

REGISTRO DE ASISTENCIA PARA LOS LABORATORIOS ETM DE LA UNIVERSIDAD SANTO TOMÁS HACIENDO USO DE LECTORES NFC Y CARNET INSTITUCIONAL

Zinedine Comas – Oscar Roncancio



Universidad Santo Tomás
Proyecto de grado

REGISTRO DE ASISTENCIA PARA LOS LABORATORIOS ETM DE LA UNIVERSIDAD SANTO TOMÁS HACIENDO USO DE LECTORES NFC Y CARNET INSTITUCIONAL

**ZINEDINE KAMEL COMAS HERNANDEZ
OSCAR JULIAN RONCANCIO CARREÑO**

Universidad Santo Tomás
Facultad de Ingeniería Electrónica
Bogotá, Colombia
2024



REGISTRO DE ASISTENCIA PARA LOS LABORATORIOS ETM DE LA UNIVERSIDAD SANTO TOMÁS HACIENDO USO DE LECTORES NFC Y CARNET INSTITUCIONAL

**ZINEDINE KAMEL COMAS HERNANDEZ
OSCAR JULIAN RONCANCIO CARREÑO**

Proyecto de grado presentado como requisito para optar al título de:

Ingeniero Electrónico

Director:

Dario Alejandro Segura Torres M. Sc

Codirector:

Carlos Javier Mojica Casallas M. Sc

Grupo de Investigación MEM (Modelado-Electrónica-Monitoreo)

Línea de Investigación: Automática

Universidad Santo Tomás

Facultad de Ingeniería Electrónica

Bogotá, Colombia

2024

Autoridades de la universidad

RECTOR GENERAL

FRAY, Álvaro José ARANGO RESTREPO, O.P.

VICERRECTOR ADMINISTRATIVO Y FINANCIERO GENERAL

Fray, Hernán Yesid RIVERA ROBERTO, O.P.

VICERRECTOR ACADÉMICO GENERAL

Fray, Mauricio Antonio CORTÉS GALLEGO, O.P.

SECRETARIA GENERAL

Dra. Lucero Galvis Cano

DECANO DIVISIÓN DE INGENIERÍAS

Fray, Javier Antonio HINCAPIE ARDILA, O.P.

SECRETARIA DE DIVISIÓN

EC. Luz Patricia Rocha Caicedo

DECANO FACULTAD DE INGENIERÍA ELECTRÓNICA

PhD. Carlos Andrés Torres Pinzón

Nota de aceptación

Firma del director

Firma del codirector

Firma del jurado

Firma del jurado

BOGOTÁ D.C. _____ DE 2024

ADVERTENCIA

La Universidad Santo Tomás no se hace responsable de las opiniones y conceptos expresados en el trabajo de grado, solo velará por qué no se publique nada contrario al dogma ni a la moral católica y porque el trabajo no tenga ataques personales y únicamente se vea el anhelo de buscar la verdad científica.

Capítulo III –Art. 46 del Reglamento de la Universidad Santo Tomás.

Dedicatoria

Dedico este trabajo principalmente a Dios y a mis padres por el constante apoyo en la realización y culminación de este proyecto de grado, por todo el esfuerzo que han puesto en mi educación a lo largo de los años.

Zinedine Kamel Comas Hernández

Dedico este trabajo a Dios y a la virgen de Chiquinquirá, a mis padres por su apoyo tanto emocional como económico para poder llegar hasta estas instancias del proyecto de grado, y a mi hermano quien ha sido mi principal compañía en este proceso.

Oscar Julián Roncancio Carreño

AGRADECIMIENTOS

Le agradezco especialmente al Ingeniero Dario Segura, porque gracias a él, este proyecto salió adelante con su conocimiento y consejos a lo largo del desarrollo de este, al Ingeniero Carlos Mojica, porque él fue quien inicialmente nos dio la idea con un comentario hecho en una clase, también le agradezco a todos los docentes que han sido involucradas durante el desarrollo del proyecto.

Zinedine Kamel Comas Hernández

Le agradezco, primeramente, a Dios y a la virgen de Chiquinquirá por otorgarnos vida y salud a mi compañero y a mi hasta el día de hoy, al ingeniero Dario Segura por brindarnos sus conocimientos y consejos en el desarrollo de este proyecto, y al ingeniero Carlos Mojica por su ayuda y permiso de ejecutar una idea de su propiedad. También agradezco a mi compañero Zinedine por ser ese gran amigo tanto en la vida como en lo académico.

Oscar Julián Roncancio Carreño

CONTENIDO

RESUMEN	XIII
ABSTRACT.....	XIV
LISTA DE FIGURAS	XV
LISTA DE TABLAS	XVI
LISTA DE GRÁFICAS	XVII
LISTA DE ILUSTRACIONES.....	XVIII
GLOSARIO	XX
1. INTRODUCCIÓN	1
1.1. Planteamiento del Problema	2
1.2. Objetivos	3
1.2.1. Objetivo General.....	3
1.2.2. Objetivos Específicos.....	3
1.3. Justificación.....	3
1.4. Impacto Social.....	5
2. ESTADO DEL ARTE	6
2.1. Proyectos relevantes basados en NFC	6
2.2. Proyecto relevante con implementación de lectores de código de barras..	7
2.3. Aplicaciones móviles y web en relación con el registro de asistencia.....	7
2.4. Distintas implementaciones de registros de asistencia en instituciones académicas.....	8
3. MARCO TEÓRICO	10
3.1. Near Field Communication	10
3.2. RFID.....	10
3.3. Aplicaciones web	11
3.4. Servicios web	12
3.5. Aplicaciones de escritorio.....	12
3.6. Arquitecturas de aplicaciones web	13
3.7. NodeMCU ESP8266	15
3.8. Protocolo SPI	16
3.9. Protocolo I2C	16
3.10. Arduino IDE.....	17
3.11. Bases de datos	17

3.12. Normativa para PCB	18
4. DISEÑO METODOLÓGICO.....	20
4.1. Fase 1 – Caracterización del sistema	21
4.1.1. Actividad 1	21
4.1.2. Actividad 2	21
4.1.3. Actividad 3	21
4.2. Fase 2 - Diseño	21
4.2.1. Actividad 4	21
4.2.2. Actividad 5	22
4.2.3. Actividad 6	22
4.2.4. Actividad 7	22
4.3. Fase 3 - Desarrollo e implementación de la aplicación	22
4.3.1. Actividad 8	22
4.3.2. Actividad 9	22
4.3.3. Actividad 10	22
4.4. Fase 4 - Validación del sistema	22
4.4.1. Actividad 11	22
4.5. Fase 5 – Escritura del documento.....	23
5. DESARROLLO CONCEPTUAL	24
5.1. Desarrollo fase I – Caracterización del sistema	24
5.1.1. Procesos identificados	26
5.1.2. Casos de uso	31
5.1.3. Componentes del sistema (Diagrama UML)	32
5.2. Desarrollo fase II – Diseño	33
5.2.1. Diseño PCB	33
5.2.2. Diseño del modelo de datos	36
5.2.2.1. Grupo de Estudiante	38
5.2.2.2. Grupo Programa.....	40
5.2.2.3. Grupo Responsable	42
5.2.2.4. Grupo Sesión	43
5.2.2.5. Administrador	44
5.2.2.6. Temporal	45
5.2.2.7. Carga_Archivo.....	45
5.2.3. Diseño Aplicación modo configuración	46

5.2.4.	Diseño aplicación web	47
5.2.4.1.	Arquitectura	47
5.2.4.2.	Wireframe interfaz de usuario	47
5.3.	Desarrollo fase III – Desarrollo e implementación de la aplicación	49
5.3.1.	Desarrollo de backend	49
5.3.2.	Desarrollo de frontend	51
5.3.3.	Configuración NodeMCU	53
5.4.	Desarrollo fase IV – Validación del sistema	54
5.4.1.	Preparación del sistema para pruebas	54
5.4.2.	Descripción de las pruebas.....	55
5.4.3.	Resultados de las pruebas	56
6.	RESULTADOS Y DISCUSIÓN	60
6.1.	Ingreso a clase.....	61
6.2.	Salida de clase.....	61
6.3.	Registro de clases en laboratorios ETM.....	62
6.4.	Condiciones del responsable	63
6.5.	Registro de asistencia	63
6.6.	Envío de asistencia	64
6.7.	Carga de archivo HTML	64
7.	CONCLUSIONES Y TRABAJOS FUTUROS.....	66
7.1.	Conclusiones.....	66
7.2.	Trabajos futuros	67
8.	REFERENCIAS BIBLIOGRÁFICAS.....	68
9.	ANEXOS	71
9.1.	Anexo 1	71
9.2.	Anexo 2	71
9.3.	Anexo 3	72
9.4.	Anexo 4	72
9.5.	Anexo 5	73
9.6.	Anexo 6	74
9.7.	Anexo 7	75
9.8.	Anexo 8	76
9.9.	Anexo 9	76
9.10.	Anexo 10	77

9.11. Anexo 11	77
9.12. Anexo 12	77

RESUMEN

En la Universidad Santo Tomás se lleva un sistema de registro de asistencia manual, en el cual cada docente se le solicita llevar un registro de la asistencia de los estudiantes a los distintos espacios académicos registrados. El registro de asistencia para los laboratorios ETM de la Universidad Santo Tomás haciendo uso de lectores NFC y carnet institucional brinda una solución para que esta actividad se haga de forma tal que los docentes no deban llevar de forma manual todo este proceso, incorporando tecnologías innovadoras dentro de un sector específico de la Universidad Santo Tomás Sede Principal.

Durante la ejecución de este proyecto se llevó a cabo una metodología cuantitativa, ya que es fundamental obtener datos experimentales que requieren la recolección de información tanto antes como después de la implementación del sistema. Dentro de las actividades realizadas en esta metodología se encuentra el reconocimiento y análisis de los distintos procesos inmersos en un registro de asistencia, seguido a ello, el análisis de que elementos serían usados para el diseño de un prototipo que realice la lectura del identificador serial de cada carnet institucional añadido a la estructura de una base de datos y aplicación a usuario con las características obtenidas en el análisis de procesos de un registro de asistencia (caracterización del sistema).

Luego de la ejecución del sistema de registro de asistencia automático en uno de los laboratorios ETM de la Universidad Santo Tomás se obtienen a los usuarios registrados en el sistema, tanto docentes como estudiantes, logrando que los estudiantes puedan reportar su ingreso y salida de cada sesión de clase haciendo uso de su carnet institucional. Con este reporte por estudiante se obtiene un reporte por cada sesión, en donde, dependiendo de la hora de ingreso de cada estudiante se genera su estado de asistencia (presente, retardo o falla injustificada). Estos reportes se envían al docente encargado a través de su correo electrónico tras finalizar el tiempo de cada sesión, según sus condiciones de tiempo establecidas. Finalmente, cada uno de estos reportes pueden ser cargados a Moodle para tener allí la trazabilidad de la asistencia por cada sesión de clase desarrollada.

Con el cumplimiento de cada uno de los objetivos planteados para el desarrollo de un sistema de registro de asistencia automático se logra tener un sistema el cual brinda mejoras en cuanto al tiempo que se emplea en el registro de asistencia por estudiante, permitiendo, además, que el principal responsable de su reporte de asistencia sea el mismo estudiante. Por otro lado, se tiene un porcentaje bajo de error en cada reporte de asistencia entregado por el sistema y se logra que el proceso de registro de asistencia no interfiera con el correcto desarrollo de cada sesión de clase.

Palabras clave:

Registro de asistencia, NFC, asistente, estudiante, RFID.

ABSTRACT

Santo Tomas University has a manual attendance registration system, in which each teacher is required to keep a record of student attendance to the different academic spaces registered. The attendance registers for the ETM laboratories of Universidad Santo Tomas using NFC readers and institutional card provides a solution for this activity to be done in such a way that teachers do not have to manually carry out the whole process, incorporating innovative technologies within a specific sector of Universidad Santo Tomas Main Campus.

In the development of this project a quantitative methodology was used, since it is essential to obtain experimental data that require the collection of information both before and after the implementation of the system. Among the activities carried out in this methodology is the recognition and analysis of the different processes involved in an attendance register, followed by the analysis of which elements would be used for the design of a prototype that performs the reading of the serial identifier of each institutional card added to the design of a database and user application with the characteristics obtained in the analysis of the processes of an attendance register (system characterization).

After the implementation of the automatic attendance registration system in one of the ETM laboratories of Universidad Santo Tomás, the users registered in the system, both teachers and students, are obtained, allowing students to report their entry and exit of each class session using their institutional card. With this report per student, a report is obtained for each session, where, depending on the time of entry of each student, their attendance status is generated (present, late or unjustified failure). These reports are sent to the teacher in charge via e-mail after the end of each session, according to the established time conditions. Finally, each of these reports can be uploaded to Moodle to have the traceability of attendance for each class session developed.

With the fulfillment of each of the objectives set for the development of an automatic attendance registration system, it is possible to have a system that provides improvements in terms of the time spent in the registration of attendance per student, allowing, in addition, that the main responsible for the attendance report is the student himself/herself. On the other hand, there is a low percentage of error in each attendance report delivered by the system and the attendance registration process does not interfere with the correct development of each class session.

Keywords:

Attendance register, NFC, attendant, student, RFID.

LISTA DE FIGURAS

Figura 1. Arquitectura de las aplicaciones web: todo en un servidor. Fuente: [20]	13
Figura 2. Arquitectura de las aplicaciones web: separación servidor de datos. Fuente: [20].	13
Figura 3. Arquitectura de las aplicaciones web: todo en un servidor, con servicio de aplicaciones. Fuente: [20]	14
Figura 4. Arquitectura de las aplicaciones web: separación de servidor de datos, con servicio de aplicaciones. Fuente:[20]	14
Figura 5. Arquitectura de las aplicaciones web: todo separado. Fuente: [20]	15
Figura 6. Diagrama metodológico. Fuente: autores.	20
Figura 7. Proceso para el registro de usuario. Fuente: autores.	27
Figura 8. Proceso de registro de ingreso a clase. Fuente: autores.	27
Figura 9. Proceso para el registro de salida de clase. Fuente: autores.	28
Figura 10. Proceso para el registro de clases en laboratorios ETM. Fuente: autores.	28
Figura 11. Proceso para el registro de condiciones del responsable. Fuente: autores.	29
Figura 12. Proceso para el registro de asistencia. Fuente: autores.	30
Figura 13. Proceso para el envío de asistencia. Fuente: autores.	30
Figura 14. Proceso para carga de archivo HTML. Fuente: autores.	31

LISTA DE TABLAS

Tabla 1. Relación entre voltaje y distancia entre pistas. Fuente:[27].....	18
Tabla 2. Relación entre ancho de pista y corriente. Fuente:[27]	19
Tabla 3. Medición inicial de tiempo en llamado de asistencia. Fuente: autores. ...	25
Tabla 4. Matriz PUGH. Fuente: autores.	33
Tabla 5. Número de asistentes por espacio académico. Fuente: autores.....	56
Tabla 6. Cantidad de estudiantes y sesiones por espacio académico. Fuente: autores.	56
Tabla 7. Cálculo de la diferencia entre ambos métodos de toma de asistencia. Fuente: autores.	57
Tabla 8. Porcentaje de error respecto al cambio de posición del enrutador. Fuente: autores.	59

LISTA DE GRÁFICAS

Gráfica 1. Respuestas docentes. Fuente: autores.	24
Gráfica 2. Tiempo cronometrado. Fuente: autores.	26
Gráfica 3. Comparación de tiempos cronometrados. Fuente: autores.	57
Gráfica 4. Porcentaje de error del sistema. Fuente: autores.	58

LISTA DE ILUSTRACIONES

Ilustración 1. Diagrama de casos de uso. Fuente: autores	32
Ilustración 2. Diagrama UML de componentes del sistema. Fuente: autores.	32
Ilustración 3. Diseño esquemático. Fuente: autores.....	34
Ilustración 4. Funcionamiento protoboard. Fuente: autores.	35
Ilustración 5. Diseño 2D. Fuente: autores.	35
Ilustración 6. Prueba de funcionamiento prototipo. Fuente: autores.	36
Ilustración 7. Modelo de base de datos implementado. Fuente: autores.	37
Ilustración 8. Grupo estudiante. Fuente: autores.....	38
Ilustración 9. Grupo Programa. Fuente: autores.	40
Ilustración 10. Grupo Responsable. Fuente: autores.	42
Ilustración 11. Grupo Sesión. Fuente: autores.	43
Ilustración 12. Tabla Administrador. Fuente: autores.....	44
Ilustración 13. Tabla Temporal. Fuente: autores.	45
Ilustración 14. Tabla Carga de Archivo. Fuente: autores.	46
Ilustración 15. Diseño app modo configuración. Fuente: autores.....	46
Ilustración 16. Arquitectura de aplicación web. Fuente: autores.	47
Ilustración 17. Wireframe para ingreso de usuario. Fuente: autores.	48
Ilustración 18. Wireframe para recepción de archivo en formato HTML. Fuente: autores.	48
Ilustración 19. Wireframe para modificación de datos de usuario. Fuente: autores.	49
Ilustración 20. Solución para implementación de capa de Negocio. Fuente: autores.	49
Ilustración 21. Funciones realizadas en el servicio web. Fuente: autores.....	50
Ilustración 22. Código para envío de correo empleando el objeto SMTPClient. Fuente: autores.	51
Ilustración 23. Código para envío del archivo de reporte de asistencia. Fuente: autores.	51
Ilustración 24. Implementación de patrón MVC en la aplicación web. Fuente: autores.	52
Ilustración 25. Diagrama de bloques MVC. Fuente: autores.	52
Ilustración 26. Diagrama de flujo programas NodeMCU. Fuente: autores	53
Ilustración 27. Diagrama de flujo modo asistencia. Fuente: autores	54
Ilustración 28. Pestaña modificación datos de usuario. Fuente: autores.....	60
Ilustración 29. Pestaña modificar. Fuente: autores.	60
Ilustración 30. Mensaje de bienvenida. Fuente: autores	61
Ilustración 31. Mensaje de salida. Fuente: autores	61
Ilustración 32. Pestaña asignación de esp. académicos. Fuente: autores.	62
Ilustración 33. Resultado de consulta SQL de la tabla asistente con resultados de la carga de lista de estudiantes. Fuente: autores.....	62
Ilustración 34. Asignación de condiciones por docente. Fuente: autores.....	63
Ilustración 35. Reporte generado. Fuente: autores.	63
Ilustración 36. Correo enviado con reporte de asistencia. Fuente: autores.....	64

Ilustración 37. Pestaña carga de horarios. Fuente: autores.	65
Ilustración 38. Resultado de consulta en SQL con resultados de la carga del HTML. Fuente: autores.	65

GLOSARIO

1. USTA: Universidad Santo Tomás.
2. UDIES: Unidad de Desarrollo Integral del Estudiante.
3. SAC: Sistema Académico.
4. ODS: Objetivos de Desarrollo Sostenible.
5. ETM: Electrónica-Telecomunicaciones-Mecánica.
6. Estudiante: Estudiante de la Universidad es la persona matriculada en algún Programa Académico.
7. Asistente: Tipo de denominación a un estudiante que asiste a una sesión de clase.
8. Grupo: Identificador del espacio académico.
9. Responsable: Usualmente un docente, o un laboratorista, quien este encargado de algún espacio académico.
10. Documento de identificación / Documento: Identificador de cada estudiante.
11. SSMS: SQL Server Management Studio, usado para hacer referencia al entorno de base de datos.
12. MSSQLServer: Microsoft SQL Server, usado para hacer referencia al SQL Server(motor de base de datos).
13. NFC: Near Field Communication, en español Comunicación de Campo Cercano.
14. HTTP: Hypertext Transfer Protocol, en español Protocolo de transferencia de hipertexto.
15. RFID: Radio Frequency Identification, en español Identificación por Radio Frecuencia.
16. IoT: Internet of Things, en español Internet de las Cosas.
17. BSSID: Basic Service Set Identifier, en español Identificador único de cada dispositivo inalámbrico.
18. UDDI: Universal Description Discovery and Integration, en español Descripción, Descubrimiento e Integración Universal.
19. XML: Extensible Markup Language, en español Lenguaje de Marcado Extensible.
20. SOAP: Simple Object Access Protocol, en español Protocolo simple de acceso a objetos.
21. WSDL: Web Services Description Language, en español Lenguaje de descripción de servicios web.
22. OSI: Open Systems Interconnection, en español Interconexión de sistemas abiertos.
23. SOA: Service Oriented Architecture, en español Arquitectura Orientada a Servicio.
24. WLAN: Wireless Local Area Network, en español Red de área local inalámbrica.

1. INTRODUCCIÓN

El registro de asistencia en las instituciones educativas es realizado para llevar un seguimiento de cuáles estudiantes están faltando a los distintos espacios académicos, esto con el fin de dar cumplimiento a normativas institucionales y a su vez, garantizar calidad en la educación de cada estudiante. Debido a lo anterior, los métodos tradicionales usados para llevar a cabo esta actividad, ya sea, por hojas de cálculo o por método manual, para ambos casos, es posible que se presenten inconsistencias, errores y limitaciones en la toma de esta.

En la USTA, para antes del año 2023, no se cuenta con un sistema automático para el registro de asistencia a clase de los estudiantes, esto produce que muchos docentes recurran a métodos tradicionales en plena era de la información y acceso a los datos, por lo anterior, se ejecuta este trabajo de grado, ofreciendo un sistema en el cual se lleva a cabo el registro de asistencia de manera automática, donde el estudiante sea quien se encargue de registrar su asistencia.

En primera instancia se realiza una caracterización del sistema a implementar, donde se identifican los distintos casos de uso y procesos a seguir que en un principio son necesarios para llevar a cabo el registro de asistencia y los distintos implicados y sus respectivas tareas para que el sistema funcione correctamente.

Al cumplir con lo anterior, se procede a diseñar e implementar un dispositivo el cual tiene como función leer el código NFC alojado en cada uno de los carnets institucionales de los estudiantes, así mismo, cumple con el propósito de enviar dicho código leído hacia un servidor alojado dentro de los laboratorios ETM.

Como tercera etapa, se desarrolla e implementa una aplicación web, donde se cumple con las funciones para el correcto funcionamiento del sistema.

Finalmente, se valida el funcionamiento del sistema, mediante la unificación del dispositivo y la aplicación desarrollada, realizando pruebas en un aula de los laboratorios ETM y con distintos espacios académicos de docentes que a lo largo de la semana hagan uso de esta.

1.1. Planteamiento del Problema

En el reglamento estudiantil de la Universidad Santo Tomás, se establece que cada estudiante está en el deber de cumplir con un mínimo del 80% de asistencia a las sesiones de clase por espacio académico matriculado, es decir, que si el estudiante cuenta con más del 20% de inasistencia se verá reflejada la nota mínima estipulada (1.0) [1]

Se ha notado como algunos estudiantes abandonan sus espacios académicos debido a distintos problemas, ya sean de tipo personal o académico, llevándolos a tomar la decisión de cancelar dicho espacio, para no afectar su promedio acumulado. A raíz de esto, desde la Unidad de Desarrollo Integral del Estudiante (UDIES), la cual tiene como uno de sus enfoques la permanencia de estudiantes en los distintos programas académicos dentro de la Universidad Santo Tomás sugiere a los docentes desde su programa de acompañamiento estudiantil que en sus respectivos espacios académicos se lleve un registro de asistencia por cada sesión de clase, para de esta manera realizar un seguimiento de manera oportuna sobre los distintos casos que se puedan presentar por cada estudiante.

En la realización del registro de asistencia, algunos docentes hacen uso de la plataforma Moodle en donde se cuenta con una herramienta de asistencia (attendance), en la cual se encuentran diligenciados todos los estados en relación con la asistencia, como lo son los apartados de Nombre, Presente, Retardo, Falla justificada, Falla injustificada.

