



**ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ
MANUEL FÉLIX LÓPEZ**

CARRERA DE INFORMÁTICA

**INFORME DE TRABAJO DE INTEGRACIÓN CURRICULAR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
INFORMÁTICA**

MECANISMO: SISTEMATIZACIÓN DE EXPERIENCIAS

TEMA:

**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A
LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL
EDIFICIO DE LA CARRERA DE COMPUTACIÓN**

AUTORES:

**LUIS EDUARDO CHICHANDA GARCÍA
OLIVER ANTONIO VERA SALAZAR**

TUTOR:

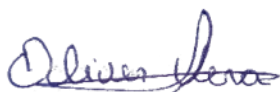
ING. FERNANDO RODRIGO MOREIRA MOREIRA, MGTR.

CALCETA, MARZO DE 2022

DECLARACIÓN DE AUTORÍA

Yo **OLIVER ANTONIO VERA SALAZAR** con cédula de ciudadanía **1315706448**; y **LUIS EDUARDO CHICHANDA GARCÍA**, con cédula de ciudadanía **1314593672**, declaramos bajo juramento que el Trabajo de Titulación titulado: **SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN** es de nuestra autoría, que no ha sido previamente presentado para ningún grado o calificación profesional, y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, concedo a favor de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos, conservando a mi favor todos los derechos patrimoniales de autor sobre la obra, en conformidad con el Artículo 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación.



OLIVER A. VERA SALAZAR

CC: 1315706448



LUIS E. CHICHANDA GARCÍA

CC: 1314593672

AUTORIZACIÓN DE PUBLICACIÓN

OLIVER ANTONIO VERA SALAZAR, con cédula de ciudadanía **1315706448** y **LUIS EDUARDO CHICHANDA GARCÍA**, con cédula de ciudadanía **1314593672**; autorizamos a la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, la publicación en la biblioteca de la institución del Trabajo de Titulación titulado: **SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y total autoría.



OLIVER A. VERA SALAZAR

CC: 1315706448



LUIS E. CHICHANDA GARCÍA

CC: 1314593672

CERTIFICACIÓN DE TUTOR

FERNANDO RODRIGO MOREIRA MOREIRA, certifica haber tutelado el Trabajo de Integración Curricular: **SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**, que ha sido desarrollado por **LUIS EDUARDO CHICHANDA GARCÍA** y **OLIVER ANTONIO VERA SALAZAR**, previa la obtención del título de Ingeniero en Informática, de acuerdo al **REGLAMENTO PARA LA ELABORACIÓN DE TRABAJO DE TITULACIÓN ESPECIAL DE PROGRAMAS DE GRADO** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

ING. FERNANDO R. MOREIRA MOREIRA, MGTR

CC: 1311726689

TUTOR

APROBACIÓN DEL TRIBUNAL

Los suscritos integrantes del tribunal correspondiente, declaran que han **APROBADO** el Trabajo de Titulación titulado: **SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**, que ha sido desarrollado por por **LUIS EDUARDO CHICHANDA GARCÍA** y **OLIVER ANTONIO VERA SALAZAR**, previo a la obtención del título de Ingeniero en Informática, de acuerdo al **REGLAMENTO PARA LA ELABORACIÓN DE TRABAJO DE TITULACIÓN ESPECIAL DE PROGRAMAS DE GRADO** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

ING. LUIS C. CEDEÑO VALAREZO, MGTR.
CC:1306246651
PRESIDENTE DEL TRIBUNAL

ING. ALFONSO T. LOOR VERA, MGTR.
CC:1311655938
MIEMBRO DEL TRIBUNAL

ING. ANGEL A. VÉLEZ MERO, MGTR.
CC:1308648565
MIEMBRO DEL TRIBUNAL

AGRADECIMIENTO

Queremos expresar nuestro infinito agradecimiento, en primer lugar, a Dios por guiarnos en nuestros caminos y por brindarnos salud, fortaleza y capacidad, a cada una de esas personas que nos han acompañado en cada una de las fases ya sea con palabras de alientos o con algunas pautas. A mi querida carrera de Computación y a todos sus docentes, por brindarnos su paciencia, orientación y sus conocimientos el cual nos fue de muchísima ayuda en el desarrollo de nuestro trabajo de titulación. Un agradecimiento en especial a nuestro guía Mgtr. Fernando Rodrigo Moreira Moreira por su valioso apoyo, sabiduría y sus sinnúmeros de consejos, los cuales nos condujeron paso a paso para lograr el objetivo planteado desde un inicio.

LOS AUTORES

DEDICATORIA

A mis padres Cruz María Salazar Zambrano y José Oliver Vera Carreño por todo su apoyo que me brindaron día a día para llegar a cumplir mi objetivo y demostrarme que a pesar de que el camino sea complicado con esfuerzo y de dedicación todo es posible.

A mis compañeros, que se convirtieron en mis hermanos y hermanas con los cuales compartí momentos muy especiales y únicos, sin importar el camino que cada cual escoja siempre recordaré con felicidad y nostalgia las amanecidas mientras hacíamos tareas que parecían que nunca acababan.

A mis amigos y familia en general por depositar la confianza y brindarme esas palabras de aliento para seguir adelante a pesar de las adversidades que se me presentaron en el transcurso de esta ardua travesía.

OLIVER A. VERA SALAZAR

DEDICATORIA

Dedico mi trabajo de titulación con todo mi cariño, a mi madre Carmen Lionis García Medranda, a mi señor padre Víctor Justino Chichanda Basurto, por ser ellos el pilar más importante y mostrarme cada día su apoyo incondicional, a cada uno de mis profesores por compartir sus conocimientos conmigo, al Ing. Kelvin Agustín Rosado Cusme, por tener fe en mí y darme su apoyo cada día en seguir adelante, a cada uno de mis compañeros de clases por apoyarnos y darnos la mano entre nosotros.

LUIS E. CHICHANDA GARCIA

CONTENIDO GENERAL

CARÁTULA	i
DECLARACIÓN DE AUTORÍA	ii
AUTORIZACIÓN DE PUBLICACIÓN.....	iii
CERTIFICACIÓN DE TUTOR	iv
APROBACIÓN DEL TRIBUNAL.....	v
AGRADECIMIENTO.....	vi
DEDICATORIA.....	vii
DEDICATORIA.....	viii
CONTENIDO GENERAL.....	ix
CONTENIDO DE CUADROS Y FIGURAS.....	xi
RESUMEN	xiii
PALABRAS CLAVE.....	xiii
ABSTRACT	xiv
KEYWORDS	xiv
CAPÍTULO I. ANTECEDENTES	1
1.1. DESCRIPCIÓN DE LA INSTITUCIÓN.....	1
1.2. DESCRIPCIÓN DE LA INTERVENCIÓN.....	3
1.3. OBJETIVOS.....	5
1.3.1. OBJETIVO GENERAL.....	5
1.3.2. OBJETIVOS ESPECÍFICOS	5
CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN	6
2.1. PLANIFICACIÓN	6
2.2. DISEÑO	6
2.3. CODIFICACIÓN O DESARROLLO.....	7
2.4. PRUEBAS.....	7
CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA	8
3.1. PLANIFICACIÓN	8
3.2. DISEÑO	13
3.3. CODIFICACIÓN O DESARROLLO.....	19
3.3.1. ITERACIÓN 1: LOGIN.....	21
3.3.2. ITERACIÓN 2 USUARIO.....	22
3.3.3. ITERACIÓN 3: AULAS	24

3.3.4. ITERACIÓN 4: BLOQUE	25
3.3.5. ITERACION 5: DISPOSITIVO INTELIGENTE EXTERNO	26
3.3.6. ITERACION 6: ACONDICIONADOR DE AIRES	27
3.3.7. ITERACIÓN 7: RESERVA.....	28
3.3.8. ITERACION 8: SOLICITUD.....	29
3.3.9. ITERACION 9: CONSUMO	31
3.3.10. ITERACION 10: REPORTE	33
3.3.11. ITERACION 11: DASHBOARD.....	33
3.4. PRUEBAS.....	35
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	47
4.1. CONCLUSIONES	47
4.2. RECOMENDACIONES.....	48
BIBLIOGRAFÍA	49
ANEXOS	53

CONTENIDO DE CUADROS Y FIGURAS

Tabla 1. Análisis de la entrevista.....	8
Tabla 2. Grupo de interesado o Stakeholders.....	9
Tabla 3. Requerimientos funcionales.	9
Tabla 4. Requisitos no funcionales.	12
Tabla 5. Plan de iteraciones del sistema multiplataforma.	12
Tabla 6. APIS REST consumidas por el sistema multiplataforma y el hardware.	16
Tabla 7. Tiempo de respuestas Api Rest.	20
Tabla 8. Consumo estimado de dispositivos electrónicos.....	32
Tabla 9. Pruebas de caja negra del sistema multiplataforma.....	39
Figura 1. Caso de uso administrador.....	14
Figura 2. Caso de uso cliente.....	14
Figura 3. Diagrama de modelo entidad relación.....	15
Figura 4. Arquitectura cliente servidor.....	16
Figura 5. Modelo Vista Controlador.....	17
Figura 6. Gestión de usuarios.....	17
Figura 7. Gestión de aula.....	18
Figura 8. Dashboard Sistema web.....	18
Figura 9. Menú aula administrador.....	19
Figura 10. Menú solicitud docente.....	19
Figura 11. Esquema API REST Sistema multiplataforma.....	20
Figura 12. Login web.....	22
Figura 13. Login móvil.....	22
Figura 14. Usuario web.....	23
Figura 15. Usuario móvil.....	23
Figura 16. Aulas web.....	24
Figura 17. Aulas móviles.....	24
Figura 18. Bloque web.....	25
Figura 19. Bloque móvil.....	25
Figura 20. Arduino web.....	26
Figura 21. Arduino móvil.....	26
Figura 22. Acondicionador de aires web.....	27
Figura 23. Acondicionador de aires móvil.....	27

Figura 24. Reserva web administrador.....	28
Figura 25. Reserva móvil administrador.....	28
Figura 26. Reserva web cliente.....	29
Figura 27. Reserva móvil cliente.....	29
Figura 28. Solicitud web administrador.....	30
Figura 29. Solicitud móvil administrador.....	30
Figura 30. Solicitud web cliente.....	31
Figura 31. Solicitud móvil cliente.....	31
Figura 32. Consumo por mes.....	32
Figura 33. Consumo total por periodo académico.....	33
Figura 34. Reporte.....	33
Figura 35. Dashboard web administrador y cliente.....	34
Figura 36. Dashboard móvil administrador y cliente.....	34
Figura 37. Prueba de tiempo de respuesta a la API Usuario con la herramienta Postman.....	35
Figura 38. Prueba de tiempo de respuesta a la API Solicitud con la herramienta Postman.....	36
Figura 39. Prueba de tiempo de respuesta a la API Reserva con la herramienta Postman.....	36
Figura 40. Prueba prototipo interactuando con el sistema mediante una identificación (Tag) registrado.....	37
Figura 41. Prueba prototipo interactuando con el sistema mediante una identificación (Tag) no registrado.....	37
Figura 42. Prueba de apertura del laboratorio mediante el sistema Web.....	38
Figura 43. Prueba de apertura del laboratorio mediante la app móvil.....	39
Figura 44. Ingreso de credenciales sistema Web y app móvil.....	42
Figura 45. Ingreso con credenciales correctas al sistema web y móvil.....	43
Figura 46. Ingreso con credenciales incorrectas al sistema web y móvil.....	43
Figura 47. Alerta aula creada correctamente sistema Web.....	44
Figura 48. Alerta aula creada correctamente app móvil.....	44
Figura 49. Alerta Arduino creado correctamente sistema Web.....	45
Figura 50. Alerta Arduino creado correctamente app móvil.....	45
Figura 51. Alerta solicitud aceptada correctamente app móvil.....	46

RESUMEN

Este trabajo de titulación tuvo como objetivo desarrollar un sistema multiplataforma para la gestión y control de acceso a las aulas, luminarias y acondicionadores de aire en el edificio de la carrera de Computación. Para el cumplimiento del mismo se empleó la metodología Programación Extrema (XP) la cual cuenta con 4 fases; para la fase de planificación se crearon los casos de usos a partir de lo que se obtuvo en la entrevista que se realizó al representante de la Unidad de Docencia de Investigación y Vinculación del área de infraestructura para la obtención de los requerimientos, en la segunda fase se diseñaron las interfaces de cada uno de los módulos del sistema multiplataforma, bases de datos, además, se desarrollaron las APIS REST y se estableció el canal de comunicación entre el prototipo hardware y el sistema multiplataforma; en la fase de desarrollo se codificaron los módulos del sistema. Para ello se emplearon las tecnologías ASP.NET y Ionic; en la fase de prueba se realizaron las verificaciones necesarias para el correcto desempeño del sistema. Con el procedimiento antes descrito se logró obtener como resultado un sistema web y una plataforma móvil cuyos datos son gestionados por servicios APIS las cuales son capaces de optimizar y gestionar el acceso a las aulas, controlar las luminarias y acondicionador de aires, llegando a la conclusión que el uso de este sistema multiplataforma tubo impactos positivos en la gestión y control de las aulas de la carrera de Computación.

PALABRAS CLAVE

Ionic, APIS REST, metodología XP, ASP.NET, casos de uso.

ABSTRACT

The objective of this degree work was to develop a multiplatform system for the management and control of access to classrooms, lights and air conditioners in the Computing career building. To comply with it, the Extreme Programming (XP) methodology was used, which has 4 phases; for the planning phase, the use cases were created based on what was obtained in the interview that was carried out with the representative of the Research and Linkage Teaching Unit of the infrastructure area to obtain the requirements, in the second phase. The interfaces of each of the modules of the multiplatform system, databases were designed, in addition, the REST APIs were developed and the communication channel between the hardware prototype and the multiplatform system was established; in the development phase the system modules were coded. For this, the ASP.NET and Ionic technologies were used; in the test phase, the necessary verifications were carried out for the correct performance of the system. With the procedure described above, it was possible to obtain as a result a web system and a mobile platform whose data is managed by APIS services which are capable of optimizing and managing access to classrooms, controlling lights and air conditioners, reaching the conclusion that the use of this multiplatform system had positive impacts on the management and control of the classrooms of the Computing major.

KEYWORDS

Ionic, REST APIS, XP methodology, ASP.NET, use cases.

CAPÍTULO I. ANTECEDENTES

1.1. DESCRIPCIÓN DE LA INSTITUCIÓN

“La Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López (ESPAM MFL) fue creada en abril de 1999 para contribuir con otros organismos, en el desarrollo del país, por medio de la investigación científica, la enseñanza universitaria y el emprendimiento.” Tanto el entorno rural y socioeconómico de Manabí, resultaron importantes para la elección de todas las carreras, las mismas que están asociadas a las empresas dedicadas al agro [1].

La creación de esta universidad, nació con la demanda y necesidad de la zona agropecuaria; a partir de esto se procedió con la inauguración de las carreras de Agrícola, Pecuaria, Medio ambiente, Agroindustria, y después en el transcurso de un tiempo se crearon las carreras de Informática, Administración de Empresas y Administración Pública. Finalmente, se fundó la carrera de Turismo tomando en cuenta el sector con más potencial y apogeo en los momentos actuales en el Ecuador. “En la actualidad en acuerdo de los sectores importantes y las industrias priorizadas por el cambio de la matriz productiva, las carreras de la ESPAM MFL (siguiendo las directrices del CES (Consejo de Educación Superior), son:” Ingeniería Agrícola, Ingeniería Ambiental, Medicina Veterinaria, Turismo, Agroindustria, Administración Pública, Administración de Empresas y Computación) estas hallan su justificación, en lo que aportan a la enseñanza del talento humano para el cambio de la misma [1].

La carrera de Computación está relacionada “con el sector de bienes y servicios tecnológicos (hardware, software y servicios informáticos), aporta al cambio de la matriz productiva porque esta es de aplicación de todos los sectores e industrias estratégicas y además es un producto versátilmente exportable” [1]. La misma que se encuentra ubicada al oriente de la Escuela Superior Politécnica Agropecuaria de Manabí, en el sector 8. A una latitud de -0.826 grados norte y longitud de -80.182 grados sur, en el sector conocido como sitio El Limón [2]. El edificio de la carrera de Computación fue creado en dos etapas, la primera fue construida en el 2010 y la segunda en el 2011. Pero la infraestructura tecnológica se hizo a partir del año 2012, fue posterior a la infraestructura física, aspirando a un edificio inteligente. Está conformada por tres plantas, la planta baja consta de 3 aulas de cada lado y 3 salas, la primera planta alta consta de 3 aulas de cada lado y una adicional más pequeña,

además, dos baterías sanitarias, oficinas para el área administrativa, oficina del Centro de Aprendizaje de Aplicaciones Informática, sala de profesores y por último la planta final que consta de 3 aulas de cada lado [3]. La misión de esta carrera es la “formación de profesionales íntegros que conjuguen ciencia, tecnología y valores en su accionar, comprometidos con la comunidad en el manejo adecuados de programas y herramientas computacionales de última generación”, conjuntamente la misma tiene como visión “ser referentes en la formación de profesionales de prestigio en el desarrollo de aplicaciones informáticas y soluciones de hardware” [4].

La UDIV (Unidades de Docencia, Investigación y Vinculación) está conformada por la unidad de ciencias básicas, unidad de desarrollo computacional y la unidad de infraestructura, en esta última se desarrollará el sistema multiplataforma en el aula de redes y de estructura de datos para el control y gestión de acceso, luminaria y acondicionador de aires.

1.2. DESCRIPCIÓN DE LA INTERVENCIÓN

De acuerdo a los avances tecnológicos y su aplicación, así como la urgente necesidad de reducir consumos energéticos sin reducir los niveles de confort, y el desarrollo que ha tenido la aplicación de los conceptos de edificios inteligentes en distintos lugares a nivel mundial, es posible proponer un sistema de control global de un edificio que no solo mejore el confort de los usuarios, sino que contribuya a la eficiencia energética, reduzca los costos operativos de los edificios, y ofrezca seguridad a quienes hacen uso de estas instalaciones[5].

Dentro de este progreso tecnológico en la unión de los campos seguridad, información, control, confort y comunicación, podemos situar la tecnología dentro de una institución para lograr controlar la misma, con una gestión eficiente y de manera robusta, en la que la seguridad es un factor principal, por esta razón el uso de estas tecnologías como cámaras, computadoras, celulares ayuda a mejorar la seguridad en el día a día de las personas. En la actualidad los sistemas de control de acceso son los más utilizados cuando de seguridad se habla, su propósito es detectar, registrar y reconocer la presencia de un individuo e identificarlo de manera única [6][7][8][9].

El acceso a espacios requiere tener dispositivos automatizados seguros y confiables que identifiquen a las personas que ingresen a los mismo. Esto permite tener un registro de seguimiento de acceso de los clientes que utilizan el lugar ya que si existen pérdidas o daños de los elementos dentro del sitio se tomará medidas a los involucrados. Los sistemas de control de acceso brindan una seguridad mucho mejor, el acceso no autorizado se ve disminuido en gran medida. Esto conduce a un proceso a nivel de seguridad mediante el uso de una combinación de algoritmos, motivo por el cual, es indispensable contar con tecnologías que permitan confirmar los usuarios que acceden a sitios de gran seguridad [10][11].

Actualmente los métodos de manejo del control de acceso en los sitios no han cambiado durante las últimas décadas. Para empresas de gran envergadura suele existir un área de recepción, donde el personal que ingresa al edificio debe registrarse con la recepcionista si es un nuevo visitante, o poder presentar el pase necesario autorizado por recursos humanos para acceder a las áreas seguras del edificio [12]. Es por esto que las técnicas de control tienen como objetivo mejorar la supervisión y

la administración de los sistemas mediante la comunicación bidireccional ya sea a través de tecnologías inalámbricas o por cable.

Hoy en día la carrera de Computación cuenta con un procedimiento biométrico de control de acceso, que no se emplea de manera adecuada dado que no se dispone de una plataforma donde se lleve un registro de control de quienes ingresan a las aulas/laboratorios. Por ende, la dirección de la carrera de Computación establece la importancia de poseer un sistema multiplataforma que gestione el control de acceso a las aulas/laboratorios que soporte el proceso de restricción de acceso, así mismo, el apagado de las iluminarias y aires acondicionados por los que es necesario incursionar en opciones que reduzcan este consumo energético para así tener una mejor gestión y control automatizado, que aportará a la seguridad reforzando la identificación del personal y alumnos formalmente activos como miembros de la comunidad universitaria [13][14][15].

El trabajo de titulación pertenece a un proyecto mayor del área de la UDIV de infraestructura, la misma que tiene como objetivo "Ofrecer al estudiante politécnico una sólida formación teórica, metodológica, y práctica que lo capacite en el análisis de problemas relacionados con los procesos del computador, auditoria y redes". Por esto, en este trabajo de titulación se desarrolló el software, el mismo que está involucrado con otro trabajo de titulación definido como "SISTEMA AUTOMATIZADO PARA CONTROL DE ACCESO A AULAS, LUMINARIAS y ACONDICIONADOR DE AIRES DE LA CARRERA DE COMPUTACION DE LA ESPAM MFL", el cual sirvió de sensor y actuador para generar datos a través de este prototipo y así el software lo pudiera consumir y realizar las acciones respectivas, cabe mencionar que este proyecto se desarrollará en un ambiente web y móvil los mismos que se enlazarán a servicios propios del sistema y a servicios con los que cuenta la institución [16].

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Desarrollar un sistema multiplataforma para la gestión y control de acceso a las aulas, luminarias y acondicionador de aires en el edificio de la carrera de Computación.

1.3.2. OBJETIVOS ESPECÍFICOS

- Recopilar los requerimientos funcionales y no funcionales.
- Establecer la arquitectura de comunicación con el hardware.
- Codificar el Sistema utilizando las herramientas de desarrollo.
- Implementar el sistema multiplataforma en la carrera de Computación.

CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN

Este trabajo tuvo como objetivo crear un sistema multiplataforma para la gestión y control de acceso a las aulas, luminarias y acondicionador de aires en el edificio de la carrera de Computación. Para cumplir con este objetivo se empleó la metodología Programación Extrema (XP), la misma que cuenta de 4 facetas que son: Planificación, diseño, codificación o desarrollo y pruebas. Además, se usó la entrevista como técnica de recolección de información [17].

2.1. PLANIFICACIÓN

En la planificación el cliente, en este caso El coordinador de la UDIV de infraestructura, determinó la necesidad del problema que existía en el área mencionada, también en esta fase se estimó el esfuerzo necesario que se efectuó en cada entrega [17]. Mediante las entrevistas realizadas al cliente, se analizó la información obtenida para describir los requisitos del sistema a desarrollar, con el fin de elaborar un documento ERS (Especificación de Requisitos del Sistema) el mismo que es una técnica de especificación de requisitos para darle a conocer al cliente de una forma más detallada la interacción de las funcionalidades del sistema; lo que conlleva al cumplimiento de los objetivos finales[18]. Además, se realizó el plan de iteraciones que muestra las horas de duración, prioridades y el número de iteraciones.

2.2. DISEÑO

“En la fase de diseño la metodología XP sugiere obtener diseños simples y elementales, procurar hacerlo todo lo menos complejo posibles para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos desarrollar” [19]. Una vez establecidos los requerimientos se diseñó un bosquejo de base de datos, cabe recalcar que la misma es una diversidad de datos, información y conocimientos almacenados de manera organizada que se pueda acceder a ellos, administrarlos y actualizarlos [20]. Se establecieron las arquitecturas de comunicación y de desarrollo para el software las cuales son las siguientes.

- La arquitectura cliente-servidor tiene como fin reducir el espacio en memoria que utiliza la base de datos, toda esta información como el consumo de los usuarios

se almacena en un servidor con una base de datos externa. Además, permites consumir los servicios en simultaneo proporcionados por el mismo[21].

- El Modelo-Vista-Controlador nació con los inicios de las interfaces gráficas de usuario, para abrigar las necesidades en la rama del desarrollo de software, el mismo que aligera y da robustez al período de vida del desarrollo [22].

Luego de establecer las arquitecturas de software y del modelo entidad relación se procedió a realizar un esquema de casos de uso de las funcionalidades del producto, seguidamente se diseñaron las interfaces del sistema multiplataforma.

2.3. CODIFICACIÓN O DESARROLLO

“El desarrollo debe ser basado en estándares ya creados, manteniendo el código consistente y fácil de comprender, este debe cumplir con la funcionalidad obteniendo una optimización del mismo” [18]. Luego de establecer los requerimientos y definir la arquitectura del software se procedió al desarrollo del sistema multiplataforma [23].

También se desarrollaron las API Rest que se necesitaban para el funcionamiento del sistema , ya que estas permiten conseguir información o crear acciones en todas las formas posibles: HTTP, JSON ,XML, etc. [24]. Todos estos datos son consumidos por el sistema web, por la aplicación móvil y por el prototipo hardware que hace uso de estas API para su correcto funcionamiento.

2.4. PRUEBAS

En esta fase se realizó la conexión con el prototipo físico para verificar el correcto funcionamiento del sistema multiplataforma, antes de ser implementado se realizaron las pruebas de cajas negra ya que esta es una de las estrategias más utilizadas en la actualidad. Las pruebas de caja negra o pruebas funcionales, se abstraen el contenido, estructura y se basan solamente en las entradas y las salidas. Sólo resta concentrarse en encontrar las circunstancias en las cuales el módulo no responde a las especificaciones esperadas. Para ello se realizan ensayos con grandes cantidades de datos de entrada posible, para determinar un posible error a la salida correspondiente en cada uno [25].

CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA

En esta sección se trabajó en cada fase de la metodología con base a lo planificado en el desarrollo metodológico.

3.1. PLANIFICACIÓN

En primera instancia para el desarrollo del sistema multiplataforma se realizaron entrevistas tanto presencial como virtual (Anexo 1) al encargado de la UDIV de infraestructura de la carrera de Computación, se realizaron diversas preguntas referentes a temas como: acceso a las aulas, secciones de aulas, tecnología a utilizar, servicios existentes, control de luminarias y acondicionadores de aire, aplicación móvil para el acceso. Con lo antes mencionado se realizó un análisis de dichas preguntas lo cual sirvió como referencia para conocer las necesidades de dicha área y poder realizar la toma de los requerimiento funcionales y no funcionales. En la tabla 1 se puede evidenciar el análisis obtenido por cada una de las preguntas realizadas a través de la entrevista.

De acuerdo a la entrevista realizada el cliente sugirió utilizar ASP.NET como tecnología de desarrollo web de la mano con SQL Server como gestor de base de datos, ya que esta se viene empleando en el desarrollo de los sistemas de la institución. Además, para la parte móvil se sugirió utilizar Ionic como framework para el desarrollo de la misma.

Tabla 1. Análisis de la entrevista.

N	Pregunta	Análisis
1	¿Cómo se lleva a cabo el proceso de ingreso a las aulas actualmente?	El proceso actual de ingreso a las aulas se lleva a cabo mediante un sistema biométrico y llaves convencionales.
2	¿Cuál es el problema a resolver?	Llevar una bitácora de gestión y control de acceso.
3	¿Qué funcionalidades tendrá la aplicación móvil?	La aplicación móvil pretende ser lo más intuitiva posible, con las características de que podrá abrir la puerta de un aula, pedir acceso, entre otras.
4	¿A quiénes beneficiará el sistema multiplataforma?	Esto beneficiará tanto al personal administrativo como a los estudiantes de dicha entidad.
5	¿Qué espera del sistema multiplataforma?	Un sistema eficiente capaz de resolver las necesidades requeridas por el cliente.

6	¿Cuáles son los reportes que le gustaría que el sistema multiplataforma contemple?	Se realizarán reporte de consumo, acceso, solicitud y reserva.
---	--	--

Fuente: Los Autores.

Con base a los resultados adquiridos de la entrevista se logró determinar la problemática para el desarrollo del sistema multiplataforma. Por lo consiguiente se identificó cada una de las personas o grupos de interesados (stakeholders) con su cargo y relevancia en el desarrollo del proyecto, el mismo que se evidencia en la tabla 2 [26].

Tabla 2. Grupo de interesado o Stakeholders.

Nombre	Cargo	Relevancia en el proyecto
Moreira Pico Ramón Joffre	Director de la carrera de Computación	5
López Zambrano Javier Hernan	Encargado de la UDIV	5
Chichanda García Luis	Desarrollador	4
Vera Salazar Oliver	Desarrollador	4
Moreira Moreira Fernando	Tutor	5

Fuente: Los Autores.

Una vez realizado el análisis de la entrevista realizada se logró determinar la información necesaria para el desarrollo de un ERS que se muestra en el Anexo 2, en el mismo que se enlistan los requisitos funcionales que se evidencian en la tabla 3. y los no funcionales se evidencian en la tabla 4.

Tabla 3. Requerimientos funcionales.

	Ref: #	Requerimiento	Función	Rol	Plataforma
Sesión	RF1	Login	Ingreso de las credenciales para identificación de ingreso de los usuarios.	Administrador Usuarios	Web-Móvil
Usuario	RF2	Usuarios	Permite ver todos los usuarios, añadir y visualizar los tag que se le han asignado a dicho usuario.	Administrador	Web-Móvil
Aulas	RF3	Crear Aula	Permite el ingreso de nueva aula	Administrador	Web-Móvil
	RF4	Modificar aula	Permite la modificación de la información del aula.	Administrador	Web-Móvil

	RF5	Cargar aula	Permite mostrar las aulas.	Administrador Cliente	Web-Móvil
	RF6	Eliminar aula	Permite eliminar un aula.	Administrador	Web-Móvil
Reserva	RF7	Crear reserva	Permite agregar un docente y asignarlo a un aula específica en un horario asignado.	Administrador	Web-Móvil
	RF8	Modificar reserva	Permite la modificación de la información de la reserva.	Administrador	Web-Móvil
	RF9	Cargar Reserva	Permite Mostrar la reserva.	Administrador Cliente	Web-Móvil
	RF10	Eliminar reserva	Permite quitar una reserva.	Administrador	Web-Móvil
Solicitud	RF11	Aceptar y denegar solicitudes	Permite aceptar o denegar las solicitudes.	Administrador	Web-Móvil
	RF12	Cargar solicitudes	Permite mostrar las solicitudes realizadas.	Administrador Cliente	Web-Móvil
	RF13	Crear solicitud	Permite el ingreso de los datos del cliente, que aula quiere reservar y el horario.	Cliente	Web-Móvil
	RF14	Modificar solicitud	Permite la modificación de la información de una solicitud.	Cliente	Web-Móvil
	RF15	Eliminar solicitud	Permite quitar una solicitud.	Cliente	Web-Móvil
Bloques	RF16	Crear bloque	Permite el ingreso de un nuevo bloque.	Administrador	Web-Móvil
	RF17	Modificar bloque	Permite la modificación de la información del bloque.	Administrador	Web-Móvil
	RF18	Cargar bloque	Permite mostrar los bloques.	Administrador	Web-Móvil
Dispositivo inteligente externo	RF19	Crear Arduino	Permite el ingreso de un nuevo Arduino.	Administrador	Web-Móvil
	RF20	Modificar Arduino	Permite la modificación de la información del Arduino.	Administrador	Web-Móvil
	RF21	Cargar Arduino	Permite mostrar los Arduinos.	Administrador	Web-Móvil
	RF22	Eliminar Arduino	Permite quitar Arduino.	Administrador	Web-Móvil

Acondicionador de aires	RF23	Encendido y apagado	Al momento de hacer uso de un aula el sistema automáticamente manda una señal ya sea de apagado o encendido del acondicionador de aires.	Sistema	Web
	RF24	Crear Aire	Permite el ingreso de un nuevo Aire.	Administrador	Web-Móvil
	RF25	Modificar Aire	Permite la modificación de la información del Aire.	Administrador	Web-Móvil
	RF26	Cargar Aire	Permite mostrar los Aires.	Administrador	Web-Móvil
	RF27	Eliminar Aire	Permite quitar Aire.	Administrador	Web-Móvil
Envío de correos	RF28	Enviar correos	Al crear una solicitud, automáticamente se le envía un correo al administrador notificándole que se ha generado una solicitud.	Sistema	Web
Consumo	RF29	Consumo energético	Se guarda el consumo eléctrico de cada aula para su respectivo reporte.	Sistema Administrador	Web
Reporte	RF30	Generar reporte	Permitirá generar reportes del aulas, solicitudes y reservas.	Administrador	Web
Dashboard	RF31	Dashboard	Permite interactuar y visualizar el estado de las diferentes aulas(Acceso, consumo, cámaras, temperatura, estado de Arduino)	Administrador Cliente	Web-Móvil
Información tiempo real	RF32	Información tiempo real	El sistema deberá enviar y recibir información en tiempo real.	Sistema	Web

Fuente: Los Autores.

En la tabla 3 se muestra un total de doce módulos subdivididos en 32 requerimientos funcionales que describen las funciones de cada uno, a su vez se describen los tipos

de usuarios que existen y el acceso que tendrán en los módulos descritos. Así mismo en la tabla 4 se muestra un total de tres requisitos no funcionales.

Tabla 4. Requisitos no funcionales.

Ref: #	Función	Rol
RNF1	El sistema web tendrá una interfaz sencilla e intuitiva que ayudará al fácil manejo para los usuarios.	Administrador
RNF2	El sistema tendrá un manual de uso que facilitará el manejo del mismo por parte de los usuarios.	Administrador
RNF3	El sistema se desarrollará teniendo en cuenta que estará disponible las 24 horas del día y accesible desde cualquier lugar.	Administrador

Fuente: Los Autores.

Una vez realizado los requerimientos se definió un plan de iteraciones con cada una de las actividades a realizar por cada miembro del equipo de desarrollo. A continuación, en la tabla 5 se muestran las 13 iteraciones.

Tabla 5. Plan de iteraciones del sistema multiplataforma.

Plan de Iteraciones					
# Iteración	Módulo	Actividades	Requerimiento	Prioridad	Tiempo (Horas)
1	Inicio sesión	Login	RF1	Alta	48
2	Usuario	Validar tipo de usuario Ver usuarios Añadir Tag Ver tag de cada usuario	RF2	Alta	72
3	Aula	Crear Aula Editar Aula Eliminar Aula Abrir Aula Ver Aulas	RF3, RF4, RF5, RF6	Alta	168
4	Bloque	Crear Bloque Editar Bloque Ver Bloques	RF14, RF15, RF16	Alta	144
5	Dispositivo inteligente externo	Crear Arduino Editar Arduino Eliminar Arduino Ver Arduinos	RF17, RF18, RF19, RF20	Alta	168

6	Aire	Crear Aire	RF21, RF22, RF23, RF24, RF25	Alta	192
		Editar Aire			
		Eliminar Aire			
		Ver Aires			
7	Reserva	Crear Reserva	RF7, RF8, RF9, RF10	Alta	216
		Editar Reserva			
		Eliminar Reserva			
		Ver Reserva			
8	Solicitud	Crear Solicitud	RF11, RF12, RF13, RF14, RF15	Alta	264
		Ver Solicitud			
		Editar Solicitud			
		Eliminar Solicitud			
		Aceptar Solicitud			
		Rechazar Solicitud			
9	Información tiempo real	Enviar y recibir Información tiempo real	RF32	Alta	96
10	Reporte	Reporte de reservas	RF30	Alta	144
11	Dashboard	Estado Aulas	RF31	Alta	360
12	Envíos de correos	Enviar correos	RF28	Media	72
13	Consumo	Ver consumo del edificio	RF29	Media	96

Fuente: Los Autores.

3.2. DISEÑO

- Siguiendo con lo planificado se procedió realizar los casos de uso los cuales da una visión del funcionamiento del sistema y de cómo interactúan los diferentes tipos de usuario con el mismo. A continuación, en las figuras 1 y 2 se muestran los casos de usos generarles de cada uno de los usuarios que interactuaran con el sistema, en el Anexo 3 se denotan los siguientes casos de usos más específicos.

