

**UNIVERSIDAD AUTÓNOMA DE TAMAULIPAS**  
FACULTAD DE INGENIERÍA "ARTURO NARRO SILLER"



Tesis

**DESARROLLO DE UN SISTEMA PARA EL PAGO DE  
TRANSPORTE PUBLICO MEDIANTE NFC**

Que para obtener el título de  
**Ingeniero en Sistemas Computacionales**

Presenta

*Gustavo de Jesús Márquez Estrada*

Director

*Dr. Alejandro Humberto García Ruiz*

Co-Director

*Dr. Salvador Ibarra Martínez*

Asesor

*M. C. Emilio Castán Rocha*



**UAT**  
Universidad Autónoma  
de Tamaulipas



Facultad de Ingeniería  
Tampico  
Secretaría Académica

Tampico, Tamaulipas, a

15 de agosto de 2022

Asunto: **Aceptación de tema de investigación**

Nº de Oficio 33-SAC- 048 -2022

**Márquez Estrada Gustavo de Jesús**

**A l u m n o**

Facultad de Ingeniería "Arturo Narro Siller"

**P r e s e n t e**

Por medio de la presente me permito informarle que ha sido revisado y aceptado por la academia de: **Arquitectura de computadoras**

su tema tentativo de investigación denominado:

***Desarrollo de un sistema para el pago de transporte público mediante NFC***

con fines de obtención del título de licenciatura de:

**Ingeniero en Sistemas Computacionales**

y le han sido asignados como Asesores, los siguientes profesores:

Director: **Dr. Alejandro Humberto García Ruiz**

Co-Director: **Dr. Salvador Ibarra Martínez**

Asesor: **M.C. Emilio Castan Rocha**

Usted dispone hasta de un año para la realización de este trabajo, a partir de esta fecha de aceptación del tema y deberá apegarse al *Reglamento de Recepción Profesional* vigente y a la *Guía para la Presentación de Anteproyectos e Informes de Investigación* de esta Facultad. De requerir más tiempo del especificado, su Director de tesis deberá gestionar este recurso. De no continuar con su investigación, el trabajo pasará a ser propiedad de su Director de tesis.

Sin más, me es grato quedar de usted.

Atentamente

**FI** Facultad de Ingeniería  
Arturo Narro Siller  
SECRETARÍA ACADÉMICA

"Verdad, Belleza, Probidad"

*Dra. Esperanza Martínez Cano*  
Secretaria Académica

c. p. Archivo

\*EMC/emc

FACULTAD DE INGENIERÍA TAMPICO  
Centro Universitario Sur (833) 241-2000, ext. 3451  
Tampico  
C.P. 89109.

Tampico, Tamaulipas a 25 de septiembre de 2023

**DRA. ESPERANAZA MARTÍNEZ CANO**  
SECRETARIA ACADÉMICA  
FACULTAD DE INGENIERÍA TAMPICO  
UNIVERSIDAD AUTÓNOMA DE TAMAULIPAS

**P R E S E N T E**

Por medio de la presente nos permitimos hacer de su conocimiento que después de haber revisado exhaustivamente la tesis denominada:

**DESARROLLO DE UN SISTEMA PARA EL PAGO DE TRANSPORTE PUBLICO  
MEDIANTE NFC**

Que presenta el **C. Gustavo de Jesús Márquez Estrada**, la cual es correcta en su desarrollo, se aprueba para que sirva de base en la presentación del Examen Profesional para obtener el título de **Ingeniero en Sistemas Computacionales**.

Agradeciendo la atención que sirva brindar a la presente, nos es grado quedar de usted.

**Atentamente**



**Dr. Alejandro Humberto García Ruiz**  
Director



**Dr. Salvador Ibarra Martínez**  
Co-Director



**Dr. Emilio Castán Rocha**  
Asesor



**UAT**  
Universidad Autónoma  
de Tamaulipas



Facultad de Ingeniería  
Tampico  
Secretaría Académica

Tampico, Tamaulipas a 27 de septiembre de 2023

Asunto: **Autorización de Examen Profesional para obtener el título de Licenciatura**

Nº de Oficio 33-SAC- RP030 -2023

**M. A. I. Hyasseliny Alejandra Hurtado Mora**  
Coordinadora de Recepción Profesional  
Facultad de Ingeniería Tampico

Por este conducto le notifico que los siguientes catedráticos, quienes han sido propuestos por la academia de *Arquitectura de computadoras* como

Director: **Dr. Alejandro Humberto García Ruiz**  
Co-Director: **Dr. Salvador Ibarra Martínez**  
Asesor: **M.C. Emilio Castan Rocha**

después de haber revisado el trabajo de investigación denominado:

***Desarrollo de un sistema para el pago de transporte público mediante NFC***

que realizó el egresado **Márquez Estrada Gustavo de Jesús**  
de la carrera de **Ingeniero en Sistemas Computacionales**  
manifiestan que han se ha dado cumplimiento a lo dispuesto en el *Reglamento de Recepción Profesional* vigente en cuanto al desarrollo de este trabajo de investigación para la presentación del examen profesional bajo la **modalidad de Tesis**, por lo cual se autoriza continuar con el proceso de recepción profesional.

Sin más, me es grato quedar de usted.

Atentamente

"Verdad, Belleza, Probidad"



Facultad de Ingeniería Tampico  
Secretaría Académica

  
**Dra. Esperanza Martínez Cano**  
Secretaría Académica

c. p. Archivo  
EMC/emc

FACULTAD DE INGENIERÍA TAMPICO  
Centro Universitario Sur  
Tampico  
C.P. 89109.