Debido a lo anterior, la realización del registro de asistencia causa una pérdida de tiempo en algunos casos puede ser superior a 5 minutos en donde se presentan diversos factores que retrasan el ejercicio de esta actividad, así como también existen casos en los que el tiempo utilizado es menor a 1 minuto, como se puede evidenciar en una encuesta realizada a los docentes de la Facultad de Ingeniería Electrónica, observada en el Anexo 1. Esta pérdida de tiempo se da por distintos factores, los cuales pueden ser que el número de estudiantes registrados por espacio académico es elevado, la velocidad del internet de la universidad en algunos momentos es lenta, factores ambientales como lo puede ser el ruido alrededor de la Universidad, la atención prestada por los estudiantes durante el desarrollo de esta actividad, el retardo que pueden tener los estudiantes al llegar al aula, entre otros; como consecuencia de los factores anteriormente nombrados, algunos docentes tienden a realizar el registro de asistencia de manera manual u omiten el proceso de registro de asistencia.

Según [2], hay alrededor de 717 estudiantes pertenecientes a las Facultades de Ingeniería Electrónica, Ingeniería de Telecomunicaciones e Ingeniería Mecánica, los cuales cuentan con un total de 11 laboratorios especializados para el desarrollo de sus actividades académicas, los cuales son denominados "Laboratorios ETM", lugar en donde se observa mayor dispersión en los estudiantes al momento del registro de asistencia debido a que llevan su concentración hacia la realización de sus prácticas en estos laboratorios.

Por lo anterior descrito se plantea la siguiente pregunta problema:

¿Cómo implementar el registro de asistencia para los estudiantes de Ingeniería Electrónica, haciendo uso de distintas herramientas informáticas y electrónicas, de modo que esta actividad se realice de manera automática en las distintas sesiones de los espacios académicos tomados en los laboratorios ETM de la Universidad Santo Tomás?

1.2. Objetivos

1.2.1. Objetivo General

Implementar un sistema de registro de asistencia automática mediante el uso de lectores NFC con el fin de dar un reporte de fallas totales a los docentes al finalizar cada corte académico en los laboratorios ETM de la Universidad Santo Tomás.

1.2.2. Objetivos Específicos

Caracterizar el proceso del sistema de registro de asistencia de manera que su ejecución sea simultánea a las sesiones de clase, logrando así la independencia entre el registro de asistencia y el desarrollo de cada sesión de clase.

Diseñar el prototipo de un dispositivo para la lectura del carnet estudiantil haciendo uso de tecnología NFC, que permita el registro de asistencia del estudiante a las sesiones de clase tomadas en los laboratorios ETM.

Implementar una aplicación que soporte el proceso caracterizado y permita la integración con el prototipo desarrollado, con el cual se busca generar informes de registro de asistencia de los estudiantes y enviarlos automáticamente a las direcciones de correo de los docentes.

Validar el desempeño del sistema mediante la unificación del prototipo y la aplicación para la comprobación de las características y el funcionamiento esperado.

1.3. Justificación

El procedimiento de registro de asistencia realizado de manera manual en la Universidad Santo Tomás trae consigo dificultades en el proceso de seguimiento estudiantil, ya que no se tiene certeza sobre la cantidad de fallas o asistencias por cada estudiante a los espacios académicos, teniendo en cuenta que los datos obtenidos para UDIES, son los registrados por cada docente en el Sistema Académico (SAC).

Con la implementación de un sistema de lectura de tarjetas NFC en la entrada de un laboratorio ETM, se buscará reducir el tiempo que se gasta en el registro de asistencia, y permitir que los estudiantes dirijan su atención a la realización de sus prácticas, llevando esta actividad a desarrollarse de manera paralela a la clase, además, se busca que esta actividad ya no sea dependiente de los docentes y lo sea solo de los estudiantes; añadido a esto se podrá tener un registro adecuado y veraz de la asistencia en cada espacio académico desarrollado en los laboratorios, ayudando al programa de acompañamiento estudiantil a realizar una mejor labor con cada uno de los estudiantes de la Universidad.

Realizar la implementación de un sistema de registro de asistencia haciendo uso de lectores NFC en la entrada a cada aula proporcionará que se pueda realizar un registro de asistencia estudiantil en cada una de estas aulas, según [2] alrededor de 212 estudiantes y 12 docentes pertenecientes a la Facultad de Ingeniería Electrónica vinculados a las sesiones de clase realizadas en los laboratorios ETM se podrán ver beneficiados, así como en [3], donde se realizó un trabajo en una mayor proporción con el fin de hacer una función de control de acceso, tanto para docentes, estudiantes y empleados del Instituto Técnico de Iguala ubicado en México, todo esto haciendo uso de lectores NFC.

Se espera que al ser un proceso en el cual se toma una menor cantidad de tiempo, tanto los estudiantes como los docentes lo realicen con mayor frecuencia, con el objetivo de obtener datos de la asistencia a clase por cada estudiante que se puedan manejar de una manera acertada y fiable.

Debido a lo anterior, se dispone de este proyecto como propuesta de opción de grado ya que es una manera de innovación en un sistema que actualmente genera mayor carga laboral a los docentes durante y después de las sesiones de clase en la Universidad Santo Tomás; proyecto en el cual se buscará aplicar conocimientos adquiridos en el ámbito de la programación y de sistemas digitales para registro de asistencia; añadido a esto se va a requerir de la investigación sobre el uso de bases de datos, el diseño de aplicaciones de tipo web y móvil, y emplear el uso de lectores de NFC que resulta innovador respecto a lo realizado comúnmente en el proceso de registro de asistencia.

La implementación de este sistema puede traer consigo posibilidades a mejoras futuras, en las cuales se pueden agregar sistemas cerrados de grabación, y poder así monitorear el estado de los elementos en las aulas. Así mismo, se puede mejorar la seguridad en cuanto a la identificación de cada estudiante al ingresar al aula, implementando un sistema de detección facial con las cámaras que se puedan allí instalar y/o con un lector de huellas digitales, evitando así el fraude de presentar un carnet ajeno en el lector NFC.

1.4. Impacto Social

Los Objetivos de Desarrollo Sostenible (ODS) propuestos por la Organización de las Naciones Unidas (ONU) en [4], tienen la finalidad de abordar desafíos a nivel global y fomentar un desarrollo sostenible en los ámbitos económico, social y ambiental. La razón fundamental de los ODS es mejorar la calidad de vida de las personas y preservar el planeta, enfrentando desafíos apremiantes que afectan a la humanidad. Estos objetivos abarcan desde la abolición de la pobreza y el hambre hasta el fomento de la igualdad de género, el acceso a la educación, la acción climática y la paz y la justicia. La premisa es establecer un marco global que oriente a los países hacia un desarrollo equitativo, respetuoso con el medio ambiente y sostenible a largo plazo. Los ODS representan un compromiso compartido para construir un mundo más justo, saludable y sostenible para las generaciones presentes y futuras.

El Objetivo 4 de Desarrollo Sostenible se enfoca en asegurar una educación de calidad, inclusiva y equitativa, así como en fomentar oportunidades de aprendizaje a lo largo de la vida para todos. En el contexto del registro de asistencia, este ODS impulsa el acceso generalizado a la educación, la igualdad en la participación, la mejora constante de la calidad educativa a través del análisis de datos de asistencia, la identificación de oportunidades de aprendizaje continuo, y la promoción de la inclusión, asegurando que el sistema de registro sea accesible para todos; así mismo, contribuye a poder explorar las distintas causas que pueden llegar a provocar la inasistencia de los estudiantes a sus distintos espacios académicos.

Actualmente en la sede principal de la Universidad Santo Tomás, en cuanto al tema de registro de asistencia no se tiene algún sistema que permita realizar un registro automático de la misma por cada estudiante. Dicha actividad se realiza de manera manual dentro de la plataforma de Moodle, seleccionando cada una de las posibles opciones al marcar un registro de asistencia.

La población docente y estudiantil de la Universidad Santo Tomás son los principalmente involucrados en esta actividad, además, se encuentra la problemática desde el programa de acompañamiento estudiantil, en donde no se recibe una totalidad de registros de asistencia de las sesiones de clase dadas, causando un análisis incompleto en cuanto a la permanencia o abandono de estudiantes en distintos espacios académicos. Por ende, al implementar el sistema que se propone en este proyecto, se lleva a innovar esta tarea diaria común en toda la Universidad, logrando que, además de ayudar a los estudiantes y docentes a aprovechar mejor su sesión de clase, se pueda brindar apoyo a el programa de acompañamiento estudiantil brindando un registro de asistencia generado para el total de las sesiones de clase dadas y de esta manera lograr un análisis correcto sobre la situación de los estudiantes en los espacios académicos.

2. ESTADO DEL ARTE

El identificador que se posee personalmente en la Universidad Santo Tomás está dado mediante un carnet, el cual tiene dos maneras de validarse electrónicamente; la primera es un código serial leído por radio frecuencia (RFID) haciendo uso de la tecnología de lectura NFC (*Near Field Connection*) que generalmente es utilizada en los torniquetes de ingreso y salida de las instalaciones del edificio, y la segunda manera, utilizada en la biblioteca y en los laboratorios para hacer algún tipo de solicitud de material, es por medio del uso de lector de código de barras, puesto que cada carnet tiene un código de barras en la parte trasera del carnet con la información del número de identificación de quien lo porta.

A través del tiempo se ha podido observar que, para la implementación de tecnologías, ya sean de acceso o de registro de asistencia en distintos lugares, se ha hecho uso de NFC y así mismo de lectores por código de barras. Posterior a la implementación de estos sistemas, con el auge de las tecnologías controladas desde un dispositivo móvil, se encuentra la necesidad de implementar en estos sistemas el uso de aplicaciones móviles en donde se pueda observar y manipular la información obtenida.

Estas tecnologías se han empleado en proyectos de registro y control de acceso, en ámbitos similares a los propuestos, como lo son en su mayoría empresariales, para el manejo los tiempos en los cuales cada persona está presente dentro del sitio haciendo uso del sistema, como lo es en [5] y [6].

2.1. Proyectos relevantes basados en NFC

Se han encontrado implementaciones de tecnologías NFC mayormente en controles de acceso a edificios o establecimientos, así como en [3] en donde se utilizó esta tecnología con el fin de dar control al acceso de la población en general del Instituto Técnico de Iguala, pensando en mejorar el ingreso y salida al parqueadero y al edificio, y así mismo con el fin de mejorar la seguridad de este. Esto a nivel de instituciones, por otro lado, se encuentran implementaciones a nivel hogar, así como en [7], trabajo en el cual se implementa un sistema de lector por NFC para el bloqueo de la puerta principal de un hogar buscando dar ingreso y bloqueo a este, además, se establece un sistema tal que al determinar que el individuo esta por fuera de su hogar corta la energía de la mayor parte del hogar dejando únicamente con conexión a aquellos electrodomésticos con necesidad de conexión continua tal como la nevera. Estos avances llevan a que el uso de los lectores NFC vayan más allá, dando paso a nuevas implementaciones tecnológicas haciendo uso de sistemas micro controlados tales como Arduino, por ejemplo, en [8], en donde se hace uso de la tecnología de RFID que se había observado solo en la agroindustria, esto con el fin de implementarlo en la acción de toma de asistencia, lo cual lleva a que esta tarea se vuelva

autónoma haciendo uso de lo que hoy se conoce como IoT (*Internet of Things*).

Teniendo en cuenta las referencias mencionadas anteriormente, se observa que la tecnología NFC tiene múltiples aplicaciones en donde se realizan implementaciones que van desde el hogar hasta grandes empresas, las cuales se enfocan en el control de acceso a las instalaciones permitiendo mejorar la seguridad y la automatización de procesos.

2.2. Proyecto relevante con implementación de lectores de código de barras

Los lectores de código de barras se pueden observar principalmente en lugares donde tener un inventario es necesario, así como en empresas, bodegas, bibliotecas, etc. Muchas veces esta implementación puede resultar algo costosa comparándole con otras tecnologías, llevando a que se busquen maneras de hacer más fácil y/o económico esta acción. Es lo que se desea realizar en [9], donde se busca implementar un sistema de lectura de código de barras a partir del uso de un teléfono móvil, haciendo que sea más sencillo y así mismo que sea una actividad que no se ancle a un solo sitio de trabajo, pues el teléfono móvil será más fácil de transportar libremente.

Tal como se menciona anteriormente, el registro de asistencia no se realiza de manera automática debido a que es un sistema portátil, es decir, no se encuentra instalado en un lugar específico, a diferencia de esto la implementación de lectores NFC en los laboratorios ETM busca realizar el registro de asistencia de los estudiantes por medio del carnet institucional.

2.3. Aplicaciones móviles y web en relación con el registro de asistencia

El uso del teléfono móvil es algo muy cotidiano en el mundo actual, en donde la mayoría de las acciones y tareas que se realizan en el día a día son mediante el uso de este dispositivo. El uso de este también se ha podido aprovechar en cuánto a la ejecución de registros de asistencia en empresas y escuelas, así como en [10] en donde se diseña e implementa una aplicación móvil para el sistema operativo Android, y añadido a esto se usa la tecnología BSSID (*Basic Service Set Identifier*) la cual se basa en identificar distintos dispositivos dentro de una red inalámbrica local (WLAN). En este ámbito se pudo detectar una serie de problemáticas en las cuáles se encuentran las pérdidas deliberadas de tiempo, errores humanos y sobrecarga laboral en docentes y administrativos, las cuales se solucionan mediante la implementación de esta tecnología de aplicación móvil.

Así mismo, en el ámbito laboral, se han desarrollado proyectos en los cuáles se lleva un control sobre las horas trabajadas por los empleados de una empresa, así como en [11], donde se implementó una aplicación de tipo web

y de tipo móvil para la empresa Delonia Software SL, En este proyecto se relata que para la implementación de todo lo relacionado con el front-end de las aplicaciones móvil y web hacen uso de React en JavaScript, y por otro lado de Xamarin Forms en C# respectivamente; mientras que para lo relacionado con el back-end se utilizó PostgreSQL como gestor de base de datos, Tomcat como contenedor web y Uaithne para el desarrollo del servidor.

De acuerdo con lo mencionado anteriormente estas implementaciones se realizan mediante aplicaciones móviles, con el propósito de identificar la asistencia de estudiantes a las escuelas o las horas trabajadas por los empleados de una empresa, no obstante, al realizar el registro de asistencia en los laboratorios ETM se busca que sea con una aplicación dispuesta al usuario.

2.4. Distintas implementaciones de registros de asistencia en instituciones académicas

Existen una gran variedad de proyectos que han optado por hacer uso de la tecnología para realizar el registro de asistencia, haciendo uso de carnet universitario como en [12], que hace uso de este registro para el análisis de rendimiento académico y control de asistencia de los estudiantes; de igual forma, se realiza el control de asistencia para docentes haciendo uso de los lectores NFC y RFID mediante un microcontrolador Arduino, donde su función es realizar el registro de la asistencia haciendo un almacenamiento en una base de datos como se hace en [13].

Así mismo, existen aplicaciones institucionales como en [14], donde se implementa un control de asistencia para evitar el fraude, en donde se busca validar la identidad del estudiante haciendo uso de su identificación institucional, inclusive existen aplicaciones como en [15] en donde se lleva a cabo el control y registro de asistencia para el personal presente dentro de una institución académica.

Existen implementaciones a gran escala de dispositivos con lectores NFC con la finalidad de dar control de acceso a la institución, así como en [16], en donde se implementó un sistema con tecnología de lectura de los carnets institucional de los estudiantes, docentes y administrativos de la Universidad Distrital Francisco José de Caldas, permitiendo un mayor flujo de entrada y seguridad en el ingreso a la misma. También existen implementaciones en las cuáles se busca la seguridad para los niños, así como lo es en [17], en donde se implementó un sistema para el control de ingreso y retiro de los niños pertenecientes a primaria en las instituciones de la Ciudad de Guayaquil, haciendo uso de la tecnología NFC en manillas que son utilizadas por dichos niños, brindando así seguridad en la integridad de los niños.

De acuerdo con lo anterior, se observa la realización de soluciones enfocadas a la academia, estas se centran en el desarrollo de implementaciones que permitan llevar tanto el control de asistencia de los estudiantes como el control de acceso de toda la comunidad educativa a las instalaciones de la institución; por su parte el desarrollo del registro de asistencia en los laboratorios ETM se hace con el carnet institucional, además de que solo busca realizar un control de asistencia a los estudiantes y no a ningún otro miembro de la comunidad educativa.

3. MARCO TEÓRICO

3.1. Near Field Communication

La tecnología NFC ha emergido como una solución versátil y segura en el uso de tarjetas inteligentes para el uso de ellas en una variedad de aplicaciones. Basada en la inducción electromagnética, NFC permite la comunicación inalámbrica a corta distancia, lo que la hace ideal para sistemas de acceso, pagos móviles y otras aplicaciones que requieren intercambio de datos de manera rápida y segura.

Desde el punto de vista de la adopción de tecnología, NFC ha demostrado ser fácil de usar para los usuarios finales, lo que ha contribuido a su creciente aceptación en el mercado. Su capacidad para funcionar sin requerir una fuente de energía externa y la simplicidad de su infraestructura la hacen una opción atractiva para una variedad de aplicaciones.

En términos de seguridad, NFC ha mejorado significativamente gracias a la implementación de estándares de cifrado y autenticación. Esto garantiza la integridad de las transacciones y protege la información del usuario contra posibles ataques cibernéticos.

A pesar de sus numerosas ventajas, la implementación de NFC para la lectura de tarjetas también enfrenta desafíos. La compatibilidad entre distintos sistemas y la protección de los datos son factores fundamentales que deben considerarse para garantizar una adopción extensa y segura de esta tecnología [18].

La capacidad de la tecnología NFC para combinarse con biometría genera avances en el sector de la salud y la seguridad de datos, dado que se ha dado uso de aplicaciones que integran esta tecnología con autenticación biométrica para poder tener un control de acceso en áreas restringidas o validación de identidad en tiempo real. Añadido a esto, NFC ha sido utilizado para pagos móviles o control de dispositivos IoT[19].

3.2. RFID

RFID (Identificación por Radiofrecuencia) es una tecnología que identifica objetos, animales o personas por medio de ondas de radio. Consiste en etiquetas RFID (también conocidas como tags o transpondedores) con un microchip y una antena en su interior. Estas pueden ser pasivas (no requieren alimentación propia) o activas (tienen una fuente de energía interna).

En el instante que una etiqueta RFID se encuentra en el alcance de un lector RFID, la antena del lector emite una señal de radiofrecuencia que energiza la etiqueta. La etiqueta responde a la señal del lector transmitiendo su información almacenada, que puede ser un número de identificación único u otros datos

relevantes. El lector recibe esta información y la procesa, permitiendo la identificación del objeto, animal o persona que lleva la etiqueta RFID.

RFID se utiliza en unas distintas aplicaciones, como control de inventario, seguimiento de activos, identificación de mascotas, control de acceso, pago sin contacto y logística, entre otras. Es una tecnología versátil y eficiente que ha revolucionado la forma en que se gestionan y rastrean los objetos en diferentes industrias [20].

Dentro del internet de las cosas (IoT) se ha realizado la integración de esta tecnología, permitiendo una amplia conectividad y automatización de procesos, como puede ser en el sector logístico en donde RFID se convierte en una herramienta de trazabilidad de productos y seguimiento de estos en tiempo real sin intervención manual. Aunque la tecnología RFID es vulnerable a clonación de etiquetas u otras acciones que provoquen pérdidas de datos, es una tecnología que brinda un amplio número de usos en la automatización de procesos[21].

3.3. Aplicaciones web

Una aplicación web es aquella en la que un usuario puede, a través de un explorador, realizar una petición a través de internet y recibe una respuesta. Dichas aplicaciones surgen a partir del salto de páginas web con contenidos estáticos a páginas en las que se tienen contenidos dinámicos provenientes de bases de datos.

Dentro de una aplicación web se pueden distinguir tres niveles o capas: el nivel superior, en el cual se interactúa con el usuario; el nivel inferior, en donde se proporciona los datos provenientes de una base de datos; y finalmente el nivel intermedio, en donde se procesan los datos (servidor web). Entre estos tres niveles se pueden distinguir dos actores principales: cliente y servidor.

Una aplicación web también puede ser descrita como una clase de aplicación cliente/servidor, teniendo como cliente al navegador y como servidor a un servidor web. Estos se comunican bajo el protocolo HTTP (*HyperText Transfer Protocol*) el cual se sitúa en el nivel 7 del modelo OSI (*Open Systems Interconnection*). En este protocolo el cliente es un programa mediante el cual se realizan solicitudes de recursos a un servidor web, mientras que dicho servidor espera las solicitudes de conexión constantemente mediante el protocolo HTTP [22].

La incorporación de frameworks como React, Angular, y Vue han brindado facilidad en el desarrollo de aplicaciones web, permitiendo el diseño de aplicaciones robustas y de fácil escalamiento. Esto, junto con la incorporación de tecnologías como AJAX y WebSockets permiten que las aplicaciones sean de mayor velocidad y se puedan realizar acciones sin tener que recargar la

página. Con ello, se logra la aplicabilidad dentro del sector de comercio electrónico, la educación, entre otras posibles áreas[23].

3.4. Servicios web

Los servicios web son software que permiten intercambiar datos y funciones sobre una red. Su gran insumo es el lenguaje extensible de marcado XML. Se pueden definir como un conjunto de tecnologías para la transferencia de información tales como SOAP, WSDL y UDDI.

La arquitectura más utilizada para implementar servicios web es SOA (Service Oriented Architecture) la cual se define como un modelo arquitectónico de software para el diseño de modelos de negocio empaquetados como servicios. Puede ser caracterizada por las siguientes propiedades:

- Vista lógica: proporciona una imagen de los componentes del sistema con una explicación de ellos y llevándolos a la operación dentro del negocio.
- Orientación al mensaje: se define según el intercambio de información entre el solicitante y el proveedor.
- Orientación a la descripción: es descrito por metadatos procesables en una máquina.
- Granularidad: deben tener tendencia a procesar una gran cantidad de mensajes con un número pequeño de operaciones o recursos.
- Orientación a la red: deben tender a ser usados sobre una red, puede no ser obligatorio.
- Plataforma neutral: los mensajes deben ser bajo un lenguaje estándar a través de las interfaces, como puede serlo el lenguaje XML.

Existe un grupo de servicios y protocolos denominado “Web Services Protocol Stack” los cuáles se usan para definir, localizar, implementar y hacer que un servicio web tenga interacción con otro. Dicho servicio se conforma por cuatro subconjuntos: Servicio de transporte, Mensajería XML, Descripción del servicio y Descubrimiento de servicios[24].

3.5. Aplicaciones de escritorio

Las aplicaciones de escritorio son aquellas que necesariamente deben estar instaladas sobre el sistema operativo y su uso es orientado a ordenadores personales. Este tipo de aplicaciones consumen recursos directamente del ordenador donde se instala, y además dificulta el tema de portabilidad de la aplicación y mantenimiento.

Para el desarrollo de estas aplicaciones, aunque los más conocidos son Java y Python, se debe considerar el sistema operativo sobre el cual se vaya a manejar, por ejemplo, para Windows se tiene Visual C++ y Visual Basic, C/C++ con Qt o GTK y Java con AWT o Swing; para Mac se tiene Objective-C/Swift con Cocoa y para Linux se tiene C/C++ con Qt o GTK.

Estas aplicaciones pueden ser mucho más estables y robustas, así mismo más rápidas, más económicas, se puede tener una mayor seguridad[25].

3.6. Arquitecturas de aplicaciones web

Básicamente se puede decir que las aplicaciones web se realizan bajo la arquitectura cliente/servidor, aun así, existen variantes de esta:

- **Todo en un servidor:** se aloja la lógica de negocio como se observa en la Figura 1, la lógica de datos y la base de datos en un único ordenador.