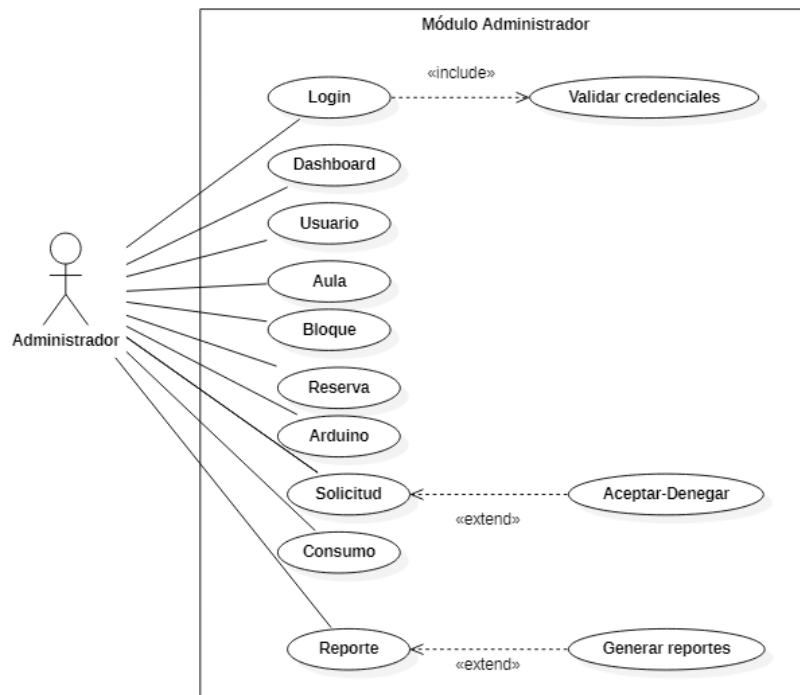


Figura 1. Caso de uso administrador

Fuente: Los Autores.

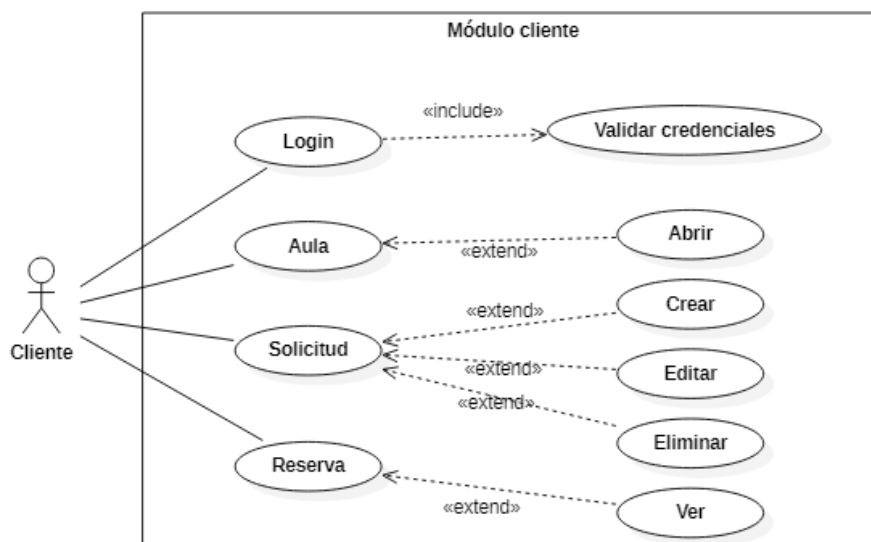


Figura 2. Caso de uso cliente

Fuente: Los Autores.

- Luego de haber realizado los casos de uso se diseñó un esquema de base de datos para el sistema multiplataforma, este fue realizado en el gestor de base de datos SQL Server, el mismo que cuenta con 19 tablas relacionadas entre sí para

su correcto funcionamiento. A continuación, se muestra el diagrama en la figura 3.

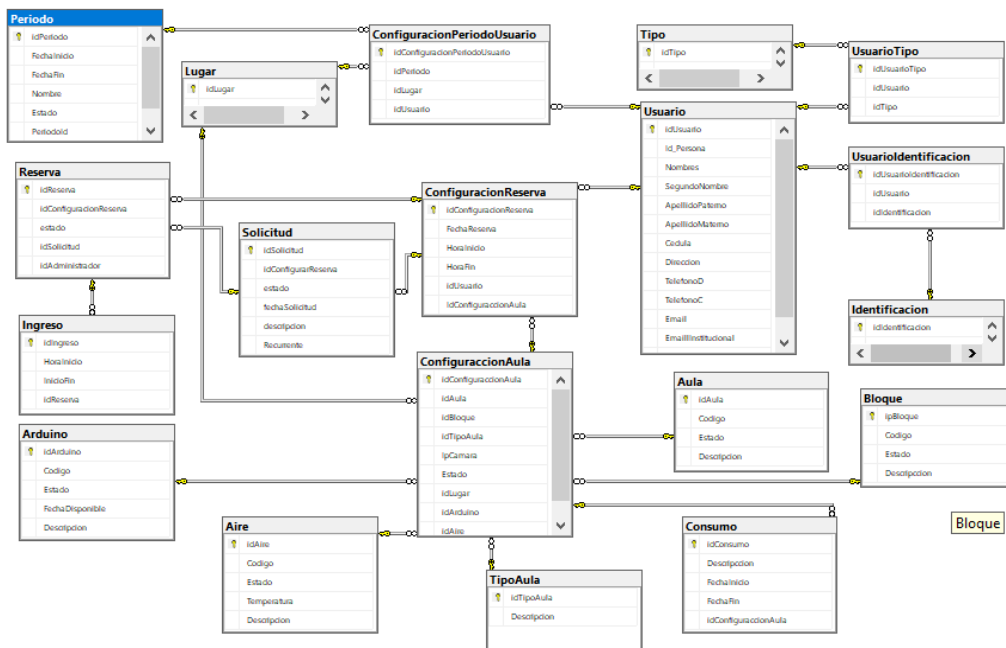


Figura 3. Diagrama de modelo entidad relación
Fuente: Los Autores.

- Además, escogió un modelo de conectividad de intercambio de datos a través del back-end y seguidamente se escogió el protocolo de mensajería HTTP. En la figura número 4 se presenta la arquitectura cliente servidor que se utilizó, en la parte del servidor se muestran las API REST que son los servicios que serán consumidos con mayor facilidad por el sistema web, móvil y el hardware, las cuales se evidencian en la tabla 6.

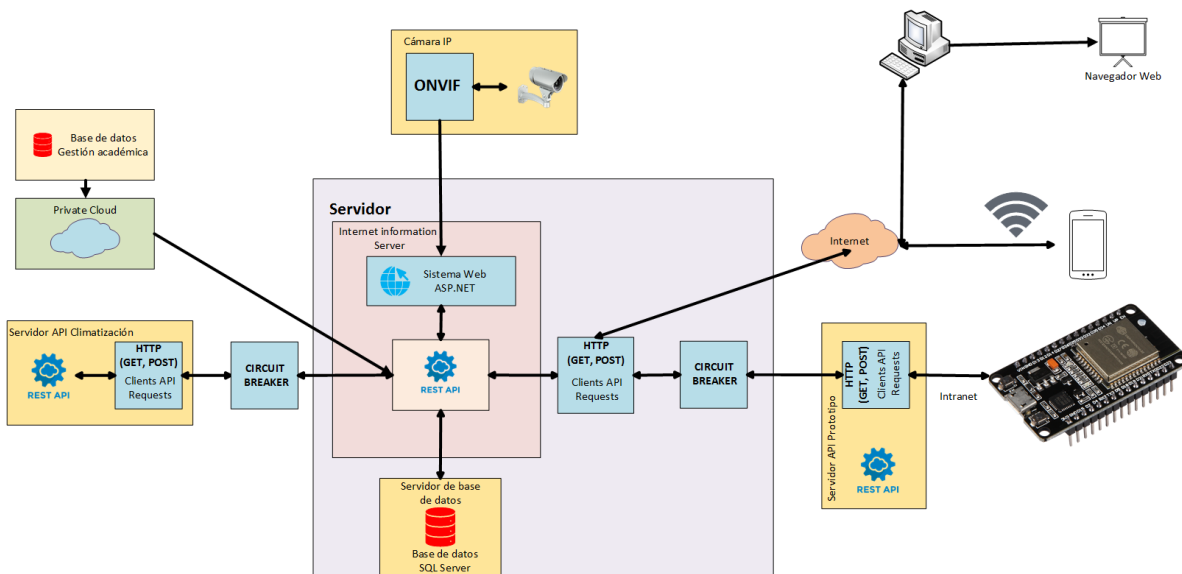


Figura 4. Arquitectura cliente servidor
Fuente: Los Autores.

Tabla 6. APIS REST consumidas por el sistema multiplataforma y el hardware.

Módulo	API REST
Inicio sesión	/Login
Usuario	/Usuario
Aula	/Aula
Bloque	/Bloque
Dispositivo inteligente externo	/Arduino
Aire	/Aire
Reserva	/Reserva
Solicitud	/Solicitud
Consumo	/Consumo
Reporte	/Reporte
Dashboard	/Home
Envío de correos	/Correo

Fuente: Los Autores.

- En la figura 5 se evidencia el esquema Modelo, Vista, Controlador que se diseñó para el desarrollo del sistema multiplataforma, la misma que realiza las siguientes funciones: las peticiones del usuario se reciben desde alguna ventana (Vista), luego se direcciona a ejecutar alguna instrucción específica al controlador para luego ser llevadas al modelo y hacer peticiones a las API REST y así devolver una respuesta a la vista.

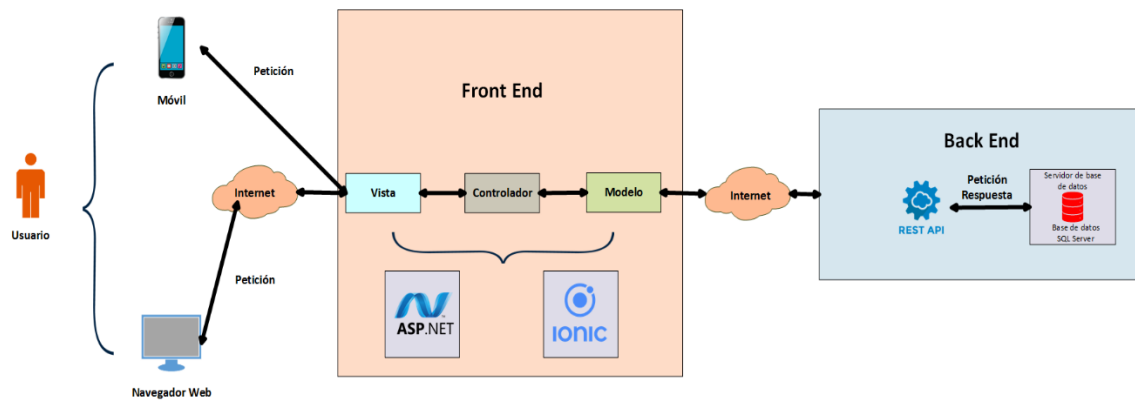


Figura 5. Modelo Vista Controlador

Fuente: Los Autores.

- Una vez definida las arquitecturas y teniendo un panorama claro del funcionamiento del sistema se procedió al diseño de las interfaces del sistema web y de la parte móvil, la misma que se realizó en la herramienta de Google Lucidchart la cuales esquematizan la forma de cómo quedaría el producto software final. En las figuras 6, 7, 8, 9, 10 se muestran las interfaces más importantes a considerar del sistema multiplataforma, así mismo, en el Anexo 4 se evidencian todas las interfaces diseñadas.

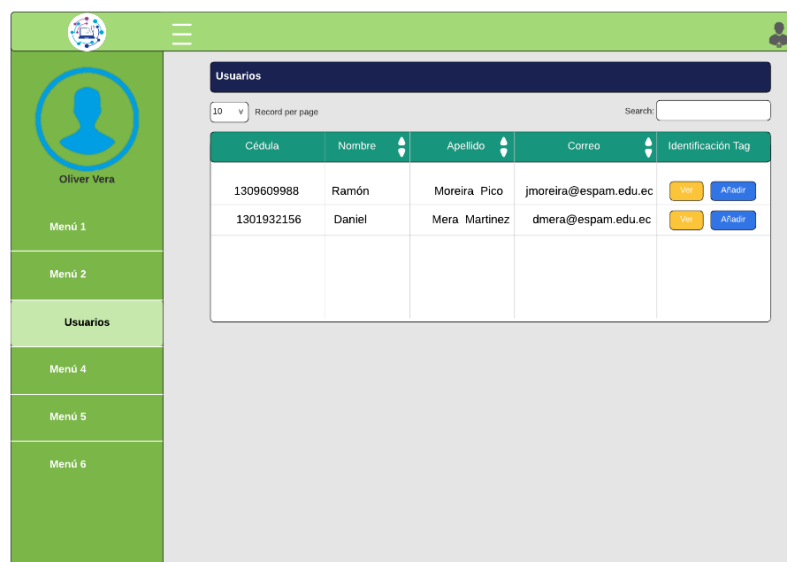


Figura 6. Gestión de usuarios

Fuente: Los Autores.

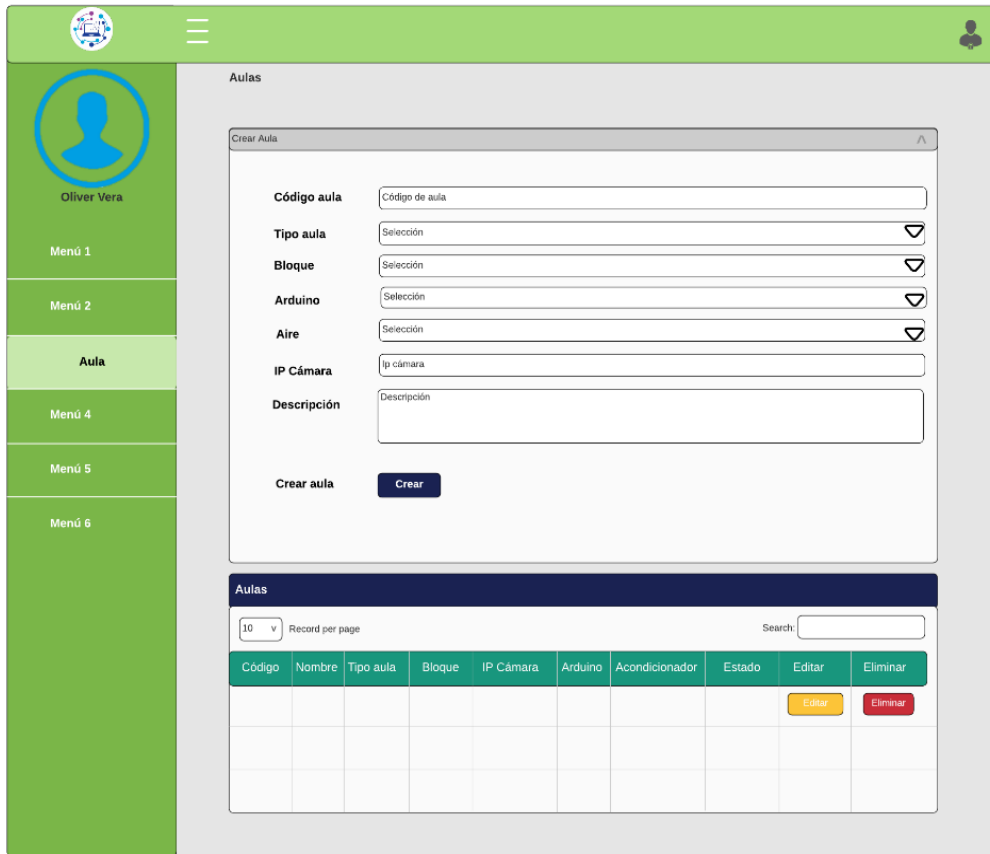


Figura 7. Gestión de aula
Fuente: Los Autores.

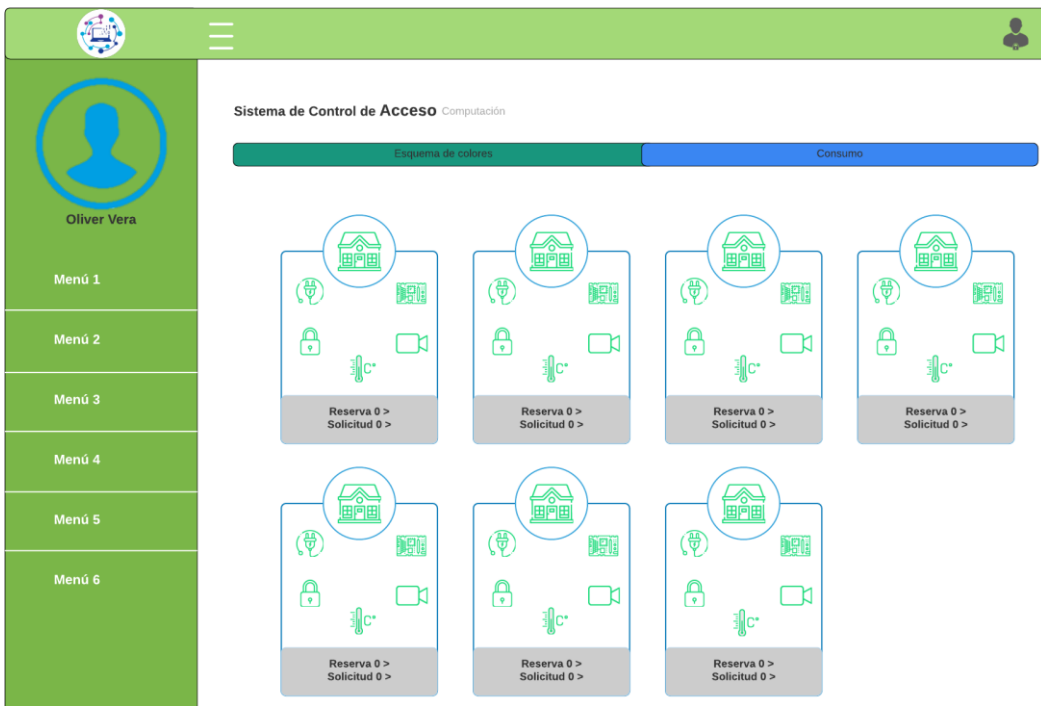


Figura 8. Dashboard Sistema web
Fuente: Los Autores.

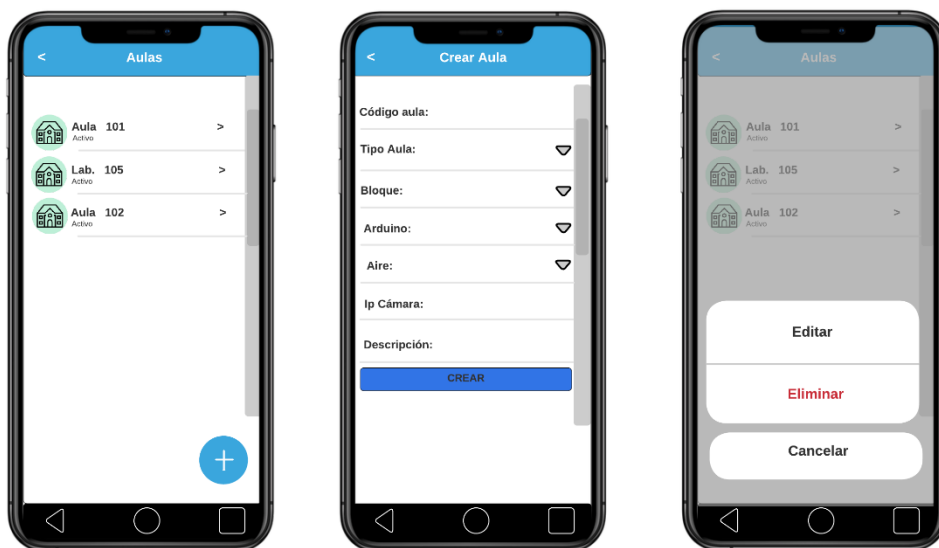


Figura 9. Menú aula administrador
Fuente: Los Autores.

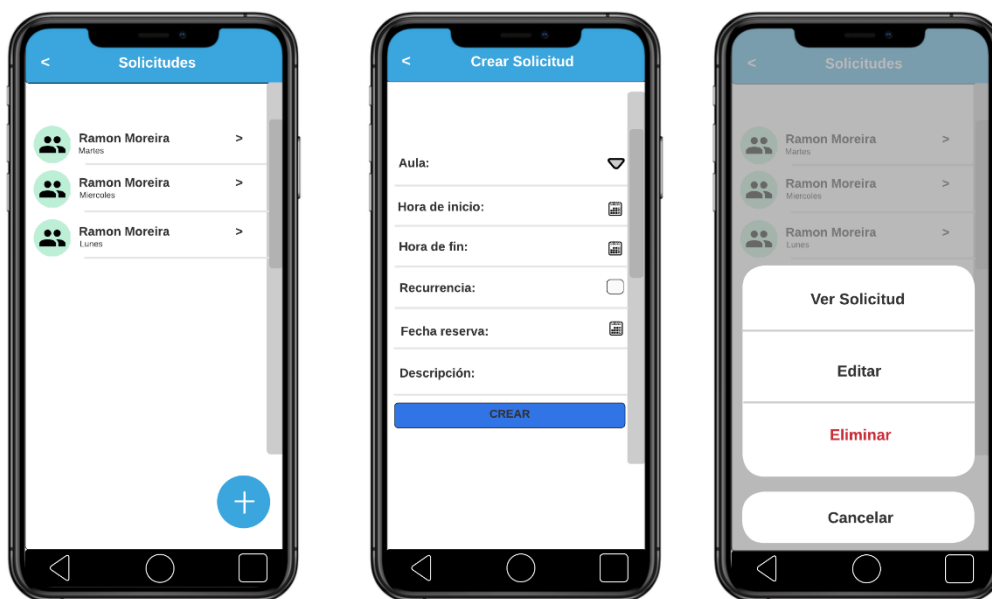


Figura 10. Menú solicitud docente
Fuente: Los Autores.

3.3. CODIFICACIÓN O DESARROLLO

En base a los requerimientos se procedió a la creación de las API REST que fueron desarrolladas en ASP.NET Framework, las cuales son consumidas por el sistema multiplataforma, en la figura 11 se simboliza el bosquejo general de las APIS REST.

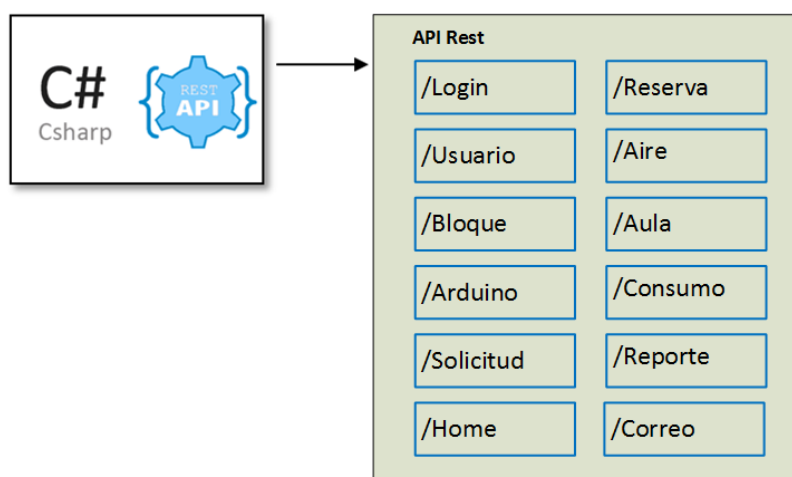


Figura 11. Esquema API REST Sistema multiplataforma

Fuente: Los Autores.

Luego del desarrollo de las Apis Rest se procedió a realizar las pruebas respectivas para estimar el tiempo de respuesta de cada una de ellas y verificar el correcto funcionamiento de las mismas, en la tabla 7 se plasman detalladamente el tiempo de respuesta y el estado de estas.

Tabla 7. Tiempo de respuestas Api Rest.

Módulos	Api Rest	Método	Tiempo de respuesta	Estado
Sesión	/Login	Post	146 ms	Correcto
Usuarios	/Usuario	Get	101 ms	Correcto
Reservas	/Reserva	Get	233 ms	Correcto
	/Reserva	Post	101 ms	Correcto
	/Reserva	Put	143 ms	Correcto
	/Reserva	Delete	82 ms	Correcto
Aulas	/Aula	Get	201 ms	Correcto
	/Aula	Post	305 ms	Correcto
	/Aula	Put	149 ms	Correcto
	/Aula	Delete	97 ms	Correcto
Bloque	/Bloque	Get	124 ms	Correcto
	/Bloque	Post	172 ms	Correcto
	/Bloque	Put	114 ms	Correcto

Consumo	/Consumo	Get	304 ms	Correcto
Arduino	/Arduino	Get	156 ms	Correcto
	/Arduino	Post	159 ms	Correcto
	/Arduino	Put	108 ms	Correcto
	/Arduino	Delete	72 ms	Correcto
Aire	/Aire	Get	53 ms	Correcto
	/Aire	Post	110 ms	Correcto
	/Aire	Put	75 ms	Correcto
	/Aire	Delete	84 ms	Correcto
Solicitud	/Solicitud	Get	258 ms	Correcto
	/Solicitud	Post	348 ms	Correcto
	/Solicitud	Put	131 ms	Correcto
	/Solicitud	Delete	103 ms	Correcto
Reporte	/Reporte	Get	94 ms	Correcto
Dashboard	/Home	Get	60 ms	Correcto
Envío de correos	/Correo	Post	75ms	Correcto

Fuente: Los Autores.

A continuación, se detallará cada una de las iteraciones que se utilizó para el desarrollo del sistema web y aplicación móvil, mismas donde se utilizaron diferentes lenguajes y framework tales como: ASP.NET, Ionic, CSS, TypeScript, JavaScript, Ajax, Angular.

3.3.1. ITERACIÓN 1: LOGIN

El sistema multiplataforma debe ser capaz de permitir al usuario iniciar sesión para acceder a los diferentes roles que tiene cada uno de los usuarios.

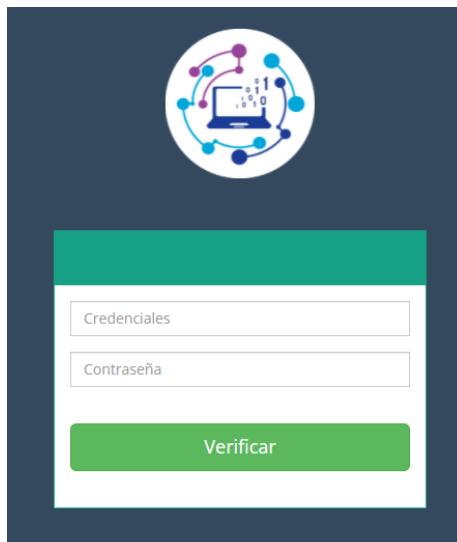
The image shows a web login interface. At the top center is a circular logo with a laptop and binary code. Below the logo is a green horizontal bar. Underneath is a white form with two input fields: 'Credenciales' and 'Contraseña'. Below these fields is a green button labeled 'Verificar'.

Figura 12. Login web
Fuente: Los Autores.

The image shows a mobile login interface. It features a white form centered on a blue-to-teal gradient background. At the top of the form is the same circular logo as in Figure 12. Below the logo are two input fields: 'Cédula' and 'Contraseña'. The 'Contraseña' field has a small eye icon to its right. Below the fields is a blue button labeled 'Verificar'.

Figura 13. Login móvil
Fuente: Los Autores.

3.3.2. ITERACIÓN 2 USUARIO

El sistema debe permitir al administrador ver los usuarios, añadir, ver y eliminar Tags.

Usuario				
10	records per page			Search: <input type="text"/>
Cedula	Nombre	Apellido	Correo	Identificación Tag
0921757282	MARIA	PINCAY CANTOS	maria.pincay@espam.edu.ec	ver Añadir
1301932156	DANIEL	MERA MARTINEZ	dmera@espam.edu.ec	ver Añadir
1306246651	LUIS	CEDEÑO VALAREZO	lcedeno@espam.edu.ec	ver Añadir
1306391614	RICARDO	VELEZ VALAREZO	rvelezvalarezo@espam.edu.ec	ver Añadir
1307723286	GUSTAVO	MOLINA GARZON	gmolina@espam.edu.ec	ver Añadir
1308542099	KELVIN	ROSADO CUSME	krosado@espam.edu.ec	ver Añadir
1308555190	MARYURY	ZAMORA CUSME	mzamora@espam.edu.ec	ver Añadir
1308648565	ANGEL	VELEZ MERO	avelez@espam.edu.ec	ver Añadir
1308755238	JAVIER	LOPEZ ZAMBRANO	jlopez@espam.edu.ec	ver Añadir
1309609988	RAMON	MOREIRA PICO	jmoreira@espam.edu.ec	ver Añadir

Showing 1 to 10 of 18 entries

← Previous 1 2 Next →

Figura 14. Usuario web

Fuente: Los Autores.

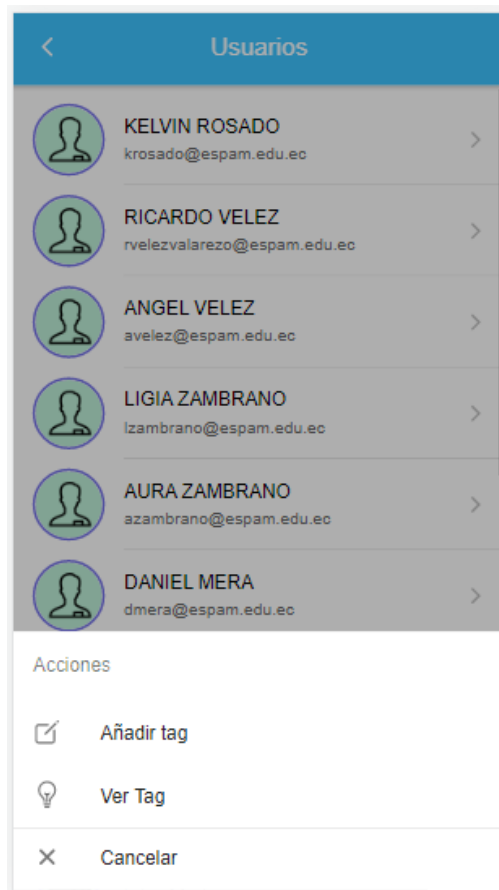


Figura 15. Usuario móvil

Fuente: Los Autores.

3.3.3. ITERACIÓN 3: AULAS

El administrador podrá visualizar de forma más detallada la información del aula, como también crear, editar y eliminar la misma.

Código	Nombre	TipoAula	Bloque	IpCamara	Arduino	Acondicionador	Estado	Editar	Eliminar
Aula 101	Aula 101	Aula	1	185.10.80.33:8082	24 DA CA 5A D7 E8		Activo	Editar	Eliminar
Lab. 101	Laboratorio de Electrónica	Laboratorio	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 102	Laboratorio de Programación y Análisis	Laboratorio	1	185.10.80.33:8082	24 DA CA 5A D7 E8		Activo	Editar	Eliminar
Aula 102	Aula 102	Aula	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 103	Laboratorio de Lenguaje, Programación y Base de Datos	Laboratorio	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 104	Laboratorio de Física	Laboratorio	2	185.10.80.33:8082	24 DA CA EA 4B E8		Activo	Editar	Eliminar
Lab. 203	Laboratorio de Diseño Gráfico2	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 204	Laboratorio de Redes	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 205	Laboratorio de Estructura y Análisis de Algoritmo	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Aula 302	Aula 302	Aula	3	185.10.80.33:8082			Activo	Editar	Eliminar

Figura 16. Aulas web
Fuente: Los Autores.

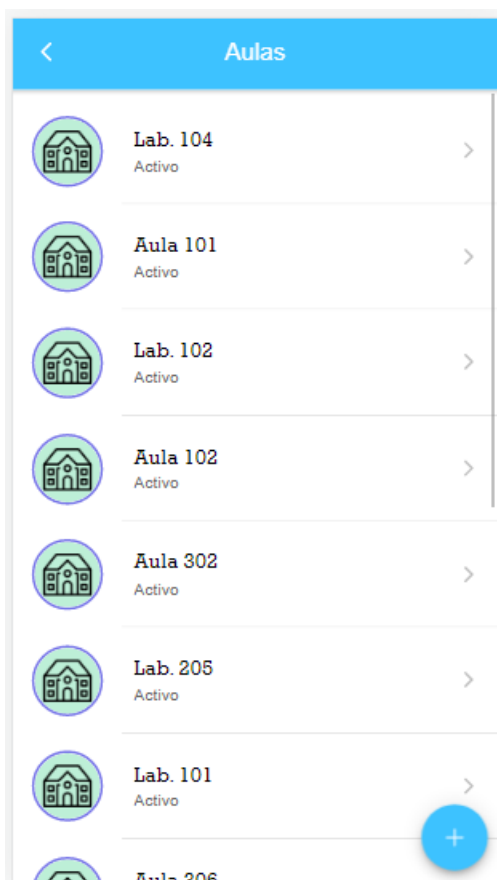


Figura 17. Aulas móviles
Fuente: Los Autores.

3.3.4. ITERACIÓN 4: BLOQUE

El sistema debe ser capaz de ver la información de los bloques, así como editar y crear uno nuevo,

Crear Bloque

Codigo Bloque

Descripción

Crear Bloque

Bloque

10 records per page Search:

Codigo	Estado	Descripción	Editar
1	Activo	1	<input type="button" value="Editar"/>
2	Activo	2	<input type="button" value="Editar"/>
3	Activo	3	<input type="button" value="Editar"/>

Showing 1 to 3 of 3 entries ← Previous 1 Next →

Figura 18. Bloque web
Fuente: Los Autores.

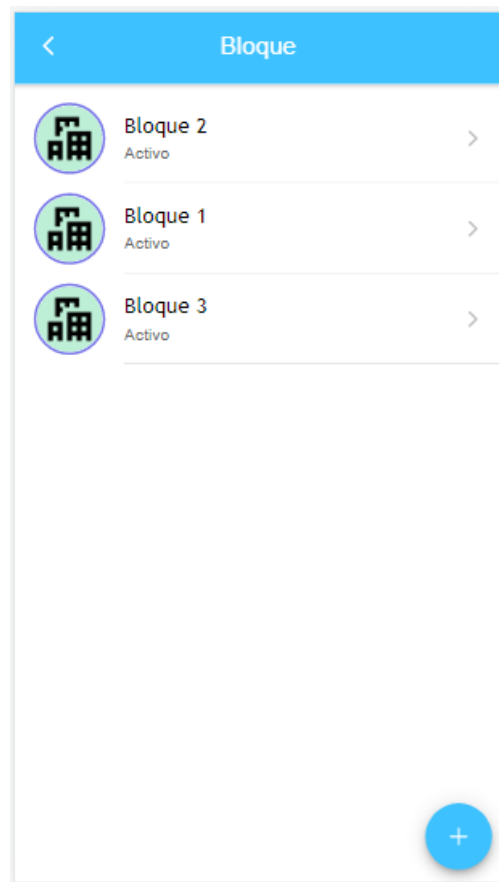


Figura 19. Bloque móvil
Fuente: Los Autores.

3.3.5. ITERACION 5: DISPOSITIVO INTELIGENTE EXTERNO

El administrador podrá visualizar de forma más detallada la información del Arduino, como también crear, editar y eliminar los mismos.

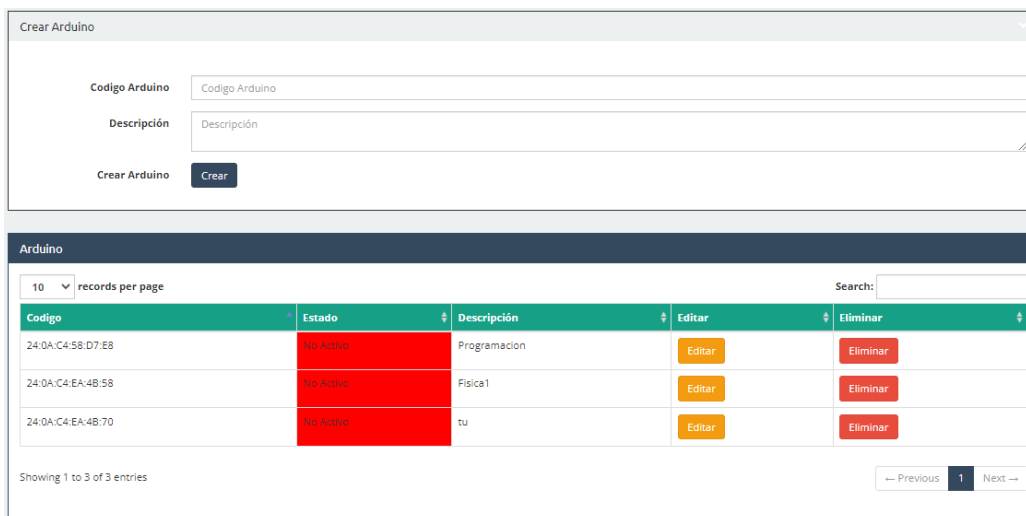


Figura 20. Arduino web
Fuente: Los Autores.