(833) 241-2000,  
ext. 3451

## Dedicatorias

*A Dios, por darme fuerzas, fortaleza y guiarme en mi camino para seguir adelante.*

*A mi padre, por ser el mejor padre que Dios pudo colocarme en mi vida, siendo siempre un ejemplo que seguir, por mostrarme siempre un camino de bondad y cariño, por apoyarme siempre en mis decisiones, por todos esos desvelos y sacrificios para que yo pueda cumplir mis sueños, por enseñarme, aun en su ausencia, lo que soy capaz de hacer. A ti, que merecías estar presente en uno de los momentos más esperados de nuestra vida. Fuiste mi héroe y ahora serás mi estrella favorita, que me guía, me ilumina y cuida de mí, te echo de menos, ahora y siempre.*

*A mi madre, por ser la mejor madre que Dios pudo colocarme en mi vida, por su amor incondicional, por mostrarme siempre su apoyo en las decisiones más importantes de mi vida, por el esfuerzo y dedicación que hace día a día para seguir adelante, por su paciencia y amor brindado todos los días de mi vida, por inculcarme, junto a mi padre, la importancia de estudiar.*

*A mis hermanos, porque siempre estar presentes en mi vida, por ser los mejores hermanos que pude haber tenido, por su apoyo incondicional a lo largo de mi carrera, por su confianza, respaldo y cariño que me impulsan a seguir adelante día con día, por ser mis mejores amigos.*

*A mis amigos, por su apoyo, por su compañía que hizo mi estadía en la universidad fuera acogedora, por todos los buenos momentos, por su amistad y cariño.*

*A mi familia, mis abuelos, tíos y primos, por confiar en mí, por ser parte de mi vida, por sus consejos, por su amor y cariño.*

## Agradecimientos

*A la Universidad Autónoma de Tamaulipas por brindarme los recursos y las herramientas para forjar mi futuro como Ingeniero.*

*A mi director de tesis, el Dr. Alejandro Humberto García Ruiz, por todo el apoyo brindado, por depositar en mí el conocimiento que me ayudo a desarrollar mi tesis, por su guía, enseñanza y, sobre todo, por la gran amistad que me brindo y ayudo a sentirme con la confianza para poder preguntar cualquier duda que me surgía, por apoyarme desde el primer momento hasta el final en el desarrollo de mi tesis.*

*A mis maestros, por otorgarme lo conocimientos y enseñanzas necesarias que me ayudaron a forjarme como una mejor persona y como un futuro ingeniero.*

*A todas esas personas que me apoyaron de una u otra manera en la realización de la tesis.*

## Resumen

El gran avance tecnológico que se ha venido generando en los últimos años ha marcado un antes y un después en el cómo vivimos nuestro día a día. La tecnología se ha integrado cada vez más en nuestro entorno, facilitando la realización de una gran variedad de actividades en las que antes se requería una mayor interacción con los usuarios.

No obstante, existen diversos sectores en los que la tecnología aún no ha alcanzado un elevado nivel de integración. Un ejemplo de esto es el transporte público, en donde el uso de la tecnología en este aun es escaza. En este sentido, en este trabajo de investigación se consideró adecuado el desarrollo de un sistema que permita mejorar la forma con la que el usuario realiza el pago del transporte público.

Con el sistema propuesto se busca que con la integración de la de la tecnología NFC se disminuya la necesidad de traer consigo efectivo para el pago del boleto del transporte público, facilitando así este proceso entre los operarios del transporte público y los usuarios. El propósito del sistema propuesto es optimizar el pago del transporte público, esto debido a que pagar con efectivo requiere de un mayor tiempo para buscar el efectivo, proporcionarlo al operador y recibir la feria de este, además de que portar con efectivo es más inseguro para los usuarios.

De esta forma, gracias al uso de la tecnología, se buscar mejorar la manera en que se realiza el pago del transporte público, quitando así la preocupación de tener que disponer de efectivo en el momento de requerir hacer uso del transporte.

El proceso de validación se realizó a través de la aplicación de pruebas de funcionalidad al sistema para corroborar su adecuado funcionamiento, a través de las cuales se logró corroborar la eficiente funcionalidad del sistema para el propósito planteado en este trabajo de investigación.

# Índice

|   |            |
|---|------------|
| <i>Dedicatorias .....</i>                           | <i>IV</i>  |
| <i>Agradecimientos .....</i>                        | <i>V</i>   |
| <i>Resumen.....</i>                                 | <i>VI</i>  |
| <i>Índice.....</i>                                  | <i>VII</i> |
| <i>Lista de Figuras.....</i>                        | <i>X</i>   |
| <i>Lista de Tablas.....</i>                         | <i>XIV</i> |
| <b>Capítulo 1.    Introducción .....</b>            | <b>1</b>   |
| <b>1.1     Antecedentes .....</b>                   | <b>1</b>   |
| <b>1.2     Definición del problema .....</b>        | <b>3</b>   |
| <b>1.3     Objetivos de la investigación .....</b>  | <b>5</b>   |
| 1.3.1 Objetivo general .....                        | 5          |
| 1.3.2 Objetivos específicos .....                   | 6          |
| <b>1.4     Preguntas de la investigación .....</b>  | <b>6</b>   |
| <b>1.5     Justificación .....</b>                  | <b>7</b>   |
| <b>1.6     Delimitación .....</b>                   | <b>8</b>   |
| <b>Capítulo 2.    Análisis de fundamentos .....</b> | <b>9</b>   |
| <b>2.1     Marco conceptual.....</b>                | <b>9</b>   |
| 2.1.1 Hardware.....                                 | 9          |
| 2.1.2 Software.....                                 | 11         |



|  |           |
|--|-----------|
| 2.1.3 Base de datos .....  | 12        |
| <b>2.2 Marco teórico .....</b>                                       | <b>14</b> |
| 2.2.1 Tecnología NFC.....  | 14        |
| 2.2.2 El transporte publico.....                                     | 17        |
| 2.2.3 Sistema operativo Android.....                                 | 18        |
| <b>2.3 Marco contextual .....</b>                                    | <b>20</b> |
| <b>2.4 Estado del arte .....</b>                                     | <b>22</b> |
| <b>Capítulo 3. Metodología de la Investigación .....</b>             | <b>24</b> |
| 3.1 Enfoque de la investigación .....                                | 24        |
| 3.2 Tipo de investigación.....                                       | 24        |
| 3.3 Método de investigación.....                                     | 25        |
| 3.4 Categorías, variables e indicadores .....                        | 28        |
| 3.5 Población y muestra .....  | 29        |
| 3.7 Instrumentos de recolección de datos.....                        | 29        |
| <b>Capítulo 4. Análisis de Resultados .....</b>                      | <b>31</b> |
| 4.1 Análisis de sistemas de transporte existentes en el mercado..... | 31        |
| 4.2 Determinación de los módulos del sistema propuesto .....         | 38        |
| 4.3 Selección del Hardware y Software para el sistema .....          | 41        |
| 4.3.1 Hardware.....  | 41        |
| 4.3.2 Software.....  | 44        |
| 4.3 Esquema general del sistema propuesto .....                      | 52        |