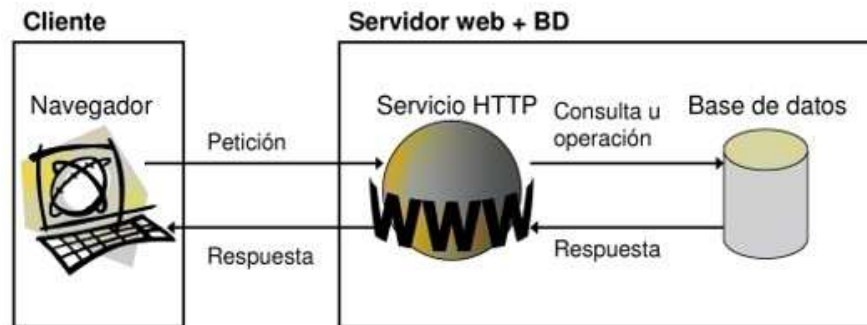


Figura 1. Arquitectura de las aplicaciones web: todo en un servidor. Fuente: [22]

- **Servidor de datos separado:** como se observa en la Figura 2, a diferencia de la arquitectura "todo en un servidor" la lógica de datos y la base de datos se operan en ordenadores distintos, teniendo uno de ellos destinado específicamente a las bases de datos.

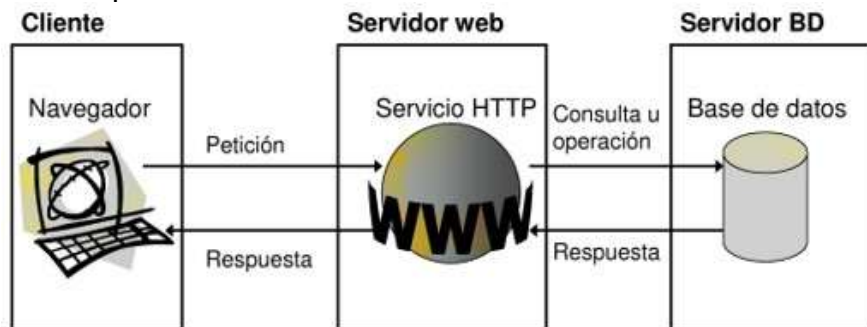


Figura 2. Arquitectura de las aplicaciones web: separación servidor de datos. Fuente: [22].

- **Todo en un servidor, con servicio de aplicaciones:** se distingue de la arquitectura “todo en un servidor” en que ya no se usa la lógica del servicio de HTTP e integra el servicio de aplicaciones en donde se gestionan los procesos de dicha lógica de negocio observado en la Figura 3.

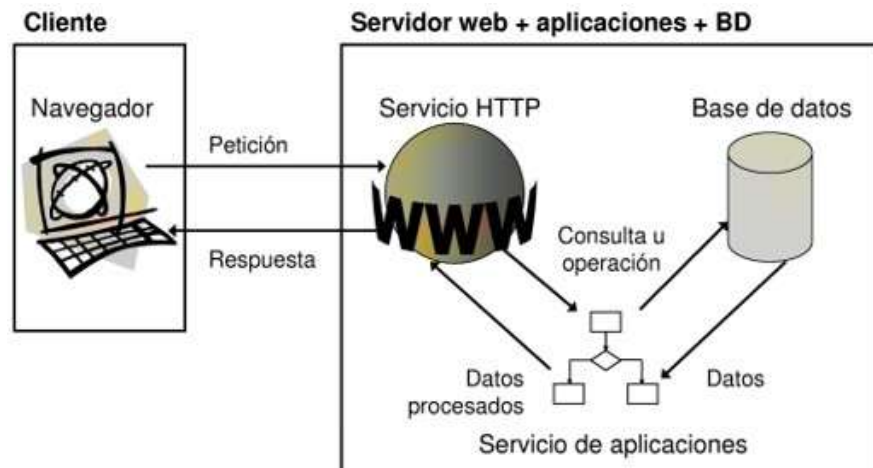


Figura 3. Arquitectura de las aplicaciones web: todo en un servidor, con servicio de aplicaciones.
Fuente: [22]

- **Servidor de datos separado, con servicio de aplicaciones:** como se observa en la Figura 4, esta arquitectura surge a partir de la arquitectura “Todo en un servidor, con servicio de aplicaciones”, y ahora se separa la lógica de datos y las bases de datos a un ordenador distinto.



Figura 4. Arquitectura de las aplicaciones web: separación de servidor de datos, con servicio de aplicaciones. Fuente:[22]

- **Todo separado:** en esta arquitectura ya se tiene todo totalmente separado como se observa en la Figura 5: las bases de datos, el servidor web, el servidor de aplicaciones y el cliente. Todas las funcionalidades básicas del servidor web se dividen en tres servidores específicos.

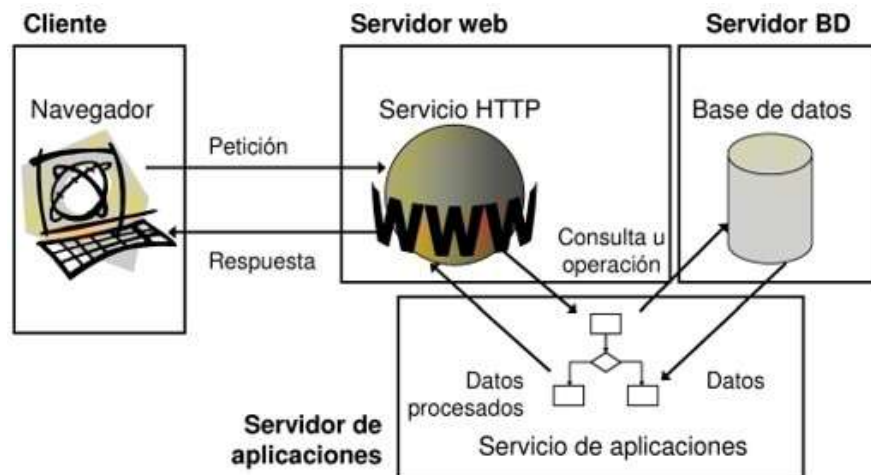


Figura 5. Arquitectura de las aplicaciones web: todo separado. Fuente: [22]

La separación de estas distintas funcionalidades se da con el fin de potenciar el desempeño del sistema permitiendo que cada servidor pueda consumir la menor cantidad de recursos para su funcionamiento; así mismo, otra ventaja que se obtiene con esta separación viene dada en términos de seguridad de la información, pues al tener los servidores separados se vuelve algo complejo el acceso a los datos [22].

3.7. NodeMCU ESP8266

El NodeMCU es una placa de desarrollo basada en el ESP8266 que es totalmente abierta en cuanto a software y hardware, en esta placa todo está dispuesto para facilitar la programación de un microcontrolador o MCU (*MicroController Unit*). Se debe tener claro que el NodeMCU no es un microcontrolador, es una placa de desarrollo que contiene un chip denominado SoC (*System on a Chip*) que dentro tiene un microcontrolador o MCU.

Esta placa de desarrollo permite implementaciones de IoT (*Internet of Things*) ya que tiene incorporada una conexión WiFi, pues con esto se pueden enviar y recibir datos o controlar entradas y salidas de forma remota e inalámbrica.

Esta placa incorpora una MCU de 32 bit de bajo consumo bajo el nombre de Tensilica L106, un módulo WiFi de 2.4GHz, una memoria RAM de 50kB, una entrada analógica de 10 bit (ADC) y 17 pines I/O (*Input/Output*) de propósito general (GPIO: *General Purpose Input/Output*) [26].

3.8. Protocolo SPI

La Interfaz Periférica Serial (SPI) es un método de comunicación digital empleado para conectar de manera eficiente diversos componentes electrónicos, sincronizando la transmisión de datos mediante un reloj común.

El protocolo SPI utiliza al menos cuatro líneas de señal:

1. SCLK (Serial Clock): esta es la señal de reloj generada por el maestro y utilizada para sincronizar la transferencia de datos.
2. MOSI (Master Out Slave In): esta es la línea a través de la cual el maestro envía datos al esclavo.
3. MISO (Master In Slave Out): esta es la línea a través de la cual el esclavo envía datos de vuelta al maestro.
4. SS (Slave Select): esta línea es utilizada por el maestro para determinar con que esclavo desea interactuar, permitiendo la comunicación con múltiples dispositivos en el mismo bus SPI.

El protocolo SPI es síncrono, lo que significa que las transmisiones de datos están sincronizadas con una señal de reloj. Además, es full-duplex, es decir, los datos se transmiten en ambas direcciones al tiempo.

SPI es ampliamente utilizado debido a su simplicidad y eficiencia en la transferencia de datos a alta velocidad. Sin embargo, a diferencia de otros protocolos como I2C, no define un esquema de direccionamiento o protocolo de manejo de colisiones, lo que puede limitar su uso en ciertas aplicaciones [27].

3.9. Protocolo I2C

El protocolo I2C, diseñado para interconectar circuitos integrados, facilita el intercambio de datos entre distintos dispositivos electrónicos, desde microcontroladores hasta sensores o manejo de memorias EEPROM a través de un bus sencillo y flexible.

El protocolo I2C utiliza dos líneas de señal:

1. SDA (Serial Data Line): Esta es la línea a través de la cual se transmiten los datos entre los dispositivos conectados.
2. SCL (Serial Clock Line): Esta es la línea de reloj para sincronizar la transferencia de datos entre los dispositivos.

I2C es un protocolo de bus de dos cables, lo que significa que todos los dispositivos están conectados a las mismas líneas de señal. Cada dispositivo

tiene una dirección única en el bus, lo que permite la comunicación directa con un dispositivo específico.

El protocolo I2C es bidireccional y soporta múltiples maestros y esclavos en el mismo bus. También es un protocolo de comunicación de corto alcance y alta velocidad, lo que lo hace ideal para aplicaciones donde se necesita una comunicación eficiente entre varios dispositivos en un sistema.

Una característica importante del protocolo I2C es su capacidad para operar en diferentes modos de velocidad (Standard Mode, Fast Mode, Fast Mode Plus, High-Speed Mode) para adaptarse a los requisitos de velocidad de diferentes aplicaciones [28].

3.10. Arduino IDE

Este entorno de desarrollo integrado (*IDE*) es una aplicación multiplataforma donde se pueden escribir y cargar programas en placas Arduino y compatibles. Gracias a librerías realizadas por terceros, es posible cargar programas en placas de desarrollo ajenas a Arduino, como es el caso del NodeMCU usado en este proyecto.

Es una aplicación que se puede instalar y usar en distintos sistemas operativos como Windows, Linux o Mac. El lenguaje más común de programación de este IDE está basado en C [29].

3.11. Bases de datos

Una base de datos es un sistema que permite el almacenamiento ordenado de datos brindando la recuperación, actualización y gestión eficiente de la información. Está compuesta por una colección de datos vinculados y un conjunto de aplicaciones para acceder a dicha información.

Hay varios tipos de bases de datos, entre los más comunes se encuentran:

Bases de datos relacionales: se almacenan los datos por medio de tablas, donde cada fila representa un registro y cada columna representa un atributo. Utilizan un lenguaje de consulta estructurado (SQL) para consultar y operar en los datos. Ejemplos de bases de datos relacionales incluyen MySQL, PostgreSQL, Oracle Database y Microsoft SQL Server.

Bases de datos NoSQL: estas bases de datos utilizan modelos de datos diferentes a las tablas relacionales, como documentos, grafos, columnas o clave-valor. Son especialmente útiles para casos de uso donde se requiere escalabilidad y flexibilidad en el esquema de datos. Ejemplos de bases de datos NoSQL incluyen MongoDB, Cassandra, Redis y Neo4j.

Bases de datos en memoria: almacenan datos en la memoria principal en lugar de en discos, lo que permite un acceso mucho más rápido a los datos. Son útiles para aplicaciones de mayor rendimiento en la lectura y escritura. Ejemplos de bases de datos en memoria incluyen Redis, Memcached e IBM solidDB.

Bases de datos distribuidas: estas bases de datos distribuyen la información entre múltiples nodos en una red, lo que permite escalabilidad horizontal y alta disponibilidad. Son útiles para sistemas que manejan grandes volúmenes de datos y requieren alta disponibilidad. Ejemplos de bases de datos distribuidas incluyen Google Bigtable, Amazon DynamoDB y Apache Cassandra.

Estos son solo algunos ejemplos de bases de datos, y hay muchos otros tipos y sistemas disponibles, cada uno con sus propias características y casos de uso específicos. La elección de una base de datos depende de los requisitos y las necesidades específicas de cada aplicación o sistema [30].

3.12. Normativa para PCB

Para el diseño de la placa de circuito impreso PCB se hace uso de la norma IPC 2221 principalmente en su capítulo 6, en donde se tienen consideraciones de tipo eléctricas para el diseño. Como se puede observar en la Tabla 1, se tiene como consideración el rango de tensión o voltaje brindando una distancia mínima entre pistas. Para el caso del dispositivo diseñado, al tener una alimentación máxima de 9 voltios en su entrada, se realiza un diseño con una distancia mínima de 0.2 milímetros.

Tabla 1. Relación entre voltaje y distancia entre pistas. Fuente:[31]

Voltaje (V)	Distancia (mm)
0 – 15	0.2
15 – 50	0.4 – 1.0
50 – 120	8.0 – 10.0

Por otro lado, en la Tabla 2 se tiene la relación del ancho mínimo de la pista, dependiendo de la cantidad de corriente que fluye por este mismo. Así como previamente se definió una distancia mínima dependiendo de la tensión máxima del sistema, también se define un ancho mínimo de la pista, en el cual se utiliza un ancho de 0.5 mm debido a que la corriente máxima medida es de 0.7 amperios. Estos valores utilizados para el dimensionamiento de las pistas se toman en relación con la norma IPC 2152, en la cual se usan estos valores presentados en las tablas para pistas con un peso de cobre de 1 onza/pie². Así mismo, estos valores son para el uso del sustrato de fibra de vidrio.

Tabla 2. Relación entre ancho de pista y corriente. Fuente:[31]

Ancho de pista (mm)	Corriente (A)
0.25	0.3
0.38	0.4
0.5	0.7
0.8	1.0

A partir de las consideraciones dadas por la norma en su capítulo 6, también se aplica el diseño a la PCB realizando una distribución entre sus sectores de potencia, de control y de circuitos de alta, media y baja frecuencia.

4. DISEÑO METODOLÓGICO

Para el desarrollo de este proyecto, se aplicó una metodología cuantitativa, ya que es fundamental obtener datos experimentales que se requieren para la recolección de información tanto antes como después de la implementación del sistema como se observa en la Figura 6.

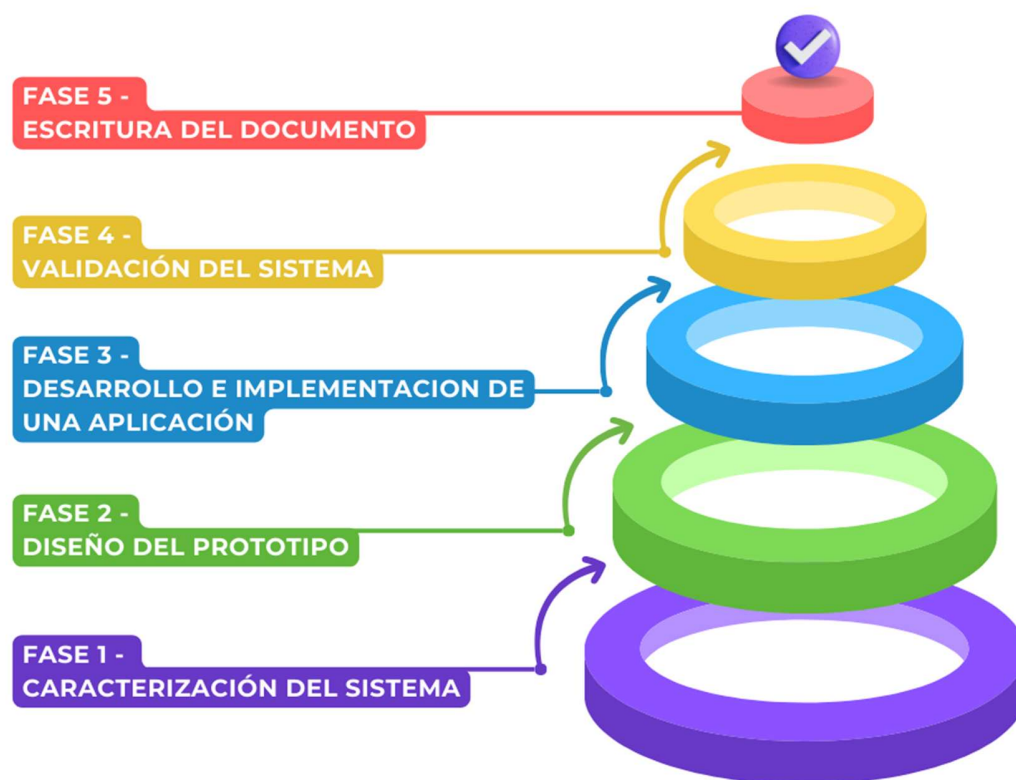


Figura 6. Diagrama metodológico. Fuente: autores.

Se desarrollaron distintas etapas en donde se buscó implementar progresivamente un sistema en el cual se propone una alternativa del método actual usado para la toma de asistencia para los estudiantes de los laboratorios ETM de la Universidad Santo Tomás, haciendo uso de herramientas informáticas para el diseño y desarrollo del hardware y software necesarios para la implementación del sistema.

Las etapas desarrolladas y sus actividades fueron:

4.1. Fase 1 – Caracterización del sistema

En esta fase, se identificaron y definieron los requisitos clave que el sistema implementado debía cumplir, así como se examinaron las características sobresalientes presentes en los proyectos previamente implementados, como se nota en [32], [14]y [17]; donde estos son implementaciones realizadas en instituciones académicas.

4.1.1. Actividad 1

Medir el tiempo empleado actualmente para el registro de asistencia realizando el conteo de manera empírica en distintos espacios académicos teniendo en cuenta que la cantidad de estudiantes en los espacios académicos varía, de esta manera poder establecer un análisis estadístico para el registro de asistencia.

4.1.2. Actividad 2

Determinar los componentes tanto físicos como digitales que debe contener el sistema, tales como un lector de tecnología NFC, el dispositivo microcontrolador a usarse, las herramientas digitales para el diseño del circuito del dispositivo, el software necesario para el desarrollo de las aplicaciones y la adecuación de la carcasa sobre la cual se implementará el hardware diseñado.

4.1.3. Actividad 3

Detallar las funciones con las que debe contar el sistema a implementar, en donde se puedan observar detalles de su funcionalidad como puede ser un identificador visual del carnet mediante una pantalla y pilotos de iluminación en donde se pueda visualizar si fue correcta la validación del registro de asistencia; añadido a esto, las características tanto en backend (capa del programa desarrollado correspondiente al funcionamiento de la aplicación y el tratamiento de datos dentro de la misma) como en frontend (capa del programa desarrollado que contiene todos los datos de visualización e interacción con el usuario) de la aplicación a desarrollar.

4.2. Fase 2 - Diseño

Esta etapa contempla el desarrollo del hardware encargado de la lectura y validación del carnet estudiantil, y para la interacción entre el sistema y los usuarios.

4.2.1. Actividad 4

Seleccionar las herramientas que se utilizan para el diseño del hardware a implementar, en donde se contemplan herramientas digitales relacionadas con el diseño y simulación de circuitos eléctricos, y así mismo el diseño de una placa PCB y la adecuación de su correspondiente carcasa.

4.2.2. Actividad 5

Diseñar y simular el circuito eléctrico a implementar, haciendo uso de la herramienta digital seleccionada, cumpliendo con las funcionalidades descritas en la fase de caracterización del sistema.

4.2.3. Actividad 6

Diseñar la placa PCB para su posterior impresión, lo cual permite llevar a cabo la implementación física de los componentes necesarios en el hardware diseñado mediante el proceso de soldadura.

4.2.4. Actividad 7

Comprobar experimentalmente el funcionamiento del hardware, asegurando que cumpla con las diversas funciones previamente identificadas en la fase de caracterización del sistema.

4.3. Fase 3 - Desarrollo e implementación de la aplicación

Se realiza el desarrollo de una aplicación que permita un acceso sencillo e interacción fluida para los usuarios al momento de visualizar el informe de asistencia generado.

4.3.1. Actividad 8

Seleccionar el lenguaje de programación y la herramienta digital apropiada para llevar a cabo el desarrollo del software.

4.3.2. Actividad 9

Desarrollar la aplicación donde se permita una interacción entre el usuario y el sistema que cumpla con los requerimientos planteados en la fase de caracterización del sistema.

4.3.3. Actividad 10

Desplegar la aplicación para el acceso de toda la comunidad que cuente con un rol de administrador dentro del sistema y que se realice un intercambio de información con el dispositivo implementado.

4.4. Fase 4 - Validación del sistema

Se verifican las características implementadas del sistema desarrollado, en donde se permita realizar el registro de asistencia haciendo uso del dispositivo piloto, donde mediante la lectura del carnet institucional de cada estudiante se realice el registro de asistencia.

4.4.1. Actividad 11

Validar el funcionamiento del sistema, donde se vean en conjunto el hardware desarrollado con la aplicación implementada para generar el registro de asistencia, así mismo, probarlo ya en una población estudiantil.

4.5. Fase 5 – Escritura del documento

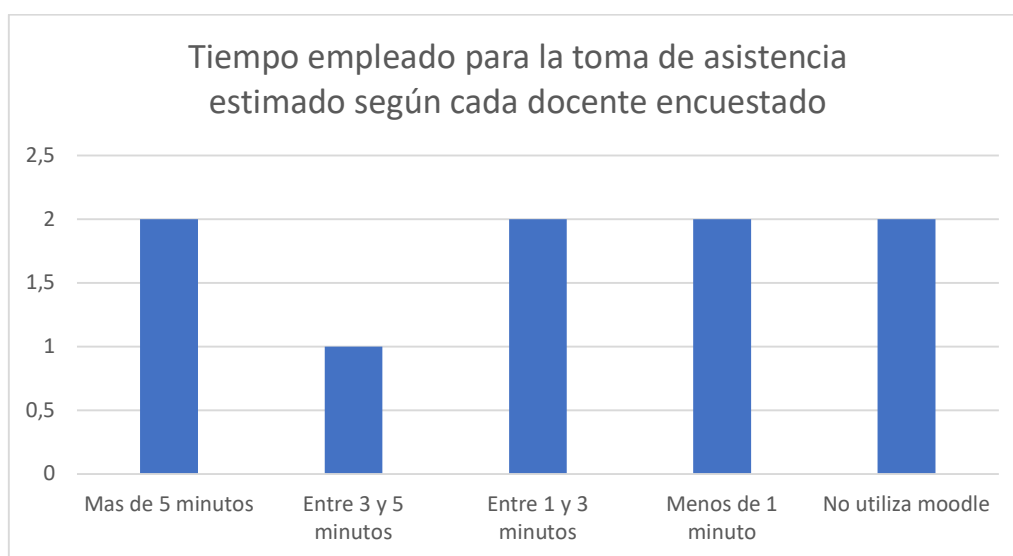
Se plantea realizar la escritura del documento a lo largo del desarrollo de las distintas actividades y las fases y de esta manera llevar un orden progresivo en donde se evidencie el desarrollo del proyecto.

5. DESARROLLO CONCEPTUAL

5.1. Desarrollo fase I – Caracterización del sistema

Se realiza el proceso de caracterización del sistema, donde se obtuvieron datos relacionados con el proceso que se lleva a cabo actualmente para el llamado a lista realizado por los distintos docentes de la Facultad de Ingeniería Electrónica de la Universidad Santo Tomás, siguiendo esto, de 11 docentes pertenecientes a la facultad de ingeniería electrónica, se lograron identificar diferentes perfiles de comportamiento en el momento de la toma de asistencia, por ejemplo, los que por lo visto no toman asistencia y los que si lo hacen, dentro de estos últimos se encuentran los que toman asistencia por Moodle al inicio de cada sesión de clase llamando una sola vez por cada estudiante (asistencia e inasistencia), los que llaman asistencia por Moodle al final de la clase (asistencia e inasistencia), los que realizan el llamado mediante una hoja en físico, donde realizan el llamado una primera vez, marcando como asistente o ausente, y al final de la sesión marca a los que llegan en el transcurso de la clase como retardo, también existe otra forma con papel pero este realiza el llamado a lista al finalizar la sesión y únicamente con asistente o ausente, de igual manera otros que hacen uso de hojas de cálculo y llenan una columna por sesión, es la misma modalidad que el de papel, únicamente que realizada mediante una hoja de cálculo.