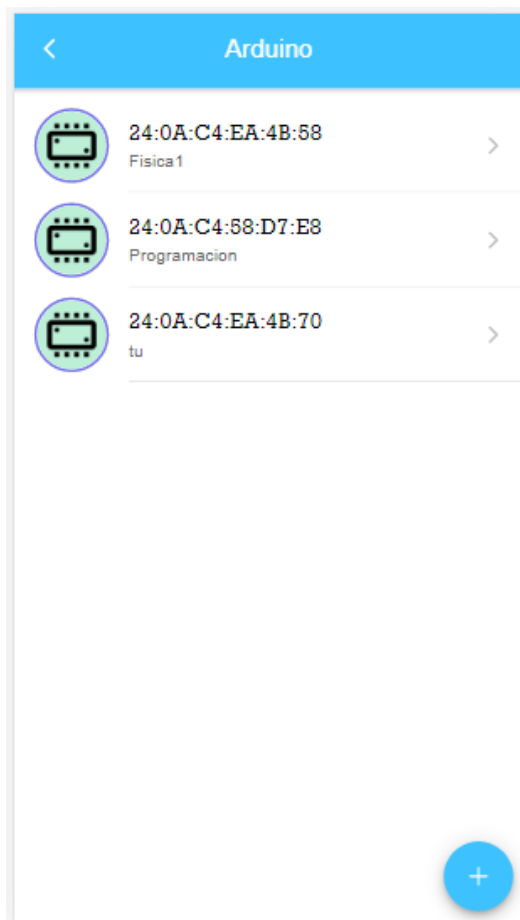


Figura 21. Arduino móvil
Fuente: Los Autores.

3.3.6. ITERACION 6: ACONDICIONADOR DE AIRES

El sistema debe ser capaz de ver la información de los aires, así como crear, editar y eliminar un aire.

The screenshot shows a web interface for managing air conditioning units. At the top, there is a form titled "Crear Aire" with two input fields: "Codigo Aire" and "Descripción". Below the form is a "Crear Aire" button. Below the form is a table titled "Aire" with the following columns: "Codigo", "Estado", "Temperatura", "Descripción", "Editar", and "Eliminar". The table contains one entry with "Codigo" 1, "Estado" "No Activo", "Temperatura" 29,55, and "Descripción" "Aire 1". The "Editar" button is yellow and the "Eliminar" button is red. Below the table, it says "Showing 1 to 1 of 1 entries" and there are "Previous" and "Next" navigation buttons.

Codigo	Estado	Temperatura	Descripción	Editar	Eliminar
1	No Activo	29,55	Aire 1	Editar	Eliminar

Figura 22. Acondicionador de aires web

Fuente: Los Autores.

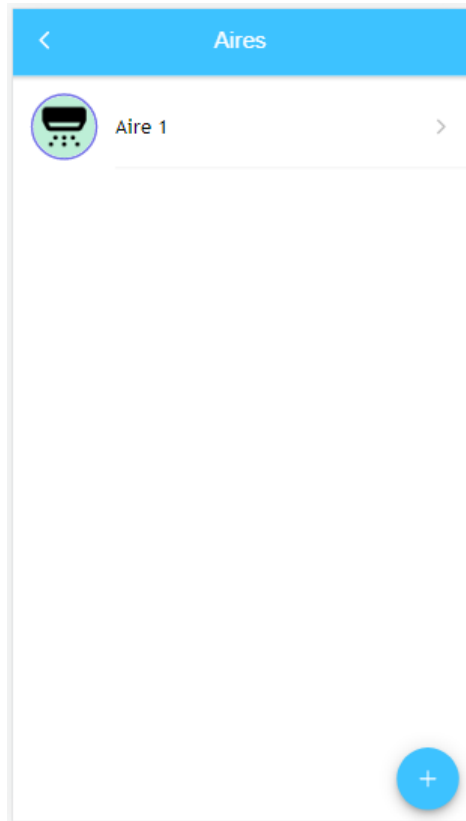


Figura 23. Acondicionador de aires móvil

Fuente: Los Autores.

3.3.7. ITERACIÓN 7: RESERVA

En esta iteración el sistema multiplataforma está compuesto por dos vistas, una para el administrador donde se podrá crear, visualizar, editar y eliminar la reserva, como se visualiza en la figura 24 y 25. Además en la vista del cliente solo se podrá visualizar la información de la reserva al cual está asignado el mismo, esta se evidencia en las figuras 26 y 27.

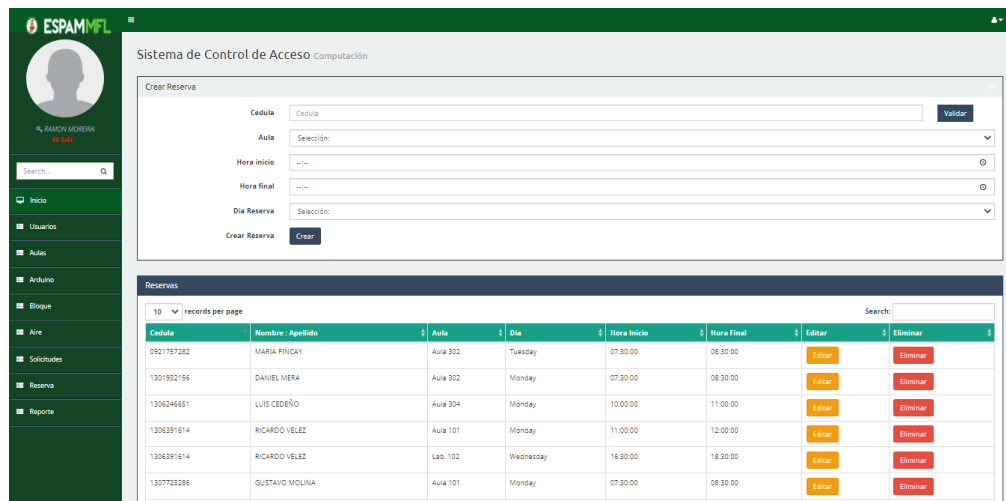


Figura 24. Reserva web administrador

Fuente: Los Autores.

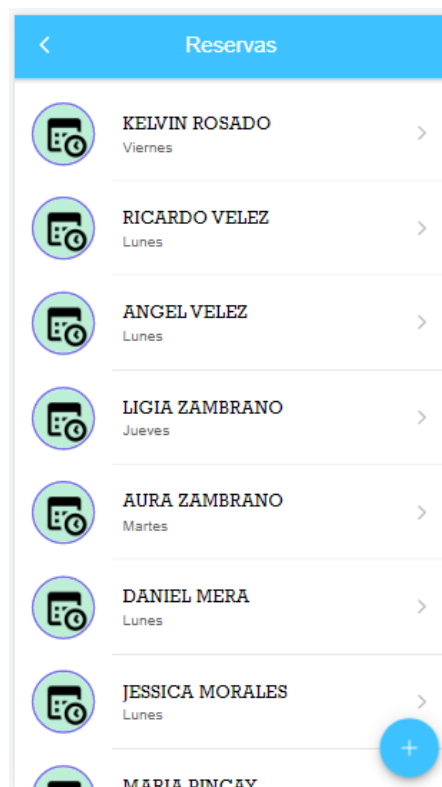


Figura 25. Reserva móvil administrador

Fuente: Los Autores.

Reservas				
Aula	Día	Hora Inicio	Hora Fin	Frecuentemente
Lab. 204	Miercoles	11:00:00	12:00:00	No
Aula 101	Lunes	12:33:00	12:33:00	No
Aula 101	Lunes	12:33:00	12:33:00	No
Aula 101	Lunes	12:33:00	12:33:00	No
Lab. 101	Jueves	19:42:00	22:45:00	No

Figura 26. Reserva web cliente
Fuente: Los Autores.

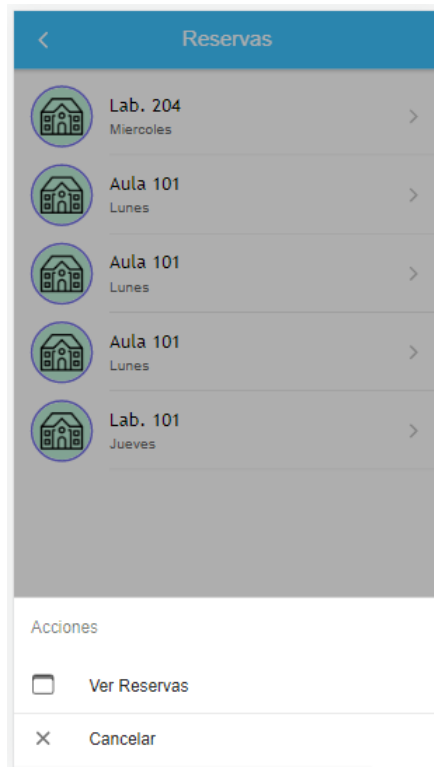


Figura 27. Reserva móvil cliente
Fuente: Los Autores.

3.3.8. ITERACION 8: SOLICITUD

En este módulo el sistema multiplataforma está compuesto por dos vistas, una para el administrador donde se podrá visualizar, aceptar o rechazar una solicitud, como se visualiza en las figuras 28 y 29. Además en la vista del cliente se podrá visualizar, crear, editar y eliminar una solicitud realizada el mismo, estas se muestran en las figuras 30 y 31.

Solicitudes							
10	records per page		Search: <input type="text"/>				
Cedula	Nombre : Apellido	Aula	Dia	Hora Inicio	Hora Final	Aceptar	Rechazar
1308542099	KELVIN ROSADO	Aula 304	Martes	16:34:00	17:34:00	Aceptar	Rechazar
1308648565	ANGEL VELEZ	Lab. 102	Viernes	00:00:00	18:38:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Martes	13:16:00	15:16:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 102	Martes	12:18:00	12:19:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 102	Miércoles	12:20:00	12:21:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Lunes	12:21:00	12:22:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Lunes	12:32:00	12:33:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Aula 101	Lunes	12:33:00	12:33:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Aula 101	Lunes	12:33:00	12:33:00	Aceptar	Rechazar

Showing 1 to 9 of 9 entries

← Previous 1 Next →

Figura 28. Solicitud web administrador

Fuente: Los Autores.









Solicitud	
<p>KELVIN ROSADO</p> <p>16:34:00 a 17:34:00</p>	<p>✖</p> <p>✔</p>
<p> RAMON MOREIRA</p> <p>De 13:16:00 a 15:16:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:18:00 a 12:19:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:20:00 a 12:21:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:21:00 a 12:22:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:32:00 a 12:33:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:33:00 a 12:33:00</p>	
<p> RAMON MOREIRA</p> <p>De 12:33:00 a 12:33:00</p>	
<p> ANGEL VELEZ</p> <p>De 00:00:00 a 18:38:00</p>	

Figura 29. Solicitud móvil administrador

Fuente: Los Autores.

Sistema de Control de Acceso Computación

Crear Solicitud

Aula: Selección: [dropdown]

Hora inicio: [time picker]

Hora final: [time picker]

Recurrencia:

Fecha Solicitud: dd/mm/aaaa [calendar icon]

Descripción: Descripción [text area]

Crear Solicitud [Create button]

Solicitud

Aula	Día	Hora Inicio	Hora Final	Descripción	Recurrencia	Editar	Eliminar
Lab. 104	Tuesday	13:16:00	15:16:00	wascasc	False	[Editar]	[Eliminar]
Lab. 102	Tuesday	12:18:00	12:19:00	acccas	False	[Editar]	[Eliminar]
Lab. 102	Wednesday	12:20:00	12:21:00	sacaca	True	[Editar]	[Eliminar]
Lab. 104	Monday	12:21:00	12:22:00	uAkKx	False	[Editar]	[Eliminar]
Lab. 104	Monday	12:32:00	12:33:00	invswsqvs	False	[Editar]	[Eliminar]
Aula 101	Monday	12:33:00	12:33:00	edowdwod	False	[Editar]	[Eliminar]

Figura 30. Solicitud web cliente
Fuente: Los Autores.

Solicitudes

- RAMON MOREIRA Martes
- RAMON MOREIRA Martes
- RAMON MOREIRA Miercoles
- RAMON MOREIRA Lunes
- RAMON MOREIRA Lunes

Acciones

- Ver Soicitud
- Editar
- Eliminar
- Cancelar

Figura 31. Solicitud móvil cliente
Fuente: Los Autores.

3.3.9. ITERACION 9: CONSUMO

En la vista web del administrador se podrá visualizar los consumos tanto por mes como por periodo académico, cabe mencionar, que se realizó una investigación de

los consumos de cada dispositivo que se encuentra en las aulas, obteniendo así un consumo por horas, los mismo que se muestran en la tabla 8 y en las figuras 32 y 33.

Tabla 8. Consumo estimado de dispositivos electrónicos

Nombre	Consumo Estimado(kw/h)	Link
Lámparas Led o de tubos	0,576 kWh	https://procoen.com/cuanta-energia-consume-una-lampara/
Proyector Epson	0.498 kWh	https://www.eii.uva.es/servicios/sinfor/apoyoDocencia/proyectors/Epson_EMP-400W.pdf
Aire acondicionado	1,459 kWh	https://www.panasonic.com/co/support/discontinued/aires-acondicionados/cs-s12tkv.specs.html
Ordenadores(Todo en uno)	5.4 kWh	https://chcenergia.es/blog/cuanto-consume-un-ordenador-o-pc/
Router Tp-Link	0,02 kWh	https://tarify.es/noticias/cuanta-electricidad-consume-router

Fuente: Los Autores.

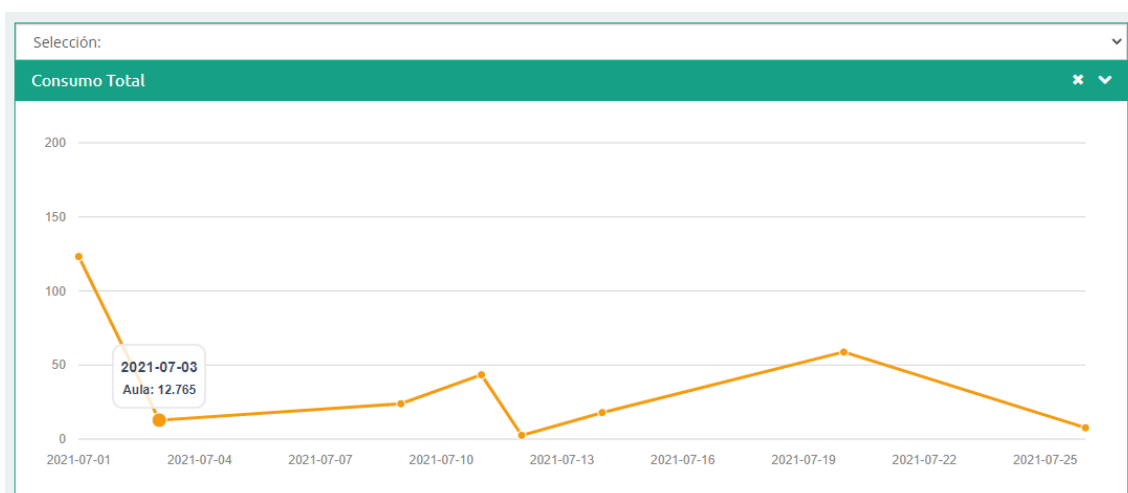


Figura 32. Consumo por mes

Fuente: Los Autores.

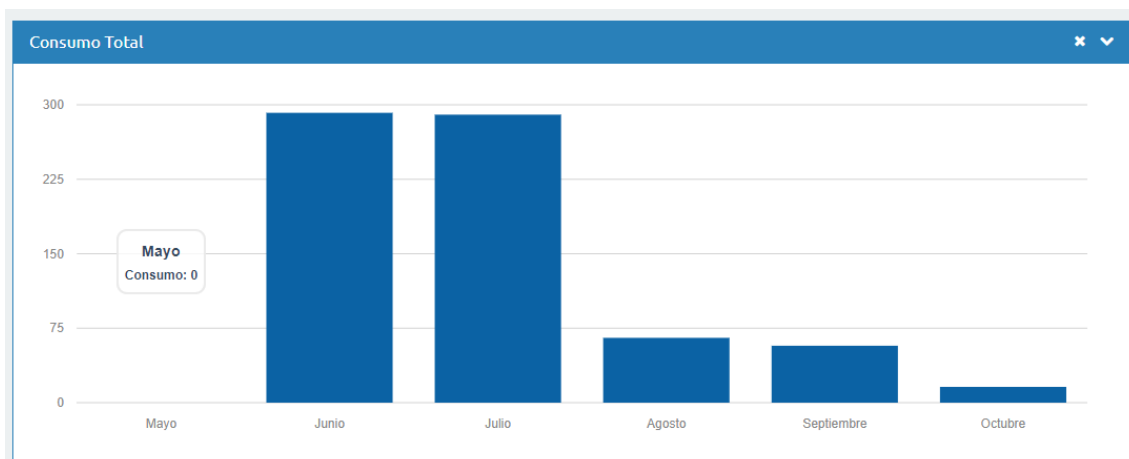


Figura 33. Consumo total por periodo académico

Fuente: Los Autores.

3.3.10. ITERACION 10: REPORTE

El administrador podrá generar un reporte del consumo energético.

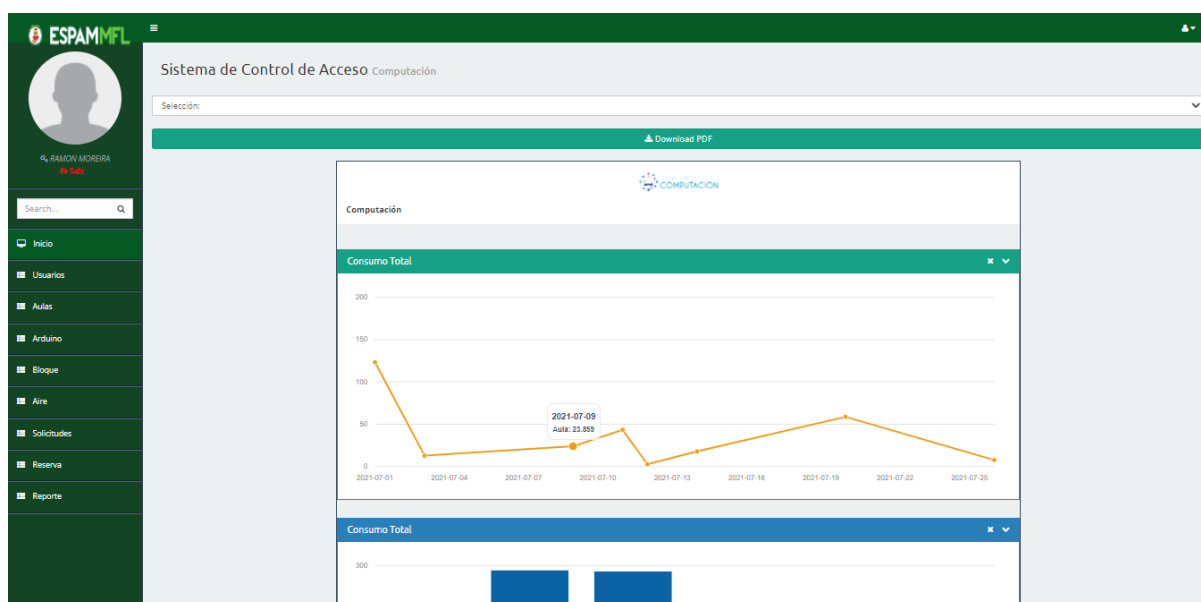


Figura 34. Reporte

Fuente: Los Autores.

3.3.11. ITERACION 11: DASHBOARD

En esta iteración los diferentes usuarios podrán visualizar e interactuar con las funcionalidades del sistema de una manera más rápida, como se muestra en las figuras 35 y 36

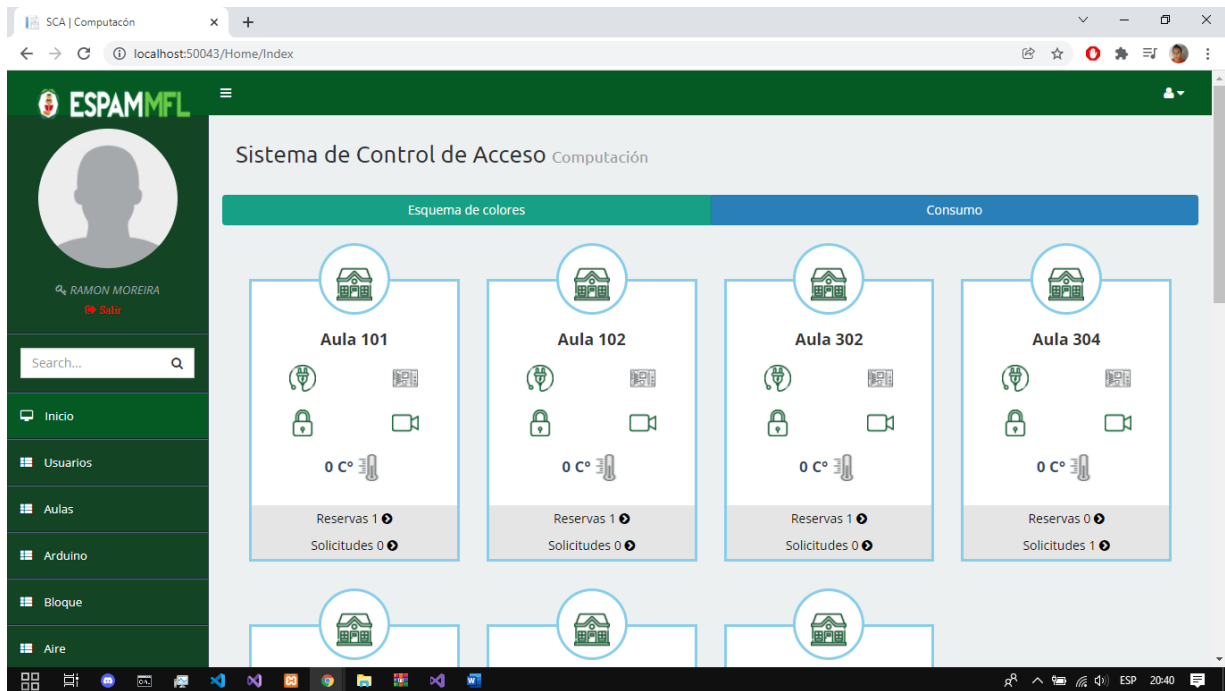


Figura 35. Dashboard web administrador y cliente
Fuente: Los Autores.

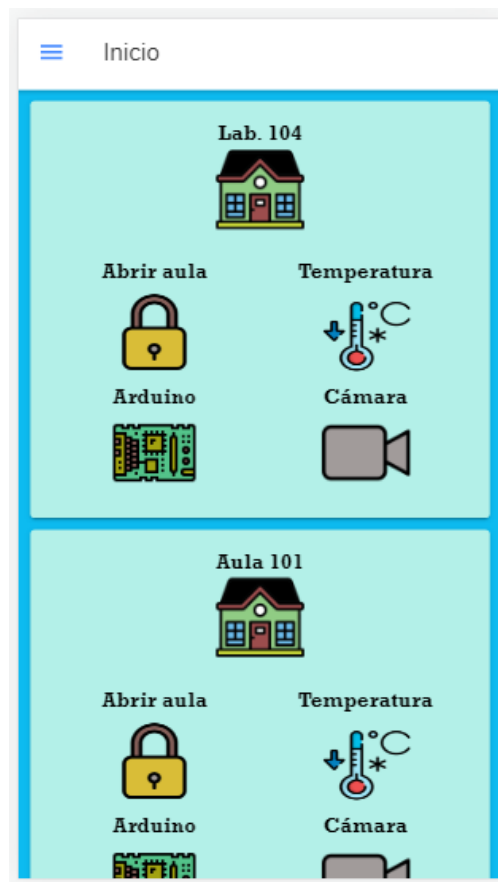
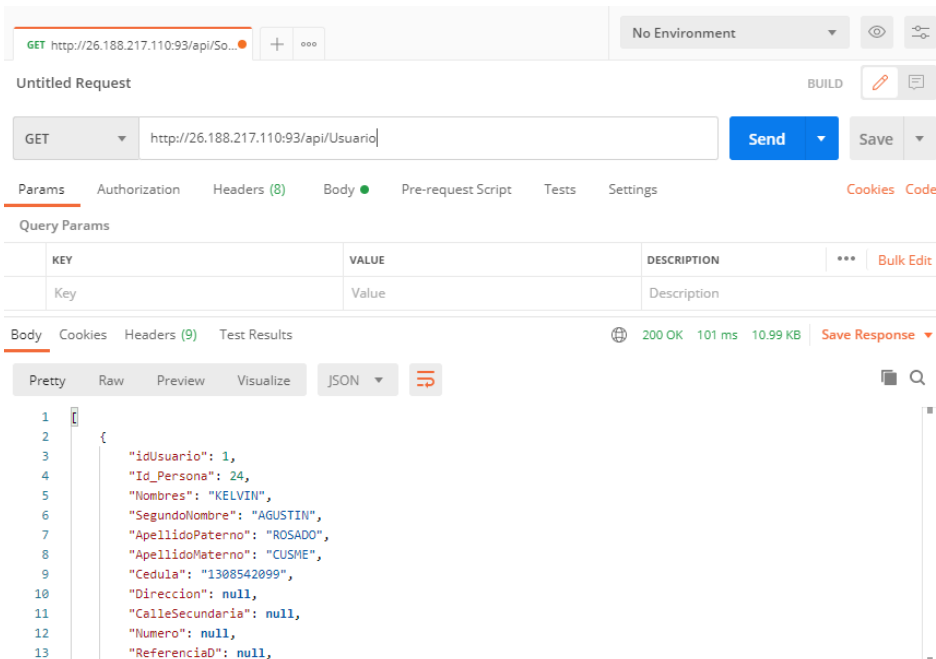


Figura 36. Dashboard móvil administrador y cliente
Fuente: Los Autores.

3.4. PRUEBAS

Para las pruebas del micro servicio APIS se utilizó la herramienta de testeo Postman, ya que este permite realizar peticiones Http. Además, se realizaron las pruebas de comunicación e interacción entre los prototipos que se encuentran implementados en el laboratorio 204 y 205 con el sistema multiplataforma, en las cuales se muestra de manera correctamente la apertura de dichos laboratorios mediante una identificación tag y a través del sistema web y móvil. En las figuras 37, 38 y 39 se evidencia algunas de las pruebas realizadas a las APIS, también, en las figuras 40, 41, 42 y 43 se visualizan las pruebas de interacción del sistema con el prototipo hardware.



The screenshot displays the Postman interface for an API test. At the top, the request method is set to GET and the URL is `http://26.188.217.110:93/api/Usuario`. The response status is 200 OK, with a response time of 101 ms and a size of 10.99 KB. The response body is shown in JSON format, containing user details.

KEY	VALUE	DESCRIPTION
Key	Value	Description

```
1 {
2   "idUsuario": 1,
3   "Id_Persona": 24,
4   "Nombres": "KELVIN",
5   "SegundoNombre": "AGUSTIN",
6   "ApellidoPaterno": "ROSADO",
7   "ApellidoMaterno": "CUSME",
8   "Cedula": "1308542099",
9   "Direccion": null,
10  "CalleSecundaria": null,
11  "Numero": null,
12  "ReferenciaD": null,
```

Figura 37. Prueba de tiempo de respuesta a la API Usuario con la herramienta Postman

Fuente: Los Autores.

GET http://26.188.217.110:93/api/Solicitud

Send Save

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies Code

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (9) Test Results 200 OK 258 ms 3.65 KB Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "idSolicitud": 16,
3   "ConfiguracionReserva": {
4     "idConfiguracionReserva": 214,
5     "FechaReserva": "Tuesday",
6     "HoraInicio": "16:34:00",
7     "HoraFin": "17:34:00",
8     "idUsuario": 0,
9     "Usuario": "1308542099",
10    "Nombres": "KELVIN",
11    "Apellido": "ROSADO",
12    "ApellidoM": null,
13  }

```

Figura 38. Prueba de tiempo de respuesta a la API Solicitud con la herramienta Postman

Fuente: Los Autores.

GET http://26.188.217.110:93/api/Reserva

Send Save

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies Code

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (9) Test Results 200 OK 233 ms 10.86 KB Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "idReserva": 1,
3   "ConfiguracionReserva": {
4     "idConfiguracionReserva": 1,
5     "FechaReserva": "Friday",
6     "HoraInicio": "10:00:00",
7     "HoraFin": "11:00:00",
8     "idUsuario": 0,
9     "Usuario": "1308542099",
10    "Nombres": "KELVIN",
11    "Apellido": "ROSADO",
12    "ApellidoM": null,
13  }

```

Figura 39. Prueba de tiempo de respuesta a la API Reserva con la herramienta Postman

Fuente: Los Autores.

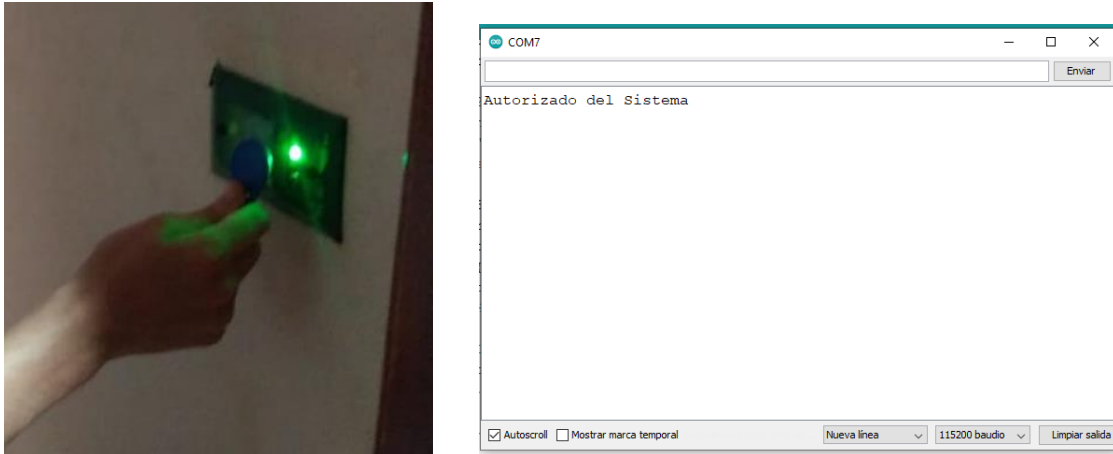


Figura 40. Prueba prototipo interactuando con el sistema mediante una identificación (Tag) registrado

Fuente: Los Autores.

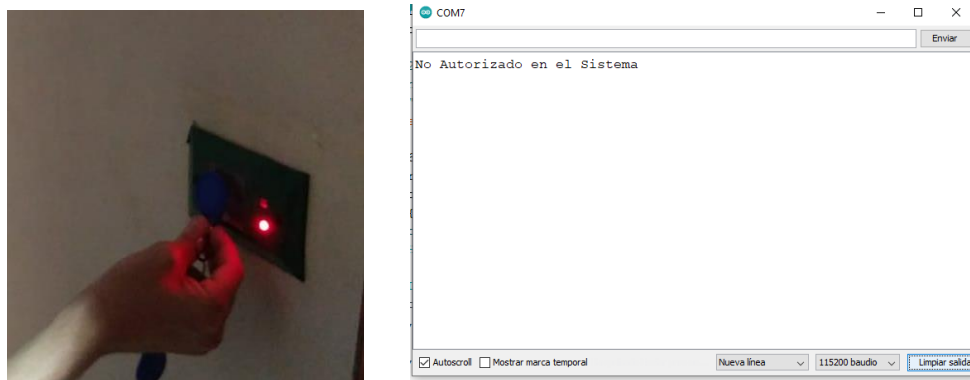
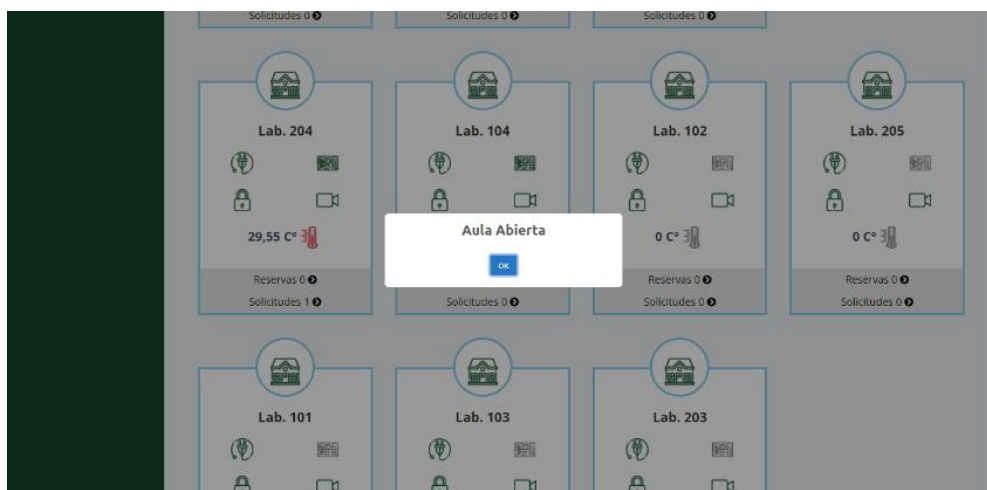


Figura 41. Prueba prototipo interactuando con el sistema mediante una identificación (Tag) no registrado

Fuente: Los Autores.



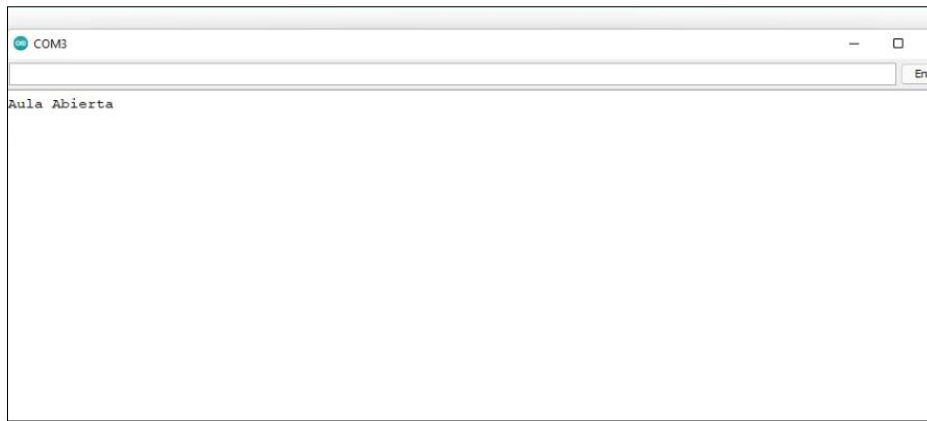


Figura 42. Prueba de apertura del laboratorio mediante el sistema Web

Fuente: Los Autores.

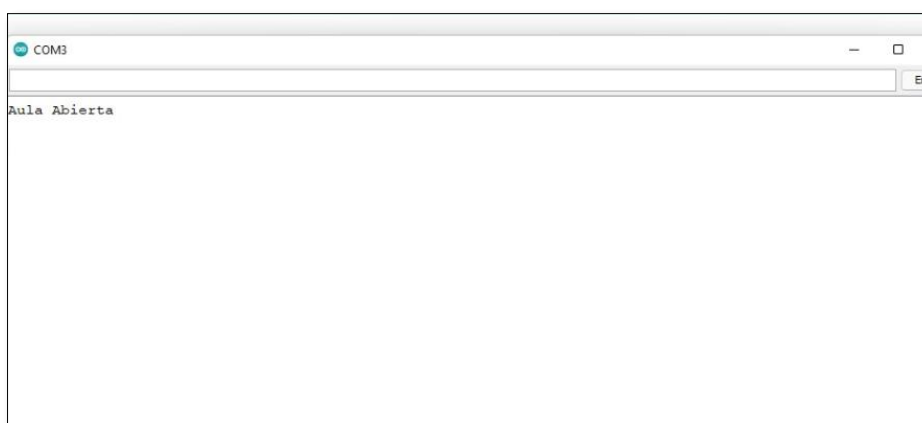
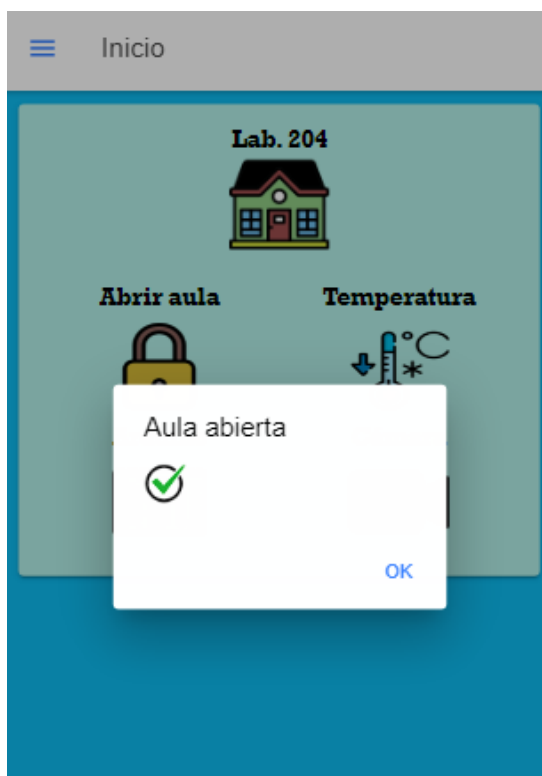


Figura 43. Prueba de apertura del laboratorio mediante la app móvil

Fuente: Los Autores.

