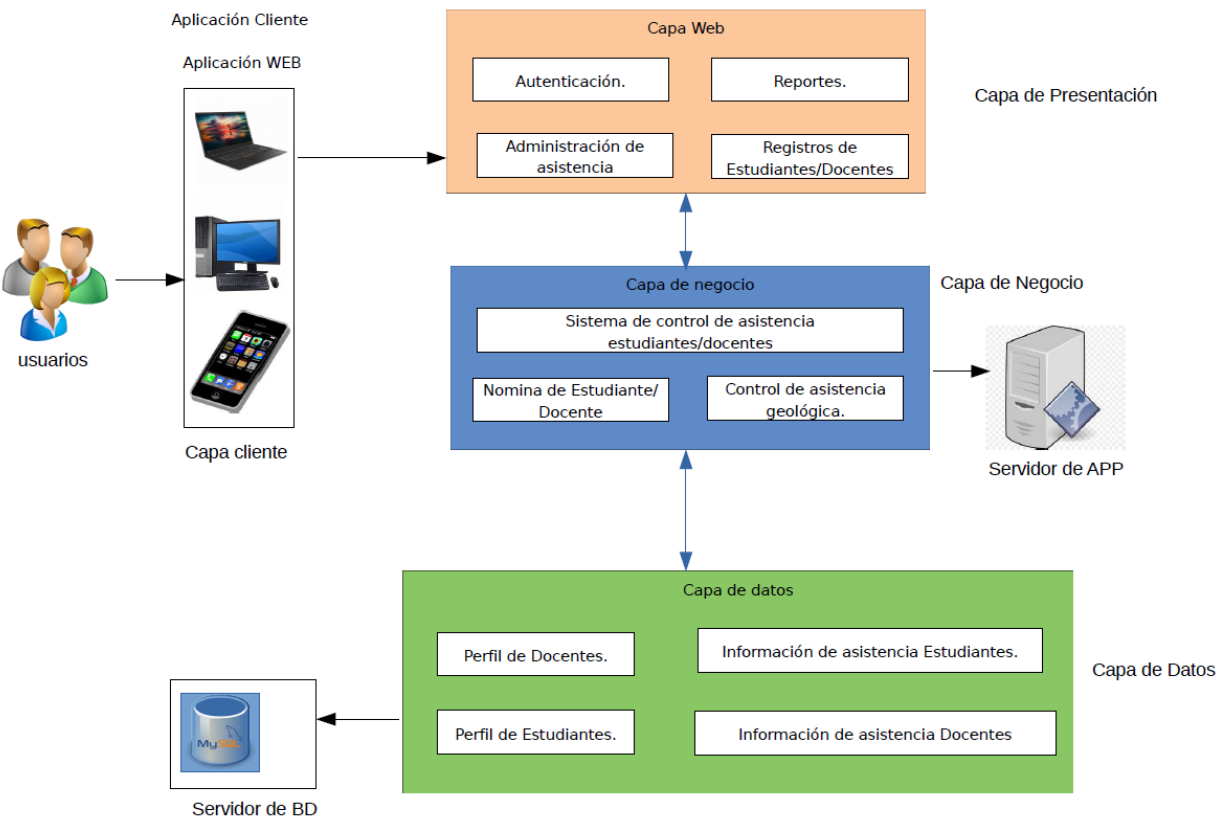


Figura 2. Arquitectura N-Capas Sistema de Registro

15.Prototipado