Dentro del tiempo empleado para la toma de asistencia, se toman distintos espacios académicos en donde existan un número de estudiantes variado y a su vez se realiza una encuesta a los docentes, para de esta manera poder establecer un tiempo promedio que es empleado por los docentes para realizar la toma de asistencia. Los resultados obtenidos por este ejercicio se pueden observar en la Gráfica 1.



Gráfica 1. Respuestas docentes. Fuente: autores.

Posterior a la encuesta realizada a los distintos docentes, se procede a realizar una medición por espacios académicos del tiempo que se tardan en el llamado de asistencia, dentro del cual se tienen los datos observados en la Tabla 3.

Tabla 3. Medición inicial de tiempo en llamado de asistencia. Fuente: autores.

Espacio Academico	Tiempo en segundos
1	108
2	168
3	159
4	212
5	128
6	301
7	215
8	308
9	224

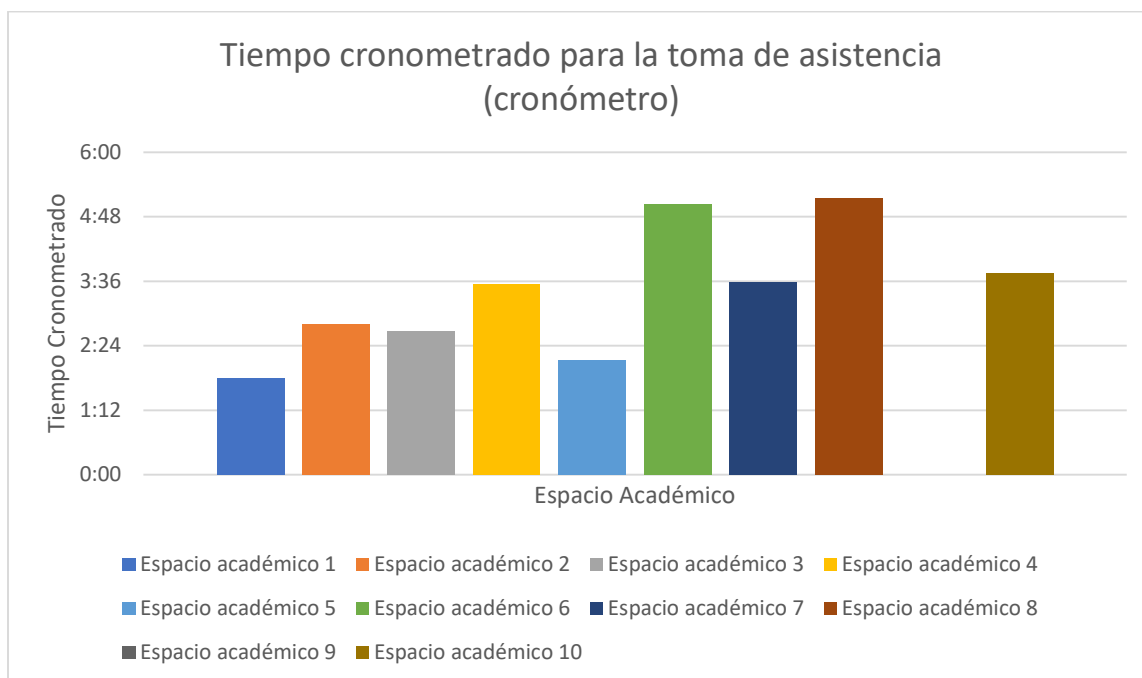
Con los datos obtenidos previamente se plantea una ecuación con el fin de obtener el tamaño de la muestra necesaria para la cantidad de sesiones que deben ser evaluadas, así como se muestra a continuación:

Ecuación 1. Ecuación para obtener el tamaño de la muestra. Fuente: autores.

$$n = \left(\frac{Z * \sigma}{E} \right)^2$$

Para este caso, $Z = 1,96$ debido a que se desea un nivel de confianza del 95%, $E = 60$ ya que es el valor máximo en segundos que se prevé que haya de error o diferencia, y $\sigma = 69.9$ ya que es el tiempo en segundos que se tiene de desviación estándar con los valores observados en la Tabla 3. Con estos valores se tiene que el número de muestras es de $n = 5.21$.

No obstante, dentro de las actividades planteadas, existe una toma del tiempo empleado por espacios académicos, por ende, se seleccionaron 10 espacios al azar, con un número de estudiantes matriculados variado y se realiza un cronometraje de 5 sesiones que corresponden a la cantidad de sesiones tomadas en una misma aula durante un periodo de cinco semanas según lo estimado en la Ecuación 1, se toma el tiempo y se obtiene el promedio de esos valores, los cuales se muestran en la Gráfica 2 y se expresan en forma de Minutos: Segundos.



Gráfica 2. Tiempo cronometrado. Fuente: autores.

5.1.1. Procesos identificados

Para el proceso de caracterización del sistema se dividen los distintos perfiles de docentes, son tres principalmente, el que llama lista con derecho a retardo, el que llama una vez lista, marca asistencia o inasistencia, en este caso no se contempla a los que no llaman a lista, puesto que se busca que, con el sistema, la asistencia se realice de forma automática. Lo único que se debe tener en cuenta es el tiempo en el que el docente quiera que la toma de asistencia este abierta y si después de finalizado este tiempo quiere que cuente retardos o no.

Se realizan diagramas de flujo horizontales, en donde se identifican los distintos procesos que se quieren llevar a cabo para la nueva toma de asistencia propuesta en este proyecto, en donde dichos procesos identificados son:

- Registro de Usuario

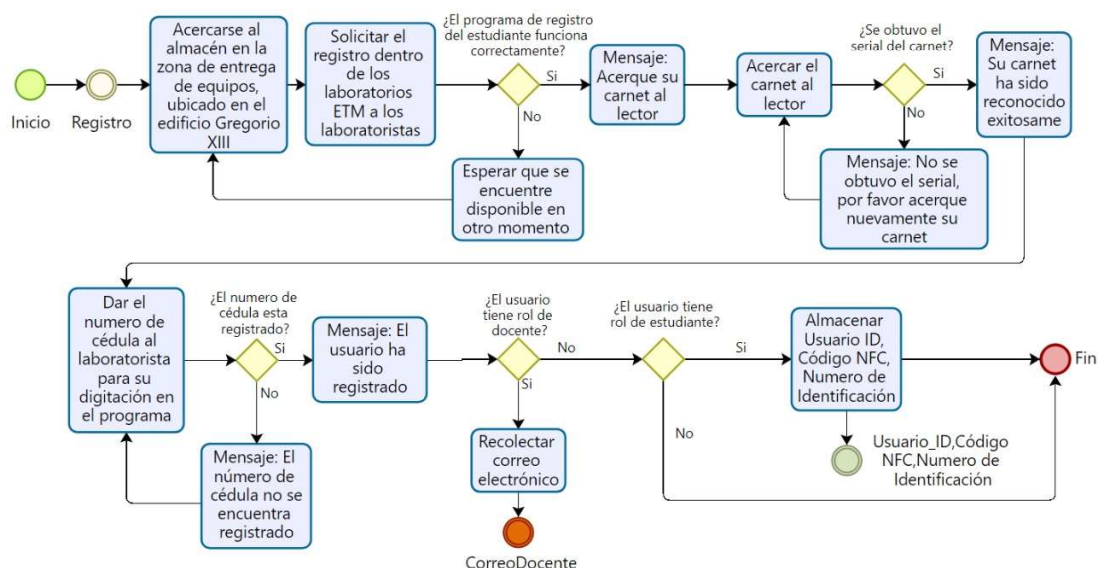


Figura 7. Proceso para el registro de usuario. Fuente: autores.

Este proceso observado en la Figura 7 será necesario para guardar el código NFC que tiene cada uno de los estudiantes que hacen uso de los laboratorios ETM, este proceso es el pilar del desarrollo de la toma de asistencia debido a que con este código NFC se busca identificar a cada uno de los estudiantes, su formato de caso de uso se puede observar en el Anexo 1.

Dentro de este proceso, el estudiante debe acercarse al almacén ubicado en los laboratorios ETM, y realizar el proceso de registro de usuario con los responsables del almacén, donde el procedimiento es acercar el lector al dispositivo, seguido de esto, buscar el número de documento y asignar el código leído con dicho documento.

- Registro de ingreso a clase

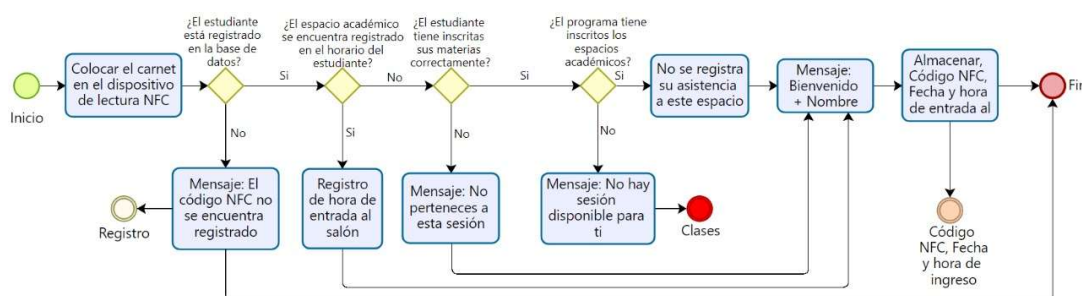


Figura 8. Proceso de registro de ingreso a clase. Fuente: autores.

Para el correcto flujo de este proceso, el estudiante tiene que cumplir con el caso de uso descrito en la Figura 8, tiene que existir una sesión en curso y a su vez, que dicha sesión corresponda a uno de los espacios académicos donde

el estudiante se encuentre inscrito, de esta manera, quedan almacenados los datos necesarios para identificar al estudiante y su hora de ingreso a la sesión. Su formato de caso de uso se puede observar en el Anexo 2.

- Registro de salida de clase

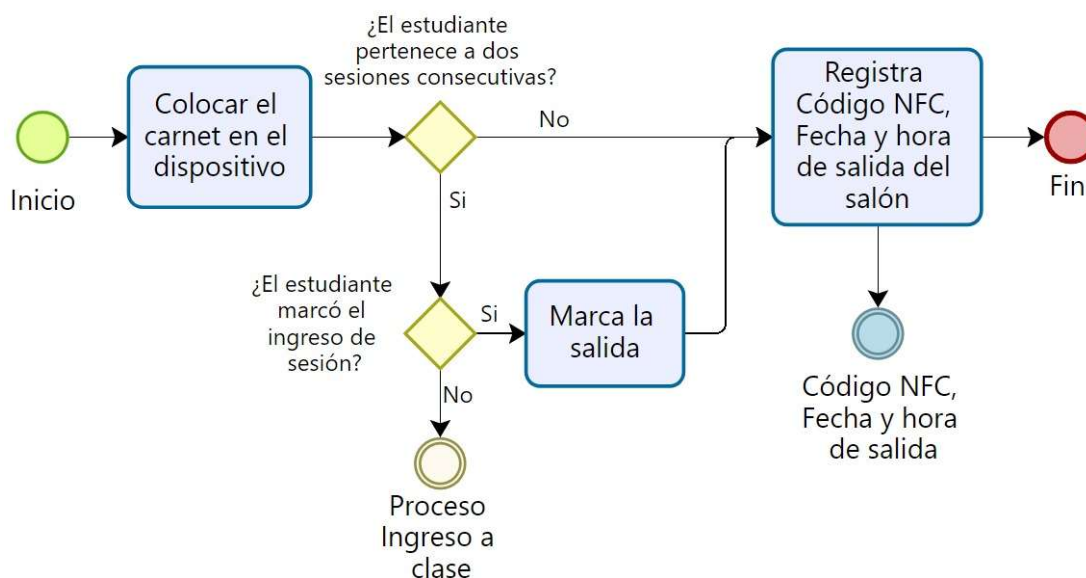


Figura 9. Proceso para el registro de salida de clase. Fuente: autores.

El proceso de la Figura 9 es necesario para asegurar que los estudiantes sí estuvieron en el aula en un tiempo considerable de la sesión, de igual forma se realiza una condición para saber el estado de asistencia si en dado caso que un estudiante pertenezca a dos sesiones consecutivas en el mismo laboratorio ETM, donde para ambos casos se haga el respectivo registro en cada sesión, su formato de caso de uso se puede observar en el Anexo 3.

- Registro de clases en laboratorios ETM

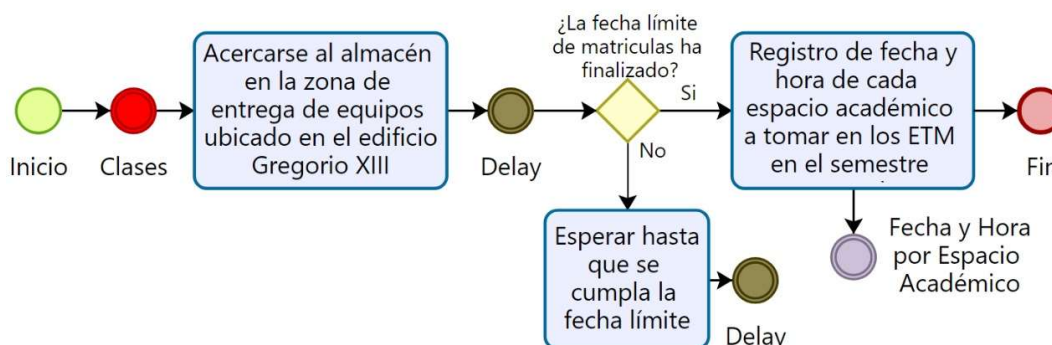


Figura 10. Proceso para el registro de clases en laboratorios ETM. Fuente: autores.

Este proceso en la Figura 10, lo deben realizar los estudiantes al principio de cada semestre para que sean asignados a los distintos espacios académicos que se toman en los laboratorios ETM, los datos correspondientes, deben ser almacenados en un archivo recopilado por el docente al inicio de cada semestre y cargado por el mismo, su formato de caso de uso se puede observar en el Anexo 4.

- Registro condiciones del responsable

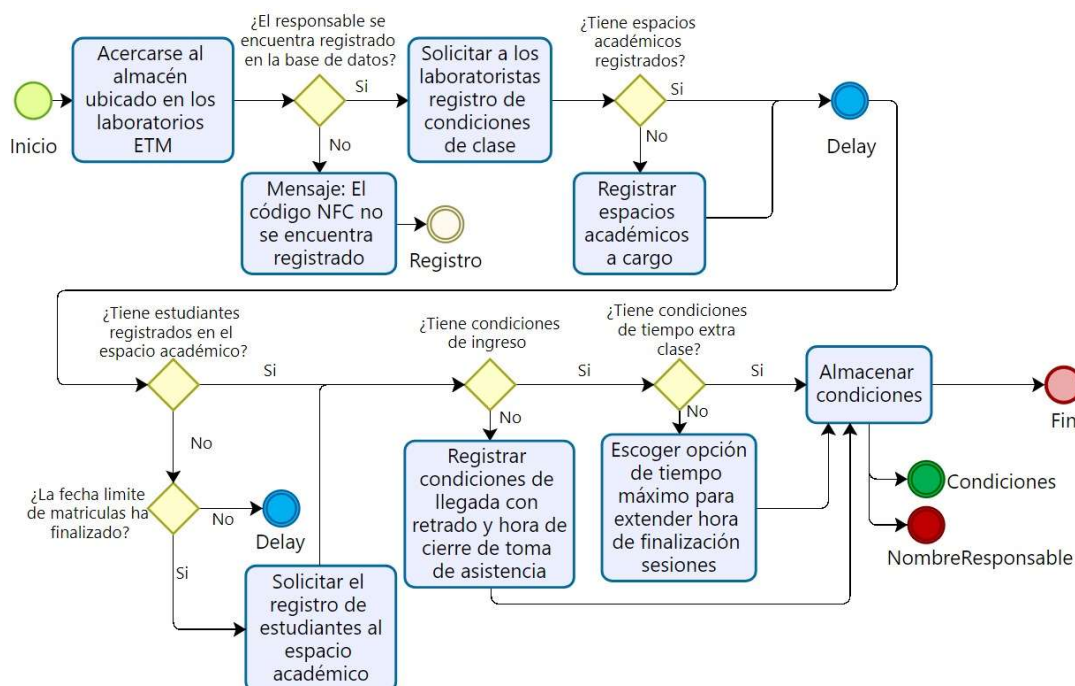


Figura 11. Proceso para el registro de condiciones del responsable. Fuente: autores.

El flujo correspondiente a este proceso consta de la verificación que el docente se encuentre registrado y que tenga espacios académicos asignados para el semestre en curso, que ya se encuentren estudiantes registrados y finalmente, aplica sus condiciones de entrada y salida de estudiantes. Su formato de caso de uso se puede observar en el Anexo 5.

- Registro de asistencia

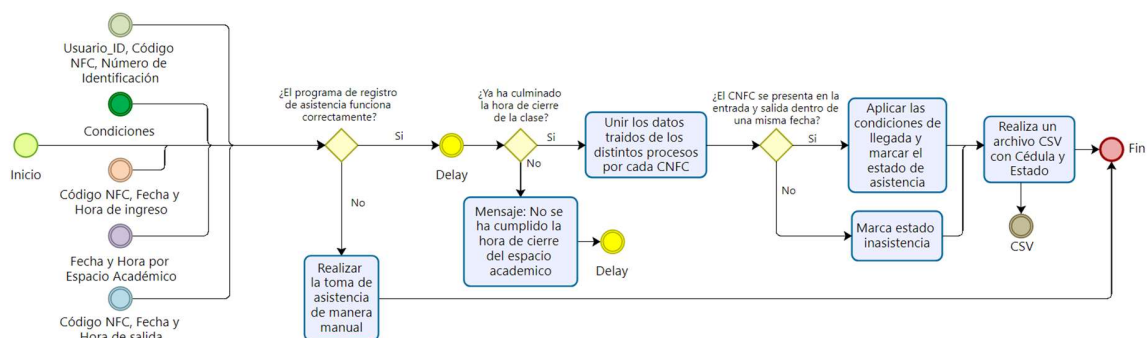


Figura 12. Proceso para el registro de asistencia. Fuente: autores.

El flujo que sigue este proceso consta de la verificación de final de sesión, cuando sea verdadero, se realizan las validaciones de los datos almacenados durante los procesos descritos anteriormente, y finalmente se pregunta si el estudiante realizó los procesos de entrada y salida adecuadamente, se concluye con el marcado de su estado de asistencia y se construye un archivo CSV. Su formato de caso de uso se observa en el Anexo 6.

- Envío de asistencia

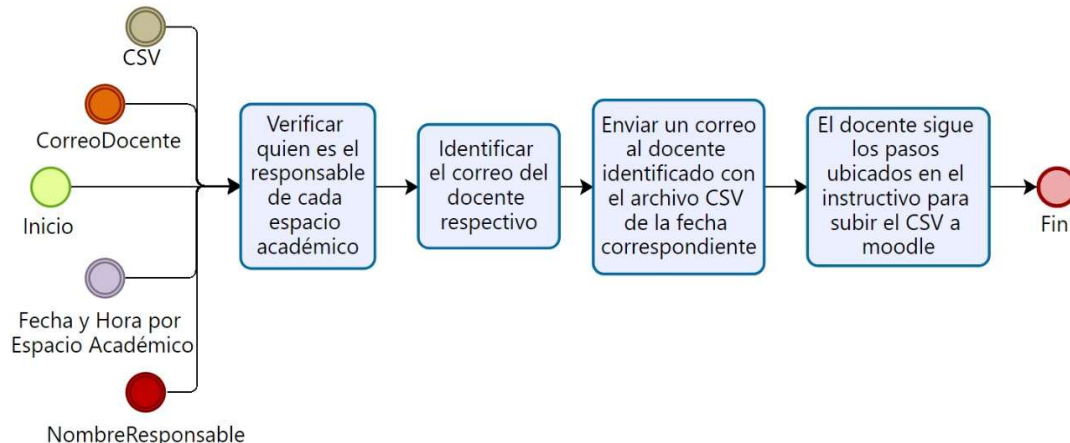


Figura 13. Proceso para el envío de asistencia. Fuente: autores.

En la Figura 13, se puede observar el proceso llevado a cabo para el envío de un archivo CSV al correo registrado del docente, este archivo se puede subir al apartado de asistencia dentro de la plataforma de Moodle, su formato de caso de uso se puede observar en el Anexo 7.

- Carga de archivo con la información de horario de grupos por programa en extensión HTML

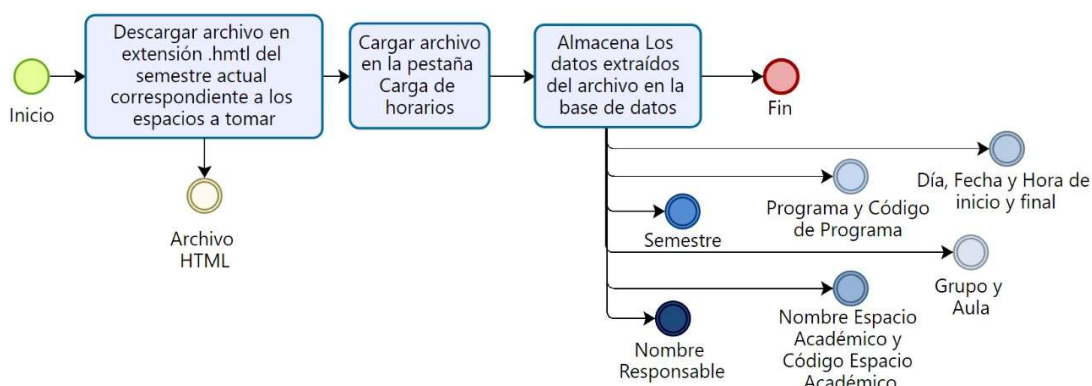


Figura 14. Proceso para carga de archivo HTML. Fuente: autores.

Para la implementación de este proceso visto en la Figura 14, es necesario obtener desde el sistema académico (SAC) información general como lo es el periodo académico, código y nombre del programa académico, de igual forma, se encuentra información específica por cada espacio académico como el nombre del espacio académico, grupo, docente a cargo, fecha de la primera sesión, fecha de la última sesión, aula, días, hora de inicio y hora de finalización en un archivo con extensión “.html” generado desde la pestaña “Horario de grupos por programa”; el cual puede ser obtenido por la comunidad académica al principio de cada semestre o por los administrativos correspondientes a cada facultad.

Este proceso cubre el transporte de información desde el SAC hasta el sistema implementado, donde se almacena toda la información descrita previamente para ser usada a lo largo de los distintos procesos de validación necesarios en el registro de asistencia, su formato de caso de uso se puede observar en el Anexo 8.

5.1.2. Casos de uso

Cada uno de los procesos descritos en la sección anterior, abarcan principalmente a 3 actores, estudiante, responsable y administrador, donde cada uno de estos, tienen distintas tareas dentro del sistema implementado como se puede observar en la Ilustración 1, estas distintas tareas deben ser realizadas en algún momento para cumplir con los distintos procesos identificados que a su vez se encuentran acompañados de un formato de caso de uso por cada proceso.