Además, para la verificación del correcto funcionamiento del sistema multiplataforma los autores realizaron las pruebas de caja negra en las cuales se testearon cada uno de los módulos desarrollados con el fin de diagnosticar posibles errores y así poder corregirlos.

Tabla 9. Pruebas de caja negra del sistema multiplataforma.

Usuarios	Módulo	Datos de entrada	Salida	Estado
		El usuario ingresará su número de cédula	Si las credenciales de ingreso son	

Administrador, cliente	Login	y contraseña para verificar si está registrado para así poder escoger el rol pertinente.	incorrectas el sistema arroja una alerta de error, caso contrario ingresará sin inconvenientes al sistema (Figuras 40, 41, 42).	Correcto
Administrador	Arduino	El administrador ingresa los datos como el código y la descripción para poder registrar un Arduino, adicionalmente este podrá visualizar, editar y eliminar un Arduino.	Al momento de registrar, editar o eliminar un Arduino el sistema multiplataforma arrojará una alerta de exitoso o error. (Figuras 45, 46)	Correcto
Administrador	Aire	El administrador ingresa los datos como el código y la descripción para poder registrar un aire, adicionalmente este podrá visualizar, editar y eliminar un aire.	Al momento de registrar, editar o eliminar un aire el sistema multiplataforma arrojará una alerta de exitoso o error.	Correcto
Administrador	Usuario	El administrador podrá visualizar los usuarios que han ingresado al sistema, adicionalmente también podrá añadir y eliminar un tag a los mismos.	Al momento de añadir o eliminar un tag el sistema multiplataforma arroja una alerta de exitoso o error.	Correcto
Administrador	Bloque	El administrador ingresa los datos como el código y la descripción para poder registrar un bloque, adicionalmente este podrá visualizar y editar el mismo.	Al momento de registrar o editar un bloque el sistema multiplataforma arroja una alerta de exitoso o error.	Correcto
		El administrador ingresa los datos como el código de aula, tipo de aula, el bloque al que va a	Al momento de añadir, editar, eliminar o abrir un	

Administrador	Aula	pertenecer, el Arduino, el aire, la ip de la cámara y la descripción para poder registrar un aula, adicionalmente este podrá visualizar, editar, eliminar, acceder a la cámara de esa aula y abrir la misma.	aula el sistema multiplataforma arroja una alerta de exitoso o error (Figuras 43, 44).	Correcto
Administrador	Reserva	El administrador ingresa los datos como el usuario, hora de inicio, hora de fin, fecha de reserva, etc. Adicionalmente este podrá visualizar y editar y eliminar la misma.	Al momento de añadir, editar, eliminar o abrir un aula el sistema multiplataforma arroja una alerta de exitoso o error.	Correcto
Administrador	Solicitud	El administrador podrá visualizar las solicitudes que han realizado los clientes, adicionalmente este podrá aceptar o rechazar dicha solicitud.	Al momento de aceptar o rechazar una solicitud el sistema multiplataforma arroja una alerta. Además, dichas solicitudes se convertirán en reservas (Figura 47).	Correcto
Administrador	Consumo	El administrador podrá seleccionar cualquier mes del periodo actual en que se encuentre para así poder desplegar dicho consumo, cabe recalcar que por defecto se muestra el mes actual.	Al momento de seleccionar un aula el administrador visualiza el consumo energético de la misma a través de un gráfico estadístico, además el consumo total del edificio.	Correcto
Administrador	Reporte	El administrador podrá seleccionar cualquier mes del periodo actual para poder generar el reporte.	El reporte generado muestra si se han hechos uso de las reservas y las solicitudes rechazadas.	Correcto
		El cliente puede abrir el aula dependiendo de su horario de	Al momento de abrir un aula el sistema	

Cliente	Aula	clases, además puede visualizar la cámara de dicha aula y la temperatura del aire de la misma.	muestra una alerta de exitoso o de error.	Correcto
Cliente	Solicitud	El cliente puede crear una solicitud, editar o eliminar la misma.	Al momento de crear, editar o eliminar una solicitud el sistema muestra una alerta de exitoso o de error.	Correcto
Cliente	Reservas		El cliente puede visualizar las reservas que están asignadas a su horario.	Correcto

Fuente: Los Autores.

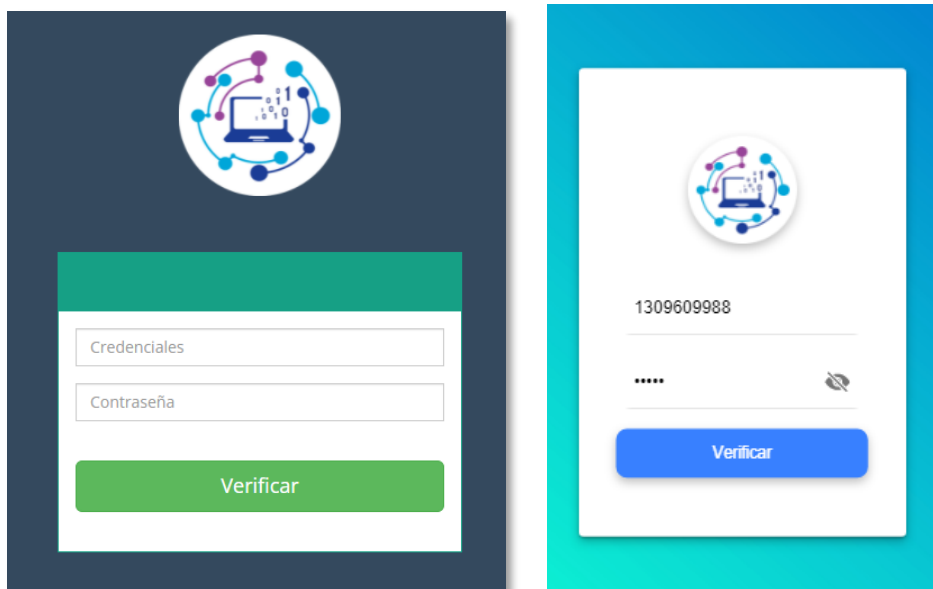


Figura 44. Ingreso de credenciales sistema Web y app móvil

Fuente: Los Autores.

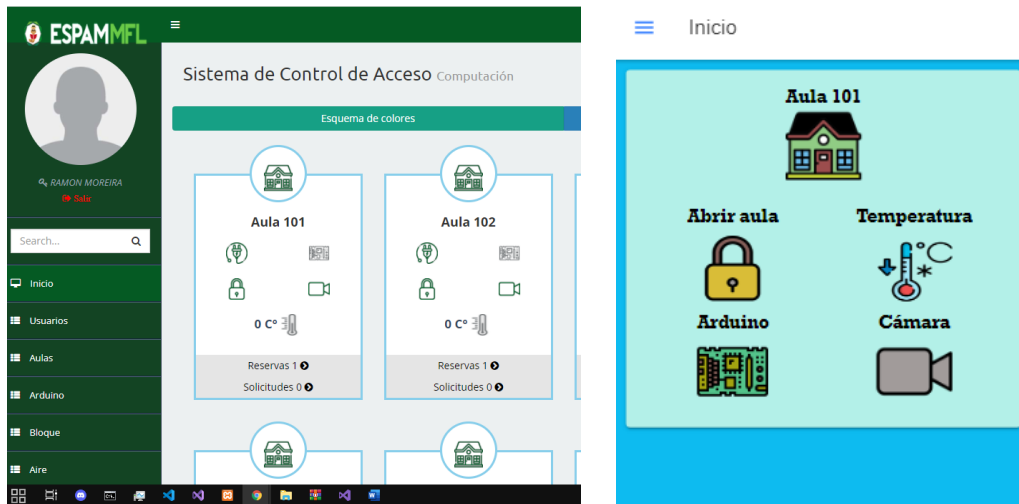


Figura 45. Ingreso con credenciales correctas al sistema web y móvil.

Fuente: Los Autores.

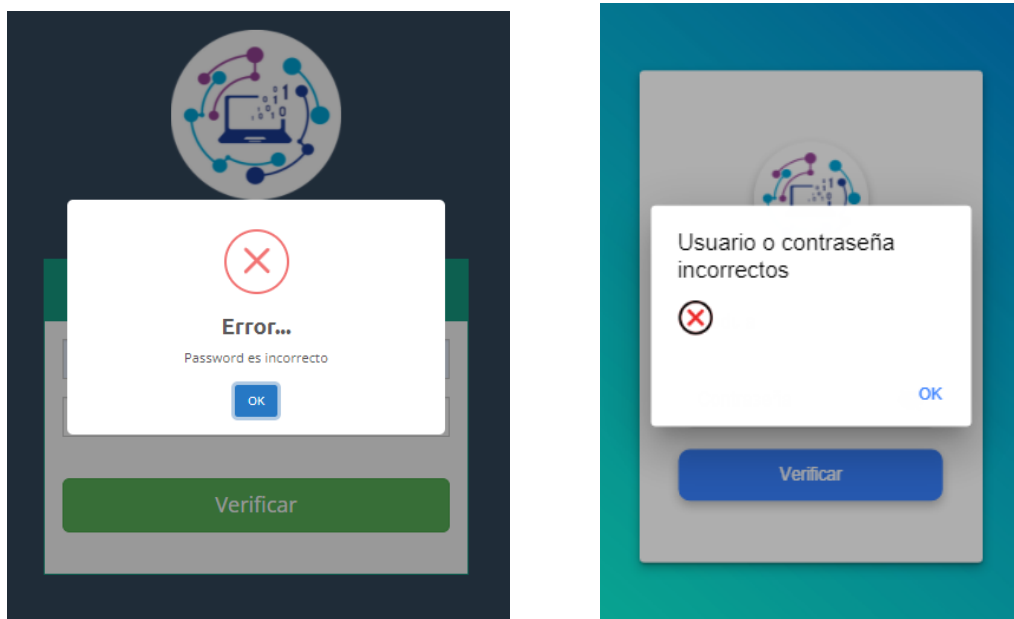
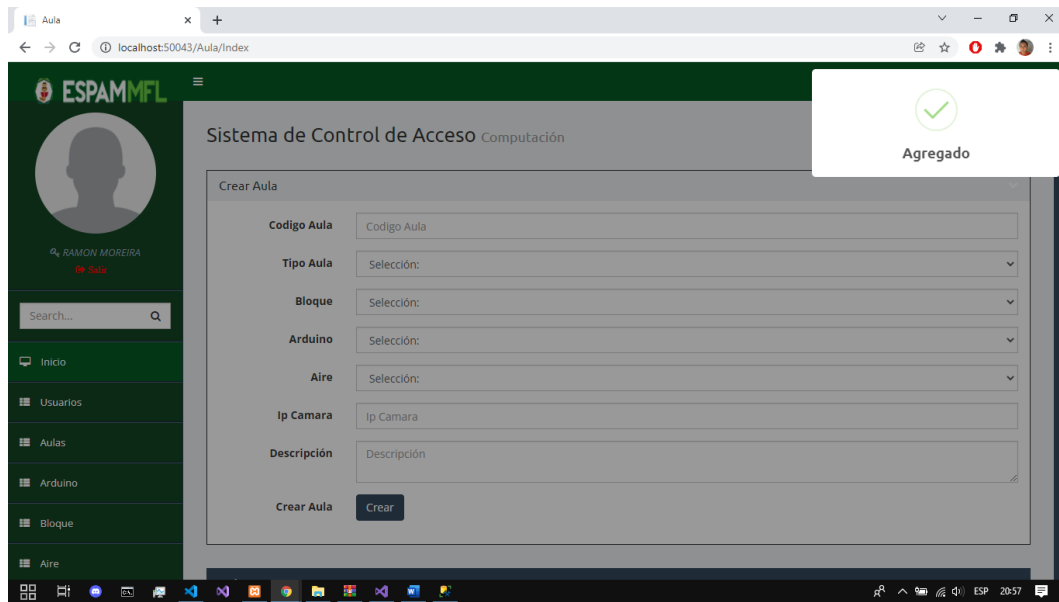


Figura 46. Ingreso con credenciales incorrectas al sistema web y móvil.

Fuente: Los Autores.



Aula 302	Aula	3	185.10.80.33:8082	Activo	Editar	Eliminar		
rh-aula-5	Física cuántica	Laboratorio	2	192.168.0.4	Selección:	Activo	Editar	Eliminar

Showing 1 to 10 of 14 entries

← Previous 1 2 Next →

Figura 47. Alerta aula creada correctamente sistema Web

Fuente: Los Autores.

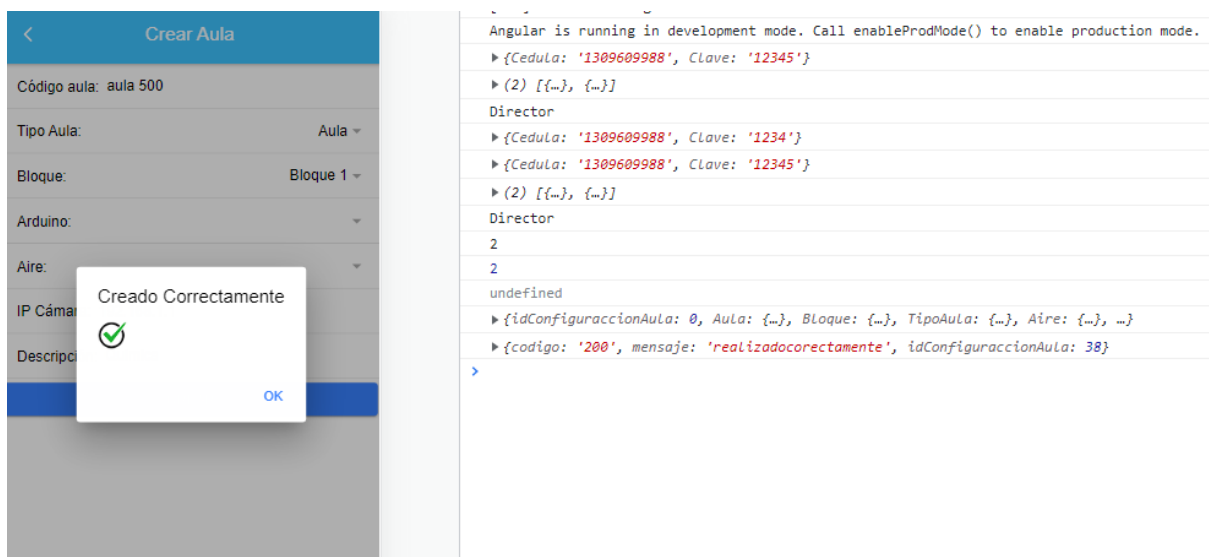


Figura 48. Alerta aula creada correctamente app móvil

Fuente: Los Autores.

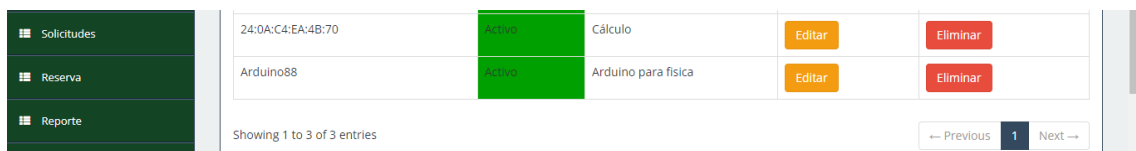
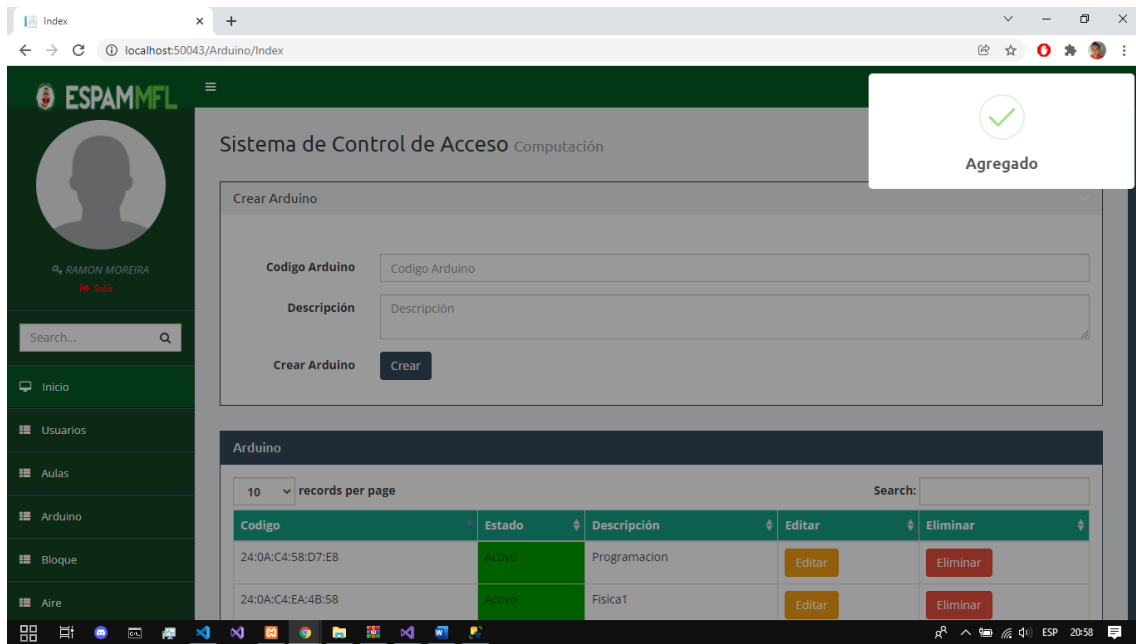


Figura 49. Alerta Arduino creado correctamente sistema Web

Fuente: Los Autores.

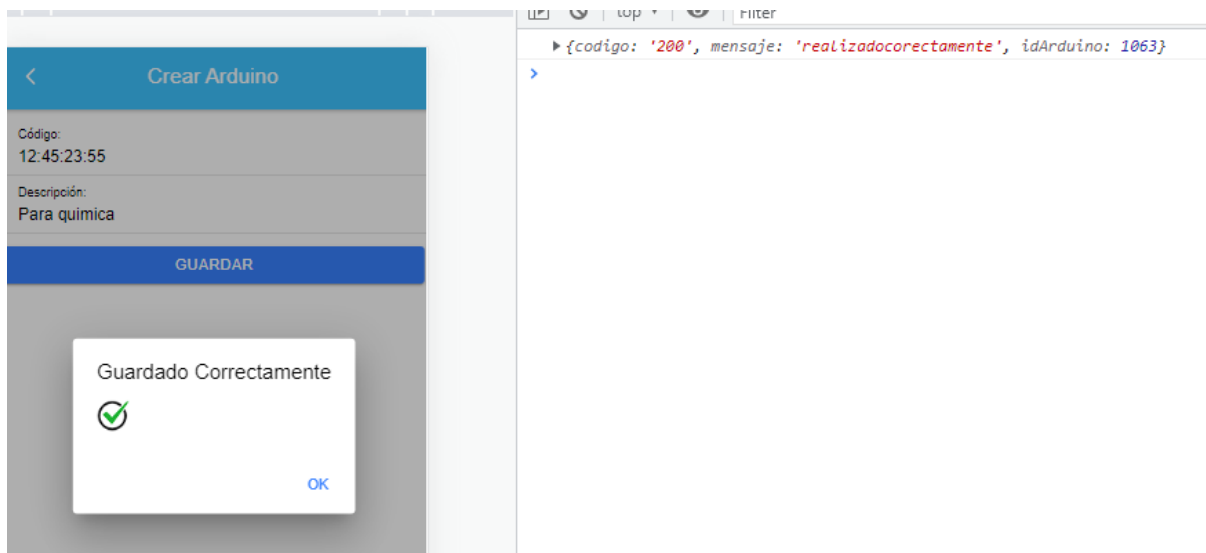


Figura 50. Alerta Arduino creado correctamente app móvil

Fuente: Los Autores.

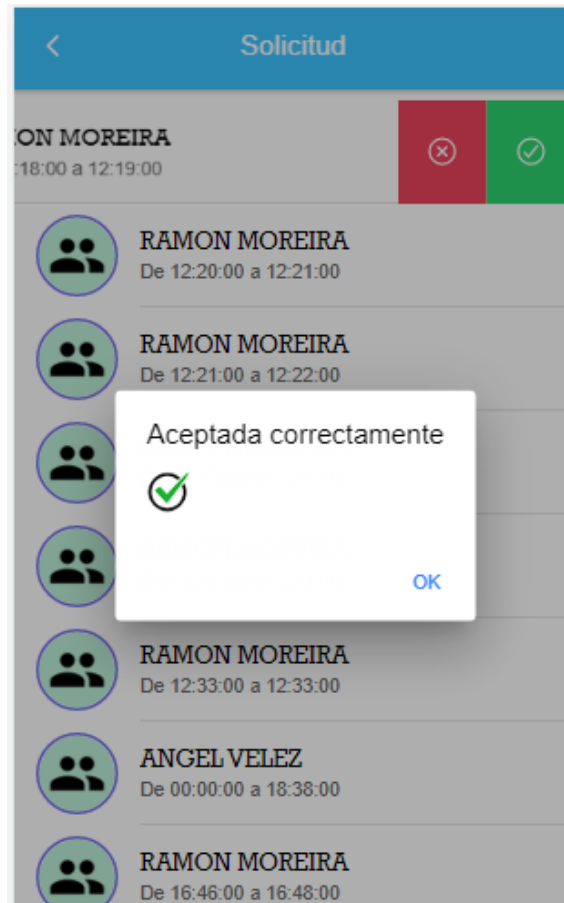


Figura 51. Alerta solicitud aceptada correctamente app móvil

Fuente: Los Autores.

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Las entrevistas realizadas al encargado de la UDIV, permitieron recopilar los requisitos de funcionamiento para el desarrollo del sistema multiplataforma, de esta manera se elaboró un documento de especificación de software (ERS) que ayudó a tener un proyecto estructurado.
- Se diseñó la arquitectura para llevar a cabo la comunicación entre el sistema multiplataforma y el prototipo hardware, la misma que permitió tener una eficacia en el proceso de envío y recepción de datos.
- El uso de la metodología Extreme Programming (XP) fue viable en el desarrollo del sistema multiplataforma, la cual permitió establecer una mejor comunicación entre el equipo de desarrollo y el cliente. Así mismo, viabilizar la reutilización del código, ya que con sus 4 fases ayudó a tener una programación ordenada que favoreció con el cumplimiento de los requisitos.
- Las pruebas de caja negra realizadas permitieron verificar el cumplimiento de los requisitos de funcionamiento establecidos por el cliente.

4.2. RECOMENDACIONES

Se recomienda:

- Utilizar un documento de especificación de software (ERS) estructurado con el estándar IEEE Recommended Practice for Software Requirements Specification ANSI/IEEE 830 1998, ya que este permite tener toda la información de una manera organizada. Además, sirve como medio de consulta técnica para tener una mejor comunicación entre el cliente y los desarrolladores.
- Diseñar una buena arquitectura de comunicación que permita visualizar y entender de mejor manera como el sistema multiplataforma se va a comunicar e interactuar con la base de datos, otros sistemas y el prototipo hardware.
- Emplear la metodología Extreme Programming (XP) en el desarrollo de software, ya que el proceso metodológico que brinda cada una de sus fases permite una mejor adaptabilidad entre el cliente y el equipo de desarrollo alcanzando así los resultados esperados.
- Realizar pruebas y corrección de errores necesarios para garantizar un producto software funcional y de calidad.

BIBLIOGRAFÍA

- [1] ESPAM MFL (Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López), *Modelo Educativo*. Calceta, 2016.
- [2] E. Gómez and J. Zambrano, “sistema de control eléctrico remoto en la galería histórica de la carrera de computación de la ESPAM-MFL,” ESPAM MFL, 2018.
- [3] J. D. Patiño Preciado, “Historia del edificio de Computación,” 2020.
- [4] ESPAM MFL (Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López), “Misión y Visión,” 2020, [Online]. Available: <http://www.espam.edu.ec/web/oferta/grado/computacion.aspx>.
- [5] J. Stiven, “Diseño energético, eléctrico y de comunicaciones para un edificio inteligente residencial: caso de estudio,” Universidad de la salle, 2018.
- [6] B. A. Molina Aquino, M. Á. Quiroz Vera, C. E. Cedeño Macías, and G. G. Molina Garzón, “MONITORING AND ACCESS CONTROL IN A COMPANY LOCATED IN PORTOVIEJO -ECUADOR,” *IUSH*, no. 27, pp. 49–54, 2016, [Online]. Available: <https://revistas.proeditio.com/iush/quid/article/view/1217>.
- [7] J. Bernal Bernabe, J. L. Hernandez Ramos, and A. F. Skarmeta Gomez, “TACIoT: multidimensional trust-aware access control system for the Internet of Things,” *Soft Comput.*, vol. 20, no. 5, pp. 1763–1779, 2016, doi: 10.1007/s00500-015-1705-6.
- [8] A. Agape and M. Postolache, “Internet-enabled access control system using a mobile application,” *IEEE*, pp. 244–249, 2018, doi: 10.1109/ICSTCC.2018.8540687.
- [9] M. Rodríguez Diego, “Gestión domótica de una casa unifamiliar basada en Arduino,” Universidad de Valladolid, 2017.
- [10] R. de la Rosa, D. Munoz Morales, E. Pérez Mendoza, and H. Mora Flores, “Automatización para el control de acceso utilizando dispositivos móviles y

- RFID,” *CITEC*, vol. 39, no. 04, pp. 205–216, 2017, [Online]. Available: <http://www.itcelaya.edu.mx/ojs/index.php/pistas/article/download/1056/886>.
- [11] V. Susukailo and Y. Lakh, “Access control system based on encryption in QR-Code technology,” *IEEE*, pp. 158–161, 2018, doi: 10.1109/IDAACS-SWS.2018.8525779.
- [12] T. Y. Teh, Y. S. Lee, Z. Y. Cheah, and J. J. Chin, “IBI-Mobile Authentication: A Prototype to Facilitate Access Control Using Identity-Based Identification on Mobile Smart Devices,” *Wirel. Pers. Commun.*, vol. 94, no. 1, pp. 127–144, 2017, doi: 10.1007/s11277-016-3320-y.
- [13] E. A. Parada Prieto, M. J. Illera Bustos, S. B. Sepúlveda Mora, D. Guevara Ibarra, and B. Medina Delgado, “Sistema de control domótico de bajo costo: un respaldo a la generación ecológica de energía eléctrica en Colombia,” *Rev. Tecnura*, vol. 20, no. 49, p. 120, 2016, doi: 10.14483/udistrital.jour.tecnura.2016.3.a08.
- [14] O. M. Cornelio, B. B. Fonseca, and P. M. Puig Díaz, “Sistema para el control de acceso centralizado en los laboratorios productivos de la Universidad de las Ciencias Informáticas System for centralized access control in production laboratories of the University of Information Science,” *ResearchGate*, vol. 9, no. 7, pp. 2306–2495, 2016, [Online]. Available: <http://publicaciones.uci.cu>.
- [15] J. Rico Moreno, M. Arroyo Almaguer, M. Rodríguez Vargas, and F. Nito Frías, “App prototipo gestor de identificación de alumnos a través de códigos QR,” *Sist. Comput. y TIC ’s*, vol. 2, no. 3, pp. 9–14, 2016, [Online]. Available: http://www.ecorfan.org/spain/researchjournals/Sistemas_Computacionales_y_TICs/vol2num3/Revista_de_Sistemas_Computacionales_y_TIC%60S_V2_N3_2.pdf.
- [16] T. De La Rosa Martín and G. M. García Rubio, “Sistema para la Gestión y Control de marcaciones del personal de la Universidad Metropolitana del Ecuador,” *Rev. Metrop. Ciencias Apl.*, vol. 2, no. 2, pp. 171–181, 2019,

[Online]. Available:
<http://remca.umet.edu.ec/index.php/REMCA/article/view/149/226>.

- [17] E. Fernandez Guevara and H. Q. Flores Chura, "Implementación de la aplicación 'Control Residente', bajo el marco de trabajo SCRUM y la metodología de desarrollo ágil Extreme Programming, para mejorar el control de servicios brindados a los estudiantes residentes de la Universidad Peruana Unioón, Fili," Universidad Peruana Unión Facultad De Ingeniería Y Arquitectura, 2019.
- [18] V. del R. Angamarca, "Implementación de un sistema de gestión para los procesos de capacitación y educación cristiana de la parroquia santuario 'Nuestra señora del cisne de la provincia de Imbabura,'" Universidad Técnica del Norte, 2018.
- [19] P. Chimba Pilatasig and M. Simba Guamushig, "Implementación de un sistema de control y administración de procesos aplicado la metodología Extreme programming para el colegio catorce de julio ubicado en la parroquia San Buenaventura del cantón Latacunga, provicia de Cotopaxi, durante el año 2013," Universidad Técnica De Cotopaxi, 2018.
- [20] A. Malik, A. Burney, and F. Ahmed, "A Comparative Study of Unstructured Data with SQL and NO-SQL Database Management Systems," *Sci. Res. an Acad. Publ.*, vol. 8, no. 4, 2020, [Online]. Available: <https://www.scirp.org/journal/paperinformation.aspx?paperid=99539>.
- [21] A. P. Dorantes, M. L. Sánchez, and M. H. Salinas, "Sistema de Recomendación sobre el Consumo Nutrimental basado en una Arquitectura Cliente-Servidor Utilizando un Dispositivo Android," *C/SCI*, pp. 90–95, 2017.
- [22] A. C. Aranda, "Modelo-Vista-Controlador. Lenguaje Uml," 2019.
- [23] W. M. García Pérez, "Metodología en el desarrollo de software," 2015.
- [24] A. Arsaute, F. Zorzan, M. Daniele, A. González, and M. Frutos, "Generación automática de API REST a partir de API Java, basada en transformación de Modelos (MDD)," *XX Work. Investig. en Ciencias la Comput.*, no. Mdd, pp. 629–633, 2018, [Online]. Available:

<http://sedici.unlp.edu.ar/handle/10915/67777>.

- [25] H. J. Oria-aguilera, Y. Valeriano, S. Luis, and M. Omar, "Pruebas a los módulos en código C de un modelo dinámico de 6 GDL para un catamarán," *Conv. Científica Ing. y Arqut.*, pp. 0–11, 2016.
- [26] C. Rincón González, F. N. Díaz Piraquive, and H. Castro Silva, "STAKEHOLDERS IMPACT ON THE PERFORMANCE OF PROJECTS IN THE COLOMBIAN MILITARY FORCES," *Int. Congr. Proj. Manag. Eng.*, pp. 71–83, 2019, [Online]. Available: <http://dspace.aepro.com/xmlui/handle/123456789/2253>.

ANEXOS

ANEXO 1. ENTREVISTA REALIZADA AL ENCARGADO DEL LA UDIV

Anexo 1-A. Formato de la entrevista



ENTREVISTA DIRIGIDA A: Mgs: Javier López Zambrano; persona encargada del Área del UDIV (Unidad de Docencia de Investigación y Vinculación) de la carrera de Computación de la ESPAM MFL.

OBJETIVO DE LA ENTREVISTA: Recopilar información para los requerimientos necesarios para determinar el alcance del “**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**”.

1. ¿Cuál es el problema a resolver?
2. ¿Cómo se lleva a cabo el proceso de ingreso a las aulas actualmente?
3. ¿Qué funcionalidades tendrá la aplicación móvil?
4. ¿Cuáles son los reportes que le gustaría que el sistema multiplataforma contemple?
5. ¿Desde su punto de vista cual sería el lenguaje de programación adecuado para desarrollar el sistema multiplataforma?
6. ¿Qué motor de base de datos emplea el sistema actual de la Universidad?
7. ¿Quién utilizará el sistema?
8. ¿Qué información va hacer procesada?
9. ¿A quiénes beneficiará el sistema multiplataforma?
10. Para el correcto funcionamiento del sistema
¿Cuál es la sugerencia que usted podría realizar a los autores?

Anexo 1-B. Código QR del video de sesión virtual y fotografía de sesión presencial para la toma de requerimientos



ANEXO 2. ERS

ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS,
LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE
COMPUTACIÓN**

DESARROLLADORES:

Luis Eduardo Chichanda García

Oliver Antonio Vera Salazar

HISTORIAL DE REVISIONES

Fecha	Revisión	Descripción	Autor
04/12/2020	1.0	"Requerimientos del software"	Luis Chichanda – Oliver Vera

Documento validado por las partes en fecha: 07/12/2020

Por el cliente	Por la empresa suministradora
Fdo. Ing: López Zambrano Javier Hernán	Fdo. Chichanda García Luis Eduardo

CONTENIDO

Historial de Revisiones	¡Error! Marcador no definido.
Contenido	58
1 Introducción	¡Error! Marcador no definido.
1.1 Propósito	60
1.2 Alcance	60
1.3 Personal involucrado	60
1.4 Definiciones, acrónimos y abreviaturas	¡Error! Marcador no definido.
1.5 Referencias	61
1.6 Resumen	61
2 Descripción general	61
2.1 Perspectiva del producto	61
2.2 Funcionalidad del producto	62
2.3 Características de los usuarios	67
2.4 Restricciones	67
3 Requisitos específicos	67
3.1 Requisitos comunes de los interfaces	67
3.1.1 Interfaces de usuario	67
3.1.2 Interfaces de hardware	68
3.1.3 Interfaces de software	68
3.1.4 Interfaces de comunicación	68
3.2 Requisitos funcionales	68
3.2.1 Gestión de aulas	69
Requisito funcional 1 “Crear aula”	69
Requisito funcional 2 “Dar de baja un aula”	69
Requisito funcional 3 “Asignación de docentes a aulas”	69
Requisito funcional 4 “Visualización del aula”	70

	59
3.2.2 Gestión de Usuario	70
Requisito funcional 5 “Inicio de sesión”	71
Requisito funcional 6 “Solicitud”	71
3.2.3 Gestión de bloque	71
Requisito función 7 “Crear bloque”	72
3.2.4 Gestión de iluminarias y acondicionador de aires.	72
Requisito función 8 “Encendido y apagado de luminarias y acondicionador de aires”	74
3.2.5 Gestión de consumo.	74
Requisito función 9 “Consumo energético”	75
3.2.6 Gestión de Arduino	75
Requisito función 10 “Crear Arduino”	75
3.2.7 Estadísticas	¡Error! Marcador no definido.
Requisito función 11 “Reportes”	76
3.2.8 Evento	¡Error! Marcador no definido.
Requisito funcional 12 “Luces y aire acondicionado”	¡Error! Marcador no definido.
3.3 Requisitos no funcionales	76
3.3.1 Requisitos de rendimiento	76
3.3.2 Seguridad	77
3.3.3 Interfaz amigable	77
3.3.4 Disponibilidad	77
3.3.5 Mantenibilidad	77

INTRODUCCIÓN

El presente documento presentará de forma organizada los requisitos que son indispensables para desarrollar el sistema multiplataforma de control de acceso a las aulas de la carrera de Computación. Este documento está estructurado en base al estándar IEEE Recommended Practice for Software Requirements Specification ANSI/IEEE 830 1998.

PROPÓSITO

El propósito general de este documento es darle a conocer al usuario los requerimientos funcionales y no funcionales que forman parte para el desarrollo del sistema multiplataforma.