|                          |  |            |
|--------------------------|--|------------|
| 4.3.3                    | Funcionalidad del sistema.....   | 53         |
| <b>4.5</b>               | <b>Diseño del sistema electrónico y de las interfaces gráficas .....</b>         | <b>54</b>  |
| 4.5.1                    | Diseño de las interfaces graficas.....   | 55         |
| 4.5.2                    | Diseño del sistema electrónico.....  | 64         |
| <b>4.6</b>               | <b>Programación del sistema propuesto .....</b>                                  | <b>65</b>  |
| <b>4.7</b>               | <b>Validación del sistema .....</b>  | <b>93</b>  |
| 4.7.1                    | Registro de una cuenta .....   | 94         |
| 4.7.2                    | Inicio de sesión en la aplicación.....   | 99         |
| 4.7.3                    | Administración de tarjeta mediante el UID.....                                   | 102        |
| 4.7.4                    | Recarga de crédito.....  | 107        |
| 4.7.5                    | Cobro de crédito.....  | 110        |
| 4.7.6                    | Pantalla de historial .....  | 112        |
| <b>Capítulo 5.</b>       | <b>Conclusiones y recomendaciones .....</b>                                      | <b>114</b> |
| <b>5.1</b>               | <b>Respuesta al problema de investigación .....</b>                              | <b>114</b> |
| <b>5.1</b>               | <b>Cumplimiento de los objetivos específicos .....</b>                           | <b>115</b> |
| 5.1.1                    | Analizar sistemas de pago de transporte publico existentes .....                 | 115        |
| 5.1.2                    | Diseñar el sistema electrónico y el software de control del sistema de pago..... | 116        |
| 5.1.3                    | Realizar la programación de lógica del sistema de pago.....                      | 117        |
| 5.1.4                    | Validar la funcionalidad del sistema desarrollado .....                          | 118        |
| <b>5.2</b>               | <b>Aceptación del supuesto .....</b>   | <b>119</b> |
| <b>5.3</b>               | <b>Recomendaciones para el trabajo .....</b>                                     | <b>119</b> |
| <b>5.4</b>               | <b>Recomendaciones para investigaciones futuras .....</b>                        | <b>120</b> |
| <b>BIBLIOGRAFIA.....</b> |  | <b>122</b> |

## Lista de Figuras

|  |           |
|--|-----------|
| <i>Figura 4.1 Diseño de la Tarjeta MI .....</i>  | <i>32</i> |
| <i>Figura 4.2 Tarjeta Mi Movilidad .....</i>   | <i>34</i> |
| <i>Figura 4.3 Aplicación Móvil “Mi Saldo” .....</i>                                    | <i>35</i> |
| <i>Figura 4.4 Diseño de la tarjeta general .....</i>                                   | <i>36</i> |
| <i>Figura 4.5 Esquema del funcionamiento del sistema .....</i>                         | <i>52</i> |
| <i>Figura 4.6 Interfaz de inicio de sesión .....</i>                                   | <i>56</i> |
| <i>Figura 4.7 Interfaz de registro de sesión .....</i>                                 | <i>57</i> |
| <i>Figura 4.8 Interfaz gráfica principal de usuario.....</i>                           | <i>59</i> |
| <i>Figura 4.9 Historial de usos de la tarjeta .....</i>                                | <i>60</i> |
| <i>Figura 4.10 UID de usuario .....</i>  | <i>62</i> |
| <i>Figura 4.11 Recarga de crédito.....</i>   | <i>63</i> |
| <i>Figura 4.12 Diseño del sistema electrónico .....</i>                                | <i>65</i> |
| <i>Figura 4.13 Sistema electrónico .....</i>   | <i>65</i> |
| <i>Figura 4.14 Estructura de la tabla “usuarios”.....</i>                              | <i>67</i> |
| <i>Figura 4.15 Estructura de la tabla “historial” .....</i>                            | <i>68</i> |
| <i>Figura 4.16 Código para validar datos en el registro de cuentas nuevas .....</i>    | <i>70</i> |
| <i>Figura 4.17 Código para realizar una petición POST de registro de cuentas .....</i> | <i>71</i> |

|   |           |
|---|-----------|
| <i>Figura 4.18 Encapsulado de datos para registrar una cuenta .....</i>                                       | <i>71</i> |
| <i>Figura 4.19 Código en PHP para establecer una conexión con la base de datos .....</i>                      | <i>72</i> |
| <i>Figura 4.20 Código en PHP para el registro de una cuenta .....</i>   | <i>73</i> |
| <i>Figura 4.21 Validación para determinar si el inicio de sesión fue correcto.....</i>                        | <i>74</i> |
| <i>Figura 4.22 Mapeo de los datos para validar inicio de sesión .....</i>                                     | <i>75</i> |
| <i>Figura 4.23 Código para comprobar los datos de inicio de sesión.....</i>                                   | <i>76</i> |
| <i>Figura 4.24 Obtención de los datos generales de los usuarios .....</i>                                     | <i>77</i> |
| <i>Figura 4.25 Método para el cambio entre pantallas .....</i>  | <i>78</i> |
| <i>Figura 4.26 Código para la inserción de registros asociados al historial de movimientos</i>                | <i>79</i> |
| <i>Figura 4.27 Recuperación del historial asociado a un usuario.....</i>                                      | <i>80</i> |
| <i>Figura 4.28 Obtención del historial de los usuarios.....</i>   | <i>81</i> |
| <i>Figura 4.29. Código en Arduino para la lectura de las tarjetas de identificación de los usuarios .....</i> | <i>83</i> |
| <i>Figura 4.30. Código en Python para el cobro del pasaje del transporte público.....</i>                     | <i>85</i> |
| <i>Figura 4.31 Activa o desactiva controles con base en el estado del UID .....</i>                           | <i>86</i> |
| <i>Figura 4.32 Código a ejecutarse cuando se da clic al botón “Confirmar” .....</i>                           | <i>87</i> |
| <i>Figura 4.33 Código para habilitar controles de edición de UID .....</i>                                    | <i>87</i> |
| <i>Figura 4.34 Código en PHP para modificar el UID de los usuarios .....</i>                                  | <i>89</i> |
| <i>Figura 4.35 Código para dar funcionalidad a los botones de recarga de crédito .....</i>                    | <i>90</i> |