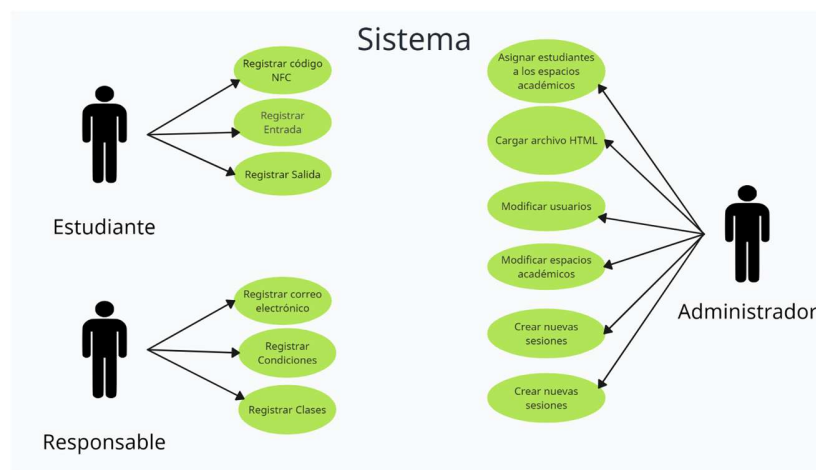


Ilustración 1. Diagrama de casos de uso. Fuente: autores

5.1.3. Componentes del sistema (Diagrama UML)

Como se puede observar en la Ilustración 2, el sistema consta de cuatro componentes principales: Computador de laboratorios, Computador de pruebas, Router (enrutador) y Dispositivo. Allí se indica la conexión que existe entre los tres primeros componentes, la cual es mediante una red local (LAN) con el nombre “usantotomas.red”. El Computador de laboratorios brinda la información contenida en su base de datos hacia el computador de pruebas por medio de esta conexión.

En el computador de pruebas se tienen tres componentes la base de datos, el servicio web y la aplicación web. La base de datos soportará el almacenamiento de los datos del sistema y a ella se conectan directamente los componentes de aplicación web y servicio web. El componente servicio web se encarga de atender las solicitudes de información de los dispositivos. Seguido, la aplicación web tiene como finalidad establecer una interacción entre el usuario y el sistema.

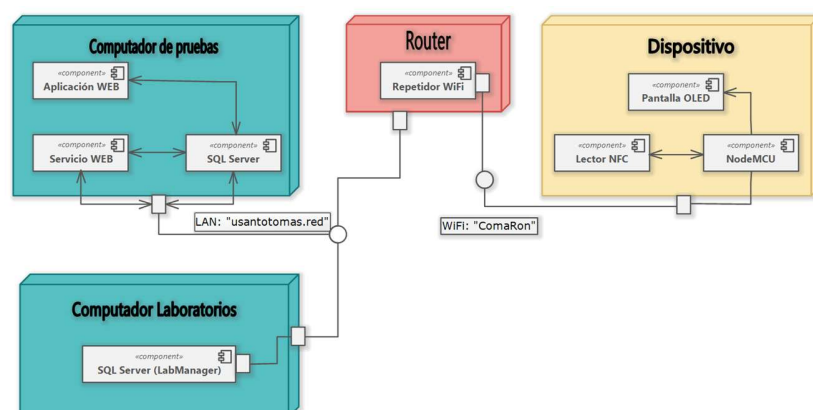


Ilustración 2. Diagrama UML de componentes del sistema. Fuente: autores.

Finalmente se tiene una conexión entre el Dispositivo y el enrutador mediante la red WiFi “ComaRon” generada por el enrutador. Este es el intermediario para el intercambio de información entre el Dispositivo y el Computador de pruebas.

Dentro del Dispositivo se tienen tres módulos de hardware: el NodeMCU, una pantalla OLED encargada de mostrar al usuario los distintos mensajes que entrega el sistema y un lector NFC (RC522) que cumple con la función de realizar la lectura a los códigos NFC de los carnets estudiantiles, todos ellos integrados en una placa PCB.

5.2. Desarrollo fase II – Diseño

A lo largo de la fase II se busca dar paso al desarrollo del proyecto en donde se desarrolla todo lo necesario para la implementación de la aplicación web.

5.2.1. Diseño PCB

Para la realización del diseño de la PCB se hace uso del software Flux¹ el cual permite el diseño, tanto del esquemático, como del modelo 2D y 3D de la placa PCB resultante.

El primer paso para la creación de la PCB es el diseño de un esquemático, donde se representen las conexiones de los componentes necesarios en el hardware, este esquemático resultante es el observado en la Ilustración 3, para el cual se tiene en cuenta que este dispositivo sea alimentado por la red eléctrica. Para ello, se implementa una configuración de regulación para un LM7805 el cual a su entrada obtiene 12V mediante un adaptador de voltaje y a la salida se obtienen 5V DC estables y con este voltaje se hace la alimentación de todos los componentes presentes en el prototipo como lo son, el NodeMCU, el lector RC522 y la pantalla OLED.

Tabla 4. Matriz PUGH. Fuente: autores.

Criterio	Microcontrolador			Lector NFC		Pantalla	
	NodeMCU ESP8266	NodeMCU ESP32	Raspberry Pi Pico W	RC522	PN532	OLED 0,96"	LCD 16x2
Precio	1	0	0	1	-1	1	1
WiFi	1	1	1	-	-	-	-
Numero de GPIO	1	1	1	-	-	-	-
Velocidad de comunicación	-	-	-	1	1	-	-
Estética	-	-	-	-	-	1	0
Evaluación	3	1	1	2	0	2	1

¹ <https://www.flux.ai/>

Los componentes que se escogieron para el diseño del prototipo son resultado de una búsqueda en cuanto a microcontroladores, lectores NFC/RFID y pantallas o dispositivos de visualización. Estos elementos se pueden observar en la Tabla 4, donde haciendo uso de la matriz PUGH se evalúan algunos de los criterios por cada elemento y se logra llegar al que se ajuste mejor a las necesidades. Allí se evalúa con “1” aquella condición que se considera buena, con “0” aquella que es aceptable y con “-1” aquella que no se considera buena opción. Así, se logra identificar el criterio de selección de los componentes vistos en el dispositivo.

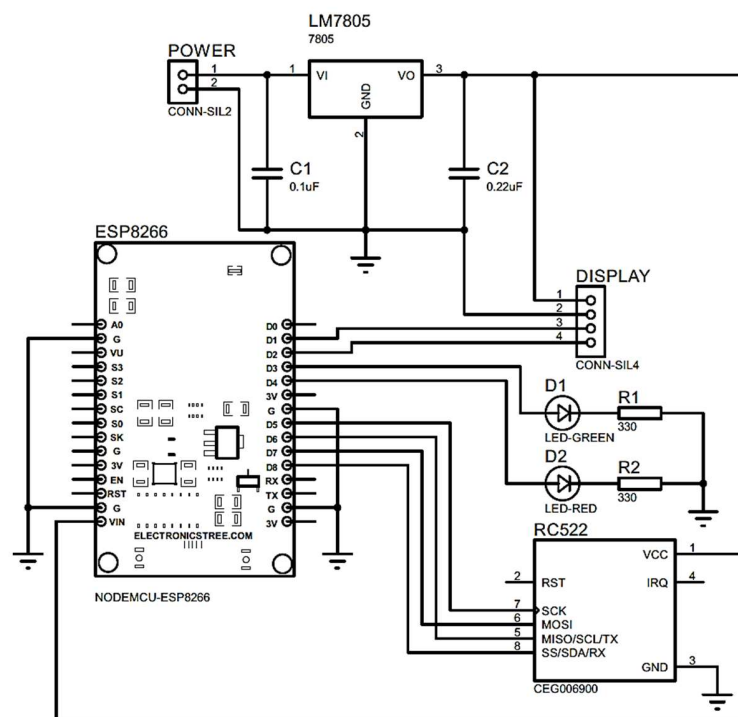


Ilustración 3. Diseño esquemático. Fuente: autores.

Como siguiente paso se realiza un montaje del circuito en una placa de pruebas (protoboard) visto en la Ilustración 4, donde se comprueba el funcionamiento del NodeMCU, el lector RC522 y la pantalla OLED, mediante una aplicación de prueba en donde se realiza la lectura del código NFC presente en una tarjeta de esta tecnología para posteriormente visualizar el código identificador en la pantalla OLED.

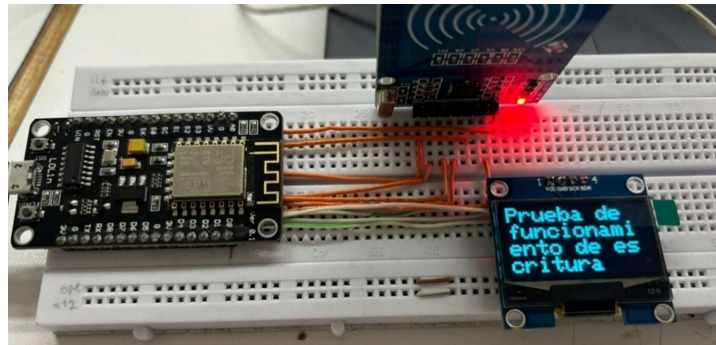


Ilustración 4. Funcionamiento protoboard. Fuente: autores.

Después de realizar las pruebas se procede a diseñar la tarjeta impresa del circuito (PCB) implementado en protoboard. El software Flux, también ofrece un apartado para empezar el diseño 2D de la PCB, en este se realiza la ubicación de los distintos componentes empleados en el esquemático y el software automáticamente traza líneas de unión entre cada componente tal cual como se hizo en el paso anterior, de esta manera se reduce la probabilidad de error al trazar los caminos y se procede a realizar el trazado como se ve en la Ilustración 5, finalmente se exporta un archivo de tipo Gerber para enviarlo a la respectiva empresa que hace la impresión de la placa.

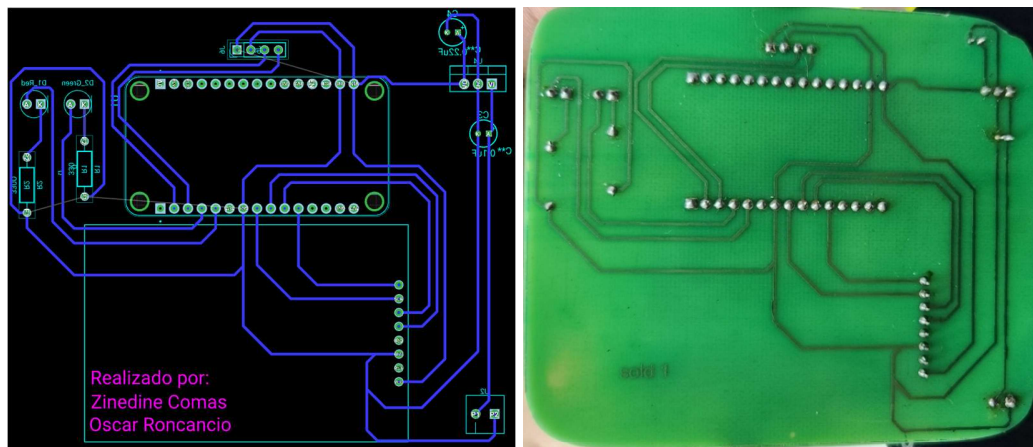


Ilustración 5. Diseño 2D. Fuente: autores.

Al momento de tener la placa, se procede a soldar los distintos componentes y armar el prototipo diseñado desde el software. Al finalizar este proceso se ensambla en la carcasa y se procede a la prueba de funcionamiento como se observa en la Ilustración 6, en el lado izquierdo se puede ver la tarjeta ubicada en el lector y en la derecha el código NFC leído.



Ilustración 6. Prueba de funcionamiento prototipo. Fuente: autores.

5.2.2. Diseño del modelo de datos

El modelo de datos se implementa en una base de datos relacional MSSQL Server, en donde se almacenan los datos necesarios para obtener los registros de las distintas sesiones de clase que se dan a lo largo del semestre y demás información necesaria para el funcionamiento del sistema. El modelo de datos propuesto se ve en la Ilustración 7, en donde se almacena toda la información necesaria. Se plantean las siguientes tablas principales:

- **Estudiante y Responsable:** Es importante dividir que parte de la población estudiantil son estudiantes y quienes serán los responsables de cada aula por espacio académico, dentro de esta tabla se encontrarán tanto docentes como los laboratoristas.
- **Programa:** Una tabla que almacene los distintos espacios académicos que se encuentren disponibles en los distintos programas.
- **Sesión:** Tabla que almacene las distintas sesiones que se tendrán por espacio académico en un semestre.

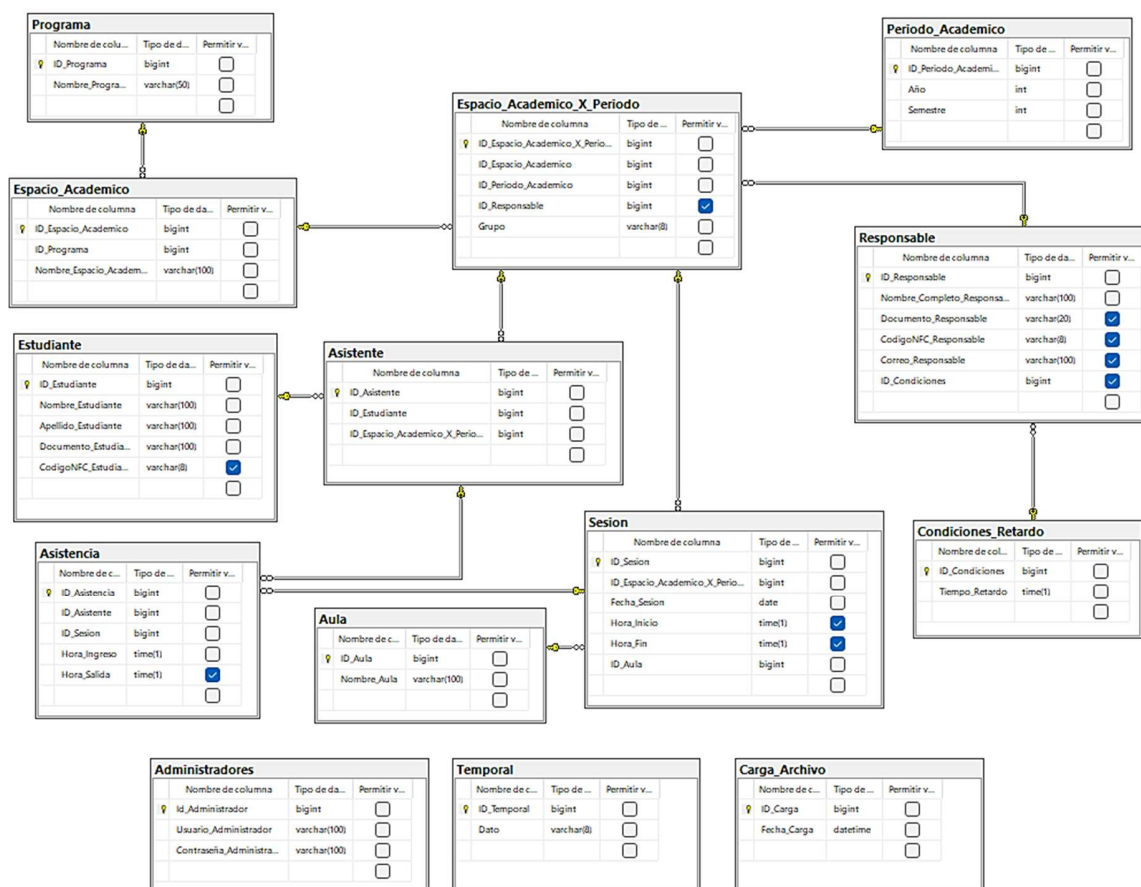


Ilustración 7. Modelo de base de datos implementado. Fuente: autores.

Las distintas tablas presentadas se llenan automáticamente por la aplicación a medida que los distintos procesos se implementan; con la carga del archivo HTML que contiene la información descrita en la fase 1, se llenan las tablas Programa, Espacio_Académico, Periodo_Académico, Responsable, Aula y Espacio_Academico_X_Periodo; la tabla Asistente se llena mediante la pestaña de asignación de espacios académicos de la aplicación que se explica más adelante.

De igual forma, este modelo de base de datos cumple con la tercera forma normal con el fin de mantener la integridad de los datos y la accesibilidad a los datos, al igual que contribuye a que el espacio de almacenamiento se reduzca y la velocidad de procesamiento se incremente [33].

Todas las tablas del modelo se explican a continuación, distribuidas en cuatro grandes grupos: grupo de estudiante, grupo de responsable, grupo de programa y grupo de sesión, y tres tablas individuales: Administrador, Temporal y Carga_Archivo.

5.2.2.1. Grupo de Estudiante

En el grupo que se observa en la Ilustración 8, se agrupan las tablas que tienen información referente al Estudiante principalmente.

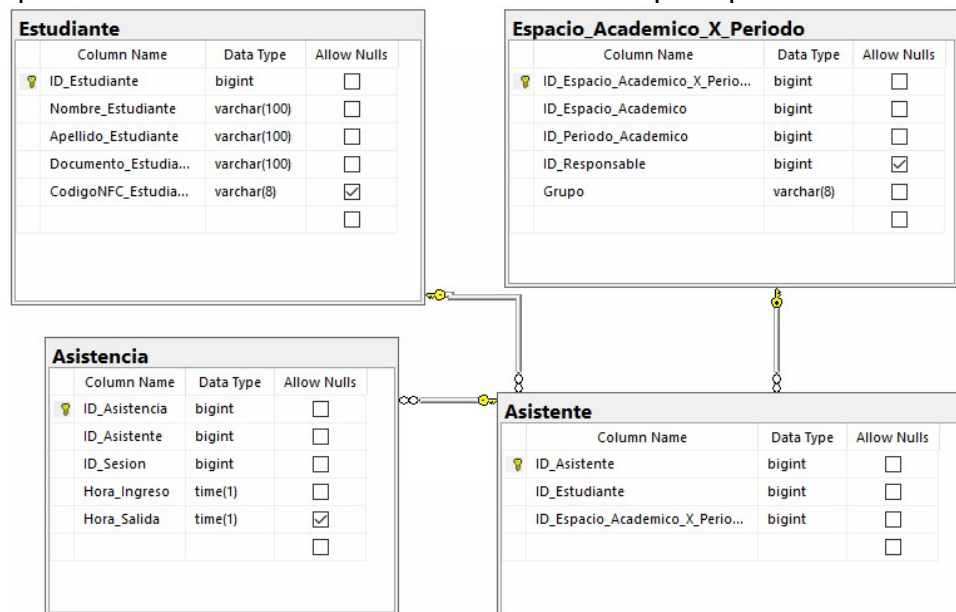


Ilustración 8. Grupo estudiante. Fuente: autores.

La primera de estas es la tabla Estudiante, a continuación, se enuncian sus campos, tipo y descripción:

- ID_Estudiante (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Nombre_Estudiante (varchar). Almacena el nombre del estudiante en un máximo de 50 letras y/o espacios.
- Apellido_Estudiante (varchar). Almacena el apellido del estudiante en un máximo de 50 letras y/o espacios.
- Documento_Estudiante (varchar). Almacena el documento del estudiante en un máximo de 50 caracteres.
- CodigoNFC_Estudiante (varchar). Almacena el código NFC del estudiante en un máximo de 8 caracteres.

Esta tabla tiene como finalidad almacenar la información de todos los estudiantes que se encuentren registrados en la base de datos del sistema de laboratorios LabManager, quienes en su mayoría son estudiantes pertenecientes a las Facultades de Ingeniería Electrónica, Ingeniería de Telecomunicaciones e Ingeniería Mecánica; de esta manera con el ID_Estudiante, es posible vincularlo a la tabla de Asistente con una relación “One-to-many”.

Así como la tabla de Asistente que cuenta con los siguientes campos:

- ID_Asistente (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- ID_Estudiante (bigint). Se obtiene el identificador de la tabla Estudiante.
- ID_Espacio_Academico_X_Periodo (bigint). Se obtiene identificador de la tabla Espacio_Academico_X_Periodo.

Esta tabla es requerida por el sistema para establecer una relación entre las tablas Estudiante y Espacio_Academico_X_Periodo, de esta manera es posible decir que muchos estudiantes pueden asistir a muchos espacios académicos en un mismo periodo.

De igual forma para la tabla de Asistencia se almacenan los campos de:

- ID_Asistencia (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- ID_Asistente (bigint). Se obtiene el identificador de la tabla de Asistente.
- ID_Sesion (bigint). Se obtiene el identificador de la tabla de Sesión.
- Hora_Ingreso (time). Almacena la hora en la que se lee el carnet por primera vez durante una sesión.
- Hora_Salida (time). Almacena la hora en la que se lee el carnet por segunda vez durante la misma sesión.

En esta tabla es en donde se almacena la información de la asistencia de un estudiante, la primera vez que el estudiante pone el carnet en el dispositivo se registra su hora de ingreso en el campo Hora_Ingreso y para la segunda vez que el estudiante registre su carnet, se actualiza el registro creado con anterioridad modificando el campo Hora_Salida, esto sin importar el tiempo entre ambos eventos, es decir, si el estudiante pasa el carnet 3 minutos después de su hora de ingreso, se le asignará automáticamente en hora de salida.

Los datos de Nombre_Estudiante, Apellido_Estudiante y Documento_Estudiante son traídos de la base de datos LabManager la cual se encuentra alojada en el almacén de los Laboratorios ETM de la Universidad Santo Tomás. El campo CodigONFC_Estudiante se llena en la pestaña “Modificación de Usuario” de la aplicación implementada.

5.2.2.2. Grupo Programa

El grupo denominado Programa, como se puede observar en la Ilustración 9, consta de 4 tablas que corresponden a Programa, Espacio_Academico, Espacio_Academico_X_Periodo y Periodo_Academico, con este grupo se obtienen los datos relacionados con cada facultad debido a que cada una maneja distintos espacios académicos y a su vez pueden o no cambiar estos a lo largo de los distintos periodos académicos.

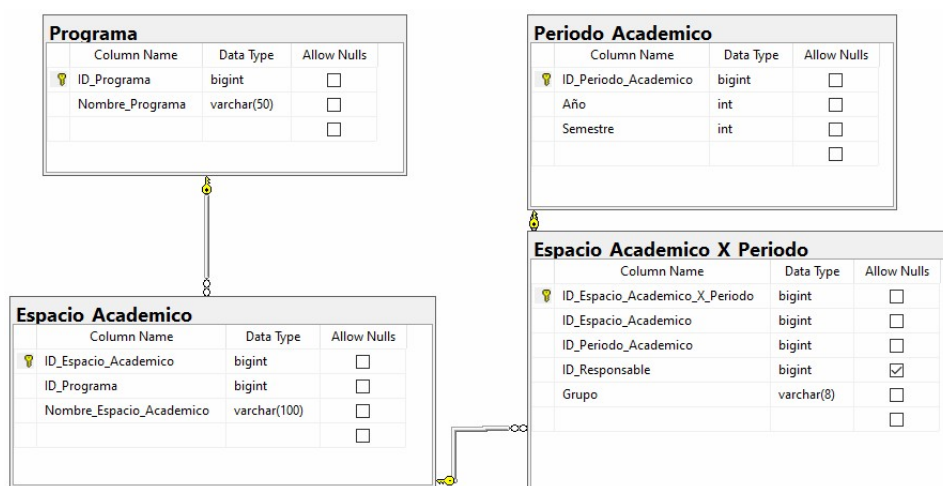


Ilustración 9. Grupo Programa. Fuente: autores.

Para la tabla Programa se manejan dos campos, que corresponden a:

- ID_Programa (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Nombre_Programa (varchar). Se almacena el nombre del programa en un máximo de 50 caracteres.

Esta tabla se llena automáticamente al realizar el proceso de Carga de Horarios con el nombre del programa obtenido del reporte obtenido del sistema académico en formato HTML.

Para la tabla Espacio_Académico se maneja los campos de:

- ID_Espacio_Académico (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- ID_Programa (bigint). Se trae el identificador de la tabla de Programa para establecer una relación entre la tabla de Programa y Espacio_Académico.

- Nombre_Espacio_Académico(varchar). Almacena el nombre del espacio académico en un máximo de 100 caracteres.

Esta tabla tiene como finalidad almacenar todos los distintos espacios académicos presentes en las distintas facultades, debido a lo anterior, se hace una relación “One-to-many” de la tabla de Programa hacia la tabla de Espacio_Académico. Por ende, se puede decir que un programa tiene muchos espacios académicos.

Para la tabla de Periodo_Académico se almacenan los campos correspondientes a:

- ID_Periodo_Academico (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Año (int). Almacena el año correspondiente al reporte cargado.
- Semestre (int). Almacena el semestre correspondiente al reporte cargado.