ALCANCE

El sistema funcionará en un entorno web y móvil, está pensado para el control de las aulas, luminaria y acondicionador de aires de la carrera de Computación, este sistema gestionará los siguientes procesos:

- Login
- Gestión de acceso a las aulas.
- Gestión de reserva.
- Gestión de bloques.
- Gestión de consumo.
- Gestión de Arduino.
- Gestión de usuarios
- Gestión de aulas
- Dashboard
- Gestión de acondicionador de aires.
- Envío de correos
- Gestión de solicitud

PERSONAL INVOLUCRADO

Nombre	Luis Eduardo Chichanda García
Rol	Jefe de desarrollo del sistema
Categoría profesional	Estudiante
Responsabilidades	Organizar: al equipo para dejar bien en claro su respectivo rol y responsabilidades. Dirigir: establecer una pauta a seguir por el equipo desarrollador. Controlar: verificar que las actividades se lleven de acuerdo al plan inicial de desarrollo.
Información de contacto	luis.chichanda@espam.edu.ec
Aprobación	Oliver Vera Salazar

Nombre	Oliver Antonio Vera Salazar
Rol	Programador
Categoría profesional	Estudiante
Responsabilidades	Elegir el código para cumplir con los requerimientos funcionales del sistema. Elegir el código que cumpla con las necesidades de interfaz según el diseño del sistema.
Información de contacto	oliver.vera@espam.edu.ec
Aprobación	Luis Chichanda García

Nombre	Ing. Javier López
--------	-------------------

Rol	Cliente
Categoría profesional	Docente, Ingeniero, Magister
Responsabilidades	Encargado de aportar con los requerimiento funcionales y no funcionales para el desarrollo del sistema multiplataforma.
Información de contacto	jlopez@espam.edu.ec
Aprobación	Oliver Vera Salazar, Luis Chichanda García

DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

HW: Hardware

SW: Software

PC: (del inglés personal computer), computadora personal, es una microcomputadora diseñada en principio para ser usada por una sola persona a la vez.

SGBD: sistema gestor de bases de datos, son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

Login: (término inglés) es el proceso mediante el cual se controla el acceso individual a un sistema informático mediante la identificación del usuario utilizando credenciales provistas por el usuario.

MVC: Es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones.

AJAX: Es una técnica de programación web que permite crear aplicaciones interactivas.

Front-end: Es la parte de un sistema que interactúa con los usuarios.

Back-end: Es la parte de un sistema encargada de procesar la entrada del front-end.

REFERENCIAS

Referencia	Título	Ruta	Fecha	Autor
-	-	-	-	-

RESUMEN

En las siguientes secciones se encontrará la descripción del producto, así como la descripción de lo que es necesario para que tenga un correcto funcionamiento.

DESCRIPCIÓN GENERAL

En esta sección se explica de forma general los objetivos que cumplirá el sistema.

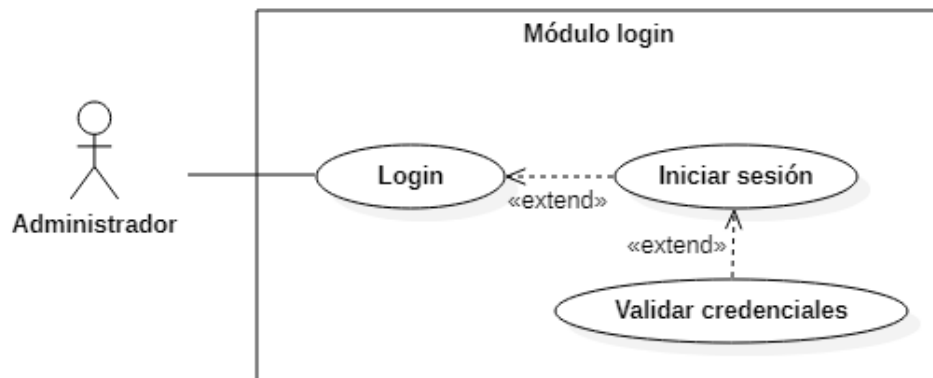
SCA es un sistema que controla el acceso a las aulas, luminarias y acondicionador de aires en el edificio de Computación de la ESPAM MFL, los datos generados se almacenarán en la base de datos para sus respectivos reportes.

PERSPECTIVA DEL PRODUCTO

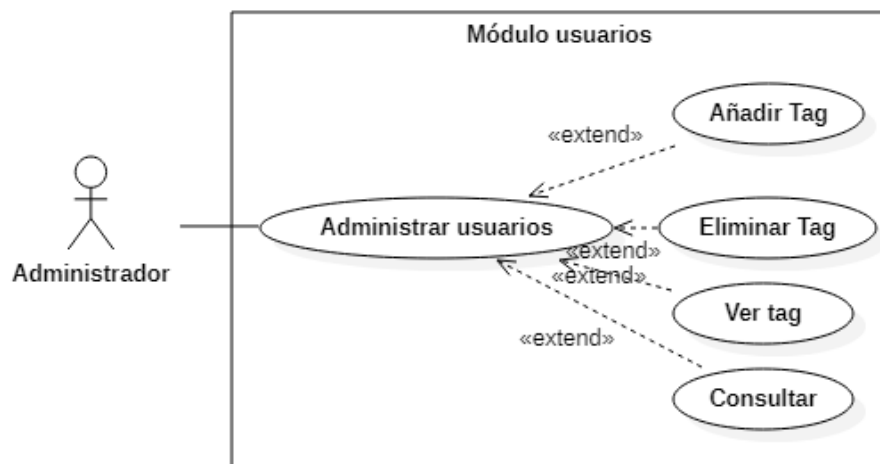
El sistema será un producto diseñado para trabajar en un entorno web y móvil administrable desde un subsistema web, lo que permitirá su utilización de forma eficaz y accesible desde cualquier equipo que se encuentre conectado a internet.

FUNCIONALIDAD DEL PRODUCTO

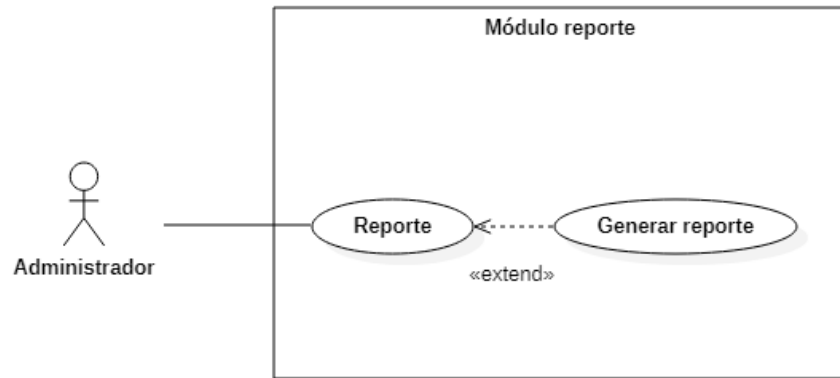
Login



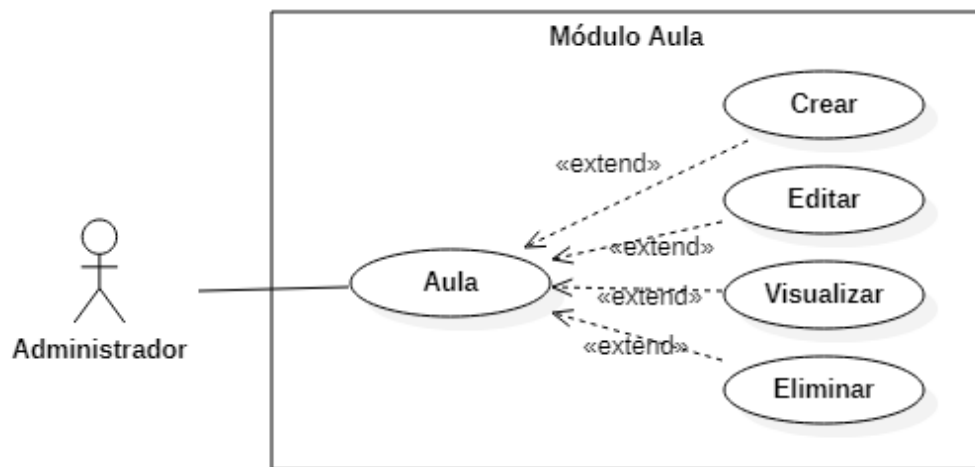
Usuarios



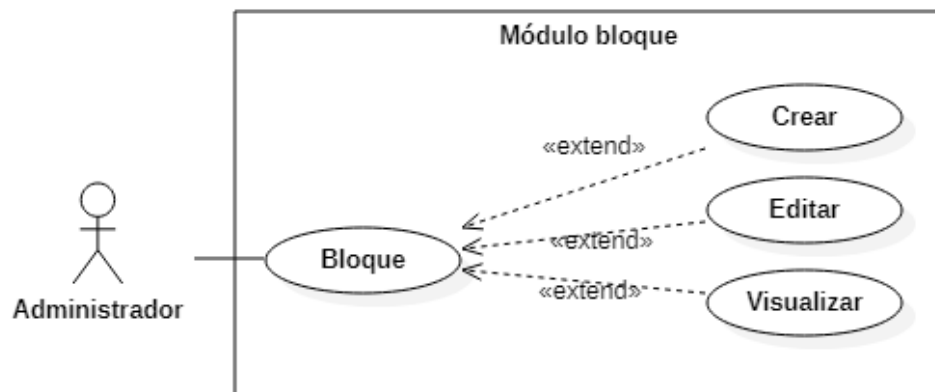
Reportes



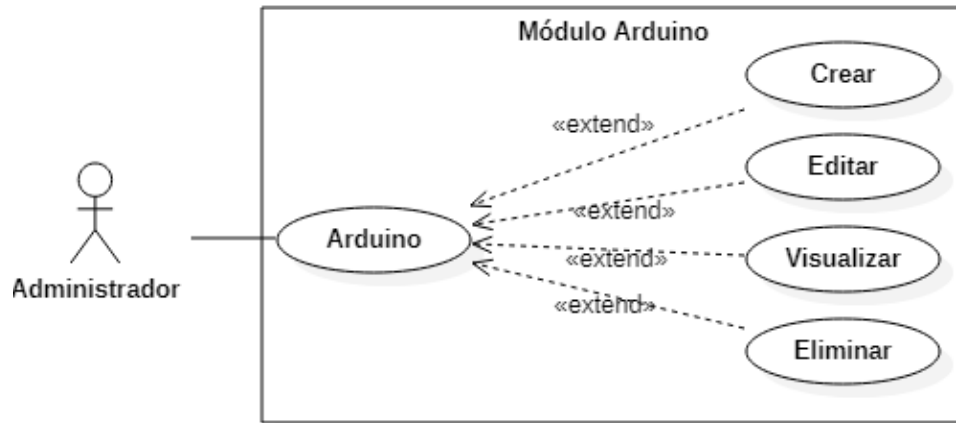
Gestión aulas



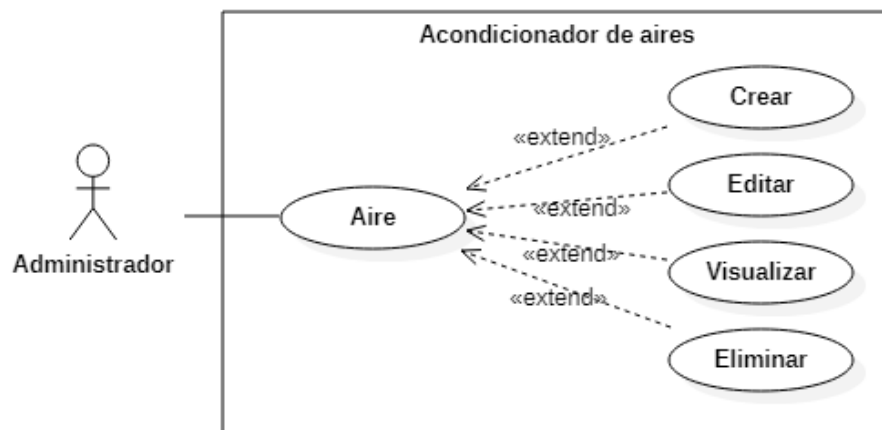
Gestión bloque



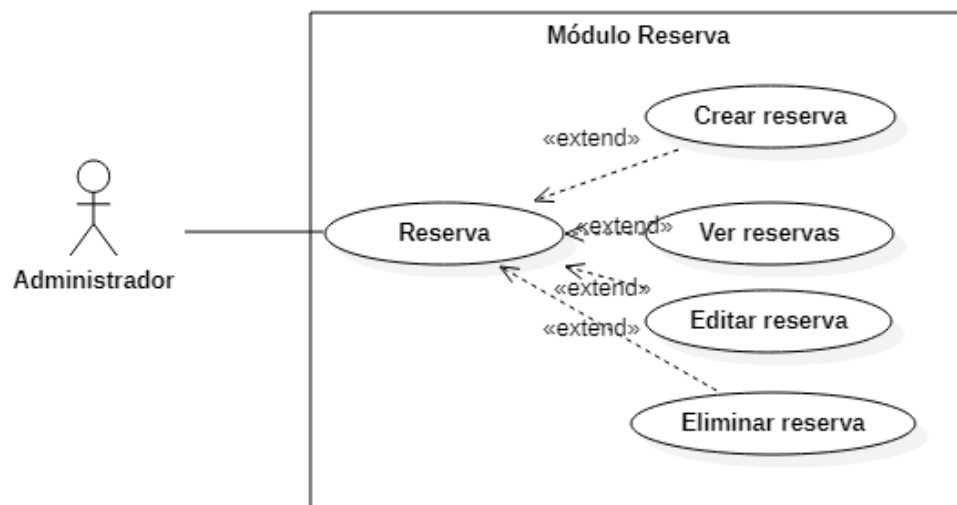
Gestión Arduino



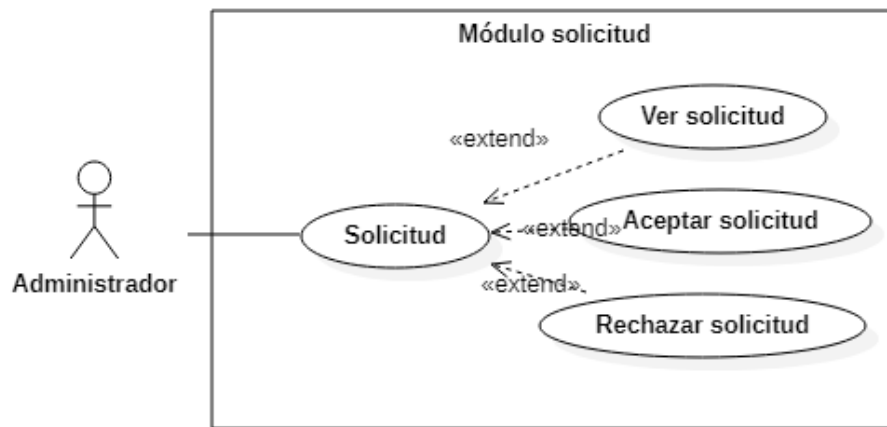
Acondicionador de aires



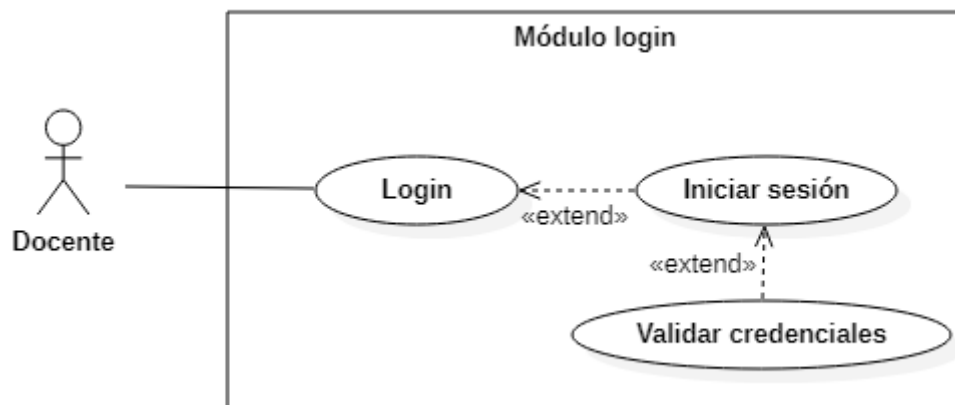
Reserva



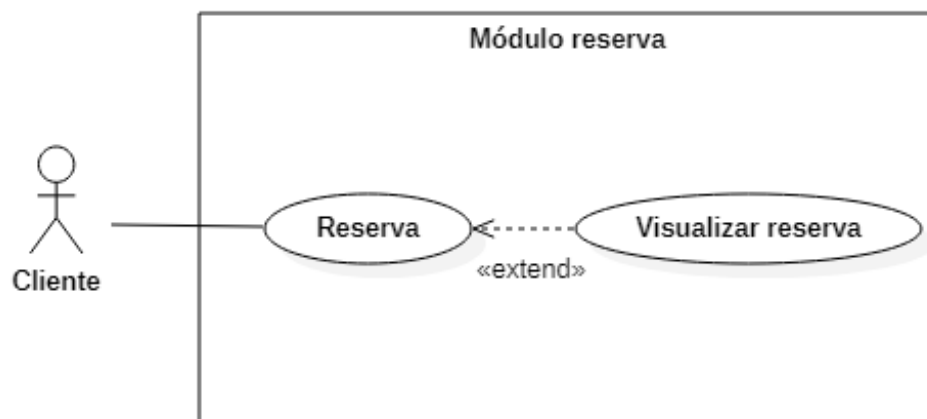
Solicitud



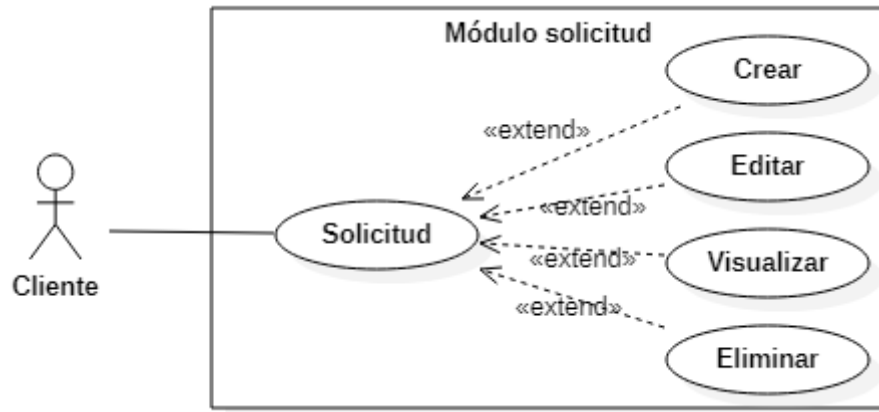
Login



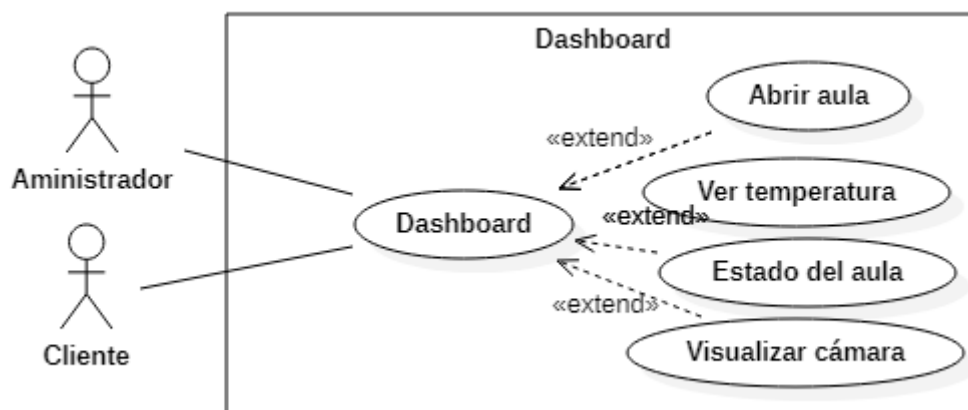
Reserva



Solicitud



Dashboard



CARACTERÍSTICAS DE LOS USUARIOS

Tipo de usuario	Administrador
Formación	Administrador
Habilidades	Habilidades básicas de manejo de una PC.
Actividades	Control total del sistema

Tipo de usuario	Usuarios
Formación	Ninguna
Habilidades	Habilidades básicas de manejo de una PC.
Actividades	Reserva de aulas. Abrir aula. Estado de las aulas.

RESTRICCIONES

- Lenguaje y tecnologías de uso: para el Front-end se utilizará (ASP.NET Framework MVC, CSS, HTML, JQuery y Ajax), y la programación del Back-end de desarrollar en (ASP.NET Framework Api)
- El sistema se diseñará utilizando la arquitectura (Modelo Vista Controlador).
- El sistema se diseñará en el modelo cliente/servidor.
- Se utilizará un servidor web para alojar codificación del sistema y un servidor de base de datos.
- El sistema debe controlar el acceso a la información, para que el cada usuario solo acceda a la información que le corresponde de acuerdo a su rol.

REQUISITOS ESPECÍFICOS

REQUISITOS COMUNES DE LOS INTERFACES

INTERFACES DE USUARIO

El sistema dispondrá de interfaces que ayudarán en los procesos que reanalizará el usuario dentro del sistema, ya que la interacción se ejecutará usando formularios, por lo que dichas interfaces incluirán:

- Botones
- Iconos relacionados a la funcionalidad de una opción
- Barras de menú

- Menús desplegables
- Mensajes de información
- Mensajes de finalización de un proceso
- Mensajes de error
- Cuadros de diálogo con ayuda

INTERFACES DE HARDWARE

Pantalla del monitor: Se necesitará disponer de una pantalla, ya que el sistema tendrá que mostrar información al Administrador y Farmacéutico a través del monitor de la computadora.

Ratón: La interacción del sistema se realizará con el movimiento y los botones del ratón, que permitirán seleccionar las opciones de los menús.

Teclado: El sistema interactuará con las pulsaciones del teclado, que permitirá realizar la entrada de datos al sistema.

Celular: Se necesitará que los clientes dispongan de un celular para que instalen la app y puedan hacer uso de los servicios.

Prototipo Arduino: se necesita una comunicación con el prototipo mediante la cual se recibirá y enviará acciones para el funcionamiento del sistema.

INTERFACES DE SOFTWARE

El sistema interactuará con la interfaz de impresión.

INTERFACES DE COMUNICACIÓN

El sistema se comunica con su base de datos a través del SGBD SQL Server y el back-end del sistema programado en ASP.NET Framework Api, por lo consiguiente se va a estar comunicando con los servicios proporcionados por el sistema de gestión académica, el sistema térmico y los servicios proporcionados por el prototipo de automatización.

REQUISITOS FUNCIONALES

GESTIÓN DE LOGIN

REQUISITO FUNCIONAL 1 “LOGIN”

Identificación del requisito:	RF1
Nombre del requisito:	Login
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador y el cliente podrán iniciar sesión con sus respectivas credenciales.
Prioridad del requisito	Alta

GESTIÓN DE USUARIOS

REQUISITO FUNCIONAL 2 “USUARIOS”

Identificación del requisito:	RF2
Nombre del requisito:	Usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

Descripción del requisito:	✓ El administrador debe poder visualizar los usuarios, así como: ver, añadir o eliminar un tag a dichos usuarios.
Prioridad del requisito	Alta

GESTIÓN DE AULAS

REQUISITO FUNCIONAL 3 “CREAR AULA”

Identificación del requisito:	RF3
Nombre del requisito:	Crear aula
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador en el módulo de crear aula podrá crearla.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 4 “MODIFICAR AULA”

Identificación del requisito:	RF4
Nombre del requisito:	Modificar aula
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá editar la información con respecto del aula
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 5 “Cargar aulas”

Identificación del requisito:	RF5
Nombre del requisito:	Cargar aulas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá visualizar las aulas que hayan sido creadas.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 6 “ELIMINAR AULA”

Identificación del requisito:	RF6
Nombre del requisito:	Eliminar aula

Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El usuario Administrador podrá eliminar aulas que hayan sido creadas..
Prioridad del requisito	Alta

GESTIÓN DE RESERVA

REQUISITO FUNCIONAL 7 “CREAR RESERVA”

Identificación del requisito:	RF7
Nombre del requisito:	Crear Reserva
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá crear una reserva para cualquier docente en un horario específico.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 8 “MODIFICAR RESERVA”

Identificación del requisito:	RF8
Nombre del requisito:	Modificar Reserva
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá modificar las reservas creadas.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 9 “CARGAR RESERVA”

Identificación del requisito:	RF9
Nombre del requisito:	Crear Reserva
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá visualizar las reservas de cada docente.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 10 “ELIMINAR RESERVA”

Identificación del requisito:	RF10
Nombre del requisito:	Eliminar Reserva
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá eliminar las reservas creadas.

Prioridad del requisito	Alta
-------------------------	------

GESTIÓN DE SOLICITUD

REQUISITO FUNCIONAL 11 “ACEPTAR Y DENEGAR SOLICITUDES”

Identificación del requisito:	RF11
Nombre del requisito:	Aceptar y denegar solicitudes
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá aceptar o denegar una solicitud hecha por el cliente.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 12 “CARGAR SOLICITUDES”

Identificación del requisito:	RF12
Nombre del requisito:	Cargar Solicitudes
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ Tanto el administrador como el cliente podrán ver las solicitudes realizadas.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 13 “CREAR SOLICITUD”

Identificación del requisito:	RF13
Nombre del requisito:	Crear Solicitud
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El cliente podrá crear solicitudes de aulas.
Prioridad del requisito	Alta

REQUISITO FUNCIONAL 14 “MODIFICAR SOLICITUD”

Identificación del requisito:	RF14
Nombre del requisito:	Modificar Solicitud
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El cliente podrá modificar alguna solicitud creada.

Prioridad del requisito	Alta
-------------------------	------

REQUISITO FUNCIONAL 15 “ELIMINAR SOLICITUD”

Identificación del requisito:	RF15
Nombre del requisito:	Eliminar Solicitud
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El cliente podrá eliminar una solicitud creada.
Prioridad del requisito	Alta

GESTIÓN DE BLOQUE

REQUISITO FUNCIÓN 16 “CREAR BLOQUE”

Identificación del requisito:	RF16
Nombre del requisito:	Crear bloque
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá crear bloques nuevos.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 17 “MODIFICAR BLOQUE”

Identificación del requisito:	RF17
Nombre del requisito:	Modificar bloque
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá modificar los bloques creados.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 18 “CARGAR BLOQUE”

Identificación del requisito:	RF18
Nombre del requisito:	Cargar bloque
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador visualizar los bloques.
Prioridad del requisito	Alta

GESTIÓN DE DISPOSITIVO INTELIGENTE EXTERNO (ARDUINO).

REQUISITO FUNCIÓN 19 “CREAR ARDUINO”

Identificación del requisito:	RF19
Nombre del requisito:	Crear Arduino
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá crear Arduinos nuevos.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 20 “MODIFICAR ARDUINO”

Identificación del requisito:	RF20
Nombre del requisito:	Modificar Arduino
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá modificar los Arduinos creados.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 21 “CARGAR ARDUINO”

Identificación del requisito:	RF21
Nombre del requisito:	Cargar Arduino
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador visualizar los Arduinos.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 22 “ELIMINAR ARDUINO”

Identificación del requisito:	RF22
Nombre del requisito:	Eliminar Arduino
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá eliminar los Arduinos creados.
Prioridad del requisito	Alta

GESTIÓN ACONDICIONADOR DE AIRES.

REQUISITO FUNCIÓN 23 “ENCENDIDO Y APAGADO DE ACONDICIONADOR DE AIRES”

Identificación del requisito:	RF23
Nombre del requisito:	✓ Encendido y apagado de acondicionador de aires
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El sistema será capaz de identificar el estado de aula y así poder enviar una señal o interactuar con las Apis de apagado o encendido los acondicionadores de aire.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 24 “CREAR AIRE”

Identificación del requisito:	RF24
Nombre del requisito:	✓ Crear aire
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá crear aires nuevos.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 25 “MODIFICAR AIRE”

Identificación del requisito:	RF25
Nombre del requisito:	✓ Modificar aire
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá modificar los aires creados.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 26 “CARGAR AIRE”

Identificación del requisito:	RF26
Nombre del requisito:	✓ Cargar aire
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador visualizar los aires creados.
Prioridad del requisito	Alta

REQUISITO FUNCIÓN 27 “ELIMINAR AIRE”

Identificación del requisito:	RF27
Nombre del requisito:	✓ Eliminar aire
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador podrá eliminar los aires creados.
Prioridad del requisito	Alta

GESTIÓN DE ENVÍO DE CORREOS.

REQUISITO FUNCIÓN 28 “ENVÍO DE CORREOS”

Identificación del requisito:	RF28
Nombre del requisito:	✓ Envío de correos
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ Al crear una solicitud, automáticamente se le envía un correo al administrador notificándole que se ha generado una solicitud.
Prioridad del requisito	Alta

GESTIÓN DE CONSUMO.

REQUISITO FUNCIÓN 29 “CONSUMO ENERGÉTICO”

Identificación del requisito:	RF29
Nombre del requisito:	Consumo energético
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El administrador del sistema podrá generar reportes del consumo que genera cada aula o las aulas en general.
Prioridad del requisito	Alta

GESTIÓN DE REPORTE

REQUISITO FUNCIÓN 30 “REPORTE”

Identificación del requisito:	RF30
-------------------------------	------

Nombre del requisito:	Reporte
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ Le permitirá al administrador del sistema generar reportes del aulas, solicitudes y reservas.
Prioridad del requisito	Alta

DASHBOARD

REQUISITO FUNCIÓN 31 “DASHBOARD”

Identificación del requisito:	RF31
Nombre del requisito:	Dashboard
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ Permite interactuar y visualizar el estado de las diferentes aulas(Acceso, consumo, cámaras, temperatura, estado de Arduino).
Prioridad del requisito	Alta

INFORMACIÓN EN TIEMPO REAL.

REQUISITO FUNCIÓN 32 “INFORMACIÓN EN TIEMPO REAL”

Identificación del requisito:	RF32
Nombre del requisito:	✓ Información en tiempo real
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Descripción del requisito:	✓ El sistema deberá enviar y recibir información en tiempo real.
Prioridad del requisito	Alta

REQUISITOS NO FUNCIONALES

REQUISITOS DE RENDIMIENTO

Garantiza las consultas, filtros o búsquedas se desarrolles de forma optimizada, para que no afecte el desempeño de la base de datos, para que no se afecte el rendimiento del tráfico de la red institucional.

El sistema deberá tener la capacidad de permitir el acceso a todos los usuarios del sistema, además deberá brindar un buen rendimiento al momento de realizar procesos de alta, media y baja demanda de recursos del sistema.

SEGURIDAD

Para poder ingresar al sistema es necesario contar con un usuario y una contraseña que se encuentren almacenados en la base de datos de gestión académica, de esta manera se controla que solo los usuarios autorizados podrán manipular el sistema y realizar cualquier consulta garantizando la seguridad y confiabilidad de la información.

El sistema contará con un mecanismo de registro de actividades, para identificar que usuario realiza una actividad dentro del sistema.

El sistema deberá controlar el acceso a todas las rutas, para garantizar que solo los usuarios permitidos puedan acceder a una ruta específica.

INTERFAZ AMIGABLE

El sistema web tendrá una interfaz de usuario sencilla e intuitiva para que sea de fácil manejo para los usuarios.

DISPONIBILIDAD

El sistema se desarrollará teniendo en cuenta que estará disponible las 24 horas del día y accesible desde cualquier lugar.

El sistema debe enviar y recibir información en tiempo real.

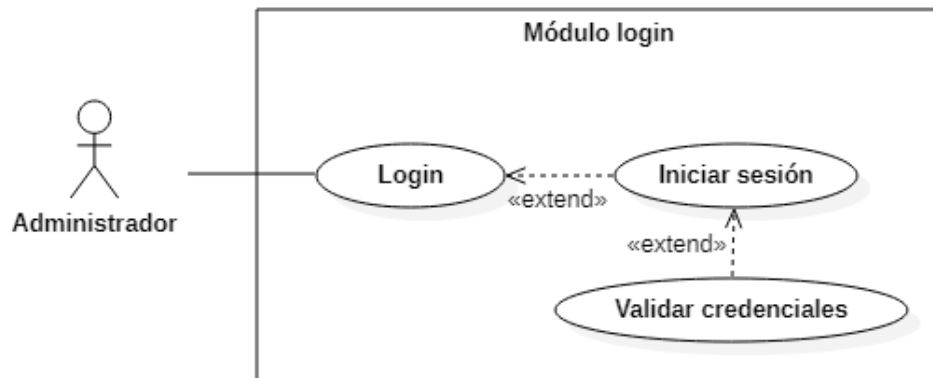
MANTENIBILIDAD

El sistema tendrá un manual de usuario para facilitar el manejo del mismo por parte de los usuarios. Además, la codificación del sistema estará comentada para garantizar un fácil entendimiento.