|   |            |
|---|------------|
| <i>Figura 4.36 Suma de crédito con la cantidad de recarga actual .....</i>                | <i>91</i>  |
| <i>Figura 4.37. Funcionalidad del botón para confirmar la recarga de crédito .....</i>    | <i>92</i>  |
| <i>Figura 4.38. Código en PHP para la actualización del crédito de los usuarios .....</i> | <i>93</i>  |
| <i>Figura 4.39 Pantalla inicio/registro de sesión .....</i>                               | <i>94</i>  |
| <i>Figura 4.40 Mensaje de error en registro de cuenta.....</i>                            | <i>95</i>  |
| <i>Figura 4.41 Las contraseñas no coinciden .....</i>                                     | <i>96</i>  |
| <i>Figura 4.42 Correo electrónico ya registrado .....</i>                                 | <i>97</i>  |
| <i>Figura 4.43 Registro exitoso .....</i>   | <i>98</i>  |
| <i>Figura 4.44 Base de datos actualizada .....</i>  | <i>98</i>  |
| <i>Figura 4.45 Mensajes de error inicio de sesión.....</i>                                | <i>99</i>  |
| <i>Figura 4.46 Campo erróneo .....</i>  | <i>100</i> |
| <i>Figura 4.47 Pantalla principal de usuario.....</i>                                     | <i>101</i> |
| <i>Figura 4.48 Ingresar UID .....</i>   | <i>102</i> |
| <i>Figura 4.49 Mensajes de error al enlazar UID .....</i>                                 | <i>103</i> |
| <i>Figura 4.50 Ingresar UID .....</i>   | <i>104</i> |
| <i>Figura 4.51 Base de datos con UID asignada.....</i>                                    | <i>104</i> |
| <i>Figura 4.52 Actualizar UID.....</i>  | <i>105</i> |
| <i>Figura 4.53 Base de datos con UID actualizado.....</i>                                 | <i>105</i> |
| <i>Figura 4.54 UID eliminado.....</i>   | <i>106</i> |

|   |            |
|---|------------|
| <i>Figura 4.55 Campos vacíos en recarga de crédito .....</i>                | <i>107</i> |
| <i>Figura 4.56 Recarga de crédito exitoso .....</i>                         | <i>108</i> |
| <i>Figura 4.57 Refrescar la pantalla .....</i>                              | <i>109</i> |
| <i>Figura 4.58 Base de datos con crédito actualizado .....</i>              | <i>109</i> |
| <i>Figura 4.59 Identificación de etiqueta.....</i>                          | <i>110</i> |
| <i>Figura 4.60 Impresión de cobro en consola .....</i>                      | <i>111</i> |
| <i>Figura 4.61 Almacenamiento de registros en la tabla “historial”.....</i> | <i>111</i> |
| <i>Figura 4.62 Cobro de crédito reflejado en la aplicación .....</i>        | <i>112</i> |
| <i>Figura 4.63 Pantalla de historial .....</i>                              | <i>113</i> |

## Lista de Tablas

|   |           |
|---|-----------|
| <i>Tabla 3.1 Categorías, variables e indicadores .....</i>  | <i>28</i> |
| <i>Tabla 3.2 Formato de la prueba de funcionalidad del sistema .....</i>                          | <i>30</i> |
| <i>Tabla 4.1 Disposición de la conexión del módulo MFRC522 con la placa de Arduino Mega .....</i> | <i>64</i> |

# Capítulo 1. Introducción

## 1.1 Antecedentes

El transporte público es un medio de desplazamiento de un punto "A" a un punto "B", del que cualquier persona puede hacer uso únicamente pagando por un boleto que da derecho a ser pasajero del transporte y que, además, funge como seguro para los usuarios.

Actualmente, el transporte público es de vital importancia en la vida de muchas personas en todo el mundo, esto debido a que muchas personas no cuentan con los recursos necesarios para adquirir un automóvil o una bicicleta. Además, el uso de este puede sacar de un apuro a las personas por el simple hecho de que el servicio de transporte está en funcionamiento la mayor parte del día.

El transporte público ha existido desde hace mucho tiempo por la necesidad de las personas de trasladarse, esto ha originado que existan diversos medios de transporte diferentes para cumplir las necesidades de las personas. El transporte público abarca a lo largo de la historia desde las carretas de caballos, barcos, trenes, tranvías, camiones, carro, hasta llegar al transporte más rápido y avanzado, los aviones.



Sin embargo, a medida que la tecnología va evolucionando se requiere que se integren nuevos mecanismos o técnicas para mejorar la funcionalidad del transporte público y brindarle el mejor servicio posible al usuario, lo que ha originado que se tenga que hacer el cobro de una comisión para poder efectuar el uso del transporte. Ocasionando así, que el usuario tenga que disponer con el efectivo necesario para hacer uso del transporte público que se le requiera.

Son cada vez más las unidades de transporte que se encuentran en circulación para así cumplir la demanda que la sociedad requiere. De esta forma, la importancia del transporte ha ido creciendo constantemente desde sus inicios hasta la actualidad, siendo un medio indispensable para el desarrollo y crecimiento de la población en general.

El aplicar a la tecnología hacia el sector productivo puede llegar a hacer que el transporte público tenga aún más relevancia para la sociedad que la que actualmente tiene, debido a que optimizaría al trabajo que se debe realizar en este sector, facilitando a las labores del ser humano y permitiendo que este último sea capaz de dejar el trabajo más tedioso a la tecnología para de esta manera enfocarse en alguna otra actividad o actividades que demanden su atención.

## **1.2 Definición del problema**

El problema principal del uso de efectivo en el transporte público es que el efectivo ha ido perdiendo importancia poco a poco en nuestro entorno, cada vez vemos más el uso de tarjetas de crédito o débito, así como, dispositivos móviles, para realizar pagos. No sorprendería que dentro de unos años el efectivo desaparezca por completo y únicamente se haga uso de medios alternativos como los citados. Es por ello por lo que se debe ir pensando en cómo actualizar los servicios en los que aún se requiera el uso de efectivo, para que de esta forma sean adaptados a las nuevas necesidades y exigencias de la sociedad.

De igual manera, el que se tenga que pagar con efectivo en los medios de transporte públicos en la actualidad puede llegar a ocasionar distintos problemas de diversas índoles. Un problema que se presenta hoy en día por el uso de efectivo en el transporte público es que a veces no contamos con el dinero para hacer uso de este medio, por lo tanto, nos vemos en la obligación de pedir prestado a algún conocido que este con nosotros o a una persona desconocida que se encuentre cerca de nosotros.