Esta tabla tiene como propósito realizar una relación “One-to-many” entre las tablas de Periodo_Académico y Espacio_Académico_X_Periodo de forma que se puede decir que dentro de un periodo académico se pueden tener muchos espacios académicos.

Para la tabla de Espacio_Academico_X_Periodo se cuentan con los campos de:

- ID_Espacio_Academico_X_Periodo (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- ID_Espacio_Académico (bigint). Almacena el identificador correspondiente al nombre de cada espacio académico.
- ID_Responsable (bigint). Se almacena el identificador del responsable de cada espacio académico durante un semestre.
- Grupo (varchar). Almacena el “Alias” del grupo en un máximo de 8 caracteres.

Esta tabla tiene como finalidad establecer la diferencia para los casos donde hay dos grupos por un mismo espacio académico, teniendo en cuenta que pueden tener el mismo docente en todos los grupos, o el docente puede variar, de esta manera se hace una relación “One-to-many” entre la tabla de Espacio_Académico y Espacio_Academico_X_Periodo, así mismo se puede decir que un

espacio académico se puede presentar en distintos espacios académicos por periodo.

5.2.2.3. Grupo Responsable

Dentro del grupo de responsable, como se puede observar en la Ilustración 10, consta de tres tablas, pero se relaciona con otro grupo, esto quiere decir que tiene relación con otro grupo anteriormente mencionado.

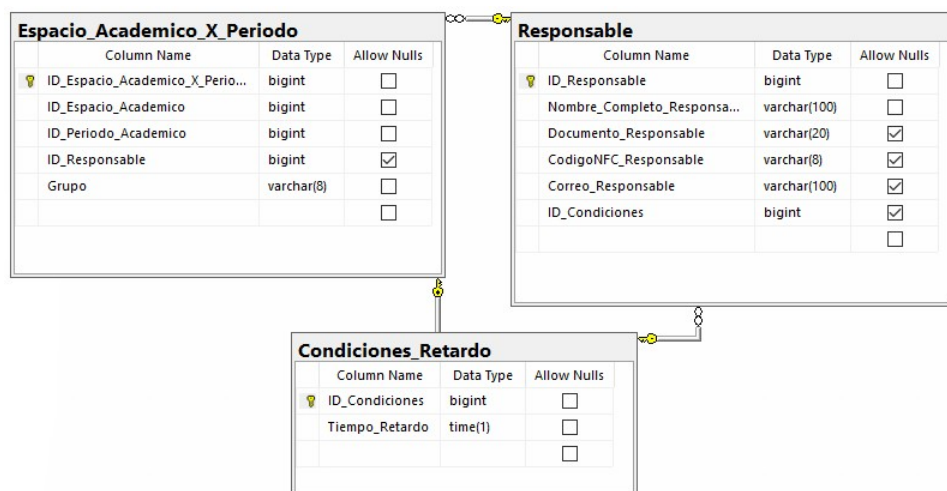


Ilustración 10. Grupo Responsable. Fuente: autores.

Para la tabla de responsable, se cuenta con los campos de:

- ID_Responsable (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Nombre_Completo_Responsable (varchar). Almacena el nombre del responsable en un máximo de 100 caracteres.
- Documento_Responsable (varchar). Almacena el documento en un máximo de 20 caracteres.
- CodigoNFC_Responsable (varchar). Almacena el código NFC en un máximo de 8 caracteres.
- Correo_Responsable (varchar). Almacena el correo del responsable en un máximo de 100 caracteres.
- ID_Condiciones (bigint). Se almacena el identificador de las condiciones seleccionadas por el docente, que se encuentran presentes en la tabla de Condiciones.

Esta tabla tiene como finalidad almacenar los datos necesarios para la identificación del perfil de cada docente; esta tabla tiene una relación “One-to-many” entre las tablas Responsable y Espacio_Academico_X_Periodo, de esta manera se puede decir que

un responsable puede pertenecer a varios espacios académicos por periodo.

Para la tabla de Condiciones_Retardo se cuenta con los campos de:

- ID_Condiciones (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Tiempo_Retardo (time). Tiempo en minutos.

Esta tabla tiene como finalidad almacenar las diferentes opciones de tiempos de retardo que el sistema va a permitir al docente elegir para sus espacios académicos. Cuenta con una relación “One-to-many” entre las tablas Condiciones_Retardo y Responsable, por ende, se dice que una condición la pueden utilizar varios docentes.

5.2.2.4. Grupo Sesión

Dentro del grupo de sesión se tiene tres tablas como se puede observar en la Ilustración 11 donde, las tablas de Aula y Sesión pertenecen a este grupo y la tabla de Espacio_Academico_X_Periodo pertenece a un grupo anteriormente mencionado.

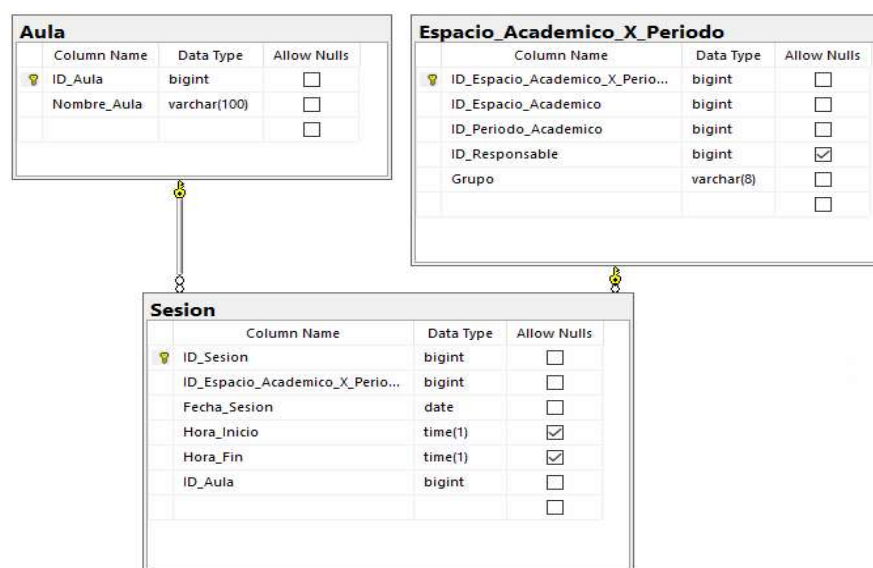


Ilustración 11. Grupo Sesión. Fuente: autores.

Para la tabla Aula se cuenta con los campos de:

- ID_Aula(bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- Nombre_Aula(varchar). Almacena el nombre del aula extraído del archivo de reporte en un máximo de 100 caracteres.

En esta tabla se almacena la información correspondiente al nombre de las distintas aulas en donde se desarrollan las sesiones de clase para cada espacio académico, el campo ID_Aula es el utilizado para la identificación del aula que es reportada desde el dispositivo de manera simple.

Para la tabla Sesión se almacenan los datos correspondientes a los campos de:

- ID_Sesion (bigint). Asigna un identificador único al registro para ser usado en otras tablas y simplificar su búsqueda.
- ID_Espacio_Academico_X_Periodo (bigint). Se almacena el identificador generado por la tabla de Espacio_Academico_X_Periodo para generar una relación entre las sesiones y los espacios académicos.
- Fecha_Sesión (date). Almacena las sesiones generadas desde la fecha de inicio y la fecha de finalización del semestre.
- Hora_Inicio (time). Almacena la hora de inicio de la sesión, obtenida del archivo de carga.
- Hora_Fin (time). Almacena la hora de finalización de la sesión, obtenida del archivo de carga.
- ID_Aula (bigint). Se trae el identificador para generar la relación “One-to-many” con la tabla de Aula.

En esta tabla se almacenan todas las sesiones de los distintos espacios académicos por semestre teniendo en cuenta que se repiten a lo largo de todas las semanas contenidas en el semestre. Estas sesiones son generadas con la fecha de inicio y fecha de finalización, presentes en el reporte obtenido desde el sistema académico.

5.2.2.5. Administrador

Se requiere de una tabla de administrador, donde sean almacenados los datos requeridos para el inicio de sesión dentro de la aplicación web desarrollada, esta es observada en la Ilustración 12.

Administrador			
	Nombre de columna	Tipo de dat...	Permitir va...
🔑	Id_Administrador	bigint	☐
	Usuario_Administrador	varchar(100)	☐
	Contraseña_Administra...	varchar(100)	☐
			☐

Ilustración 12. Tabla Administrador. Fuente: autores.

Esta tabla contiene los siguientes campos:

- Id_Administrador (bigint). Asigna un identificador único a cada registro dentro de la tabla.
- Usuario_Administrador (varchar). Contiene el usuario con el que se va a realizar login dentro de la aplicación web.
- Contraseña_Administrador (varchar). Contiene la contraseña relacionada con el usuario para realizar el login.

Esta tabla es consultada por la aplicación web, al momento de realizar la validación del usuario y contraseña para proporcionar un control de acceso al sistema y que no todas las personas con acceso a la ruta puedan hacer uso de la aplicación web.

5.2.2.6. Temporal

En la tabla presentada en la Ilustración 13, habrá únicamente un registro a la vez, donde esta cumple la función de almacenar el código NFC leído por el dispositivo diseñado, cuando se está asignando el código al estudiante en la aplicación web.

Temporal			
	Column Name	Data Type	Allow Nulls
	ID_Temporal	bigint	☐
	Dato	varchar(8)	☐
			☐

Ilustración 13. Tabla Temporal. Fuente: autores.

Cuenta con los campos de:

- ID_Temporal (bigint). Asigna un identificador único a cada registro dentro de la tabla.
- Dato (varchar). Almacena el dato leído por el dispositivo en un máximo de 8 caracteres.

Se plantea únicamente un espacio de ocho caracteres debido a que ese es el número exacto que contienen los códigos NFC de los carnets.

5.2.2.7. Carga_Archivo

En la tabla observada en la Ilustración 14, se crea un registro cada que se ejecute la función de cargar archivo en la pestaña de carga de horarios, con el objetivo de llevar un historial de fechas en las que se realiza la carga de archivos de horarios.

Carga_Archivo			
	Column Name	Data Type	Allow Nulls
🔑	ID_Carga	bigint	☐
	Fecha_Carga	datetime	☐
	Nombre_Arc...	varchar(50)	☒
			☐

Ilustración 14. Tabla Carga de Archivo. Fuente: autores.

Cuenta con los campos de:

- ID_Carga (bigint): Asigna un identificador único a cada registro dentro de la tabla.
- Fecha_Carga (datetime): Almacena la fecha y hora en la que es cargado el archivo.
- Nombre_Archivo: Almacena el nombre del archivo cargado.

5.2.3. Diseño Aplicación modo configuración

Como se observa en la Ilustración 15, se plantea el diseño de la interfaz de usuario para la aplicación de modo de configuración para el NodeMCU, donde se maneja únicamente una ventana en donde se digita un comando en la sección de “texto a enviar” seguido de un clic en el botón de “Enviar”, este procedimiento permite realizar las distintas configuraciones para los requerimientos que el programa necesite, como lo es el nombre de la red WiFi, la contraseña, y el aula desde la cual el dispositivo va a reportar, y dentro de las cajas observadas en la parte derecha se ven los datos que se encuentren almacenados actualmente en el NodeMCU, ya sea desde la memoria o desde la configuración que se vaya realizando paso a paso.



Titulo de la App

Texto a enviar

Respuesta

Puertos

Direccion SSID

Password

Aula

Enviar

Ilustración 15. Diseño app modo configuración. Fuente: autores

5.2.4. Diseño aplicación web

5.2.4.1. Arquitectura

Para el diseño de la aplicación web, se hace uso de una arquitectura basada en 3 capas, que como se puede observar en la Ilustración 16, la primera capa es la de datos, en donde se tiene el almacenamiento de información de la aplicación y se relaciona con cada una de las tablas de la base de datos; la segunda capa es la de negocio, allí, se tienen las funcionalidades básicas CRUD (*Create, Read, Update, Delete*) y brinda la comunicación entre la capa de datos y la capa de aplicación web; la tercera capa, para este proyecto, está dividida en dos componentes: Aplicación WEB y Servicio WEB. En el primero se realiza todo lo correspondiente a funcionalidad con la interacción de usuario, brindándole y a la vez recibiendo información de dicho usuario, implementado con el patrón MVC (Modelo – Vista - Controlador), donde el modelo es quien representa los datos para la aplicación, la vista corresponde a la interfaz de usuario y el controlador es el que realiza la lógica en la interacción entre los datos del modelo y la exportación de estos por la vista. Por otro lado, el Servicio WEB contiene las funciones necesarias para generar un registro de asistencia de manera automática y atender a las solicitudes realizadas por el dispositivo mediante el protocolo HTTP.

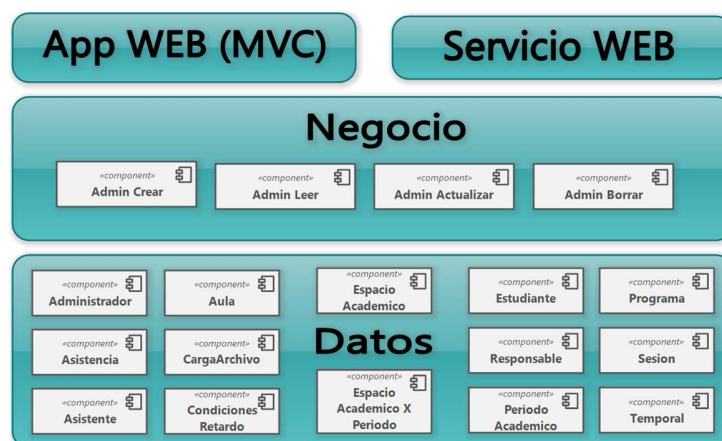


Ilustración 16. Arquitectura de aplicación web. Fuente: autores.

5.2.4.2. Wireframe interfaz de usuario

Para la implementación de la interfaz de usuario en la aplicación web se realiza el diseño de 3 distintos *wireframes* (representación visual en escala de grises de la estructura de una página web²), los cuales son la base para el diseño final. En la Ilustración 17 se tiene el diseño de la

2

<https://www.lucidchart.com/pages/es/que-es-un-wireframe-para-un-sitio-web#:~:text=En%20el%20dise%C3%B1o%20web%2C%20un,o%20pantalla%20de%20aplicaci%C3%B3n%20m%C3%B3vil.>

página de inicio de la aplicación, en donde se busca brindar al usuario el nombre del sistema junto con el logo de la Universidad Santo Tomás y con dos campos donde el usuario pueda digitar su usuario y contraseña para validar el acceso.

Registro de asistencia para los laboratorios ETM de la Universidad Santo Tomas



Ilustración 17. Wireframe para ingreso de usuario. Fuente: autores.

Como segundo *wireframe*, observado en la Ilustración 18, se realiza el diseño de la página en donde se solicitará al usuario el archivo de formato HTML obtenido desde el sistema académico, con la opción de poder visualizar en pantalla el contenido de su documento y posteriormente poderlo cargar al sistema.

Recibir HTML

Seleccione el archivo HTML

Seleccionar archivo

Visualizar archivo

Cargar archivo

Reporte a cargar

Visualización de archivo HTML

Ilustración 18. Wireframe para recepción de archivo en formato HTML. Fuente: autores.

Finalmente, se tiene el diseño de una pestaña para poder tabular la información correspondiente de cada tipo de usuario. Como se puede observar en la Ilustración 19, se presenta al usuario un desplegable con la lista de los tipos de usuario que puede allí encontrar, tabulando la información contenida en cada uno de estos, y brindando al usuario un botón para poder realizar la modificación de cada uno de estos registros tabulados.

Modificación datos de Usuario

Seleccione el tipo de usuario a modificar

Tipo de usuario v

Usuarios registrados:

Nombre	Apellido	Documento de identidad	Código NFC	Modificar
				Modificar

Ilustración 19. Wireframe para modificación de datos de usuario. Fuente: autores.

5.3. Desarrollo fase III – Desarrollo e implementación de la aplicación

Con el inicio de la fase III, se da paso al desarrollo e implementación de todo lo necesario en cuanto a software para generar un registro de asistencia automático, generando lo correspondiente a backend en donde se tiene la interacción con los datos por medio de las funciones básicas CRUD, y así mismo desde el Frontend poder incorporar estas funciones con la interfaz diseñada para el usuario.

5.3.1. Desarrollo de backend

Para iniciar el desarrollo del backend, se realiza la correspondiente capa de Datos, observada en la Ilustración 16, para tener una interacción entre la aplicación y la base de datos diseñada. Luego de ello, se realiza la solución observada en la Ilustración 20, donde, dividida en dos bibliotecas de clases, se realiza el mapeo de las tablas existentes en la base de datos con la biblioteca “Negocio” y se codifica la correspondiente lógica con los datos de estas tablas por medio de las funciones básicas CRUD con la biblioteca “Administradores”.

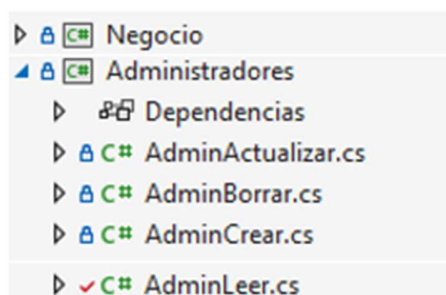


Ilustración 20. Solución para implementación de capa de Negocio. Fuente: autores.

Para implementar la capa de servicios que se ve en la Ilustración 16, se emplean dos puntos de entrada para solicitudes HTTP POST, en las cuales se realiza la correspondiente acción a las solicitudes realizadas por el dispositivo, denominadas “CrearAsistencia” y “CrearTemporal”, como se observa en la Ilustración 21, en la primera se emplea la lógica para determinar el estado de asistencia de cada estudiante que pasa su carnet por

el lector del dispositivo, soportando los procesos de “ingreso de clase” y “salida de clase”, identificados en las Figura 8 y Figura 9. Además, devuelve como respuesta un mensaje de cuerpo JSON con información sobre su estado de asistencia. Por otro lado, para la segunda solicitud, que soporta el proceso “registro de usuario”, identificado en la Figura 7, donde se recibe la solicitud del dispositivo y se genera la carga del código NFC leído hacia la tabla temporal, con el fin de posteriormente asignar este código a un estudiante en su proceso de registro de usuario

```
app.MapPost("/CrearAsistencia",
    (Servidor.CrearAsistencia asistencia, HttpContext httpContext) =>
    {
        return Servidor.Funciones.
            CrearAsistenciaServicio(asistencia, httpContext);
    });
app.MapPost("/CrearTemporal",
    (Negocio.Temporal codigo, HttpContext httpContext) =>
    {
        return Servidor.Funciones.
            CrearTemporalServicio(codigo, httpContext);
    });
```

Ilustración 21. Funciones realizadas en el servicio web. Fuente: autores.

Finalmente, se soporta el caso de uso “registro de asistencia” observado en la Figura 12, codificando la funcionalidad que genera un reporte de asistencia por cada espacio académico, en un archivo separado por comas en donde se listan las cédulas de los asistentes y se le asigna un estado de asistencia (presente, retardo o falla injustificada), según la condición asignada por el responsable. Posterior a la generación del reporte de asistencia se realiza el envío de dicho archivo al correo electrónico almacenado del responsable encargado del espacio académico, en el momento que finaliza la sesión, según la condición de tiempo máximo de final de sesión registrada, soportando el proceso “envío de asistencia” observado en la Figura 13. Para ello, como se observa en la Ilustración 22, se hace uso de un objeto SMTPClient que soporta el protocolo SMTP (*Simple Mail Transfer Protocol*), por el cual se pueden enviar correos de Gmail, empleando las credenciales para uso en aplicaciones del correo electrónico diseñado para el desarrollo de este proyecto.

```

MailMessage correo = new MailMessage();
SmtpClient smtp = new SmtpClient();

correo.From = new MailAddress(correoRemitente, nombreRemitente, System.Text.Encoding.UTF8);
correo.To.Add(correoDestinatario);

correo.Subject = asunto;
correo.SubjectEncoding = Encoding.UTF8;

correo.Body = mensaje;
correo.BodyEncoding = Encoding.UTF8;

correo.IsBodyHtml = false;

correo.Priority = MailPriority.High;

correo.Attachments.Add(new Attachment(archivoAdjunto));

smtp.Credentials = new System.Net.NetworkCredential(correoRemitente, contraseña);
smtp.Port = puerto;
smtp.Host = servidorSMTP;
smtp.EnableSsl = ssl;

smtp.Send(correo);

```

Ilustración 22. Código para envío de correo empleando el objeto SMTPClient. Fuente: autores.

En la Ilustración 23 se hace el envío del archivo separado por comas generado, incluyendo como asunto la información del espacio académico y grupo para el que se está generando el reporte para ser enviado hacia el correo del docente encargado.

```

if (asistente == asistentes.Last())
{
    Correo correo = new Correo();

    correo.CorreoRemitente = "registroasistenciaetm@gmail.com";
    correo.CorreoDestinatario = responsable.CorreoResponsable;
    correo.Contraseña = "zyoqgwaaqsiaugy";
    correo.NombreRemitente = "COMARON";
    correo.Asunto = "Asistencia-" + materia.NombreEspacioAcademico
        + "-" + espacio.Grupo + "-" + periodo.Año + periodo.Semestre + "-Sesion-"
        + sesion.FechaSesion.Year + "/" + sesion.FechaSesion.Month + "/"
        + sesion.FechaSesion.Day + "-" + Hora + ":" + Minuto;
    correo.Mensaje = "Archivo de asistencia";
    correo.ArchivoAdjunto = @"C:\Users\Comar\OneDrive\Escritorio\PUBLICACION\AsistenciaSesion"
        + materia.NombreEspacioAcademico
        + "-" + espacio.Grupo + "-" + periodo.Año + periodo.Semestre + "-"
        + sesion.FechaSesion.Year + "-" + sesion.FechaSesion.Month + "-"
        + sesion.FechaSesion.Day + "-" + Hora + "-" + Minuto + ".csv";
    correo.ServidorSMTP = "smtp.gmail.com";
    correo.Puerto = 587;
    correo.SSL = true;

    correo.Enviar();
}

```

Ilustración 23. Código para envío del archivo de reporte de asistencia. Fuente: autores.

5.3.2. Desarrollo de frontend

Tomando como referencia los *wireframes* diseñados en 5.2.4.2, se realiza la implementación de diferentes pestañas con el fin de brindar al usuario una interfaz en la cual pueda visualizar y modificar la información que sea necesaria. Para ello, se hace uso del patrón MVC. La implementación de este se observa en la Ilustración 24, para el cual se realizan 13 vistas, cada una correspondiente a una función en específico dentro de la aplicación.

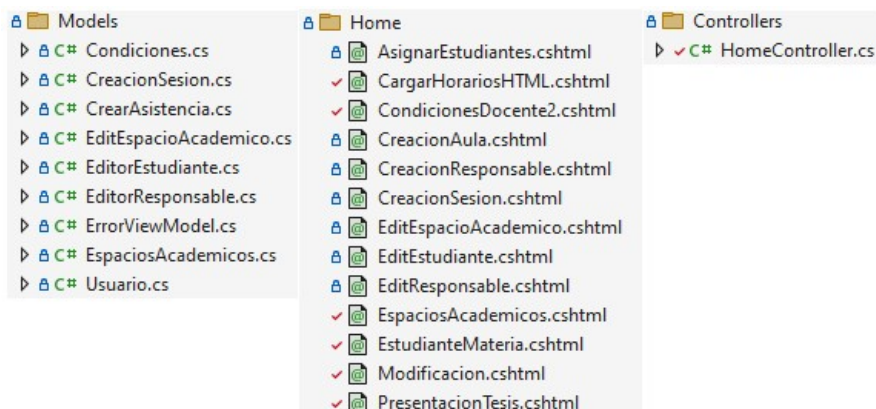


Ilustración 24. Implementación de patrón MVC en la aplicación web. Fuente: autores.