ANEXO 3. CASOS DE USO

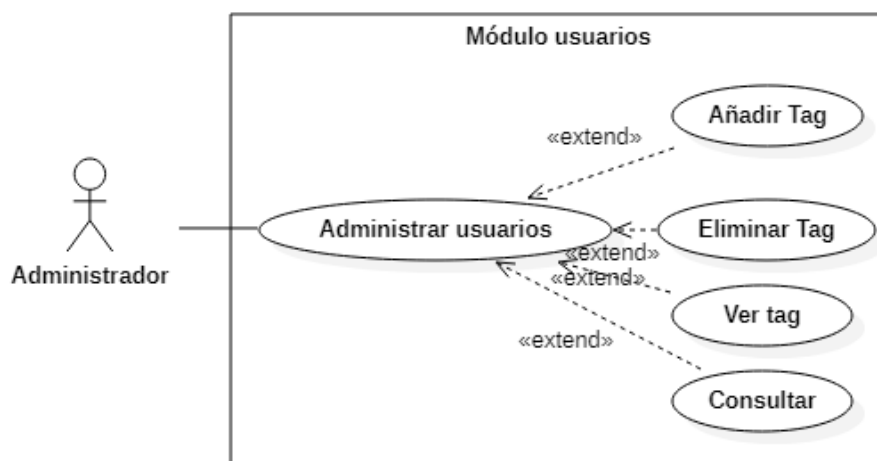
Nombre Caso de uso: Login

Actor: Administrador



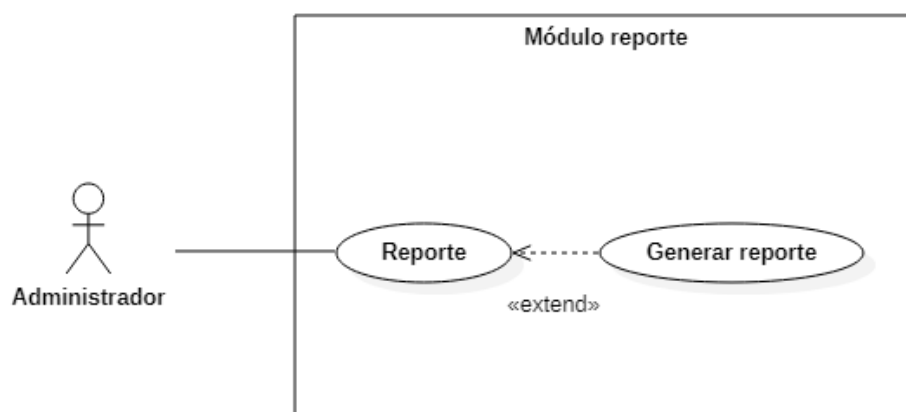
Nombre Caso de uso: Usuarios

Actor: Administrador



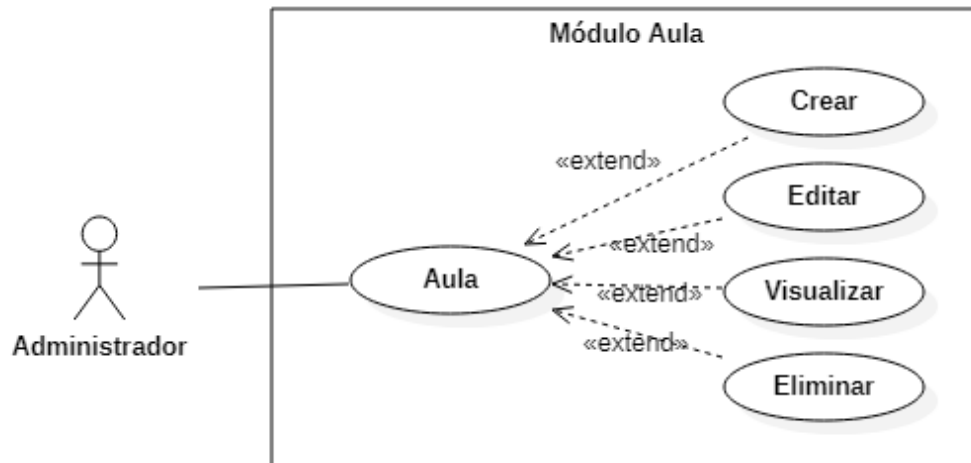
Nombre Caso de uso: Reportes

Actor: Administrador



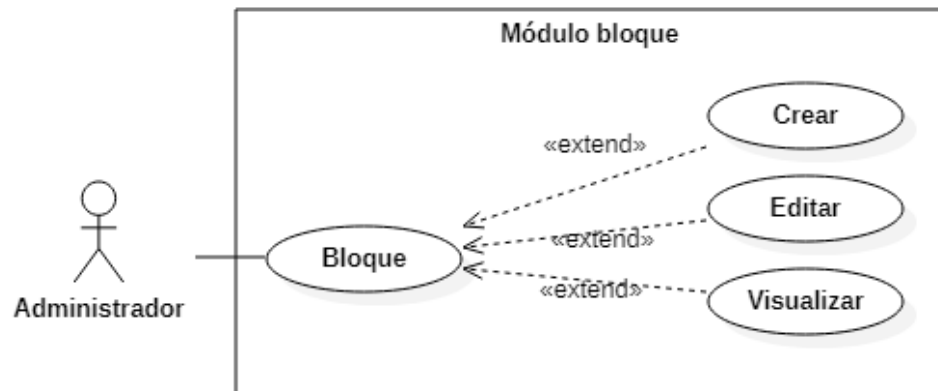
Nombre Caso de uso: Gestión aulas

Actor: Administrador



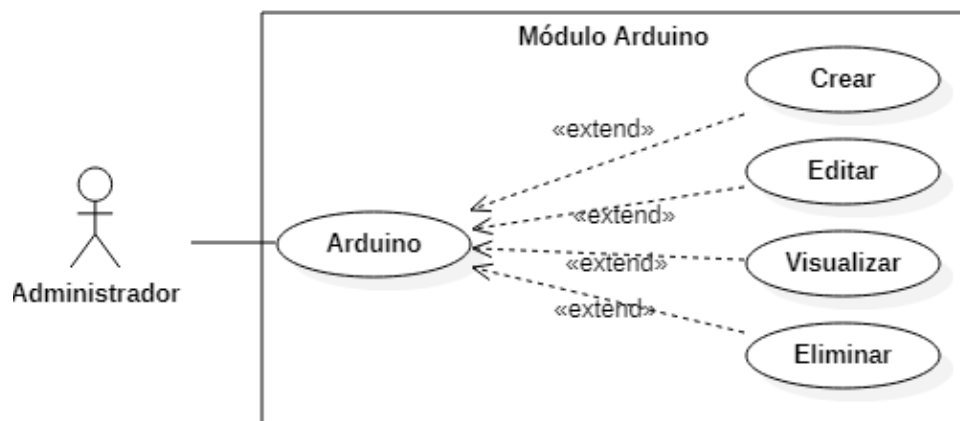
Nombre Caso de uso: Gestión bloque

Actor: Administrador



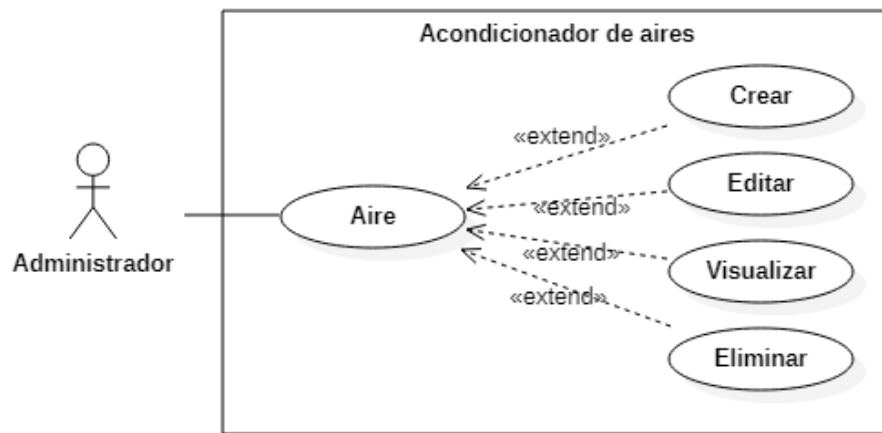
Nombre Caso de uso: Gestión Arduino

Actor: Administrador



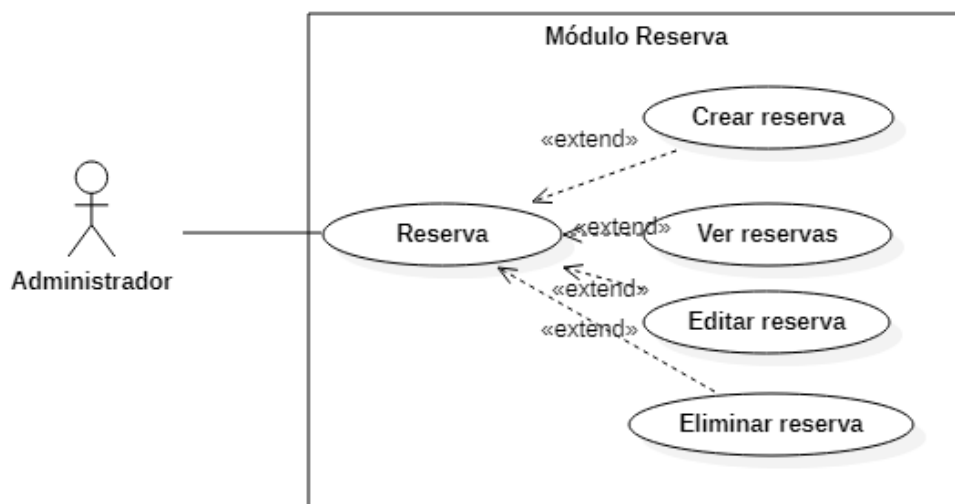
Nombre Caso de uso: Acondicionador de aires

Actor: Administrador



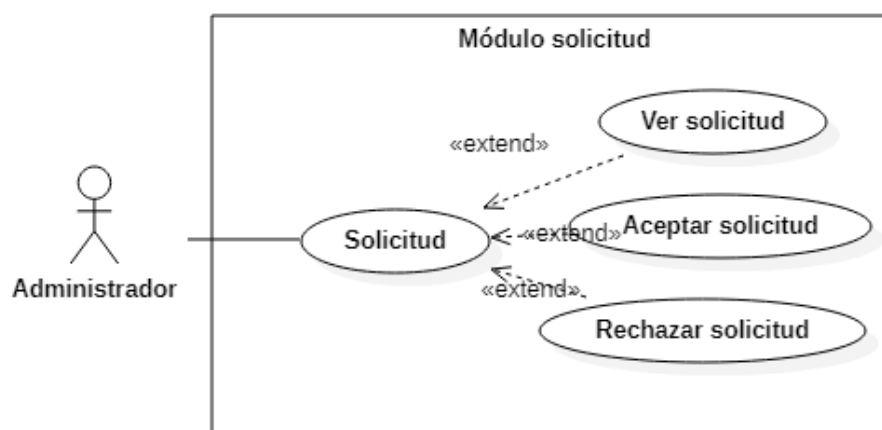
Nombre Caso de uso: Reserva

Actor: Administrador



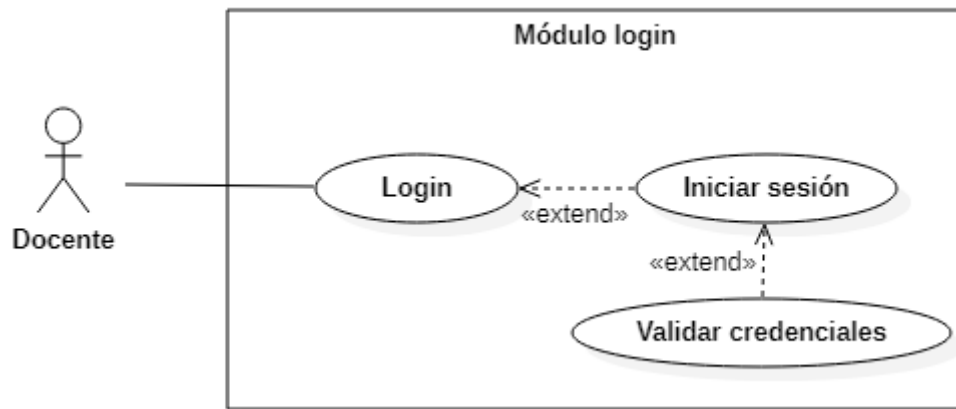
Nombre Caso de uso: Solicitud

Actor: Administrador



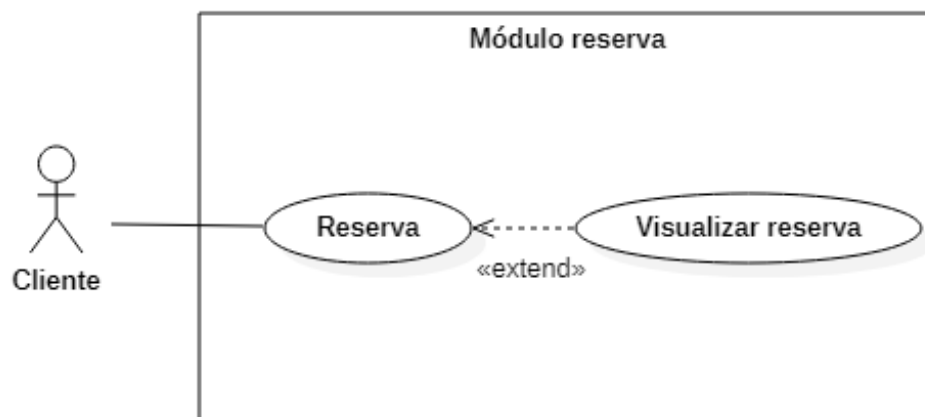
Nombre Caso de uso: Login

Actor: Docente



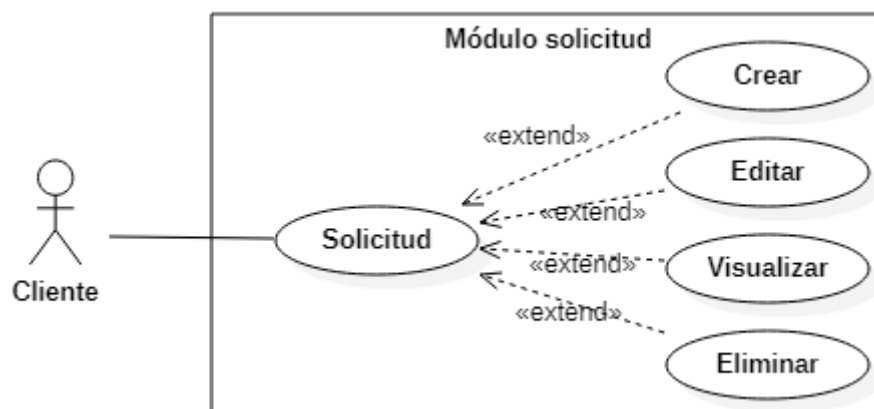
Nombre Caso de uso: Reserva

Actor: Cliente



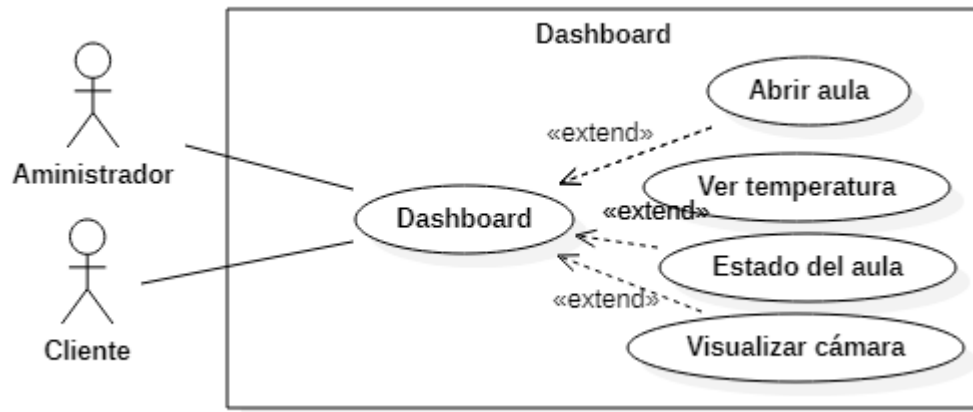
Nombre Caso de uso: Solicitud

Actor: Cliente



Nombre Caso de uso: Dashboard

Actor: Administrador-Cliente



ANEXO 4. INTERFACES DEL SISTEMA MULTIPLATAFORMA

Anexo 4-A. Interfaces Web

Módulo: Login

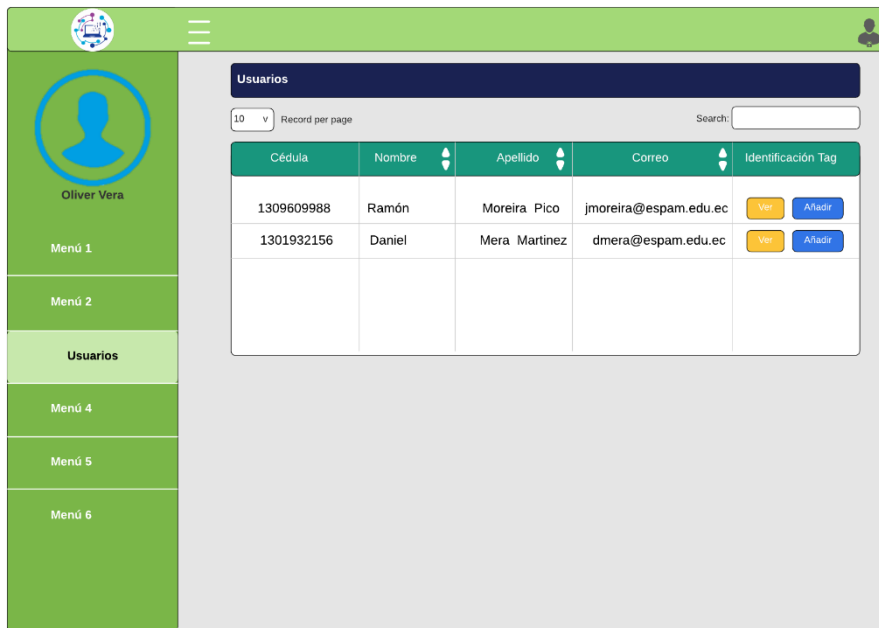
Usuario: Administrador-Docente

The login interface features a light green background. At the top center is a circular logo with a laptop, a globe, and binary code. Below the logo are two white input fields: the first is labeled 'Usuario' and the second is labeled 'Contraseña'. Below these fields is a yellow button with the text 'INGRESAR' in black capital letters.

Módulo: Pantalla de inicio

Usuario: Administrador-Docente

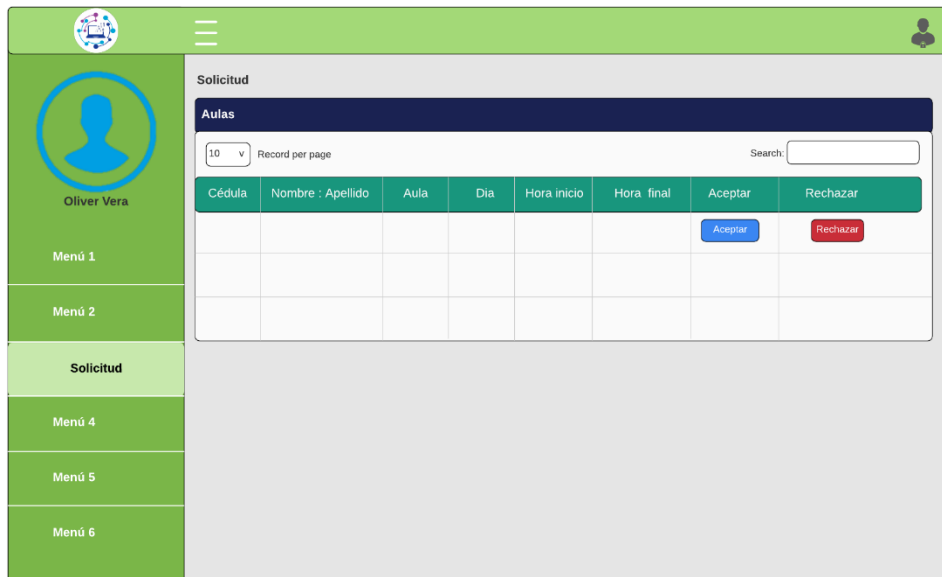
The dashboard interface has a green header bar with a logo on the left, a hamburger menu icon in the center, and a user profile icon on the right. Below the header, the main content area is titled 'Sistema de Control de Acceso' with a sub-header 'Computación'. There are two tabs: 'Esquema de cobro' (selected) and 'Contorno'. The main area displays a grid of seven cards. Each card contains a house icon, a lock icon, a video camera icon, and a person icon. Below the icons, each card shows 'Reserva 0 >' and 'Solicitud 0 >'. On the left side, there is a vertical green sidebar with a user profile icon and the name 'Oliver Vera', followed by six menu items labeled 'Menú 1' through 'Menú 6'.

Módulo: Usuarios**Usuario:** Administrador


Usuarios

10 v Record per page Search:

Cédula	Nombre	Apellido	Correo	Identificación Tag
1309609988	Ramón	Moreira Pico	jmoreira@espam.edu.ec	Ver Añadir
1301932156	Daniel	Mera Martínez	dmera@espam.edu.ec	Ver Añadir

Módulo: Solicitud**Usuario:** Administrador


Solicitud

Aulas

10 v Record per page Search:

Cédula	Nombre : Apellido	Aula	Dia	Hora inicio	Hora final	Aceptar	Rechazar
						Aceptar	Rechazar

Módulo: Reservas

Usuario: Administrador

Módulo: Reportes

Usuario: Administrador

Category	Budget	Expense	Budget - Expense
Phone	85	95	-10
Internet	200	230	-30
Electricity	300	375	-75
Movies	85	80	5
Food	300	350	-50
Fuel	650	650	0
Insurance	200	300	-100

Módulo: Bloques**Usuario:** Administrador

Oliver Vera

Menú 1

Menú 2

Bloque

Menú 4

Menú 5

Menú 6

Crear Bloque

Código

Descripción

Crear bloque

Bloque

10 v Record per page Search:

Código	Estado	Descripción	Editar
			<input type="button" value="Editar"/>

Módulo: Aulas**Usuario:** Administrador

Oliver Vera

Menú 1

Menú 2

Aula

Menú 4

Menú 5

Menú 6

Aulas

Crear Aula

Código aula

Tipo aula

Bloque

Arduino

Aire

IP Cámara

Descripción

Crear aula

Aulas

10 v Record per page Search:

Código	Nombre	Tipo aula	Bloque	IP Cámara	Arduino	Acondicionador	Estado	Editar	Eliminar
								<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>

Módulo: Arduino

Usuario: Administrador

The screenshot shows the 'Crear Arduino' form and a table of existing records. The table has the following structure:

Código	Estado	Descripción	Editar	Eliminar
			Editar	Eliminar

Módulo: Acondicionador de aires

Usuario: Administrador

The screenshot shows the 'Crear Aire' form and a table of existing records. The table has the following structure:

Código	Estado	Descripción	Editar	Eliminar
			Editar	Eliminar

Módulo: Consumo

Usuario: Administrador



Módulo: Reserva

Usuario: Docente

The screenshot displays the 'Reservas' module for a teacher. The sidebar on the left includes the user's profile 'Oliver Vera' and a menu with options 'Menú 1' through 'Menú 6', with 'Reserva' selected. The main content area features a table with the following structure:

Reservas

10 v Record per page Search:

Aula	Día	Hora inicio	Hora fin	Frecuentemente

Módulo: Solicitud

Usuario: Docente

The screenshot shows a web application interface for a teacher. On the left is a green sidebar with a user profile for 'Oliver Vera' and a menu with items 'Menú 1' through 'Menú 6', where 'Solicitud' is highlighted. The main content area is titled 'Solicitud' and contains two sections. The top section, 'Crear Solicitud', has a form with fields for 'Aula' (a dropdown menu), 'Hora inicio', 'Hora fin', and 'Fecha Solicitud' (a date picker). It also includes a 'Recurrencia' checkbox which is checked, a 'Descripción' text area, and a 'Crear solicitud' button with a 'Crear' sub-button. The bottom section, also titled 'Solicitud', features a table with columns for 'Aula', 'Día', 'Hora inicio', 'Hora fin', 'Descripción', 'Recurrencia', 'Editar', and 'Eliminar'. Above the table are controls for 'Records per page' (set to 10) and a search box. The table is currently empty.

Anexo 4-B. Interfaces Móvil

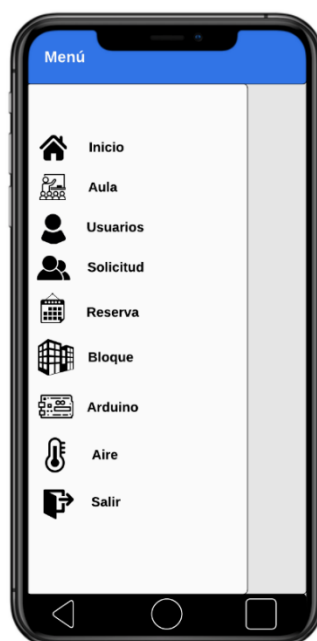
Módulo: Login

Usuario: Administrador-Docente



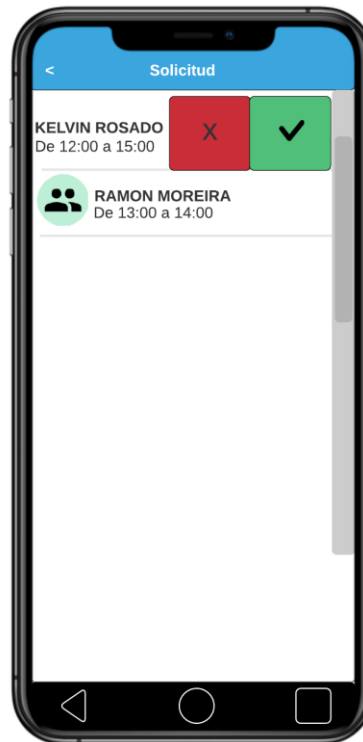
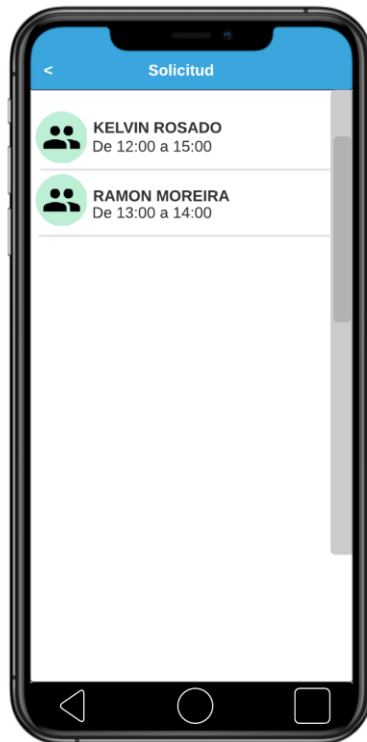
Módulo: Pantalla de inicio

Usuario: Administrador



Módulo: Solicitud

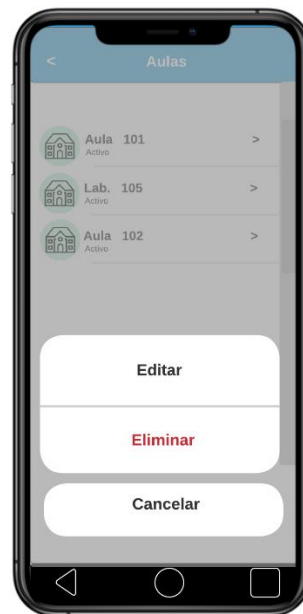
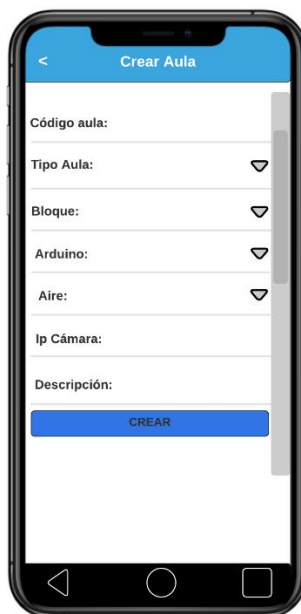
Usuario: Administrador



Módulo: Bloque

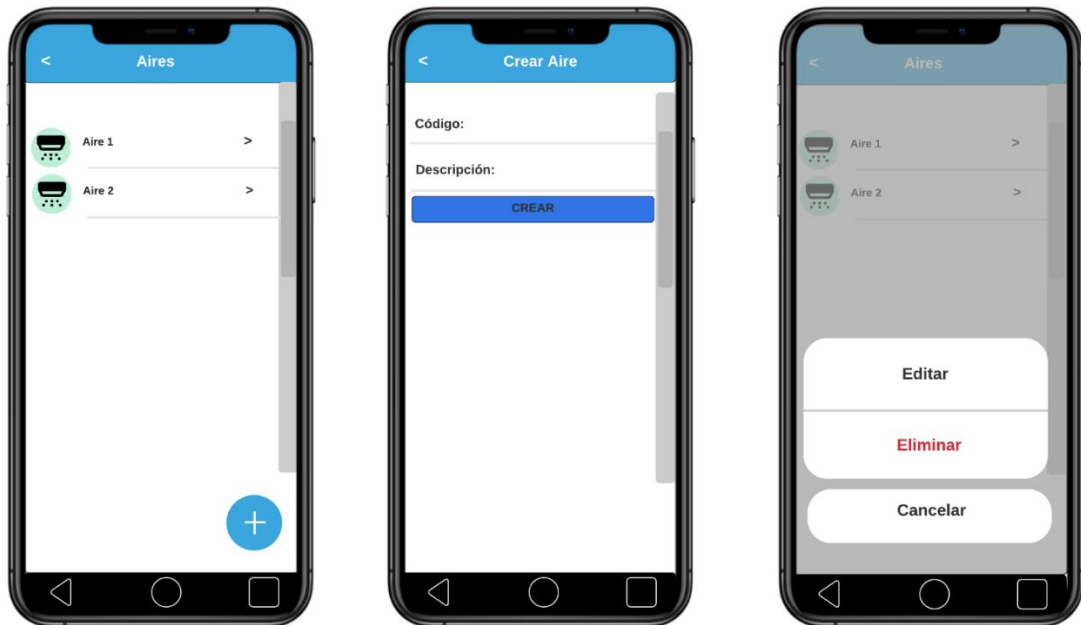
Usuario: Administrador



Módulo: Aula**Usuario: Administrador****Módulo: Arduino****Usuario: Administrador**

Módulo: Acondicionador de aires

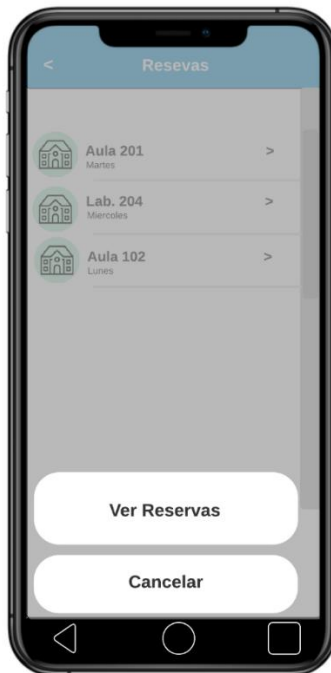
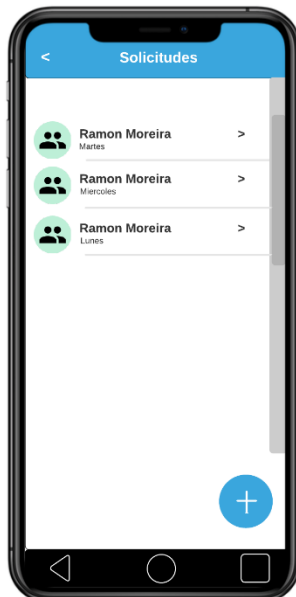
Usuario: Administrador



Módulo: Pantalla de inicio

Usuario: Docente



Módulo: Reserva**Usuario:** Docente**Módulo:** Solicitud**Usuario:** Docente

ANEXO 5. MANUAL DEL PROGRAMADOR



MANUAL DEL PROGRAMADOR

**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE
ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR
DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**

DESARROLLADORES:

Luis Eduardo Chichanda García

Oliver Antonio Vera Salazar

ENERO, 2022

INTRODUCCIÓN

La finalidad de este documento es explicar detalladamente el lenguaje de programación, el patrón de diseño, las técnicas, procedimientos que se utilizaron para realizar la codificación del trabajo de titulación “Sistema multiplataforma para el control de acceso a las aulas, luminarias y acondicionador de aires del edificio de la carrera de computación”, con el objetivo de dar a conocer al posible nuevo equipo de desarrolladores la estructura del sistema.

IDENTIFICACIÓN DEL SISTEMA

PARÁMETROS	DETALLE
Proyecto	Sistema multiplataforma para el control de acceso a las aulas, luminarias y acondicionador de aires del edificio de la carrera de computación.
Versión	1.0.
Área de desarrollo	Este sistema fue desarrollado para aplicarse en todas las áreas de la institución en donde se realicen tareas.
ERS de referencia	Se utilizó como documento de referencia el Standard IEEE 830.
Equipo	Luis Eduardo Chichanda García. Oliver Antonio Vera Salazar.
Modelo de desarrollo	La metodología utilizada fue XP (Programación Extrema).
Paradigma de programación	El patrón de diseño del sistema es MVC con ASP.NET Framework 4.8.

ALCANCE

El sistema funcionará en un entorno web y móvil, está pensado para el control de las aulas, luminaria y acondicionador de aires de la carrera de Computación.

REQUERIMIENTOS DE HARDWARE

Se recomienda que las computadoras que ejecuten el sistema web tengan las siguientes características:

- ✓ RAM mínimo de 1 GB
- ✓ Procesador Intel mínimo Core I3
- ✓ Conexión a Internet
- ✓ Disco duro de 300 GB
- ✓ Accesorios: teclado y mouse
- ✓ Tarjeta de red integrada 10/100/1000 Ethernet

REQUERIMIENTOS DE SOFTWARE

Las herramientas de software utilizadas en el sistema fueron el ASP.NET Framework 4.8, HTML, CSS3, AJAX, Ionic Framework 6.18.0, Angular 13.0.1, TypeScript 4.4.4, JavaScript y APIs, para el motor de base de datos se utilizó SQL.

CONOCIMIENTOS PREVIOS

Para que un equipo de desarrolladores realice cambios en el sistema, es necesario que tengan conocimientos acerca de:

- ✓ ASP.NET Framework 4.8
- ✓ Patrón MVC
- ✓ Recursos web
- ✓ Peticiones HTTP
- ✓ SQL Server 2019 15.0.2000.5
- ✓ TypeScript 4.4.4
- ✓ Ionic Framework 6.18.0
- ✓ Angular 13.0.1
- ✓ HTML
- ✓ CSS3
- ✓ JavaScript
- ✓ APIs

1. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

1.1. VISTA FUNCIONAL

La estructura del proyecto se basa en el cumplimiento de los requerimientos generales del mismo. Los pasos que siguen los usuarios del sistema para garantizar la correcta planificación y ejecución de las tareas se especifican en el Anexo 3 del trabajo de titulación.

1.2. VISTA LÓGICA

El sistema web está desarrollado bajo el lenguaje de programación ASP.NET, el cual utiliza el entorno MVC (Modelo, Vista, Controlador) para una mejor organización del código. En las Vistas se encuentran todas las páginas de interacción con el usuario y en los Controladores se conectan con la APIs por medio de peticiones HTTP.

Los servicios se encuentran desarrollados bajo el entorno de desarrollo ASP.NET Framework APIs, el mismo que está formado por controladores modelos conexión a la base de datos y la lógica de programación.

La aplicación móvil esta desarrollada bajo el lenguaje de programación TypeScript. Además, se utilizó Ionic y Angular de Framework para su desarrollo.

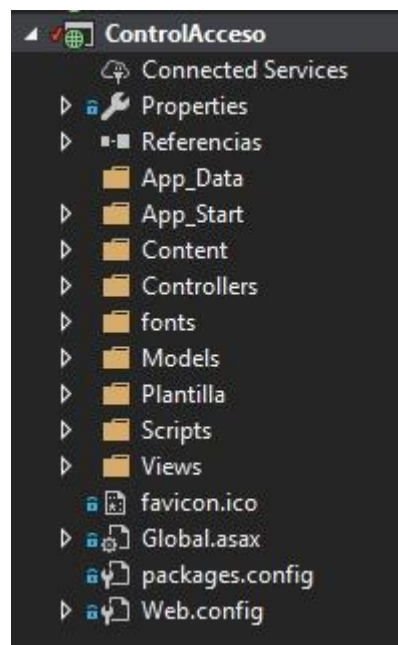


Figura 52. Estructura general del sistema web

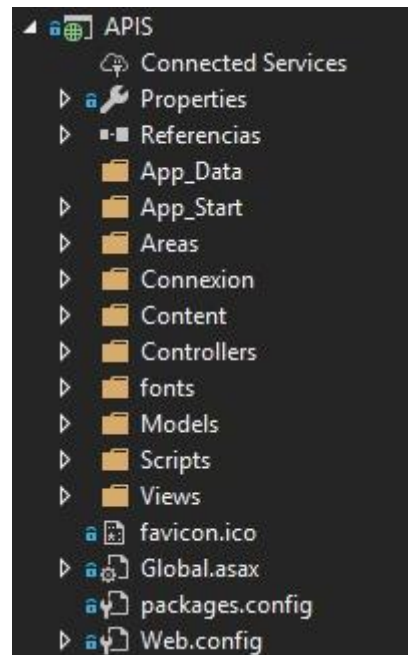


Figura 53. Estructura general de los servicios

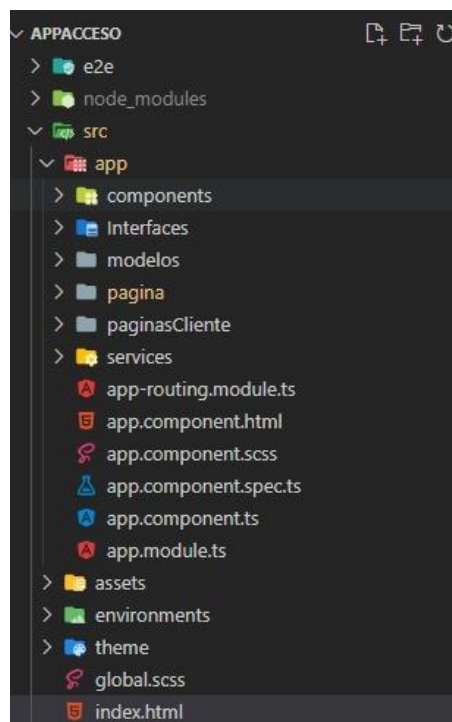


Figura 54. Estructura general de la aplicación móvil

1.2.1. ESTRUCTURA DEL SISTEMA MULTIPLATAFORMA

Para poderse comunicar con los servicios del sistema existen dos maneras las cuales son: a través de la app móvil y el sistema web, de los cuales cada uno se conectan por medio de la internet. La aplicación móvil se conecta directamente con los servicios APIs, en cambio el sistema web funciona con la arquitectura MVC.

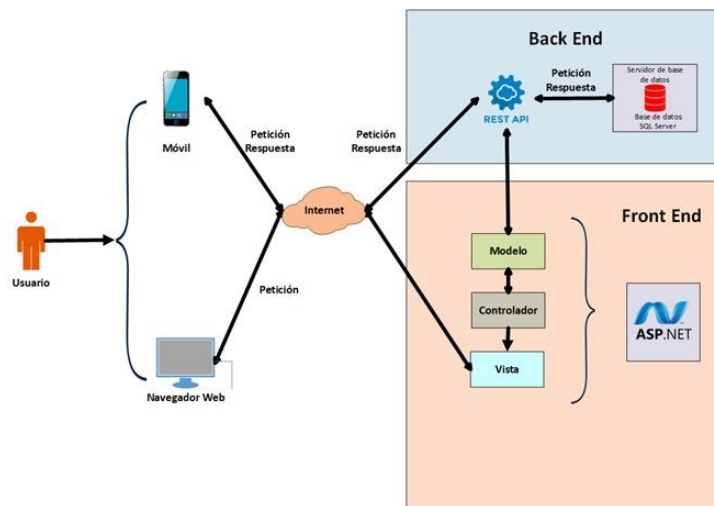


Figura 55. Modelo, Vista, Controlador

1.3. DESCRIPCIÓN DE LOS ACTORES DEL SISTEMA

Los actores del sistema (tipo usuarios) son administrador y usuario.