Del mismo modo, se presenta la situación de que el operador tiene que dedicar tiempo para recibir el efectivo por parte de los usuarios, que, si bien es cierto que no es una gran cantidad de tiempo, este mismo con una gran cantidad de personas abordando la unidad de transporte puede ocasionar que lleguemos minutos tarde a nuestro destino.

Otro problema que se nos puede presentar es el exceso de efectivo con el que a veces contamos, contar con exceso de efectivo puede ocasionar que nuestra cartera o bolsillos estén más cargados de lo normal. De esta forma, su presencia se notará más en nuestras prendas, haciéndonos más vulnerables de un asalto, además de que dependiendo de donde los guardemos podría llegar a ser incomodo y malo para nuestra salud el contar con una gran cantidad de efectivo y realizar acciones como sentarnos o apoyarnos sobre este.

Derivado de los problemas citados, así como algunos otros más que pudieran llegar a presentarse, hoy en día son más los investigadores que han comenzado a desarrollar innovadoras aportaciones para darles una solución. No obstante, en algunos países en donde ya se ha integrado a esta tecnología se han detectado algunos problemas a los que se han enfrentado los investigadores y desarrolladores, siendo los más destacados el

que la mantenibilidad del sistema tiende a ser costoso y en muchas ocasiones el sistema resulta poco intuitivo para los usuarios, además de que para recargar crédito a las tarjetas es necesario recurrir a casetas específicas ubicadas en la ciudad. Debido a esto, este tipo de tecnología aun no es aplicada en muchas partes del mundo, como ocurre en Tampico, Tamaulipas, lugar en que se realiza este trabajo de investigación.

En particular, en Tampico, algunas de las causas de que esta tecnología no se haya integrado aún en la zona, son el hecho de que esta puede llegar a generar dudas entre la sociedad por el hecho de que la gente ya está muy acostumbrada al uso de efectivo y un cambio podría generarles dudas o crear controversia. En este sentido, en este trabajo de investigación se busca el desarrollo de un sistema de pago que permita aminorar a los problemas mencionados.

### ***1.3 Objetivos de la investigación***

#### ***1.3.1 Objetivo general***

Desarrollar un sistema que haga uso de tecnología NFC para facilitar el pago del transporte público.

### **1.3.2 Objetivos específicos**

1. Analizar sistemas de pago de transporte publico existentes
2. Diseñar el sistema electrónico y el software de control del sistema de pago
3. Realizar la programación de lógica del sistema de pago
4. Validar la funcionalidad del sistema desarrollado

### **1.4 Preguntas de la investigación**

De acuerdo con los objetivos trazados en esta investigación se desprenden las siguientes preguntas de investigación:

1. ¿De qué manera se llevará a cabo el uso de la tecnología NFC para el desarrollo del sistema de pago?
2. ¿Cuáles son las principales características y funciones de los sistemas de pago de transporte publico existentes en el mercado?
3. ¿Cómo deberá ser el diseño de las interfaces gráficas con las que interactuará el usuario, así como, el diseño del sistema electrónico que será utilizado por el sistema?
4. ¿De qué manera deberá realizarse la programación de la lógica del sistema de pago para que este opere de manera eficiente?
5. ¿Cuál proceso que debe seguirse para efectuar la validación del sistema de forma eficiente?

## **1.5 Justificación**

El uso del sistema integrado con tecnología NFC para el transporte público proporcionará a los ciudadanos una mayor facilidad al momento de realizar el pago correspondiente al precio del boleto, permitiendo así que el pago del transporte público se vuelva mucho más cómodo y sencillo para las personas. Quitando así la necesidad de cargar con efectivo para pagar el derecho a este medio de transporte.

De igual forma, con el sistema propuesto las personas dejarán de preocuparse por disponer con el efectivo suficiente para hacer uso del transporte público, haciendo así que las personas cada vez vayan preocupándose menos por situaciones cotidianas por el simple hecho de que estas ya están optimizadas.

Las unidades de transporte junto con sus operadores también se verían beneficiados, pues hacer uso de un sistema que registre cada entrada de dinero mediante esta tecnología, hará que ya no sea necesario el contar a mano el efectivo que consiguió la unidad a lo largo del día, ni de gastar en la realización de boletos físicos, así como también permitirá no distraerse en recibir y entregar el efectivo a los pasajeros de la unidad.

El sistema será de bajo costo e intuitivo para las personas y no será requerido el ir específicamente a una caseta dedicada para poder recargar crédito al sistema. Haciendo posible que los usuarios puedan disponer de crédito en sus respectivas tarjetas de una manera más fácil y cómoda para estos.

### **1.6 Delimitación**

El desarrollo del sistema de pago se llevará a cabo a modo de prototipo. Por lo tanto, su alcance se verá limitado a las restricciones que el prototipo permita. En este sentido, en este primer prototipo, para realizar el pago del transporte público, se hará uso de tarjetas previamente incorporadas de un UID único, de tal forma que, al asociarse este código a un usuario, este podrá hacer uso del sistema y realizar el pago del efectivo correspondiente a su boleto de autobús.

La red a través de la cual se comunicará el sistema será una red de área local. La aplicación desarrollada se destinará a dispositivos Android debido a la amplia gama de dispositivos móviles que cuentan con este sistema operativo.

La validación del sistema propuesto se efectuará a través de la aplicación de pruebas de funcionalidad aplicadas a nivel de laboratorio.

## **Capítulo 2. Análisis de fundamentos**

### **2.1 Marco conceptual**

En este apartado se presentará a algunos conceptos que resultan esenciales para entender de manera clara y precisa lo que se desea realizar en este trabajo de investigación.

#### **2.1.1 Hardware**

Son todos los elementos físicos que forman parte de los dispositivos electrónicos o de los sistemas informáticos, es decir, el hardware son todas esas partes tangibles del sistema, que en conjunto hacen posible la comunicación entre los dispositivos electrónicos del sistema y el usuario.

##### **a) Sistema electrónico**

Conjunto de componentes electrónicos, tales como circuitos integrados, resistencias, capacitores y transistores, diseñados para trabajar en conjunto y realizar una tarea específica (Mottershead, G., 2015).