Así mismo, dependiendo de los datos que se van a manejar dentro de cada vista, se hace uso de uno de los 11 modelos realizados. Hay vistas que constan de información muy similar, por ende, se hace uso de un mismo modelo en más de una vista. Finalmente se implementa un único controlador, en el cual se emplea toda la lógica necesaria para la interacción entre el usuario y los datos que se muestran y obtienen por cada vista generada. En la Ilustración 25 se muestra un diagrama de bloques del uso de los componentes MVC de la aplicación implementada.

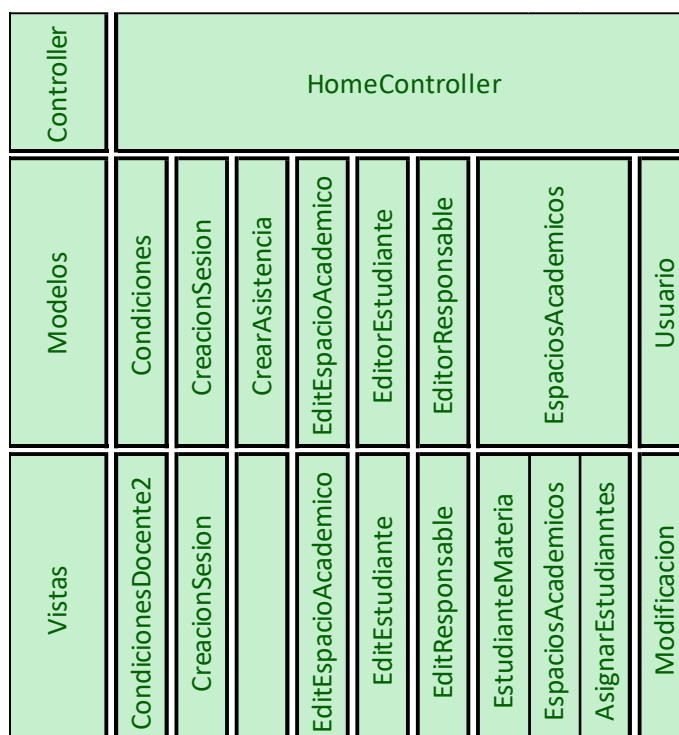


Ilustración 25. Diagrama de bloques MVC. Fuente: autores.

5.3.3. Configuración NodeMCU

Para el manejo del NodeMCU se realizan dos programas, uno con el fin de que el dispositivo se encuentre en modo registro de carnet y otro para el modo de registro de asistencia.

La programación inicial del NodeMCU se realiza mediante el uso de las librerías Arduino.h que es usada para el manejo de puertos de propósito general, como se hace uso de dos protocolos de comunicación serial: SPI e I2C, para esto se usan las librerías SPI.h, usada para el manejo de la pantalla OLED y Wire.h para el protocolo I2C. La librería MFRC522.h es usada para el manejo del lector RC522.

Para el uso del módulo WiFi, se usan las librerías de ESP8266WiFi.h y ESP8266HTTPClient.h, las cuales tienen funciones para realizar la conexión a la red WiFi y el uso del protocolo HTTP para el intercambio de información entre el servidor y el cliente, de igual forma se hace uso de la librería ArduinoJson.h para poder enviar un mensaje con cuerpo de formato JSON que es el solicitado por el servidor.

Finalmente, para el manejo de la pantalla OLED, se hace uso de dos librerías Adafruit_GFX.h y Adafruit_SH110X.h donde esta última, es complementaria a la primera y permite el manejo y configuración de los distintos mensajes que se muestran en la pantalla OLED.

5.3.3.1. Modo Registro de carnet

Como primer programa, se tiene uno en donde se hace la lectura de carnet, esto quiere decir que cada que se realiza la lectura del código NFC, se realiza una petición HTTP, en la que se aplica un método POST, adicionando el código NFC a la tabla temporal, donde se borra el registro anterior, el flujo implementado se puede observar en la Ilustración 26.

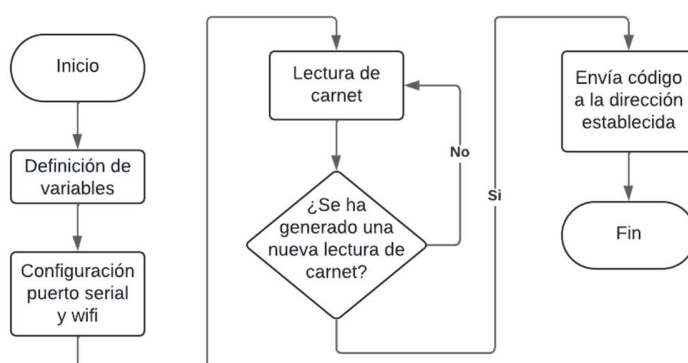


Ilustración 26. Diagrama de flujo programas NodeMCU. Fuente: autores

5.3.3.2. Modo Registro de asistencia

Dentro de este segundo programa, el dispositivo entra en modo de configuración y queda a la espera de recibir un dato por puerto serial,

de lo contrario si este dato no llega, el programa procede a consultar lo almacenado en los espacios de memoria asignados para la memoria EEPROM y continua el flujo descrito para el programa anterior, como se observa en la Ilustración 27, a diferencia que la dirección establecida para la petición en este modo modifica la tabla de asistencia.

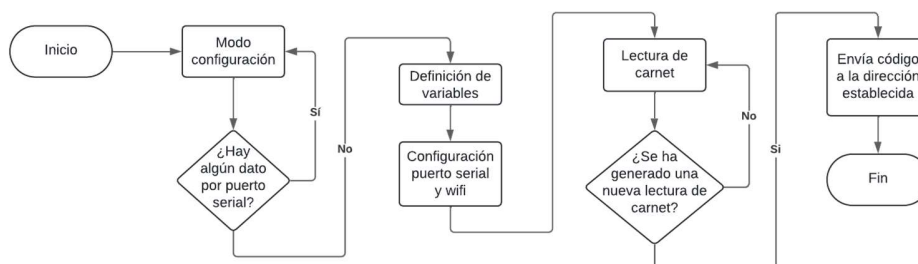


Ilustración 27. Diagrama de flujo modo asistencia. Fuente: autores

Con el manejo de la memoria EEPROM se contemplan distintos casos como lo puede ser la pérdida de energía para que no tenga que ser necesario volver a pedir los datos referentes al aula y el caso en el que el dispositivo sea cambiado de aula para que se pueda volver a configurar desde la app implementada para este y no desde Arduino IDE.

Para el manejo de la EEPROM se hace uso de una librería más, la de EEPROM.h, esta cumple con la función de usar espacios de memoria de programa como si fueran de datos permanentes, esto con el fin de mantener datos como el aula donde se encuentre el dispositivo, el nombre de la red WiFi y la contraseña de esta, todos estos datos se modifican en el proceso definido como “Modo configuración” y se leen en el proceso “Definición de variables”.

5.4. Desarrollo fase IV – Validación del sistema

5.4.1. Preparación del sistema para pruebas

Para el desarrollo de esta fase, se crea un servidor enlazado del “Computador Laboratorios” de la base de datos de LabManager en el “Computador de pruebas”³, para traer la información de los estudiantes registrados, por lo anterior, se hace necesario garantizar el correcto funcionamiento del equipo donde se encuentra almacenada la base de datos y solo se pueda acceder a una tabla en específico, que en este caso es la de Usuario (Seguridad), por ello, se crea una vista para esta información y un perfil dentro de la instancia de SSMS del “Computador Laboratorios”, que solo tenga acceso de solo lectura a dicha vista y a la tabla de Usuarios.

³ Ver la descripción de estos componentes en 5.1.3, Ilustración 2.

Para la sincronización se ejecuta un script con la consulta hecha desde la instancia del “Computador de pruebas” en donde se obtienen los datos del servidor de LabManager y añadirlos en la base de datos alojada en el equipo local con el script ejecutado desde SSMS, este script se puede ver en el Anexo 9.1.

Posterior a la sincronización se realiza el despliegue de la aplicación web desarrollada haciendo uso del servidor web IIS (*Internet Information Services*) de Microsoft. Allí se establece una dirección IP sobre la cual estará alojada la aplicación y será visible por el computador de pruebas y todo dispositivo conectado a una misma red. Teniendo en cuenta esto, se hace uso de la conexión cableada local LAN de los laboratorios ETM de la Universidad Santo Tomás. A esta misma red se conecta un enrutador que fue previamente configurado para proporcionar una red WiFi. Con esta conexión al enrutador, cualquier dispositivo que se conecte a su red WiFi podrá establecer una conexión con el computador de pruebas. Siguiendo esta idea, el dispositivo desarrollado se conecta a dicha red WiFi para poder establecer el intercambio de información entre este y el sistema.

Finalmente, se realiza una “prueba de humo” con los docentes de la Facultad de Ingeniería Electrónica durante el periodo intersemestral del año 2024 donde se identifican y realizan los ajustes necesarios para el cumplimiento de cada uno de los procesos caracterizados en 5.1. Estas pruebas se realizaron durante 22 días, donde los docentes registraban su ingreso y salida a la sala de profesores ubicada en el Laboratorio ETM 4 del edificio Gregorio XIII.

5.4.2. Descripción de las pruebas

Inicialmente, se busca información correspondiente a los espacios académicos que sean cursados en el semestre 2024-2 en los laboratorios ETM, se busca el laboratorio en el que existe una mayor concentración de espacios académicos de la Facultad de Ingeniería Electrónica y se escoge este laboratorio. El laboratorio seleccionado es el ETM8, con un total de 8 espacios académicos cursados a lo largo de la semana, cada uno con una sesión semanal, con una variedad de 5 docentes responsables y un total de 132 asistentes distribuidos en dichos espacios académicos así:

Tabla 5. Número de asistentes por espacio académico. Fuente: autores.

Espacio académico	Asistentes	Se toma asistencia	Tiempo de toma de asistencia	Método
Sistemas digitales II 4A	18	Si	138.07s	Moodle
Programación II 4A	23	Si	145.07s	Moodle
Programación II 4B	23	Si	152.73s	Moodle
Programación Web	7	Si	103.83s	Moodle
Analítica de datos	15	No	N/A	N/A
Fuentes de energía renovables	4	No	N/A	N/A
Seguridad de la información	17	No	N/A	N/A
Electrónica II	25	Si	123.68s	Hoja de cálculo
TOTAL	132	-	-	-

Luego de seleccionar los espacios académicos, se procede a socializar el proceso de pruebas con los docentes en su espacio académico, en donde, se socializa lo que se desea realizar y cuál es el objetivo por cumplir dentro la actividad.

5.4.3. Resultados de las pruebas

La etapa de pruebas se realiza por lo menos en una de las sesiones de cada espacio académico seleccionado, como se puede observar en la Tabla 6, donde se valida el proceso de registro de asistencia por cada estudiante.

Tabla 6. Cantidad de estudiantes y sesiones por espacio académico. Fuente: autores.

Estudiantes y sesiones por espacio		
Espacio académico	Estudiantes	# de sesiones
Sistemas digitales II 4A	18	4
Programación II 4A	23	2
Programación II 4B	23	2
Programación Web	7	2
Electrónica II 4B	25	3

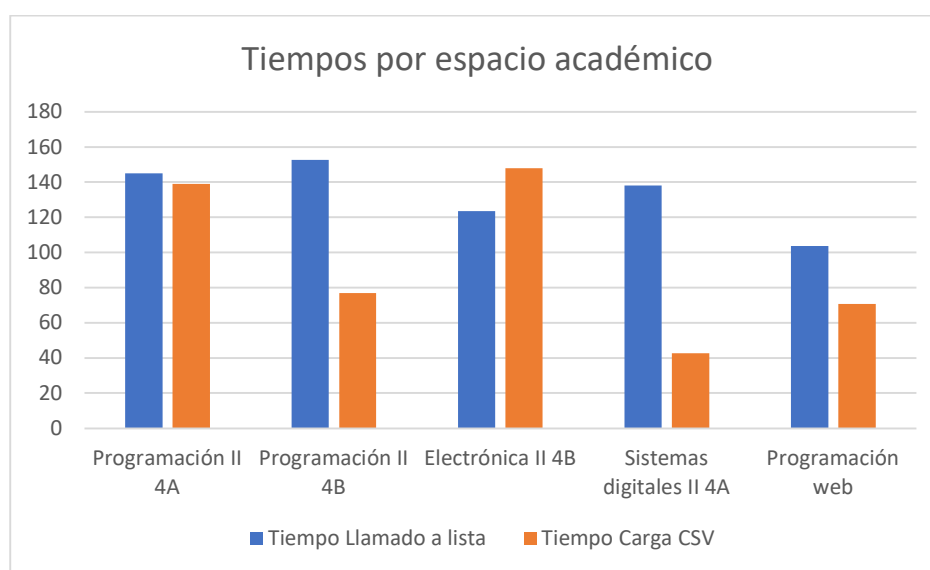
En cuanto al tiempo, durante las sesiones donde se lleva a cabo la toma de asistencia de manera tradicional, se emplea un promedio de 8.08 segundos por estudiante, donde se tiene en cuenta el tiempo de creación de la sesión correspondiente en Moodle, añadido al tiempo empleado para llamar a cada estudiante, en el Anexo 10, se presentan los datos resultantes de este proceso.

Con la implementación del sistema, el docente tiene que finalizar el proceso de envío de asistencia, descrito en la Figura 13, siguiendo los pasos para la carga del archivo CSV, donde el tiempo empleado para la carga de este archivo es un promedio de 5.56 segundos por estudiante, estos datos son obtenidos de la tabla presentada en el Anexo 11, obteniendo de esta manera una reducción de un 31.2% del tiempo empleado para el proceso de registro de asistencia, como se evidencia en la Tabla 7.

Tabla 7. Cálculo de la diferencia entre ambos métodos de toma de asistencia. Fuente: autores.

Tiempos Promedio Toma asistencia	8,080
Tiempos Promedio Carga asistencia	5,56
Diferencia	31,19%

Se realiza una comparación entre el tiempo que es tomado por el docente para las actividades de llamado a lista y para cargar el archivo CSV en la plataforma, donde se tiene en cuenta desde que se crea la sesión para ambas actividades, como se puede observar en la Gráfica 3, hay que tener en cuenta que el tiempo que emplea el docente creando la sesión es una actividad que es posible realizar una única vez durante el semestre.⁴

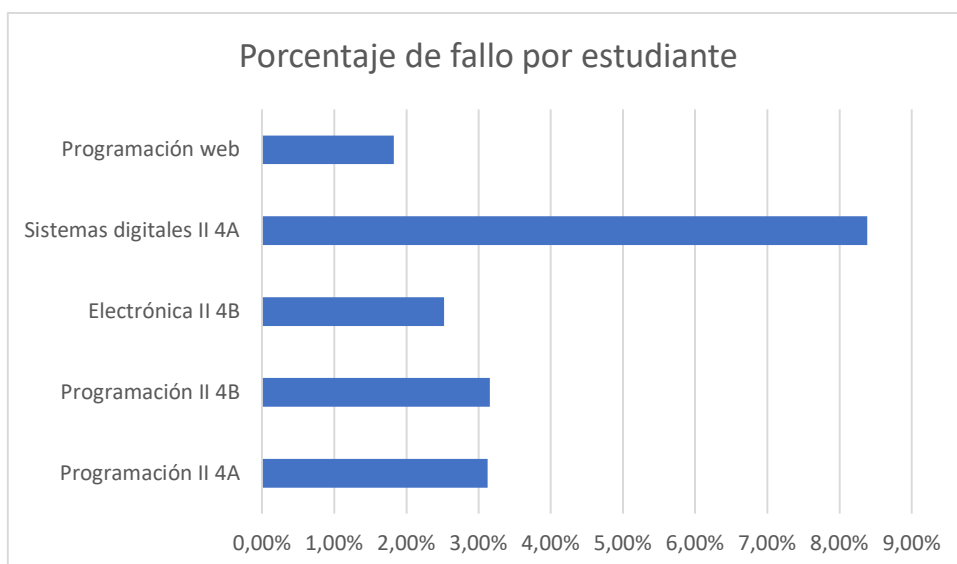


Gráfica 3. Comparación de tiempos cronometrados. Fuente: autores.

En la Gráfica 4, se observa una comparación entre el resultado obtenido desde el sistema y el resultado obtenido por el docente al realizar el llamado a lista de manera empírica, donde, si el sistema marca como estado R (retardo) y el docente marca como P o FI, se considera como error. Estos valores son obtenidos de la tabla observada en el Anexo 12, donde se realiza

⁴ https://docs.moodle.org/all/es/Gu%C3%ADa_r%C3%A1pida_de_Asistencia

una ponderación de cada espacio académico según la cantidad de estudiantes matriculados, para tener un porcentaje de error por estudiante.



Gráfica 4. Porcentaje de error del sistema. Fuente: autores.

Como es posible observar en el Anexo 12, el espacio con mayor porcentaje de error durante las sesiones de prueba fue el de Sistemas Digitales II 4A, esto debido a distintos factores observados a lo largo de la fase IV, donde el principal factor de error es la hora en la que se realiza el llamado a lista, debido a que el proceso de llamado a lista del docente se realiza después de la primera hora de la sesión, lo que produce que los estudiantes tengan un amplio tiempo para ingresar al aula y aun así obtener un estado de presente en el reporte del docente, esto genera múltiples errores durante dicho espacio.

De igual forma, se observa que hubo una gran concentración de errores en distintos espacios académicos durante la sesión 1, debido a la distancia que existía entre el enrutador y el dispositivo; lo que producía errores de comunicación entre el dispositivo y el servidor, lo cual llevaba a pérdida de información ya que, en su mayoría, el dispositivo respondía como: "Error de servidor".

En busca de solución de este problema, se cambia la ubicación del enrutador para mejorar el tiempo de respuesta entre el dispositivo y el servidor, obteniendo de esta manera una menor pérdida de información y mayor velocidad de respuesta, esto se encuentra evidenciado en la Tabla 8, dicha diferencia es obtenida calculado el promedio de errores que se presentan en las sesiones previas y posteriores al cambio de posición.

Tabla 8. Porcentaje de error respecto al cambio de posición del enrutador. Fuente: autores.

Antes del cambio	32,51%
Después del cambio	3,34%
Diferencia	29,18%

Añadido a esto, uno de los grandes problemas observados, es que el 33.2% de estudiantes registran su entrada, pero al momento que finaliza la sesión, no realizan el proceso de registro de salida, lo que genera un estado de FI en el reporte generado y a su vez, es considerado como error.

6. RESULTADOS Y DISCUSIÓN

Se realiza la validación de los distintos casos de uso identificados durante la fase de caracterización se requiere de un dispositivo lector de código NFC configurado en modo registro de carnet, como se explica en 5.3.3.1, este registro se realiza con los estudiantes que vayan a hacer uso del sistema, este proceso se lleva a cabo desde la aplicación web desarrollada dentro de la pestaña “Modificación de Usuario” como se observa en la Ilustración 28.

Ilustración 28. Pestaña modificación datos de usuario. Fuente: autores

Dentro de la columna “Modificar”, se tiene que hacer clic en el lápiz, correspondiente a la información de cada usuario, accediendo a la pestaña resultante observada en la Ilustración 29 donde se realiza la carga del código NFC del carnet dentro de la columna “Código NFC” haciendo uso del botón con el ícono de carga ubicado debajo del texto de esta columna. Para un registro exitoso, es necesario contar con el correcto registro de la cédula del usuario, ya que, con esta, se realizan las distintas relaciones e identificaciones de cada estudiante con cada espacio académico.

Editar estudiante

Ilustración 29. Pestaña modificar. Fuente: autores.

6.1. Ingreso a clase

Este proceso es realizado exclusivamente por los estudiantes pertenecientes a los espacios académicos registrados, donde la acción que deben realizar es la de poner su carnet estudiantil en el dispositivo, con esto, se da cumplimiento al caso de uso proceso de ingreso a clase, observado en la Figura 8, el mensaje obtenido por los estudiantes al realizar su primer registro dentro de la sesión es el observado en la Ilustración 30, donde el mensaje presentado es “Hola [Nombre registrado]”.



Ilustración 30. Mensaje de bienvenida. Fuente: autores

6.2. Salida de clase

El proceso de salida de clase presentado en la Figura 9, es una acción que deben cumplir los estudiantes al finalizar la sesión de clase, con el fin de validar la permanencia del estudiante dentro de la sesión, por lo anterior, al realizar un segundo registro del carnet, el dispositivo responde de la manera “Adiós [Nombre registrado]” como se observa en la Ilustración 31.



Ilustración 31. Mensaje de salida. Fuente: autores

6.3. Registro de clases en laboratorios ETM

Para dar cumplimiento al proceso de registro de clases en los laboratorios ETM presentado en la Figura 10, se solicita a cada docente que pueda brindar una lista de los estudiantes que asisten a cada espacio académico del cual dicho docente es responsable. Esta lista de estudiantes debe estar diligenciada en un archivo de Excel, en donde se tenga el nombre de cada estudiante junto con su documento de identidad. Para poder realizar la carga de dicha lista en el sistema, se busca el espacio académico que se desea en la pestaña “Asignación de esp. académicos” como se observa en la Ilustración 32, donde al oprimir el botón en la columna “Cargar estudiantes” permitirá al usuario cargar la lista de estudiantes previamente descrita. Una vez cargada la lista de estudiantes en el sistema, se indicará al usuario mediante un mensaje que el proceso se ha completado con éxito.

Carga de Horarios Modificación de Usuario Espacios Académicos Crear nueva sesión Condiciones para retardo por docente Asignación de esp. académicos

Seleccione la asignatura para cargar la lista de estudiantes:

Show 10 entries Search: Fuentes

Id	Espacio académico	Grupo	Periodo	Cargar estudiantes
34	FUENTES DE ENERGÍA RENOVABLE	7A	20242	
36	FUENTES DE ENERGÍA RENOVABLE	7A	20242	

Showing 1 to 2 of 2 entries (filtered from 59 total entries) Previous 1 Next

Ilustración 32. Pestaña asignación de esp. académicos. Fuente: autores.

Tras finalizarse el proceso de carga de la lista de estudiantes, se verificó la carga por medio de una consulta directamente a la tabla Asistente en la base de datos, empleando SSMS, como se observa en la Ilustración 33, donde se muestran los asistentes al espacio académico consultado en la aplicación en la Ilustración 32. Esta asignación de asistentes se realiza gracias al listado cargado para dicho espacio académico, pues de esta manera se logra relacionar a los estudiantes registrados en el sistema, con cada espacio académico que cursen.

	ID_Espacio_Academico	ID_Programa	Nombre_Espacio_Academico
1	34	1	FUENTES DE ENERGÍA RENOVABLE

	ID_Espacio_Academico_X_Periodo	ID_Espacio_Academico	ID_Periodo_Academico	ID_Responsable	Grupo
1	34	34	1	21	7A

	ID_Asistente	ID_Estudiante	ID_Espacio_Academico_X_Periodo
1	119	2960	34
2	121	3115	34
3	120	3132	34
4	122	3472	34

Ilustración 33. Resultado de consulta SQL de la tabla asistente con resultados de la carga de lista de estudiantes. Fuente: autores.

6.4. Condiciones del responsable

El cumplimiento del proceso condiciones del responsable observado en la Figura 11 toma importancia para validar los estados de asistencia de todos los estudiantes pertenecientes a cada espacio académico, donde, haciendo uso de la aplicación web, el responsable ingresa con su usuario y contraseña para registrar sus condiciones, para ello debe: seleccionar su nombre, registrar las condiciones de retardo para estudiante y de tiempo máximo de final de sesión. Con estos valores registrados, tal como se observa en la Ilustración 34, el sistema realiza la validación de la asistencia de cada estudiante dentro de los parámetros establecidos por el responsable.

Ilustración 34. Asignación de condiciones por docente. Fuente: autores.

6.5. Registro de asistencia

Para cumplimiento del proceso de registro de asistencia observado en la Figura 12, se obtiene del sistema un archivo separado por comas de formato CSV⁵, en el cual se tiene el listado de los estudiantes pertenecientes al espacio académico. En este archivo se tiene el estado de asistencia de cada uno de los estudiantes, así como se observa en la Ilustración 35, en donde se muestra el resultado en una hoja de cálculo donde se tiene el listado de documentos de los estudiantes registrados y al lado de cada uno de ellos su estado de asistencia según su hora de ingreso y salida de la sesión, o su inasistencia a la misma.