2. DISEÑO Y ANÁLISIS DE ALGORITMOS

2.1. ESTANDARIZACIÓN DEL CÓDIGO

La estandarización del código se muestra en el siguiente cuadro:

ELEMENTO	TIPO DE ELEMENTO	ESTILO DE ESCRITURA	EJEMPLO
Modelo	Model	Primera letra de cada palabra en mayúscula.	Persona.cs, Arduino.interface.ts

Vistas	Views	Primera letra de cada palabra en mayúscula.	Index.html, Arduino.page.html
Controladores	Controllers	Primera letra de cada palabra en mayúscula.	ArduinoController.css
Variables	Locales y globales	Todo escrito en minúsculas	rol
Objetos	-	Se antepone la letra "o" seguida de la palabra.	oArduino
Listas	-	Se antepone la palabra "list" seguido de la palabra.	listAreas
Métodos	-	Primera letra de cada palabra en mayúscula.	Personainsertar
HTML	-	En forma de etiquetas	<h1>Tareas</h1>
Services	Services	La primera palabra el tipo de servicio seguido del nombre.	getArduino

2.2. CONEXIÓN A LA BASE DE DATOS

El tipo de conexión, nombre de base de datos, usuario, contraseña, entre otros, se especifican en el archivo “.config” que se encuentra en la raíz del directorio de las APIs.

```
<connectionStrings>
  <add name="AdministracionAcademicaEntities"
connectionString="metadata=res:///Connexion.GESTIONACADEMICA.Model1.csdl|res:///
Connexion.GESTIONACADEMICA.Model1.ssdl|res:///Connexion.GESTIONACADEMICA.Model1.
msl;provider=System.Data.SqlClient;provider connection string=&quot;data source=
DESKTOP-H1B23TT;initial catalog=AdministracionAcademica;user
id=sa;password=luis456chi;MultipleActiveResultSets=True;
App=EntityFramework&quot;; providerName="System.Data.EntityClient" />
  <add name="ControlAccesoEntities" connectionString="metadata=res:///
Connexion.ControlAcceso.ConexionControlAcceso.csdl|res:///Connexion.
ControlAcceso.ConexionControlAcceso.ssdl|res:///Connexion.
ConexionControlAcceso.msl;provider=System.Data.SqlClient;provider connection
string=&quot;data source=DESKTOP-H1B23TT;initial catalog=ControlAcceso;user
id=sa;password=luis456chi;MultipleActiveResultSets=True;App=EntityFramework&quot;;
providerName="System.Data.EntityClient" />
</connectionStrings>
```

Figura 56. Conexión a la base de datos

2.3. VISTA

En esta parte se detallan todas las vistas presentadas en el sistema web y móvil.

VISTA	DESCRIPCIÓN
Usuario	Vista para mostrar todos los usuarios, añadir tag, eliminar tag.
VISTA	DESCRIPCIÓN
Aula	Vista para la administración de las aulas
VISTA	DESCRIPCIÓN
Arduino	Vista para la administración de los Arduinos
VISTA	DESCRIPCIÓN
Aire	Vista para la administración de los aires
VISTA	DESCRIPCIÓN

Bloque	Vista para visualizar, modificar y añadir bloques.
VISTA	DESCRIPCIÓN
Reserva	Vista para ver reserva.
ReservaAdmin	Vista para administración de las reservas.
VISTA	DESCRIPCIÓN
Solicitud	Vista para administrar las reservas.
SolicitudAdmin	Vista para ver, rechazar y aceptar una solicitud.
VISTA	DESCRIPCIÓN
Home	Vista para visualizar el dashboard.
VISTA	DESCRIPCIÓN
Reporte	Vista para generar reporte.

2.4. CONTROLADOR

En esta parte se detallan todos los controladores con sus parámetros y APIs consumidas.

AIRECONTROLLER.CS

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de aires	'GET', "Aire"
GET	JsonResult	Muestra datos	Código Aire	Aire	'GET', "Aire/{id}"
POST	JsonResult	Actualiza datos	Datos de aire	Dato de aire	'POST', "[form_params] => \$data]
PUT	JsonResult	Edita datos	Datos de aire	Datos de aire	'PUT', "[form_params] => \$data]
DELETE	JsonResult	Elimina datos	Id aire	Mensaje de respuesta	'DELETE', "Aire/{id}"

AulaController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de aulas	'GET', "Aula"
GET	JsonResult	Muestra datos	Código Aula	Dato de aula	'GET', "Aula/{id}"
POST	JsonResult	Actualiza datos	Datos de Aula	Dato de aula	'POST', "[form_params] => \$data]
PUT	JsonResult	Edita datos	Datos de aula	Datos de aula	'PUT', "[form_params] => \$data]
DELETE	JsonResult	Elimina datos	Id aula	Mensaje de respuesta	'DELETE', "Aula/{id}"

ArduinoController.cs

MÉTODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de Arduinos	'GET', "Arduino"
GET	JsonResult	Muestra datos	Código Arduino	Arduino	'GET', "Arduino/{id}"
POST	JsonResult	Actualiza datos	Datos de Arduino	Dato de Arduino	'POST', "[form_params] => \$data]
PUT	JsonResult	Edita datos	Datos de Arduino	Datos de Arduino	'PUT', "[form_params] => \$data]
DELETE	JsonResult	Elimina datos	Id Arduino	Mensaje de respuesta	'DELETE', "Arduino/{id}"

ReservaAdminController.cs

MÉTODO	TIPO	DESCRIPCIÓN	PARÁMETROS DE ENTRADA	PARÁMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de Reservas	'GET', "Reserva"

GET	JsonResult	Muestra datos	Código Reserva	Reserva	'GET', " Reserva /{id}"
POST	JsonResult	Actualiza datos	Datos de Reserva	Dato de Reserva	'POST',",[form_params' => \$data]
PUT	JsonResult	Edita datos	Datos de Reserva	Datos de Reserva	'PUT',",[form_params' => \$data]
DELETE	JsonResult	Elimina datos	Id Reserva	Mensaje de respuesta	'DELETE', " Reserva /{id}"

BloqueController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de Bloques	'GET', " Bloque "
GET	JsonResult	Muestra datos	Código Bloque	Bloque	'GET', " Bloque /{id}"
POST	JsonResult	Actualiza datos	Datos de Bloque	Dato de Bloque	'POST',",[form_params' => \$data]
PUT	JsonResult	Edita datos	Datos de Bloque	Datos de Bloque	'PUT',",[form_params' => \$data]

SolicitudController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de solicitudes	'GET', "solicitud"
GET	JsonResult	Muestra datos	Código solicitud	solicitud	'GET', "solicitud/{id}"
POST	JsonResult	Actualiza datos	Datos de solicitud	Dato de solicitud	'POST',",[form_params' => \$data]
PUT	JsonResult	Edita datos	Datos de solicitud	Datos de solicitud	'PUT',",[form_params' => \$data]
DELETE	JsonResult	Elimina datos	Id solicitud	Mensaje de respuesta	'DELETE', "solicitud /{id}"

USUARIOCONTROLLER.CS

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de usuarios	'GET', "usuario"
GET	JsonResult	Muestra datos	Código usuario	usuario	'GET', "usuario/{id}"

ReservaClienteController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de reservas	'GET', "reserva"
GET	JsonResult	Muestra datos	Código reserva	reserva	'GET', "reserva/{id}"

SolicitudAceptarController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
GET	JsonResult	Muestra datos	-	Listas de solicitudes	'GET', "solicitud"
GET	JsonResult	Elimina solicitud	Código solicitud	solicitud	'GET', "solicitud/{id}"
POST	JsonResult	Acepta solicitud	Datos de solicitud	Dato de solicitud	'POST', "[form_params]=> \$data]"

ReporteController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE	PARAMETROS DE	API
--------	------	-------------	---------------	---------------	-----

			ENTRADA	SALIDA	
GET	JsonResult	Muestra datos	-	Listas de reportes	'GET', "reporte"
GET	JsonResult	Muestra datos	Código reporte	reporte	'GET', "reporte/{id}"

LoginController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
POST	JsonResult	Valida datos	Datos de login	Dato de login	'POST', "[form_params]=> \$data]

AccesoController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
POST	JsonResult	Valida acceso	Identificación tag	Dato de acceso	'POST',';',['form_params'=> \$data]

ArduinoDisponibleController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
POST	JsonResult	Valida la dirección MAC del Arduino	MAC Arduino	Dato del Arduino	'POST',';',['form_params'=> \$data]

ActualizarTagController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
POST	JsonResult	Valida los tags	Datos del tag	Datos del tag	'POST',';',['form_params'=> \$data]

ArduinoEstadoSdController.cs

METODO	TIPO	DESCRIPCIÓN	PARAMETROS DE ENTRADA	PARAMETROS DE SALIDA	API
POST	JsonResult	Valida el estado de la SD	Arduino	Datos Arduino	'POST',';',['form_params'=> \$data]

ANEXO 6. MANUAL DEL CLIENTE**MANUAL DE USUARIO**

**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE
ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR
DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**

DESARROLLADORES:

Luis Eduardo Chichanda García

Oliver Antonio Vera Salazar

ENERO 2022



El presente manual de usuario es un documento elaborado para llevar a cabo el correcto uso del Sistema Multiplataforma.

Realizado por: *Luis Chichanda*.....

Firma

Luis Eduardo Chichanda García
Postulante

Realizado por: *Oliver Vera*.....

Firma

Oliver Antonio Vera Salazar
Postulante

Revisado por:

Firma

Ing. Fernando R. Moreira Moreira, Mgtr.
Tutor Académico

INTRODUCCIÓN

Este sistema se desarrolló con la finalidad gestionar y controlar el acceso a las aulas, luminarias y acondicionadores de aires en el edificio de la carrera de Computación, con el propósito de favorecer la reducción de costos, tiempo en el uso de recursos y administración de la información, dando la posibilidad de interactuar con los responsables de cada tarea.

El presente documento servirá de guía a los usuarios del sistema, describiendo en forma detallada las interfaces de usuario y los procesos que deben realizarse para ejecutar una determinada acción, además las definiciones sobre ciertos términos utilizados a lo largo de esta guía se encuentran al final en el glosario.

OBJETIVO

Brindar ayuda y soporte necesario a los usuarios del sistema web del control de acceso a las aulas, luminarias y acondicionadores de aires del edificio de la carrera de Computación.

ALCANCE

- ✓ Perspectiva general del sistema
- ✓ Detalle de cada funcionalidad
- ✓ Glosario de términos

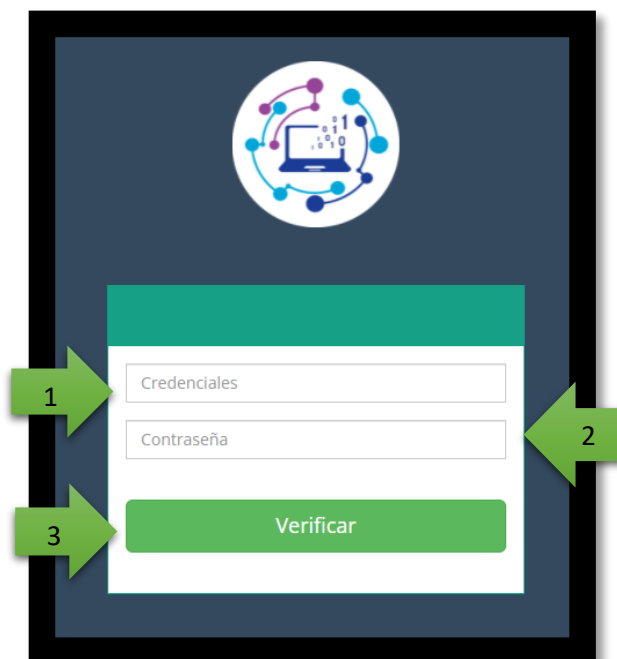
ROLES DE USUARIO

Administrador: tendrá acceso a todos los módulos del sistema.

Usuario: tendrá acceso solo a los módulos de gestión de reserva y gestión de solicitudes.

1. INGRESO AL SISTEMA WEB

Para acceder al sistema web deberá ingresar sus credenciales de usuario.



The screenshot shows a login form with a dark blue background and a circular logo at the top center. The logo features a laptop, binary code, and a network diagram. The form has a green header bar. Below the header, there are three input fields: 'Credenciales', 'Contraseña', and a green 'Verificar' button. Three green arrows with numbers 1, 2, and 3 point to the 'Credenciales' field, the 'Contraseña' field, and the 'Verificar' button, respectively.

Figura 57. Login del sistema



The screenshot shows a role selection form with a dark blue background and the same circular logo as Figure 57. The form has a green header bar. Below the header, there is a dropdown menu labeled 'Selección de Rol' and a green 'Acceder' button. Two green arrows with numbers 4 and 5 point to the dropdown menu and the 'Acceder' button, respectively.

Figura 58. Selección de Rol

INFORMACIÓN:

1. **Cédula:** Ingrese el número de cédula.

2. **Contraseña:** Ingrese su contraseña.
3. **Verificar:** Dar clic en este botón para validar sus credenciales.
4. **Selección de rol:** Dar clic en este botón para escoger su rol (Administrador o Usuario).
5. **Acceder:** Dar clic en este botón para ingresar al sistema.

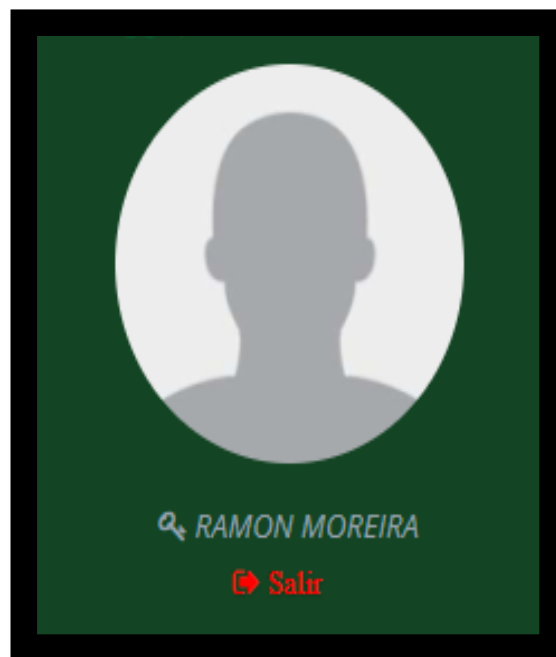


Figura 59. Inicio rol administrador

2. PAGINA PRINCIPAL

Esta es la pantalla de Inicio en donde se puede observar un Dashboard de las diferentes aulas y laboratorios con sus estados; y en la parte lateral izquierda estará el menú de opciones.

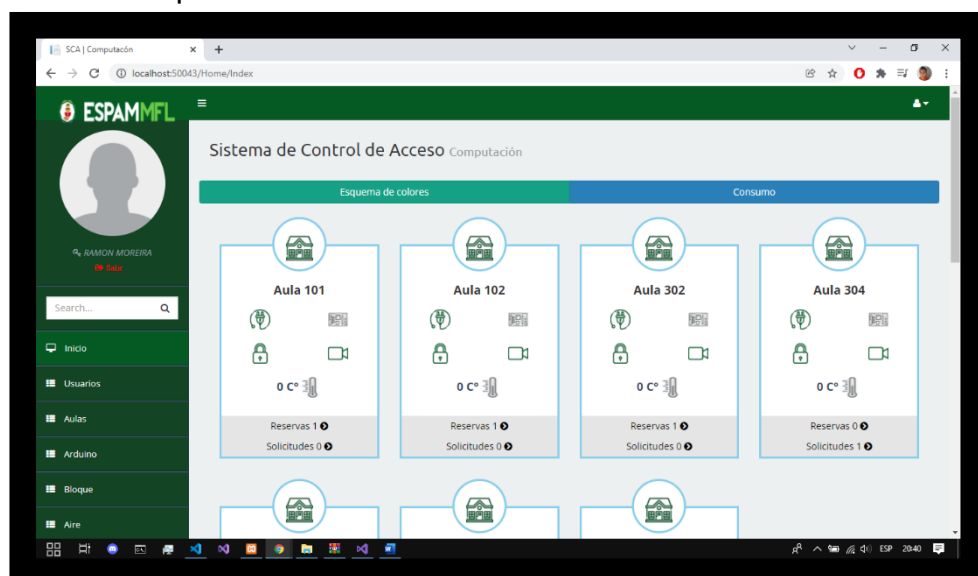


Figura 60. Página principal del sistema del administrador



INFORMACIÓN:

- 1. Esquema de colores:** Diferenciar los diferentes estados mediante colores (aulas, laboratorios, consumo, temperatura, acceso, cámara).
- 2. Consumo:** Consumo energético de las aulas y laboratorios.

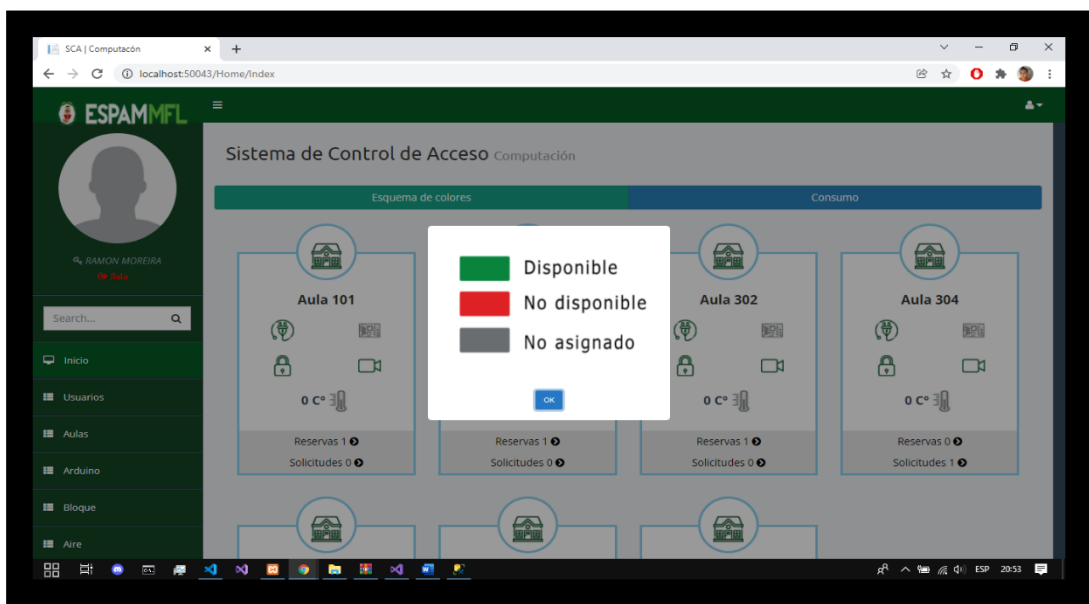


Figura 61. Esquema de colores

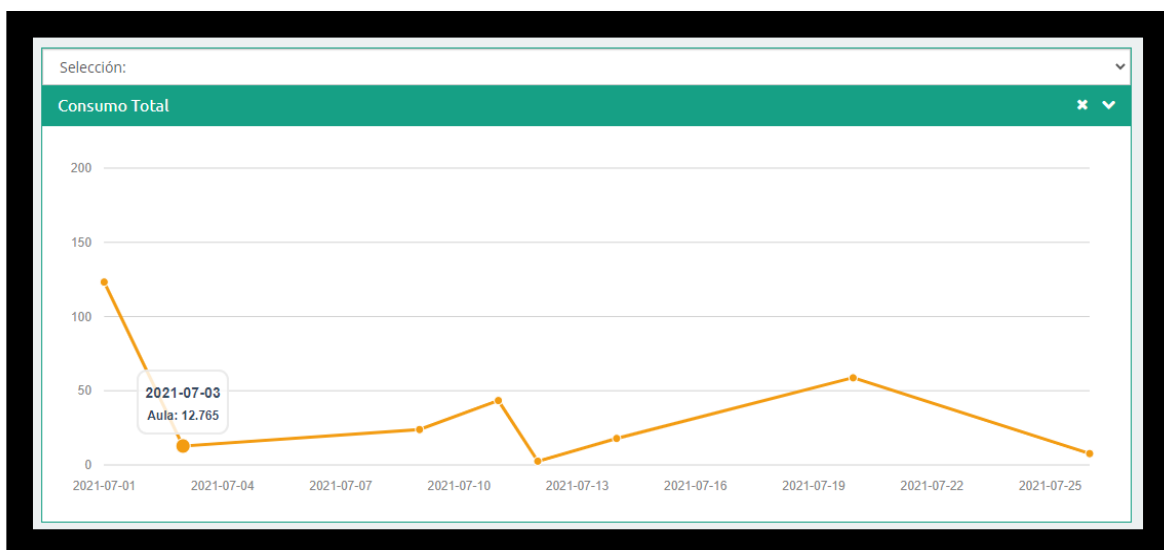


Figura 62. Consumo de energía por mes

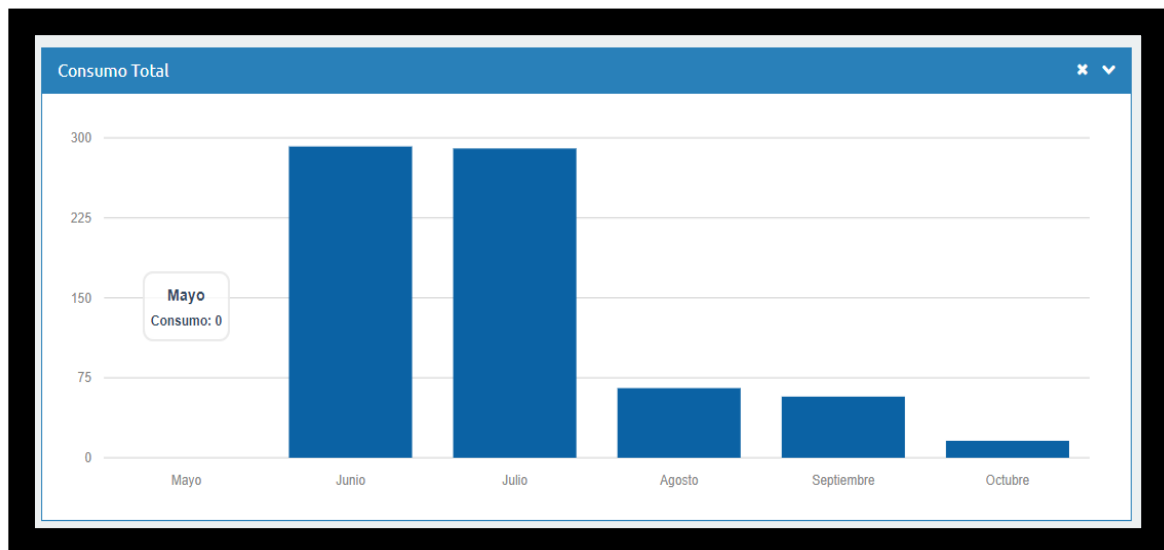


Figura 63. Consumo de energía total

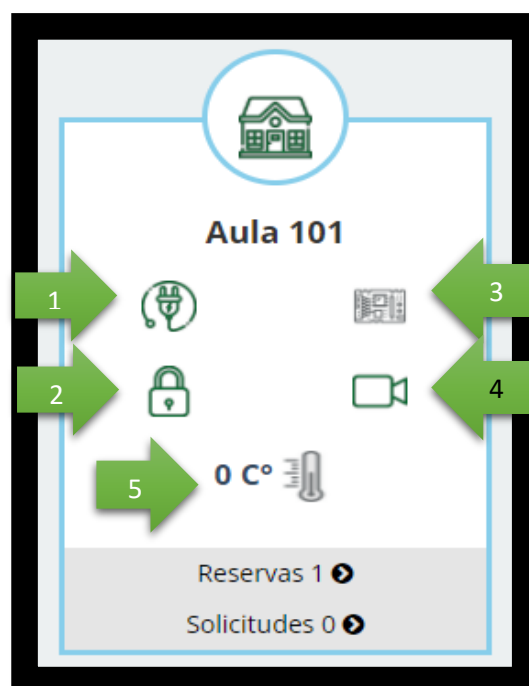


Figura 64. Visualización del aula 101

INFORMACIÓN:

1. Dar clic en este botón para ver el consumo del aula.
2. Dar clic en este botón para abrir el aula.
3. Se visualiza el estado del aula acorde al esquema de colores.

4. Dar clic en este botón para visualizar el aula por medio de la cámara.
5. Dar clic en este botón para visualizar la temperatura del acondicionador de aire.

3. MÓDULOS DEL SISTEMA(ADMINISTRADOR)

3.1. GESTIÓN USUARIOS

Cedula	Nombre	Apellido	Correo	Identificación Tag
0921757282	MARIA	PINCAJ CANTOS	maria.pincay@espam.edu.ec	ver Añadir
1301932156	DANIEL	MERA MARTINEZ	dmera@espam.edu.ec	ver Añadir
1306246651	LUIS	CEDEÑO VALAREZO	lcedeno@espam.edu.ec	ver Añadir
1306391614	RICARDO	VELEZ VALAREZO	rvelezvalarezo@espam.edu.ec	ver Añadir
1307723286	GUSTAVO	MOLINA GARZON	gmolina@espam.edu.ec	ver Añadir
1308542099	KELVIN	ROSADO CUSME	krosado@espam.edu.ec	ver Añadir
1308555190	MARYURY	ZAMORA CUSME	mzamora@espam.edu.ec	ver Añadir
1308648565	ANGEL	VELEZ MERO	avelez@espam.edu.ec	ver Añadir
1308755238	JAVIER	LOPEZ ZAMBRANO	jlopez@espam.edu.ec	ver Añadir
1309609988	RAMON	MOREIRA PICO	jmoreira@espam.edu.ec	ver Añadir

Showing 1 to 10 of 18 entries

Figura 65. Lista de usuarios

INFORMACIÓN:

1. **Botón ver:** Dar clic en este botón para ver las identificaciones(tags).
2. **Botón Añadir:** Dar clic en este botón para añadir un nuevo tag.

3.2. GESTIÓN AULA

Crear Aula

1 Codigo Aula

2 Tipo Aula Selección:

3 Bloque Selección:

4 Arduino Selección:

5 Aire Selección:

6 Ip Camara

7 Descripción

8

Figura 66. Interfaz de creación de aula

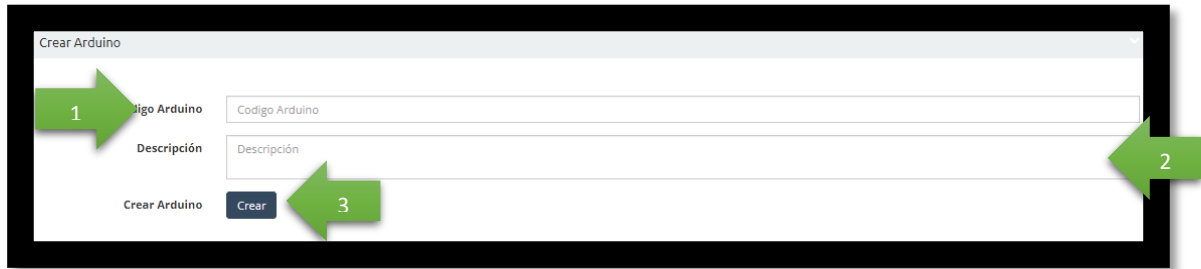
Código	Nombre	TipoAula	Bloque	IpCamara	Arduino	Acondicionador	Estado	Editar	Eliminar
Aula 101	Aula 101	Aula	1	185.10.80.33:8082	240A-C4E8-D7-58		Activo	Editar	Eliminar
Lab. 101	Laboratorio de Electrónica	Laboratorio	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 102	Laboratorio de Programación y Análisis	Laboratorio	1	185.10.80.33:8082	240A-C4E8-4B-70		Activo	Editar	Eliminar
Aula 102	Aula 102	Aula	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 103	Laboratorio de Lenguaje, Programación y Base de Datos	Laboratorio	1	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 104	Laboratorio de Física	Laboratorio	2	185.10.80.33:8082	240A-C4E8-4B-58		Activo	Editar	Eliminar
Lab. 203	Laboratorio de Diseño Gráfico2	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 204	Laboratorio de Redes	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Lab. 205	Laboratorio de Estructura y Análisis de Algoritmo	Laboratorio	2	185.10.80.33:8082			Activo	Editar	Eliminar
Aula 302	Aula 302	Aula	3	185.10.80.33:8082			Activo	Editar	Eliminar

Figura 67. Listado de Aula y laboratorios

INFORMACIÓN:

1. **Código aula:** Ingrese el nuevo código de aula.
2. **Tipo de aula:** Seleccione el nuevo tipo de aula (Aula o Laboratorio).
3. **Bloque:** Seleccione el bloque de la nueva aula.
4. **Arduino:** Seleccione el Arduino creado.
5. **Aire:** Seleccione el acondicionador de aire creado.
6. **IP Cámara:** Ingrese la dirección IP de la cámara.
7. **Descripción:** Ingrese breve descripción de la nueva aula a crear.
8. **Botón Crear:** Dar clic en este botón para crear una nueva aula.
9. **Botón Editar:** Dar clic en este botón para editar los parámetros del aula o laboratorio.
10. **Botón Eliminar:** Dar clic en este botón para eliminar el aula o laboratorio.

3.3. GESTIÓN ARDUINO



Crear Arduino

1 Código Arduino

Descripción

Crear Arduino

2

3

The image shows a web form titled "Crear Arduino". It contains three input fields: "Codigo Arduino", "Descripción", and "Crear Arduino". A dark blue button labeled "Crear" is positioned below the "Crear Arduino" field. Three green arrows with numbers 1, 2, and 3 point to the "Codigo Arduino" field, the "Crear" button, and the right side of the form respectively.

Figura 68. Interfaz creación Arduino

Codigo	Estado	Descripción	Editar	Eliminar
24:0A:C4:58:D7:E8	No Activo	Programacion	Editar	Eliminar
24:0A:C4:EA:4B:58	No Activo	Fisica 1	Editar	Eliminar
24:0A:C4:EA:4B:70	No Activo	tu	Editar	Eliminar

Figura 69. Listado de Arduino

INFORMACIÓN:

- Código Arduino:** Ingrese el nuevo código del Arduino.
- Descripción:** Ingrese descripción del Arduino.
- Botón Crear:** Dar clic en este botón para crear un nuevo Arduino.
- Botón Editar:** Dar clic en este botón para editar los parámetros del Arduino.
- Botón Eliminar:** Dar clic en este botón para eliminar el Arduino.

3.4. GESTIÓN BLOQUE

Crear Bloque

1 Codigo Bloque

2 Descripción

3 Crear

Figura 71. Interfaz creación de bloque

Codigo	Estado	Descripción	Editar
1	Activo	1	Editar
2	Activo	2	Editar
3	Activo	3	Editar

Figura 70. Listado de bloque

INFORMACIÓN:

1. **Código Bloque:** Ingrese el nuevo código del bloque.
2. **Descripción:** Ingrese descripción del Bloque.
3. **Botón Crear:** Dar clic en este botón para crear un nuevo bloque.
4. **Botón Editar:** Dar clic en este botón para editar los parámetros del Bloque.

3.5. GESTIÓN AIRE

The screenshot shows a form titled 'Crear Aire'. It contains two input fields: 'Codigo Aire' and 'Descripción'. Below the fields is a 'Crear Aire' label and a 'Crear' button. Three green arrows with numbers point to these elements: arrow 1 points to the 'Codigo Aire' field, arrow 2 points to the 'Descripción' field, and arrow 3 points to the 'Crear' button.

Figura 73. Interfaz gestión aire

The screenshot shows a table titled 'Aire' with a search bar and a dropdown for 'records per page' set to 10. The table has columns for 'Codigo', 'Estado', 'Temperatura', 'Descripción', 'Editar', and 'Eliminar'. A single row is displayed with the following data: Codigo: 1, Estado: No Activo, Temperatura: 29,55, Descripción: Aire 1. Two green arrows with numbers point to the 'Editar' and 'Eliminar' buttons in the row: arrow 4 points to the 'Editar' button and arrow 5 points to the 'Eliminar' button.

Codigo	Estado	Temperatura	Descripción	Editar	Eliminar
1	No Activo	29,55	Aire 1	Editar	Eliminar

Figura 72. Listado de aire

INFORMACIÓN:

1. **Código Aire:** Ingrese el nuevo código del Aire.
2. **Descripción:** Ingrese descripción del Aire.
3. **Botón Crear:** Dar clic en este botón para crear un nuevo Aire.
4. **Botón Editar:** Dar clic en este botón para editar los parámetros del Aire.
5. **Botón Eliminar:** Dar clic en este botón para eliminar el Aire.

3.6. GESTIÓN SOLICITUDES

Cedula	Nombre : Apellido	Aula	Dia	Hora Inicio	Hora Final	Aceptar	Rechazar
1308542099	KELVIN ROSADO	Aula 304	Martes	16:34:00	17:34:00	Aceptar	Rechazar
1308648565	ANGEL VELEZ	Lab. 102	Viernes	00:00:00	18:38:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Martes	13:16:00	15:16:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 102	Martes	12:18:00	12:19:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 102	Miércoles	12:20:00	12:21:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Lunes	12:21:00	12:22:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Lab. 104	Lunes	12:32:00	12:33:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Aula 101	Lunes	12:33:00	12:33:00	Aceptar	Rechazar
1309609988	RAMON MOREIRA	Aula 101	Lunes	12:33:00	12:33:00	Aceptar	Rechazar

Showing 1 to 9 of 9 entries

← Previous 1 Next →

Figura 74. Interfaz gestión solicitudes

INFORMACIÓN:

- Botón Aceptar:** Dar clic en este botón para aceptar la solicitud.
- Botón Rechazar:** Dar clic en este botón para rechazar la solicitud.

3.7. GESTIÓN RESERVA

Crear Reserva

1 Cedula

2 Validar

3

4 Hora inicio

5

6 Dia Reserva

7 Crear

Figura 75. Interfaz gestión creación reserva

Cedula	Nombre : Apellido	Aula	Dia	Hora Inicio	Hora Final	Editar	Eliminar
0921757282	MARIA PINCAY	Aula 302	Martes	07:30:00	08:30:00	Editar	Eliminar
1301932156	DANIEL MERA	Aula 302	Lunes	07:30:00	08:30:00	Editar	Eliminar
1306246651	LUIS CEDEÑO	Aula 304	Lunes	10:00:00	11:00:00	Editar	Eliminar
1306391614	RICARDO VELEZ	Aula 101	Lunes	11:00:00	12:00:00	Editar	Eliminar
1306391614	RICARDO VELEZ	Lab. 102	Miércoles	16:30:00	18:30:00	Editar	Eliminar
1307723286	GUSTAVO MOLINA	Aula 101	Lunes	07:30:00	08:30:00	Editar	Eliminar
1308542099	KELVIN ROSADO	Lab. 104	Viernes	10:00:00	11:00:00	Editar	Eliminar
1308542099	KELVIN ROSADO	Lab. 204	Miércoles	21:42:00	22:42:00	Editar	Eliminar
1308542099	KELVIN ROSADO	Aula 102	Lunes	16:31:00	17:31:00	Editar	Eliminar
1308555190	MARYURY ZAMORA	Lab. 102	Lunes	19:15:00	20:15:00	Editar	Eliminar

Figura 76. Listado de reservas

INFORMACIÓN:

1. **Cédula:** Ingrese número de cedula.
2. **Botón Validar:** Dar clic en este botón para validar cédula.
3. **Aula:** Seleccione aula o laboratorio a reservar.
4. **Hora Inicio:** Ingrese la hora de inicio de la reserva.
5. **Hora Final:** Ingrese la hora de finalización de la reserva.
6. **Dia Reserva:** Seleccione el día a reservar.
7. **Botón Crear:** Dar clic en este botón para crear una nueva reserva.
8. **Botón Editar:** Dar clic en este botón para editar los parámetros de la reserva.
9. **Botón Eliminar:** Dar clic en este botón para eliminar la reserva.

4. PAGINA PRINCIPAL(USUARIOS)



Figura 77. Página principal del sistema de los usuarios

INFORMACIÓN:

1. **Esquema de colores:** Diferenciar los diferentes estados mediante colores (aulas, laboratorios, consumo, temperatura, acceso, cámara).