##### **b) Dispositivo móvil**

Dispositivo electrónico portátil diseñado para facilitar a los usuarios la comunicación y el acceso a la información a distancia y en movimiento (Goggin, G., 2017).



#### **c) Puerto serie**

Es la interfaz de comunicación serial de comunicación utilizada para transferir datos entre dispositivos electrónicos, permite el envío y la recepción de mensajes, así también como el control de la depuración (FLEXBOT, 2019).

#### **d) Voltaje**

Es la medida de energía electrónica que se transfiere por unidad de carga eléctrica. Refiriéndose a la diferencia de potencial eléctrico entre dos puntos en un circuito eléctrico, medido en voltios (V) (Dorf, R., & Svoboda, J., 2015).

#### **e) Procesador**

Dispositivo electrónico capaz de recibir, interpretar y ejecutar instrucciones para la realización de operaciones aritméticas y lógicas. Se utiliza principalmente en ordenadores y dispositivos móviles, hasta sistemas embebidos en electrodomésticos y automóviles. Su función principal es el procesar los datos y ejecutar el programa (Tanzi, T., & Valero, M., 2012).

### **2.1.2 Software**

Es todo lo lógico e intangible de un ordenador. Es el conjunto de programas, herramientas y reglas destinadas a la realización de una o diversas funciones dentro del sistema informático.

#### **a) Sistema operativo**

Software encargado de gestionar los recursos del hardware y software de un dispositivo electrónico, permitiendo la comunicación entre el usuario y el sistema (Tanenbaum, A. S., & Bos, H., 2015).

#### **b) Android**

Sistema operativo de Código abierto diseñado para dispositivos móviles, está basado en el núcleo de Linux y fue desarrollado por Google (Gironés, J. T., 2019).

#### **c) Aplicación móvil**

Programa de software diseñado específicamente para ser utilizado en dispositivos móviles, como lo son teléfonos inteligentes, tabletas, relojes inteligentes, etc. Las aplicaciones están hechas para cumplir ciertas tareas específicas que sean requeridas por parte del usuario (Herazo, L., 2019).

#### **d) Interfaz de usuario**

Es la capa visual y táctil que permite a los usuarios el interactuar con el sistema o con la aplicación. En esta, los usuarios pueden ver, tocar, seleccionar y manipular los elementos gráficos que se muestran en la pantalla para realizar tareas o acciones específicas (Albornoz, M. C., Berón, M., & Montejano, G. A., 2017).

#### **e) Lenguaje de programación**

Es un conjunto de reglas, símbolos y convenciones que permiten a los programadores crear software. Se utilizan para escribir código que posteriormente será compilado o interpretado por una máquina, lo que permitirá que el software ejecute (Según Aguilar, L. J., Azuela, M. F., & Baena, L. R., 1988).

### **2.1.3 Base de datos**

Las bases de datos son un conjunto organizado de información que se almacena en un sistema informático, utilizado para recopilar, organizar, almacenar y recuperar datos relacionados entre sí, lo que permite a los usuarios el realizar consultas y extraer información específica de manera eficiente.

De acuerdo con Elmasri, Navathe, Castillo, Pérez y Espiga (2007), en la actualidad, las bases de datos son ampliamente utilizadas en diversos ámbitos y contextos. Las bases de datos relacionales son las más populares y extendidas en la actualidad, aunque hay que tener en cuenta que su uso no siempre ha sido el predominante. Las bases de datos jerárquicas fueron las primeras en surgir, evolucionando posteriormente hacia las bases de datos reticulares hasta llegar a las relacionales.

#### **a) Dato**

Es una unidad básica de información que se almacena en una tabla y que describe una entidad o una relación entre entidades. Los datos pueden ser valores numéricos, cadenas de texto, fechas, imágenes u otro tipo de información que se pueda almacenar en la base de datos (Marqués, M., 2011).

#### **b) Tabla**

Se dice que la tabla es una colección de datos organizados en filas y en columnas. Cada fila representa un registro único en la tabla, mientras que cada columna representa un atributo específico del registro (Kroenke, D. M., & Auer, D. J., 2016).

### **c) Consulta**

Es una solicitud de información específica que se realiza a través de un lenguaje de consulta estructurado, la consulta puede incluir una o varias tablas, utilizada para recuperar, insertar, actualizar o eliminar información. Las consultas permiten acceder a datos específicos y relevantes para sus necesidades sin la necesidad de revisar grandes cantidades de información (Gomez, M. D., 2013).

### **d) Índices**

Estructura de datos que mejora la velocidad de recuperación de información, creados para mejorar el rendimiento de las consultas en una base de datos, permitiendo la búsqueda de información de manera eficiente (Silberschatz, A., Korth, H. F., & Sudarshan, S. 2002).

## **2.2 Marco teórico**

### **2.2.1 Tecnología NFC**

La tecnología NFC se trata de una tecnología de comunicación inalámbrica, la cual, hace posible la transferencia de datos entre dispositivos electrónicos estando a un corto alcance, haciendo así que la tecnología se pueda utilizar de una manera sencilla y rápida. Su utilización, actualmente, es para la realización de transacciones financieras, debido a su seguridad

y a su rapidez a la hora de transferir los datos (Anaya-Cantellán, A., & López-Martínez, I., 2014).

De igual manera, en Anaya-Cantellán, A., & López-Martínez, I. (2014) se abordan tres aspectos importantes que hay que tener en cuenta siempre que se hable de tecnología NFC para comprender su funcionamiento:

- **Modos de funcionamiento**

Esta tecnología cuenta con dos modos de funcionamiento para los dispositivos que desean realizar una comunicación, el modo activo y pasivo.

a) Modo activo. Este modo requiere el uso de dos dispositivos activos, en el cual, ambos generan un campo electromagnético y se hace posible la transferencia o el intercambio de datos.

b) Modo pasivo. Para este caso, el dispositivo pasivo no cuenta con la capacidad de crear su propio campo electromagnético, sino que requiere de un dispositivo activo para poder aprovechar el campo electromagnético generado por el mismo.

- **Modos operacionales**

- a) Peer-to-peer. Funciona únicamente mediante dos dispositivos activos, permitiendo que la transferencia de información sea rápida. Un dispositivo transmite mientras el otro escucha es transmisión.
- b) Emulación de tarjeta. Utiliza un dispositivo activo y un lector de NFC. El dispositivo NFC se comporta de manera similar al de una tarjeta inteligente.