Documento	Estado
	R
	R
	FI
	FI
	FI
	FI
	FI
	P
	FI
	P
	P
	FI
	P
	R
	P
	P
	R

Ilustración 35. Reporte generado. Fuente: autores.

⁵ Valores separados por comas.

6.6. Envío de asistencia

En cumplimiento al proceso envío de asistencia observado en la Figura 13, luego de obtener el reporte de asistencia en el archivo separado por comas se realiza el envío de este por medio de la cuenta de correo electrónico “COMARON”, como se observa en la Ilustración 36, cuyo asunto es el nombre del espacio académico seguido de la fecha de ejecución de la sesión y hora de desarrollo de esta. Además, se incluye un instructivo de YouTube⁶ realizado por los autores para los docentes, y puedan visualizar como crear una sesión en Moodle, y cargar allí el archivo CSV que se les envía. Dicho correo es enviado al correo institucional ingresado por el docente registrado como responsable del espacio académico.

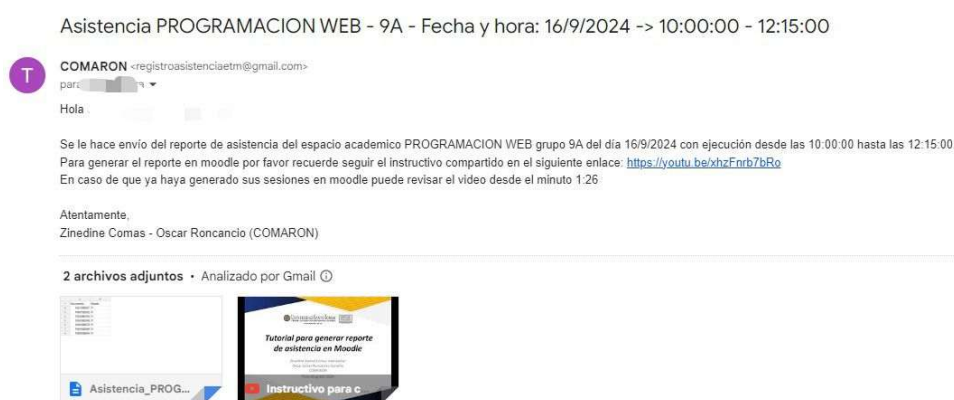


Ilustración 36. Correo enviado con reporte de asistencia. Fuente: autores.

6.7. Carga de archivo HTML

Para la carga del archivo de extensión HTML obtenido del sistema académico SAC soportando el proceso carga de archivo HTML observado en la Figura 14, se hace uso de la pestaña “Carga de Horarios” de la aplicación web desarrollada, como se observa en la Ilustración 37. Allí se selecciona el archivo que debe ser previamente descargado en el computador y se visualiza en la pestaña actual. Luego, se debe oprimir el botón “Cargar archivo” y el sistema empezará a cargar toda la información alojada en dicho archivo. Al finalizar el proceso, el usuario recibirá en pantalla un mensaje indicando que el proceso de carga del archivo HTML se ha completado con éxito.

⁶ <https://youtu.be/xhzFnrb7bRo>

Carga de Horarios
Modificación de Usuario
Espacios Académicos
Crear nueva sesión
Condiciones para retardo por docente
Asignación de esp. académicos

Recibir HTML

Seleccione el archivo .HTML

Seleccionar archivo
Reporte.html

Visualizar archivo a cargar
Cargar archivo

Reporte a cargar

UNIVERSIDAD SANTO TOMAS
Horario de grupos por programa

30 de Julio de 2024
20:11:42
MOTRPR_GWT

Programa: 11002 INGENIERÍA ELECTRÓNICA - JORNADA DIURNA
Periodo: 20242

Pensum
Nivel Asignatura Grupo Cupo Docente

Fechas
Aula Horario

20142 PENSUM DESDE 20142 - VIGENTE

ID	1	12249	FUNDAMENTOS DE CIRCUITOS
1A	16		
31/07/2024 - 20/11/2024			
LAB. ETM11 - EDIF. GREGORIO XIII	Mie	2:00 pm	4:00 pm
02/08/2024 - 22/11/2024			
SALA DE DIBUJO 302 - EDIF. LUIS J. TORRES	Vie	2:00 pm	4:00 pm

Ilustración 37. Pestaña carga de horarios. Fuente: autores.

Luego de realizar la carga del archivo desde la aplicación web, se realiza la correspondiente consulta en SQL obteniendo como resultado lo observado en la Ilustración 38, donde se verifica la información correspondiente al espacio académico presente en el archivo HTML del proceso de carga de la Ilustración 37.

ID_Periodo_Academico	Año	Semestre
1	1	2024 2

ID_Programa	Nombre_Programa
1	1 INGENIERÍA ELECTRÓNICA - JORNADA DIURNA

ID_Aula	Nombre_Aula
1	5 LAB. ETM11 - EDIF. GREGORIO XIII

ID_Espacio_Academico	ID_Programa	Nombre_Espacio_Academico
1	1	FUNDAMENTOS DE CIRCUITOS

ID_Responsable	Nombre_Completo_Responsable	Documento_Responsable	CodigoNFC_Responsable
1	1	NULL	NULL

ID_Espacio_Academico_X_Periodo	ID_Espacio_Academico	ID_Periodo_Academico	ID_Responsable	Grupo
1	1	1	1	1A

ID_Sesion	ID_Espacio_Academico_X_Periodo	Fecha_Sesion	Hora_Inicio	Hora_Fin	ID_Aula
1	4	1 2024-07-31	14:00:00.0	16:00:00.0	5
2	170	1 2024-08-07	14:00:00.0	16:00:00.0	5
3	172	1 2024-08-14	14:00:00.0	16:00:00.0	5
4	185	1 2024-11-20	14:00:00.0	16:00:00.0	5

Ilustración 38. Resultado de consulta en SQL con resultados de la carga del HTML. Fuente: autores.

7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1. Conclusiones

La caracterización del sistema realizada en 5.1, fue parte fundamental dentro del desarrollo del proyecto, determinando los pasos a seguir al momento de implementar el sistema, ya que con este el docente puede realizar su sesión sin necesidad de llamar a lista, logrando que esta sea una tarea específica de los estudiantes.

El dispositivo diseñado en 5.2.1 cumple con la funcionalidad de leer el código NFC dentro del carnet institucional de cada usuario del sistema, ofreciendo una interacción con el usuario mediante mensajes que son visualizados en la pantalla, además, permite la interacción entre el usuario y el sistema por medio del protocolo web HTTP a través de una red WiFi privada.

Con el desarrollo de la aplicación web implementada en 5.3, se logra dar soporte a cada uno de los procesos identificados en 5.1.1, logrando generar una aplicación a la cual se puede acceder con permisos de administrador, con información intuitiva para quien dispone de ella, y con funcionalidades que permiten identificar a cada uno de los actores dentro del proceso de registro de asistencia. Además, se logra emplear un sistema que es capaz de generar reportes de asistencia por cada sesión de clase desarrollada indicando el estado de asistencia de cada estudiante, permitiendo ser posteriormente compartido por correo electrónico al docente encargado de cada espacio académico.

Los componentes descritos en la arquitectura presentada en la sección 5.1.3 son suficientes para soportar el sistema implementado, el cual da como resultado una mejora significativa del 31.2% en el tiempo empleado para la toma de asistencia por estudiante, esta reducción está evidenciada en la diferencia de promedios mostrada en la Tabla 7, donde se demuestra que la implementación del sistema mejora los tiempos en el proceso de registro de asistencia.

En los reportes entregados se presenta un 3.8% de error calculado en el Anexo 12, respecto a los reportes realizados de manera manual, representado principalmente en el hecho de que los estudiantes no registran su salida, lo cual se considera un error en el proceso más que un error técnico del sistema.

Con el cumplimiento de cada uno de los objetivos específicos planteados, se logra la implementación de un sistema capaz de brindar un registro de asistencia de manera automática haciendo uso del dispositivo con lector NFC, se obtiene el estado de asistencia de cada estudiante para que posteriormente el sistema pueda generar un reporte de asistencia por cada sesión y ser entregado a los docentes que imparten clases dentro de uno de los laboratorios ETM de la Universidad Santo Tomás, dejando esta actividad como responsabilidad única

del estudiante, y permitiendo que el tiempo empleado para esta actividad no interfiera con el correcto desarrollo de las sesiones de clase.

7.2. Trabajos futuros

Es posible realizarle distintos cambios al sistema implementado, como lo pueden ser:

- Implementar reconocimiento facial o dactilar al estudiante, esto con el fin de realizar una autenticación del estudiante cuando ingrese al aula.
- Emplear un control de acceso al aula donde el estudiante tenga que poner su carnet al ingresar y al salir del aula, con el fin de mitigar el hecho de que no se ponga el carnet a la salida.
- Integrar el sistema con Moodle y SAC, con el fin de evitar los procesos que requieren actividades manuales por los involucrados.
- Alojamiento del servidor en la nube, con el fin de reducir la cantidad de elementos físicos utilizados en el sitio a implementar.
- Implementar el sistema para el resto de la comunidad académica.
- Finalización de la sesión por parte del responsable y de esta manera generar dar paso al proceso de registro de asistencia planteado en la Figura 12.

8. REFERENCIAS BIBLIOGRÁFICAS

- [1] “Acuerdo-No-50-Reglamento-Gral-Est-Preg-Multicampus legible”.
- [2] USTA, “SIIM.” Accessed: Aug. 16, 2023. [Online]. Available: <https://websiim.usta.edu.co/>
- [3] L. C. Bracamontes, S. R. Z. Barrera, S. V. Bahena, and C. A. J. Ramírez, “TARJETAS NFC COMO SISTEMA DE ACCESO A LAS INSTALACIONES DEL ITDEIGUALA,” *Edición especial*, p. 107.
- [4] N. Unidas, “La Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe,” 2030. [Online]. Available: www.issuu.com/publicacionescepal/stacks
- [5] E. Francy Irudaya Rani, R. Vedhapriyavadhana, J. S. J. M., and K. C., “Attendance Monitoring Using Face Recognition with Message Alert,” *SSRN Electronic Journal*, Mar. 2020, doi: 10.2139/ssrn.3529319.
- [6] A. Rao, “AttenFace: A Real Time Attendance System using Face Recognition,” Nov. 2022, [Online]. Available: <http://arxiv.org/abs/2211.07582>
- [7] R. L. Jorda *et al.*, “Comparative evaluation of NFC tags for the NFC-controlled door lock with automated circuit breaker,” in *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2018*, Institute of Electrical and Electronics Engineers Inc., Jul. 2018. doi: 10.1109/HNICEM.2018.8666375.
- [8] A. Shrivastava, S. J. Suji Prasad, A. R. Yeruva, P. Mani, P. Nagpal, and A. Chaturvedi, “IoT Based RFID Attendance Monitoring System of Students using Arduino ESP8266 & Adafruit.io on Defined Area,” *Cybern Syst*, 2023, doi: 10.1080/01969722.2023.2166243.
- [9] T. Sangkharat and J. La-Or, “Application of Smart Phone for Industrial Barcode Scanner,” in *2021 7th International Conference on Engineering, Applied Sciences and Technology, ICEAST 2021 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021, pp. 9–12. doi: 10.1109/ICEAST52143.2021.9426288.
- [10] A. and M. A. and E. M. M. Ababaou Nabil and Maafiri, “Android-Based Attendance Management System,” in *Digital Technologies and Applications*, B. Motahhir Saad and Bossoufi, Ed., Cham: Springer Nature Switzerland, 2023, pp. 298–307.
- [11] E. P. Superior, C. Martí González, J. A. Clavijo, B. Ponente, and S. Santini, “Universidad Autónoma de Madrid TRABAJO DE FIN DE GRADO APLICACIÓN MÓVIL Y WEB PARA EL CONTROL DE ASISTENCIA DE LOS EMPLEADOS DE UNA COMPAÑÍA,” 2019.
- [12] F. Valerio, “DESARROLLO DE PROTOTIPO MÓVIL PARA EL CONTROL Y REGISTRO DE ASISTENCIA DE ALUMNOS DE LA PUCV DIEGO IGNACIO MAYORGA MARTÍNEZ FRANCO GIOVANNI VALERIO PASQUALETTI INFORME FINAL DE PROYECTO PARA OPTAR AL TÍTULO PROFESIONAL DE INGENIERO DE EJECUCIÓN EN INFORMÁTICA,” 2017.

- [13] A. Rodríguez-González *et al.*, “Uso de tarjeta universitaria y tecnología móvil para el control de acceso a instalaciones universitarias y su posterior análisis en términos de rendimiento académico y control de asistencia en clase - [Use of university card and mobile technology to study and improve academic performance and class attendance control],” Universidad de Zaragoza, Sep. 2017, pp. 1–6. doi: 10.26754/cinaic.2017.000001_023.
- [14] Sergio Conde Gómez, “Control de asistencia mediante técnicas NFC}”.
- [15] L. Paz -Bolivia, “UNIVERSIDAD MAYOR DE SAN ANDRES FACULTAD DE TECNOLOGIA CARRERA DE ELECTRONICA Y TELECOMUNICACIONES NIVEL LICENCIATURA EXAMEN DE GRADO ‘SISTEMA DE CONTROL Y MONITOREO DE ASISTENCIA DE PERSONAL MEDIANTE RADIOFRECUENCIA Y NFC’ Postulante: Miguel Angel Alcon Baltazar,” 2016.
- [16] A. J. Balsero, M. Cristian, and G. V. Garcia, “DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL CONTROL DE ACCESO EN LA SEDE DE INGENIERÍA DE LA UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS MEDIANTE EL USO DE TORNQUETES CONTROLADOS POR CARNET CON TECNOLOGIA NFC Y LECTOR BIOMÉTRICO DE HUELLA DACTILAR.”
- [17] LEÓN PÉREZ JORDAN JARDEL. and PONCE BUITRÓN PEDRO DANIEL., “ANÁLISIS Y DISEÑO DE UN SISTEMA PARA LA MONITORIZACIÓN DE ENTRADA Y RETIRO DE NIÑOS EN PRIMARIAS EN LA CIUDAD DE GUAYAQUIL HACIENDO USO DE PULSERAS INTELIGENTES INTEGRADAS CON MICROCONTROLADORES NFC”.
- [18] Ecma International, “Near Field Communication - White paper.” Accessed: Mar. 10, 2024. [Online]. Available: <https://www.ecma-international.org/wp-content/uploads/tc32-tg19-2005-012-NFC-white-paper.pdf>
- [19] “Coskun, V., Ok, K., & Ozdenizci, B. (2018). Near Field Communication (NFC): From Theory to Practice. Wiley.”.
- [20] R. Weinstein, “RFID: A Technical Overview and Its Application to the Enterprise,” 2005. [Online]. Available: <http://www.rfiduk>.
- [21] “Raza, S., Wallgren, L., & Voigt, T. (2018). Security Considerations for the Internet of Things and RFID. Computers & Security, 77, 117-134.”.
- [22] S. Luján Mora, “Programación de aplicaciones web: Historia, Principios básicos y Clientes web.”
- [23] “Osmani, A. (2018). Learning JavaScript Design Patterns: A JavaScript and Node.js Developer’s Guide. O’Reilly Media.”.
- [24] “Alonso, G., Casati, F., Kuno, H., & Machiraju, V. (2010). Web Services: Concepts, Architectures and Applications. Springer Science & Business Media.”.
- [25] Euroinnova, “Aplicaciones de escritorio.” [Online]. Available: <https://www.euroinnova.edu.es/blog/aplicaciones-de-escritorio#que-son-las-aplicaciones-de-escritorio>
- [26] L. Del Valle Hernández, “NodeMCU tutorial paso a paso desde cero,” 2018.
- [27] E. López Pérez, “Protocolo SPI(Serial Peripheral Interface).”
- [28] M. Ruiz Méndez, “PROTOCOLO I2C COMUNICACIONES CON PLACA ARDUINO.”

- [29] Claudio Peña, *Arduino IDE: Domina la programación y controla la placa*. 2020.
- [30] Emilio Ulloa, "BASES DE DATOS Y SISTEMAS BASES DE DATOS Y SISTEMAS DE GESTION DE BASES DE DE GESTION DE BASES DE DATOS DATOS," 2013.
- [31] Al Delta, "NORMAS INTERNACIONALES BASICAS PARA DISEÑO DE CIRCUITOS IMPRESOS PCB Y PRODUCTOS ELECTRÓNICOS." [Online]. Available: www.aldelta.com.co
- [32] F. Valerio, "DESARROLLO DE PROTOTIPO MÓVIL PARA EL CONTROL Y REGISTRO DE ASISTENCIA DE ALUMNOS DE LA PUCV DIEGO IGNACIO MAYORGA MARTÍNEZ FRANCO GIOVANNI VALERIO PASQUALETTI INFORME FINAL DE PROYECTO PARA OPTAR AL TÍTULO PROFESIONAL DE INGENIERO DE EJECUCIÓN EN INFORMÁTICA," 2017.
- [33] Microsoft, "Fundamentos de la normalización de bases de datos," Fundamentos de la normalización de bases de datos.

9. ANEXOS

9.1. Anexo 1

Nombre	Registro de Persona
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	Este proceso es requerido como actividad inicial de todos los implicados dentro de la actividad de toma de asistencia
Actores	Estudiante, Docente, Administrador, Laboratorista
Precondiciones	Tiene que poseer un carnet estudiantil y un número de identificación
Flujo Normal	Figura 7. Proceso para el registro de usuario. Fuente: autores.
Flujo Alternativo	Esperar que se encuentre disponible
Postcondiciones	Almacena: Usuario_ID, Código NFC, Numero de Identificación

9.2. Anexo 2

Nombre	Ingreso a Clase
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	Proceso requerido para el inicio, se va a realizar el registro de la hora en la que ha sido puesto el carnet de cada persona
Actores	Estudiante
Precondiciones	El estudiante tuvo que realizar el Proceso de Registro
Flujo Normal	Figura 8. Proceso de registro de ingreso a clase. Fuente: autores.
Flujo Alternativo	Realizar el proceso de registro de persona
Postcondiciones	Se almacena Código NFC y Fecha_Hora de ingreso

9.3. Anexo 3

Nombre	Salida de Clase
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	Proceso que culmina el registro de salida del aula, para
Actores	Estudiante
Precondiciones	El estudiante tuvo que realizar el proceso de Ingreso a Clase
Flujo Normal	Figura 9. Proceso para el registro de salida de clase. Fuente: autores.
Flujo Alternativo	N/A
Postcondiciones	Se almacena Código NFC y Fecha Hora de salida

9.4. Anexo 4

Nombre	Registro de Espacios Académicos en ETM
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	Este proceso consta del registro de los distintos espacios académicos que se van a tomar, que día, hora y ETM por cada una de las materias
Actores	Docente, Laboratorista
Precondiciones	Se tuvo que finalizar el periodo de matricula
Flujo Normal	Figura 10. Proceso para el registro de clases en laboratorios ETM. Fuente: autores.
Flujo Alternativo	Esperar que finalice la fecha límite de matricula
Postcondiciones	Se almacena Fecha y Hora por Espacio Académico

9.5. Anexo 5

Nombre	Registro de Condiciones del responsable
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	En este proceso el docente registra como quiere manejar la asistencia, como lo es, desde el inicio de clase, cuanto tiempo quiere que valga la asistencia sin retardo; hasta que hora quiere que se tome la asistencia y cuando quiere que se le envíe el reporte de asistencia al correo
Actores	Docente, Laboratorista
Precondiciones	Se tuvo que hacer el Proceso de Registro de Persona, Registro de Clase ETM
Flujo Normal	Figura 11. Proceso para el registro de condiciones del responsable. Fuente: autores.
Flujo Alternativo	Si no se encuentra registrado hay que registrarse
	Si no ha finalizado la fecha de matrículas, hay que esperar que se cumpla la fecha
Postcondiciones	Se almacena Condiciones. Se almacena Nombre del responsable

9.6. Anexo 6

Nombre	Registro de Asistencia
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	En este proceso el dispositivo reúne los distintos datos necesarios que han sido recopilados a lo largo de los procesos para realizar un archivo separado por comas en donde se tengan los datos para el registro de asistencia
Actores	Dispositivo
Precondiciones	Se tuvo que culminar el proceso de Registro de Persona, Registro de Clases por ETM, Registro de Ingreso a clase, Registro de Salida y el Proceso de Condiciones del responsable
Flujo Normal	Figura 12. Proceso para el registro de asistencia. Fuente: autores.
Flujo Alternativo	Si no funciona el dispositivo, Realizar la toma de asistencia de manera manual
Postcondiciones	Se almacena Archivo CSV

9.7. Anexo 7

Nombre	Envío de Asistencia
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	En este proceso el dispositivo envía un correo con el archivo CSV y el docente debe de seguir los pasos para la carga del archivo en la plataforma de Moodle
Actores	Dispositivo y Docente
Precondiciones	Se tuvo que culminar el proceso de Registro de Asistencia
Flujo Normal	Figura 13. Proceso para el envío de asistencia. Fuente: autores.
Flujo Alternativo	Esperar que se cumpla hora de finalización de la sesión
Postcondiciones	Se debe de seguir el paso a paso para subir el archivo a Moodle

9.8. Anexo 8

Nombre	Carga archivo HTML
Autor	Zinedine Comas
Fecha	06/04/2024
Descripción	En este proceso el sistema requiere de un archivo obtenido desde el sistema académico, donde se encuentre la información de los espacios académicos a cursar durante un semestre
Actores	Administrador
Precondiciones	Contar con el reporte de sistema académico
Flujo Normal	Figura 14. Proceso para carga de archivo HTML. Fuente: autores.
Flujo Alternativo	Crear sesiones en la pestaña "Crear nueva sesión"
Postcondiciones	Se almacena: Semestre, Programa y código de programa; Día, Fecha y hora de inicio y final, grupo y aula, Nombre de responsable, Nombre de espacio académico

9.9. Anexo 9

```

use ProyectoLabManager
insert into Estudiante (Documento_Estudiante, Nombre_Estudiante,
Apellido_Estudiante)
select Documento, Nombres, Apellidos from
[172.18.8.166\SQLEXPRESS,51579].[LabManager].[Asistencia].[View_AsistenciaEstudiante
]
where Documento collate SQL_Latin1_General_CP1_CI_AS
not in (
select Documento_Estudiante collate SQL_Latin1_General_CP1_CI_AS from
Estudiante)

```

9.10. Anexo 10

Tiempos Promedio Toma asistencia		
Espacio Académico	Tiempo Total(S)	Tiempo X Estudiante(S)
Programación II 4A	145,07	6,307
Programación II 4B	152,73	6,640
Electrónica II 4B	123,68	4,947
Sistemas digitales II 4A	138,07	7,671
Programación web	103,83	14,833
Promedio	132,676	8,080

9.11. Anexo 11

Tiempos Promedio Carga asistencia		
Espacio Académico	Tiempo Total(S)	Tiempo X Estudiante(S)
Programación II 4A	139	6,04
Programación II 4B	77	3,35
Electrónica II 4B	148	5,92
Sistemas digitales II 4A	42,73	2,37
Programación web	70,78	10,11
Promedio	95,502	5,56

9.12. Anexo 12

Porcentaje fallo					
Espacio académico	Sesión 1	Sesión 2	Sesión 3	Sesión 4	Total
Programación II 4A	21,73%	4,34%			13,04%
Programación II 4B	26,32%	0,00%			13,16%
Electrónica II 4B	20,00%	4,00%	5%		9,67%
Sistemas digitales II 4A	22,20%	72,30%	44%	40%	44,73%
Programación web	No registrada	0,00%	No registrada	50%	25,00%
Fechas	20 al 26 ago	27 al 2 sep	3 al 9 sep	10 al 16 sep	21,12%

Porcentaje fallo	
Porcentaje por estudiante	Porcentaje de fallo
24%	3,12%
24%	3,15%
26%	2,52%
19%	8,39%
7%	1,82%
TOTAL	3,80%