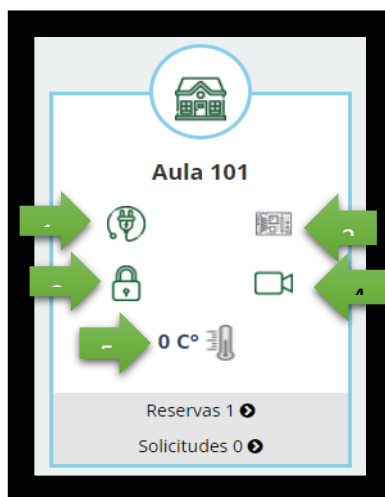


Figura 78. Visualización del aula 101

INFORMACIÓN:

1. Dar clic en este botón para ver el consumo del aula.
2. Dar clic en este botón para abrir el aula.
3. Se visualiza el estado del aula acorde al esquema de colores.
4. Dar clic en este botón para visualizar la cámara dentro del aula.
5. Dar clic en este botón para visualizar la temperatura del acondicionador de aire.

5. MÓDULOS DEL SISTEMA(USUARIOS)

5.1. GESTIÓN RESERVA CLIENTE

Reservas					
Aula	Día	Hora Inicio	Hora Fin	Frecuentemente	
Lab. 204	Miercoles	11:00:00	12:00:00	No	
Aula 101	Lunes	12:33:00	12:33:00	No	
Aula 101	Lunes	12:33:00	12:33:00	No	
Aula 101	Lunes	12:33:00	12:33:00	No	
Lab. 101	Jueves	19:42:00	22:45:00	No	

Figura 79. Interfaz reserva cliente

5.2. GESTIÓN SOLICITUD

Crear Solicitud

1 Aula Selección: [dropdown]

2 Hora inicio [time picker]

3 Hora final [time picker]

4 Recurrencia

5 Fecha Solicitud dd/mm/aaaa [calendar]

6 Descripción Descripción [text area]

7 Crear Solicitud [Crear button]

Figura 80. Interfaz gestión solicitud

Aula	Día	Hora Inicio	Hora Final	Descripción	Recurrencia	Editar	Eliminar
Lab. 104	Jueves	13:16:00	15:16:00	Tarea	False	Editar	Eliminar
Lab. 102	Jueves	12:18:00	12:19:00	Tarea	False	Editar	Eliminar
Lab. 102	Miércoles	12:20:00	12:21:00	Tarea	True	Editar	Eliminar
Lab. 104	Lunes	12:21:00	12:22:00	Tarea	False	Editar	Eliminar
Lab. 104	Lunes	12:32:00	12:33:00	Tarea	False	Editar	Eliminar
Aula 101	Lunes	12:33:00	12:33:00	Tarea	False	Editar	Eliminar
Aula 101	Lunes	12:33:00	12:33:00	Tarea	False	Editar	Eliminar

Figura 81. Listado de solicitudes

INFORMACIÓN:

- 1. Aula:** Seleccione el aula o laboratorio a solicitar
- 2. Hora inicio:** Ingrese la hora de inicio de la solicitud.
- 3. Hora final:** Ingrese la hora de finalización de la solicitud.
- 4. Recurrencia:** Activar o desactivar la recurrencia.
- 5. Fecha de solicitud:** Ingrese la fecha de la solicitud.
- 6. Descripción:** Ingrese descripción de la solicitud.

7. **Botón Crear:** Dar clic en este botón para crear una nueva solicitud.
8. **Botón Editar:** Dar clic en este botón para editar los parámetros de la solicitud.
9. **Botón Eliminar:** Dar clic en este botón para eliminar la solicitud.

6. INGRESO APP MÓVIL(ADMINISTRADOR)

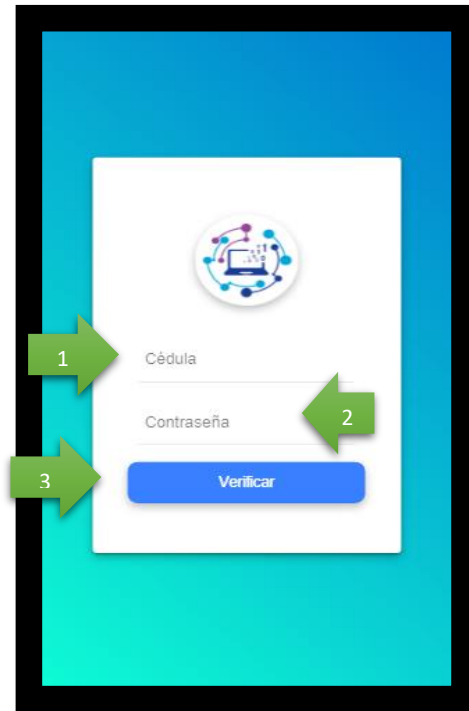


Figura 82. Login app móvil

INFORMACIÓN:

1. **Cédula:** Ingrese el número de cédula.
2. **Contraseña:** Ingrese su contraseña.
3. **Verificar:** Dar clic en este botón para validar sus credenciales.

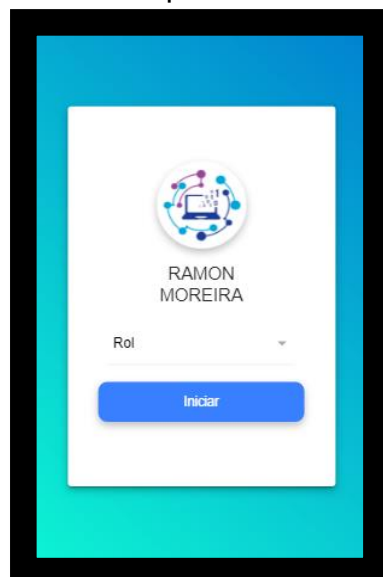


Figura 83. Elegir rol

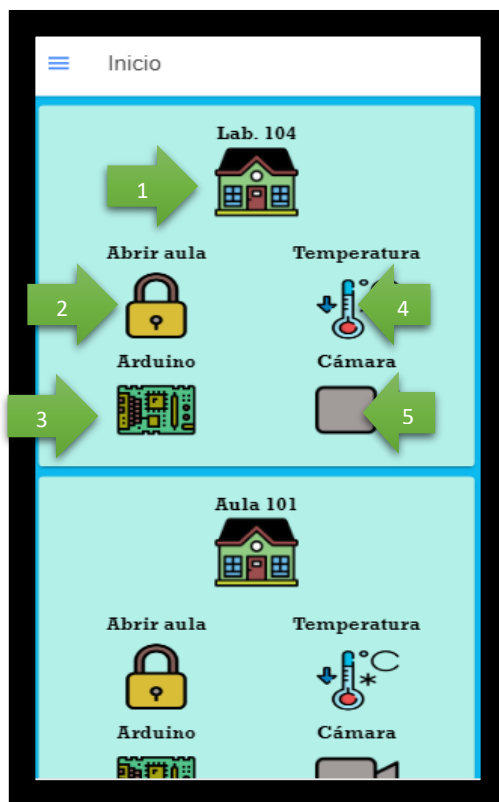


Figura 84. Ventana principal del administrador

INFORMACIÓN:

1. Dar clic en este botón para ver el consumo del aula o laboratorio.
2. Dar clic en este botón para abrir el aula.
3. Se visualiza el estado del aula acorde al esquema de colores.
4. Dar clic en este botón para visualizar la temperatura del acondicionador de aire.
5. Dar clic en este botón para visualizar el aula por medio de la cámara.

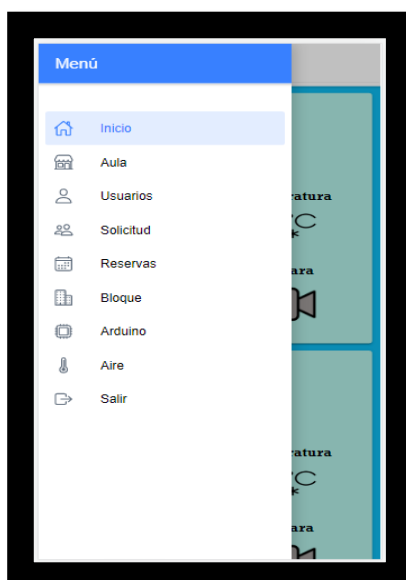


Figura 85. Menú de opciones del administrador

6.1. GESTIÓN AULA

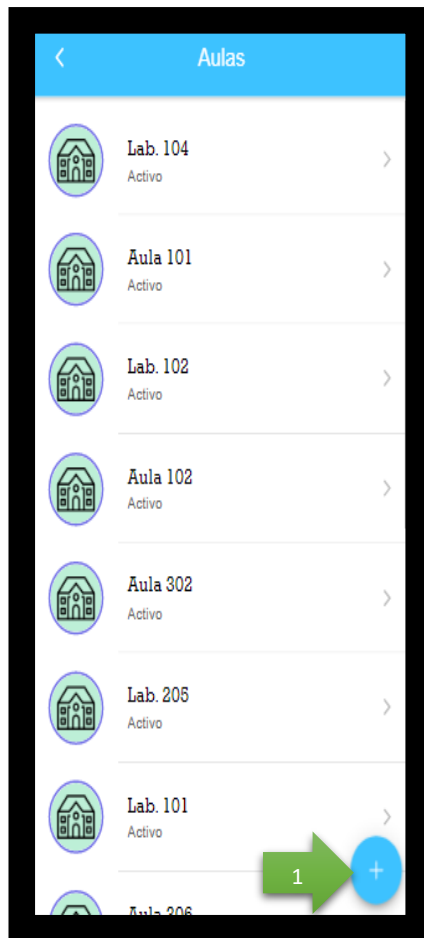


Figura 86. Interfaz aula



Figura 87. Crear aula

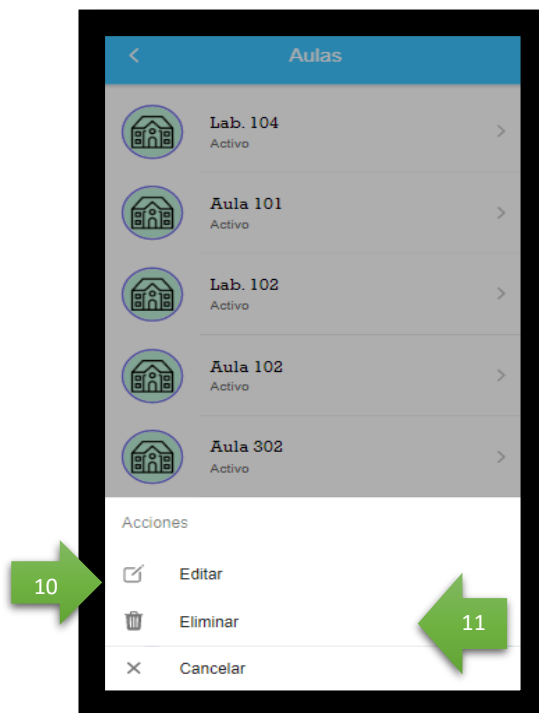


Figura 88. Listado de aulas y laboratorios

INFORMACIÓN:

1. **Botón añadir:** Dar clic en este botón para añadir una nueva aula.
2. **Código aula:** Ingrese el nuevo código de aula.
3. **Tipo de aula:** Seleccione el nuevo tipo de aula (Aula o Laboratorio).
4. **Bloque:** Seleccione el bloque de la nueva aula.
5. **Arduino:** Seleccione el Arduino creado.
6. **Aire:** Seleccione el acondicionador de aire creado.
7. **IP Cámara:** Ingrese la dirección IP de la cámara.
8. **Descripción:** Ingrese breve descripción de la nueva aula a crear.
9. **Botón Crear:** Dar clic en este botón para crear una nueva aula.
10. **Botón Editar:** Dar clic en este botón para editar los parámetros del aula.
11. **Botón Eliminar:** Dar clic en este botón para eliminar el aula.

6.2. GESTIÓN USUARIOS

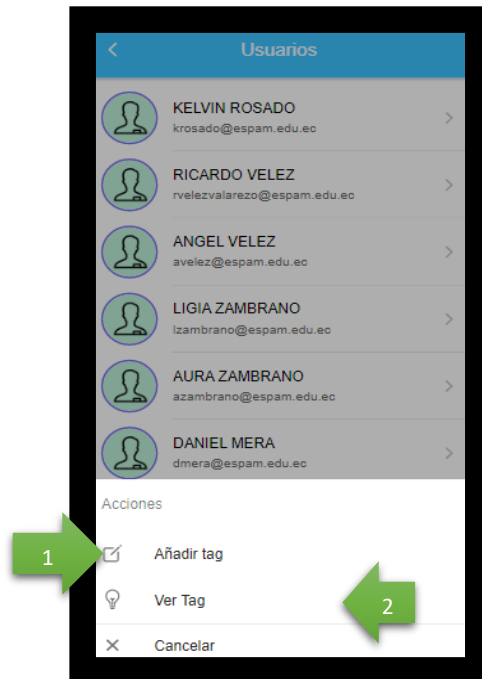


Figura 89. Interfaz gestión usuarios

INFORMACIÓN:

1. **Botón Añadir tag:** Dar clic en este botón para añadir un nuevo tag.
2. **Botón ver Tag:** Dar clic en este botón para ver las identificaciones(tags).

6.3. GESTIÓN SOLICITUD

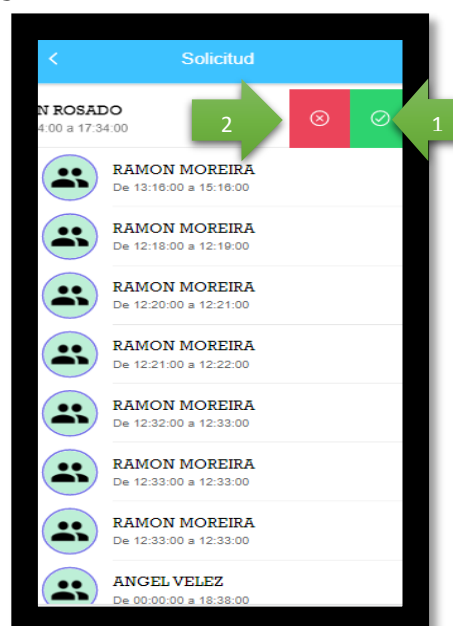
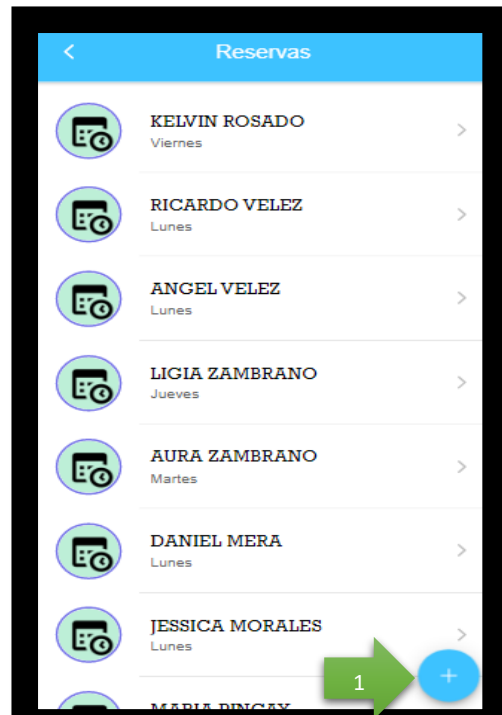
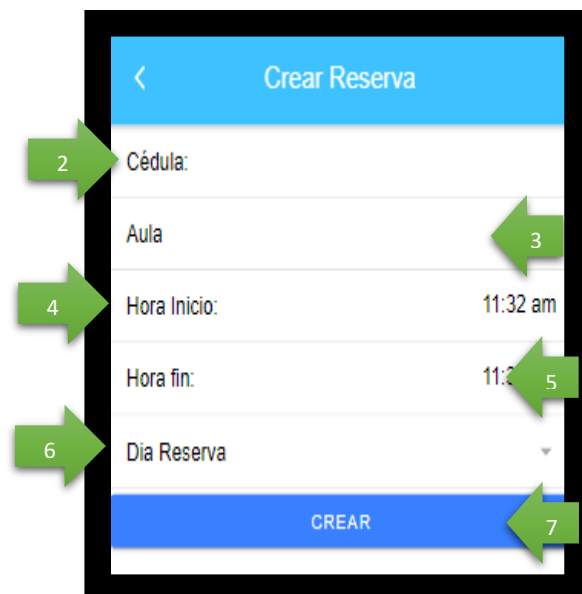


Figura 90. Interfaz gestión solicitud

INFORMACIÓN:

1. **Botón Aceptar:** Dar clic en este botón para aceptar la solicitud.
2. **Botón Rechazar:** Dar clic en este botón para rechazar la solicitud.

6.4. GESTIÓN RESERVAS**Figura 91.** Interfaz reservas**Figura 92.** Creación de reservas

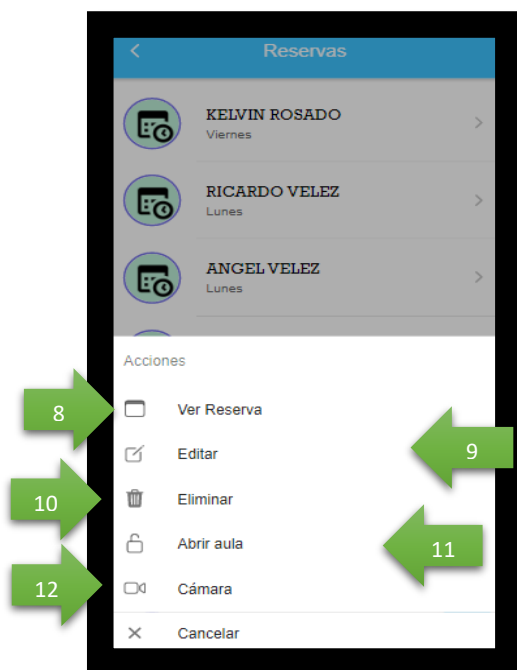


Figura 93. Listado de reservas

INFORMACIÓN:

1. **Añadir reserva:** Ingrese número de cedula.
2. **Cédula:** Ingrese número de cedula.
3. **Aula:** Seleccione aula o laboratorio a reservar.
4. **Hora Inicio:** Ingrese la hora de inicio de la reserva.
5. **Hora Final:** Ingrese la hora de finalización de la reserva.
6. **Dia Reserva:** Seleccione el día a reservar.
7. **Botón Crear:** Dar clic en este botón para crear una nueva reserva.
8. **Botón ver reserva:** Dar clic en este botón para ver el estado de la reserva.
9. **Botón Editar:** Dar clic en este botón para editar los parámetros de la reserva.
10. **Botón Eliminar:** Dar clic en este botón para eliminar la reserva.
11. **Abrir aula:** Dar clic en este botón para abrir el aula.
12. **Cámara:** Dar clic en este botón para visualizar el aula por medio de la cámara.

6.5. GESTIÓN BLOQUE

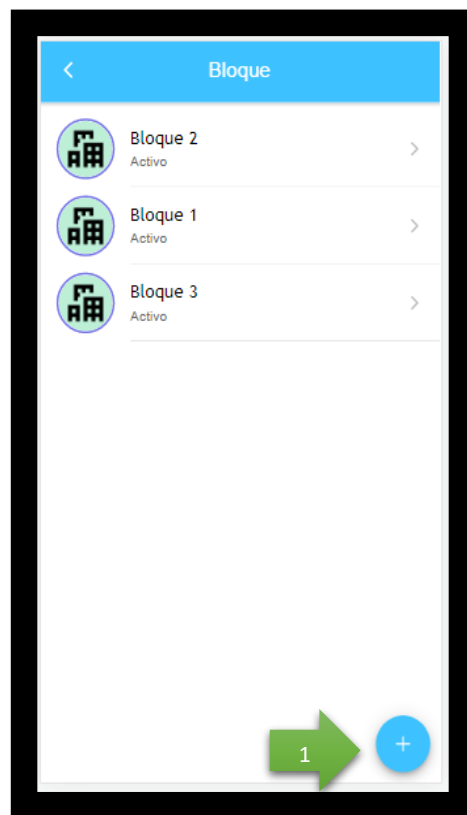


Figura 94. Gestión bloque



Figura 95. Crear bloque

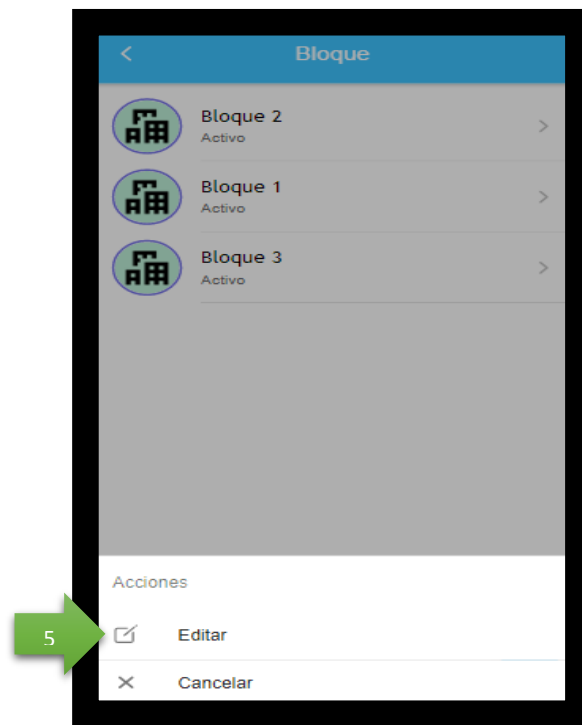
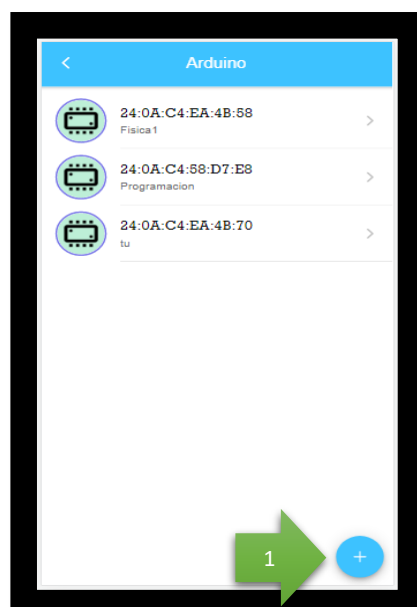


Figura 96. Listado de bloques

INFORMACIÓN:

1. **Botón añadir Bloque:** Dar clic en este botón para añadir un nuevo Bloque.
2. **Código Bloque:** Ingrese el nuevo código del bloque.
3. **Descripción:** Ingrese descripción del Bloque.
4. **Botón Guardar:** Dar clic en este botón para crear un nuevo Bloque.
5. **Botón Editar:** Dar clic en este botón para editar los parámetros del Bloque.

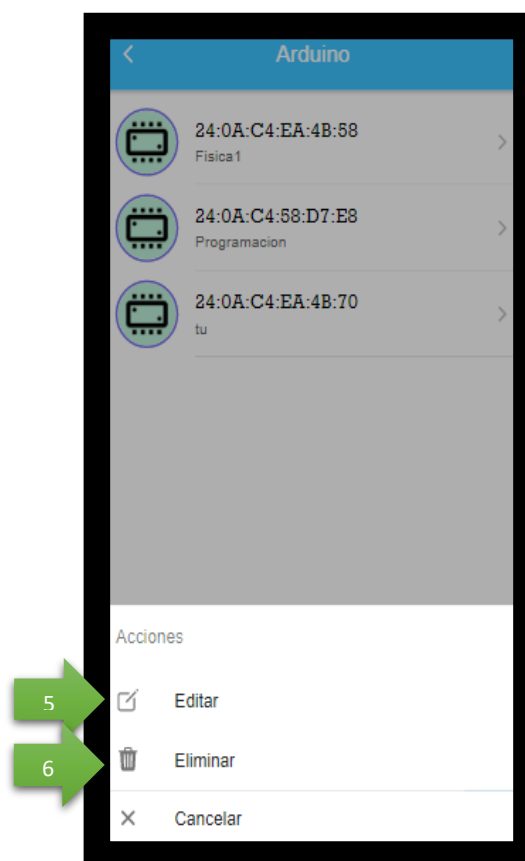
6.6. GESTIÓN BLOQUE





The screenshot shows a mobile application interface for creating a new Arduino. At the top, there is a blue header with a back arrow and the text "Crear Arduino". Below the header, there are two text input fields: "Código:" and "Descripción:". At the bottom of the form, there is a prominent blue button labeled "GUARDAR". Three green arrows with numbers 2, 3, and 4 point to the "Código:" field, the "Descripción:" field, and the "GUARDAR" button, respectively.

Figura 98. Crear Arduino



The screenshot shows a mobile application interface displaying a list of Arduinos. The header is blue with a back arrow and the text "Arduino". The list contains three items, each with an Arduino icon, a MAC address, and a name: "24:0A:C4:EA:4B:68 Fisica1", "24:0A:C4:68:D7:E8 Programacion", and "24:0A:C4:EA:4B:70 tu". At the bottom, there is a white section titled "Acciones" containing three buttons: "Editar" (with a pencil icon), "Eliminar" (with a trash icon), and "Cancelar" (with an 'X' icon). Two green arrows with numbers 5 and 6 point to the "Editar" and "Eliminar" buttons, respectively.

Figura 99. Listado de Arduino

INFORMACIÓN:

1. **Botón añadir Arduino:** Dar clic en este botón para añadir un nuevo Arduino.
2. **Código:** Ingrese el nuevo código del Arduino.

3. **Descripción:** Ingrese descripción del Arduino.
4. **Botón Guardar:** Dar clic en este botón para crear un nuevo Arduino.
5. **Botón Editar:** Dar clic en este botón para editar los parámetros del Arduino.
6. **Botón Eliminar:** Dar clic en este botón para eliminar el Arduino.

6.7. GESTIÓN AIRE

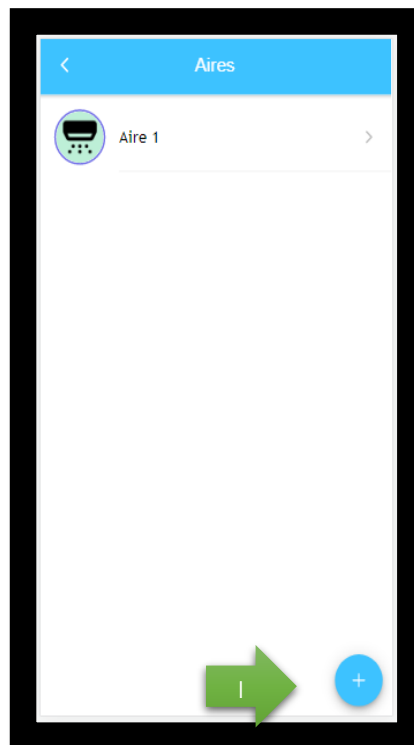


Figura 100. Gestión Aire



Figura 101. Crear Aire

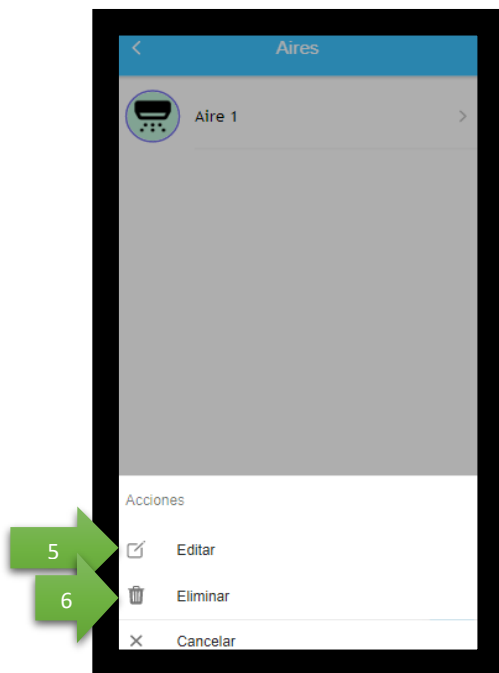


Figura 102. Listado de aire

INFORMACIÓN:

1. **Botón añadir aire:** Dar clic en este botón para añadir un nuevo aire.
2. **Código:** Ingrese el nuevo código del aire.
3. **Descripción:** Ingrese descripción del aire.
4. **Botón Guardar:** Dar clic en este botón para crear un nuevo aire.
5. **Botón Editar:** Dar clic en este botón para editar los parámetros del aire
6. **Botón Eliminar:** Dar clic en este botón para eliminar el aire.

7. PAGINA PRINCIPAL(USUARIO)

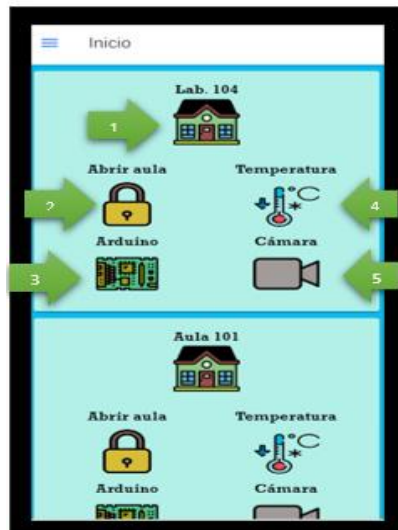


Figura 103. Ventana principal usuarios

INFORMACIÓN:

1. Dar clic en este botón para ver el consumo del aula o laboratorio.
2. Dar clic en este botón para abrir el aula.
3. Se visualiza el estado del aula acorde al esquema de colores.
4. Dar clic en este botón para visualizar la temperatura del acondicionador de aire.
5. Dar clic en este botón para visualizar el aula por medio de la cámara.

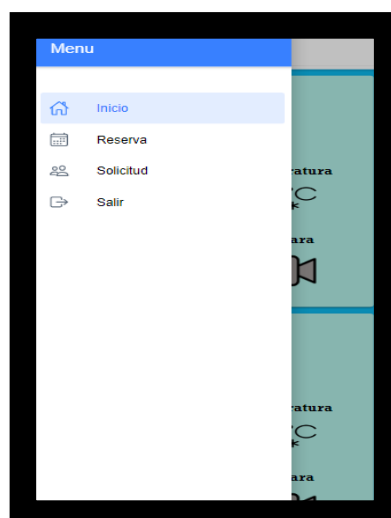


Figura 104. Menú de opciones usuario

7.1. GESTIÓN RESERVA USUARIO

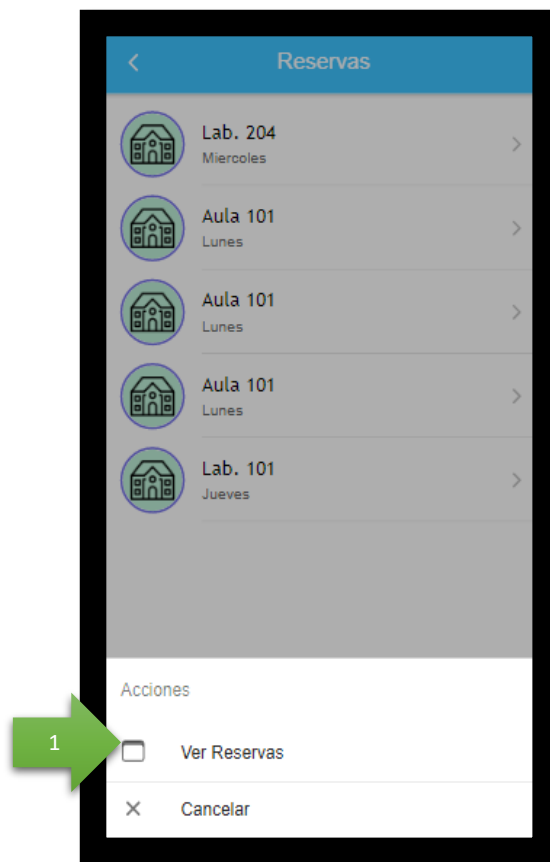


Figura 105. Gestión reserva

INFORMACIÓN:

1. **Ver Reservas:** Dar clic en este botón para ver reservas.

7.2. GESTIÓN SOLICITUD USUARIOS

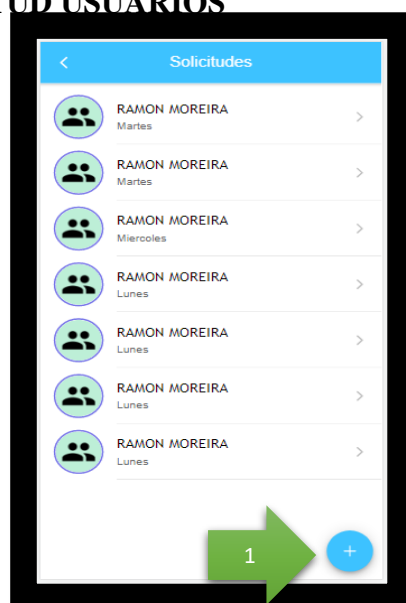


Figura 106. Gestión solicitud



Figura 107. Crear solicitud

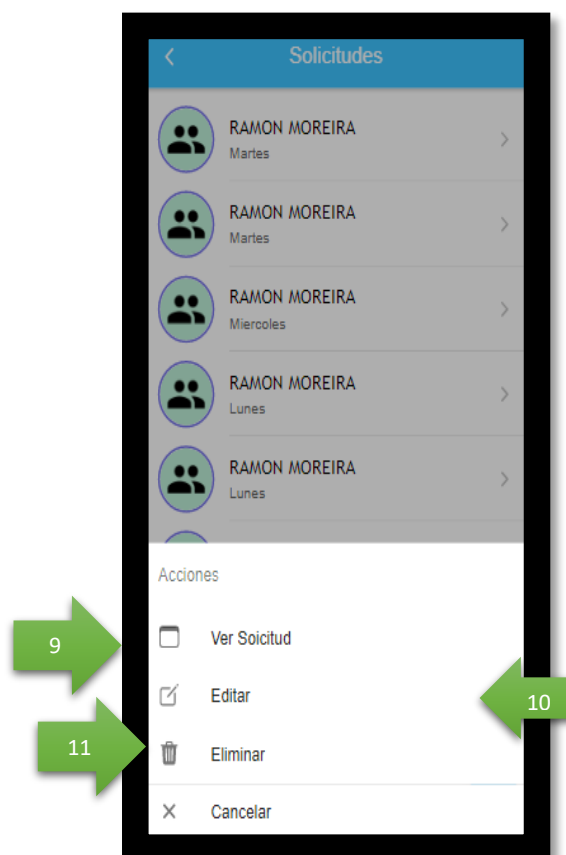


Figura 108. Listado solicitudes

INFORMACIÓN:

1. **Botón añadir:** Dar clic en este botón para añadir una nueva solicitud.
2. **Aula:** Seleccione el aula o laboratorio a solicitar.
3. **Hora inicio:** Ingrese la hora de inicio de la solicitud.

4. **Hora fin:** Ingrese la hora de finalización de la solicitud.
5. **Recurrencia:** Activar o desactivar la recurrencia.
6. **Fecha de solicitud:** Ingrese la fecha de la solicitud.
7. **Descripción:** Ingrese descripción de la solicitud.
8. **Botón Crear:** Dar clic en este botón para crear una nueva solicitud.
9. **Ver Solicitud:** Dar clic en este botón para ver la solicitud.
10. **Botón Editar:** Dar clic en este botón para editar los parámetros de la solicitud.
11. **Botón Eliminar:** Dar clic en este botón para eliminar la solicitud.

ANEXO 7. CERTIFICADO DE CUMPLIMIENTO



Calceta, 05 de enero de 2022

Ingeniero
Joffre Moreira Pico, MGS.
Director Carrera de Computación - ESPAM MFL
En su despacho. -

Asunto: CARTA AVAL

En mi calidad de encargado de la UDIV de Infraestructuras, comunico que una vez verificada la instalación y funcionamiento del software de la tesis titulada: "**SISTEMA MULTIPLATAFORMA PARA EL CONTROL DE ACCESO A LAS AULAS, LUMINARIAS Y ACONDICIONADOR DE AIRES DEL EDIFICIO DE LA CARRERA DE COMPUTACIÓN**", doy constancia de la entera satisfacción al cumplimiento de los requerimientos técnicos solicitados por esta unidad, por tal motivo cumpla con el proceso de la emisión del aval correspondiente a los estudiantes: **OLIVER ANTONIO VERA SALAZAR** y **LUIS EDUARDO CHICHANDA GARCIA**, destacando sus excelentes competencias y colaboración en el desarrollo de este proyecto de sistematización de experiencia.

Para los fines legales pertinentes, me suscribo a usted.

Atentamente,

Firmado digitalmente
por JAVIER HERNAN
LOPEZ ZAMBRANO

Mgtr. Javier Hernán López Zambrano
Encargado de la UDIV de Infraestructuras
Carrera de Computación
ESPAM MFL
Correo: jlopez@espam.edu.ec