- **Dispositivos móviles con tecnología NFC**

Primeramente, los dispositivos móviles son conocidos como las computadoras de bolsillo, pues estos ayudan a que las personas tengan la facilidad de acceder de manera remota a cualquier sitio de internet, datos, información y comunicación sin la necesidad de tener una computadora o una laptop para poder realizar estas acciones.

En la actualidad, cada celular que sale al mercado cuenta con una tecnología muy particular que normalmente las personas no son conscientes su existencia, se trata de la tecnología NFC, con la cual, podemos compartir información o transferir datos con una acción tan simple y rápida.

### ***2.2.2 El transporte publico***

En FACUA (2007) se dice que el transporte público se inspira prácticamente en criterios de solidaridad. Se debe tener en cuenta que se habita en un mundo globalizado donde las necesidades de desplazamientos entre diversos países y ciudades cada vez adquieren más trascendencia. Los sitios de producción se encuentran situados a gigantescas distancias de los puntos de vista de consumo. Los habitantes residen en una localidad, trabajan o estudian en otra y los sitios de tiempo libre y goce del descanso tienen la posibilidad de estar en otro sitio diferente. La utilización de los sectores públicos del transporte como lo son el autobús, el metro o los taxis se muestran como un medio que facilita el transporte, alejado del estrés que puede llegar a crear el uso de un automóvil.

Dicho esto, el transporte público es el medio de transporte que se basa en el pago de un boleto que da el derecho de obtener un lugar y de utilizar la unidad de transporte que fue solicitada por la persona, permitiendo así, que se pueda realizar el viaje correspondiente desde un punto A hasta un punto B.



### **Importancia de mejorar el transporte público**

Se dice que el utilizar el transporte público como medio de viaje principal trae consigo ciertas ventajas que nos ayudan en el día a día y a su vez, ayudan al planeta, tales como Imbric. (n.d.):

- El transporte público contamina menos que los vehículos personales, si bien es cierto que todo vehículo contamina debido a la emisión de gases, el utilizar el transporte público hace que el número de vehículos en circulación disminuya, ocasionando así, que el utilizar el transporte público sea una elección más ecológica para desplazarse dentro de una ciudad.
- Ocasiona menos tráfico, de la mano del punto anterior, el hecho de que haya muchos vehículos ocasiona que haya atascos en diversos puntos de la ciudad.
- Utilizar el transporte público es mucho más económico que el utilizar vehículos personales.

### **2.2.3 Sistema operativo Android**

De acuerdo con Gironés, J. T. (2019), Android vio por primera vez la luz en 2003 por la compañía Android Inc., primeramente,

se trataba de un sistema operativo el cual era prácticamente un desconocido hasta el día que la empresa Google decidió comprarlo. Con el pensamiento en el gran futuro que los teléfonos celulares iban a tener.

Debido a que Android está basado en Linux, hace posible que tenga acceso a los recursos, permitiendo gestionarlo, esto debido a que se encuentra en una capa encima del Kernel, permitiendo así, acceder a los recursos como los controladores de la pantalla, cámara, memoria, etc.

El despliegue de este sistema operativo fue lento debido a que antes el mercado de los teléfonos celulares no era tan demandante a como lo en la actualidad, no fue hasta el 2011 que llegó la versión 3.0 llamada Honeycomb.

Tal cual se tenían las expectativas, los teléfonos celulares fueron adquiriendo cada vez más importancia y consigo, el sistema Android, pues cada vez se fueron dejando cada vez más los celulares convencionales y las puertas se fueron abriendo a que los dispositivos contaran con el sistema que realizara de una manera efectiva y eficiente las tareas que se fueran demandadas.

Android actualmente, es uno de los sistemas operativos con más importancia a nivel mundial, pues Google ha sabido cumplir con las promesas relacionadas con Android, haciendo que el sistema no haya llegado solamente a celulares, sino a todo tipo de sistemas, permitiendo tener diferentes dispositivos que cuenten con este sistema.

### ***2.3 Marco contextual***

Este trabajo de investigación surgió derivado de que se observó que los estudiantes que hacen un uso diario del transporte público muchas veces olvidan o gastan todo el dinero destinado al pasaje, obligándoles esto a tener que pedir prestado o irse caminando a sus hogares, poniendo así su seguridad en riesgo, debido a una mala administración del dinero con el que contaban para su pasaje.

Al igual que, no es óptimo debido a que, si los jóvenes llegarán a llevar consigo un exceso de dinero en efectivo, este podría ocasionarle una incomodidad a causa de una sobrecarga de objetos en sus carteras o en sus bolsillos, con base en el hecho de que el pasaje usualmente se paga con monedas de baja denominación.

De igual manera, el hecho de cargar con una gran cantidad de dinero en efectivo hace que en el dado caso de que los jóvenes sean asaltados, estos se vean más afectados, pues además de poner en riesgo su integridad física, se quedarían sin la posibilidad de poder pagar el transporte público para trasladarse a sus hogares.

Debido a lo anterior, en este trabajo de investigación se consideró pertinente pensar en una manera en la cual la mayoría de estos problemas sean resueltos, de esta forma se consideró adecuado el desarrollo de un sistema el cual permita realizar el cobro del transporte público a través de tarjetas capaces de ser leídas por tecnología NFC. Además, dichas tarjetas se encontrarán vinculadas a las cuentas de usuario de los ciudadanos, de tal manera que a través de estas cuentas sea posible administrar el crédito, recargar más, monitorear el historial de movimientos, así como otros datos de los usuarios, todo desde una aplicación móvil intuitiva para los usuarios.

Por la naturaleza de este trabajo de investigación, esta tendrá lugar a nivel de laboratorio. No obstante, con esta investigación se busca crear un interés por parte de las empresas encargadas del transporte público de la zona para mejorar la efectividad de este, teniendo repercusiones

principalmente en la sociedad y en los sectores públicos, como lo es el transporte público.

De esta forma, se buscar hacer que el transporte público mejore sus servicios mediante la integración de las herramientas tecnológicas con las que hoy en día contamos y tenemos acceso.

## **2.4 Estado del arte**

Con el fin de lograr un mayor entendimiento de lo que han hecho otros investigadores en esta área, en este apartado se presentará a algunos trabajos investigación relacionados en los que de igual forma se hizo uso de la tecnología emergente NFC.

En Mora Cerezo, R. D., y Santos Bajaña, C. A. (2019), se realizó el diseño de un sistema de compra para supermercados mediante el uso de microcontroladores (Arduino) con NFC. Este consistió en utilizar NFC y Wifi para la obtención de información de productos, el observar datos de la factura y el total de compra. Todo esto surgió de la necesidad de proporcionar a los consumidores una herramienta la cual ayudará a que ellos minimicen el tiempo de espera y las largas filas en los supermercados, consiguiendo así un ahorro de tiempo para los consumidores en fechas donde haya mucha demanda.

Por su parte, en el trabajo de Acevedo Mosquera, M. E., Acevedo Mosquera, M. A., Neri Espinoza, K. A., Chávez Álvarez, A. (2015), se presenta un sistema de multas automático mediante la tecnología NFC para bibliotecas. En este se remarca la importancia de tener un control en las bibliotecas para la facilidad del préstamo o renta de libros, recalcando la automatización de un sistema que genere multas a los usuarios que no devuelvan a tiempo a algún libro de la biblioteca. Para este trabajo se utilizó a la tecnología NFC, Arduino, Lenguaje C/C++, SPI y Visual C#, para llevar a cabo el desarrollo del sistema.

Finalmente, en López Bermeo, S. P. (2019), se desarrolló una máquina dispensadora de snacks con pago por NFC. Para este trabajo, además se desarrolló una aplicación móvil, como medio de comunicación bidireccional para consultar el efectivo al final de la compra y el adquirir productos deseados de una manera rápida, fácil y efectiva. El sistema se ha desarrollado de tal forma que este pueda ser fácilmente implementado en restaurantes, bares y aeropuertos. Este trabajo, al igual que los trabajos anteriores, se desarrolló basado en la tecnología Arduino como hardware encargado de administrar el proceso del sistema de pago mediante NFC.

## **Capítulo 3. Metodología de la Investigación**

### **3.1 Enfoque de la investigación**

Debido a la naturaleza de este trabajo de investigación se identificó que este sigue un enfoque de investigación del tipo cualitativo. De acuerdo con Sampieri, R. (2014), la investigación cualitativa es la que se encarga de que los datos, la riqueza interpretativa, la dispersión, el entorno y los detalles tengan más profundidad a la hora de ser desglosados en un ámbito. Del mismo modo, Sampieri, menciona que: "El enfoque cualitativo utiliza la recolección y análisis de los datos para afinar las preguntas de investigación o revelar nuevas interrogantes en el proceso de interpretación".

### **3.2 Tipo de investigación**

En cuanto al tipo de investigación en este trabajo de investigación se hará uso del tipo investigación aplicada. Para Esteban Nieto, N. (2018), la investigación aplicada se trata de resolver problemas que llegan a presentarse en procesos de producción, distribución, circulación y servicios de las actividades humanas, orientándose principalmente a perfeccionar y optimizar a los sistemas, procedimientos y normas existentes y a generar nuevos.

### **3.3 Método de investigación**

Este trabajo de investigación tendrá una metodología basada en el método de desarrollo de software clásico. Las etapas que esta metodología utilizará son descritas a continuación.

#### **Análisis**

Se consultarán diferentes bases de datos, libros, revistas y páginas de internet de interés, en las que la investigación se centrará en la búsqueda de temas relacionados al pago del transporte público. En esta etapa la búsqueda de información se centrará en identificar a los componentes con los que debe contar un sistema de pagos para operar de manera eficiente. Posteriormente, se seleccionarán a los componentes más destacables e imprescindibles, con la intención de proponer un nuevo prototipo de sistema de pago orientado al transporte público.

#### **Diseño**

Para el diseño de la aplicación móvil se busca que ésta sea de fácil manejo para cualquier persona, de tal forma que la aplicación cuente con un diseño intuitivo, sencillo y amigable para los usuarios. Para lograr esto, la aplicación contará con las pantallas necesarias para que los usuarios puedan realizar la creación de cuentas desde la misma aplicación, iniciar



sesión con sus cuentas, si estas ya se encuentran activas, y acceder a la pantalla principal una vez se haya iniciado sesión, en donde la aplicación mostrará el crédito con el que cuenta cada usuario y habilitará a los usuarios a acceder al resto de funcionalidades de la aplicación móvil.

Del mismo modo, en esta etapa se realizará el diseño del sistema electrónico que deberá ser incorporado en el transporte público para realizar el cobro del pasaje a los usuarios desde la aplicación móvil. El diseño del sistema electrónico se busca sea lo menos intrusivo para los usuarios, de tal manera que el sistema, además de no ocupar mucho espacio, resulte de fácil manipulación y acceso.

### **Programación**

Una vez que se cuente con los diseños de las pantallas que serán utilizados por la aplicación móvil, así como, con el diseño del sistema electrónico, se procederá a realizar la programación adecuada de cada uno de sus componentes para garantizar su correcto funcionamiento e integración.

La programación del sistema de pago se realizará en tres partes: la primera de ellas con el lenguaje de programación Android para programar la lógica de la aplicación móvil con la

cual se dispondrá de la capacidad de iniciar sesión y/o registrar nuevos usuarios, visualizar los datos de los usuarios y su crédito disponible, y principalmente, realizar el pago del transporte público.

La segunda parte se efectuará en el lenguaje PHP, con este se realizarán las consultas necesarias a la base de datos en MySQL, ya sea para la inserción, modificación o eliminación de información en esta o para la solicitud de información de algún usuario en particular.

Finalmente, la tercera parte hará uso del lenguaje de programación Python y de Arduino IDE, utilizados para la programación de la lógica del sistema electrónico, permitiendo de esta forma realizar el cobro de crédito y la actualización del cargo correspondiente en la base de datos, mediante la lectura de tarjetas NFC asociadas a los usuarios.

### **Validación**

Esta etapa consistirá en la revisión del sistema de pago desarrollado mediante la aplicación de pruebas de funcionalidad para comprobar que esté cumpla con las especificaciones requeridas y establecidas en este trabajo de investigación.

Esta actividad tiene además como propósito servir como medio de apoyo para disminuir los errores que se podrían llegar a presentar durante el proceso de entrada y salida de datos. Comprobando así, si el sistema es capaz de operar eficientemente de acuerdo con los lineamientos definidos como parte del objetivo general de este trabajo de investigación.

### **3.4 Categorías, variables e indicadores**

En este apartado se presenta a las categorías, variables e indicadores que fueron definidos para llevar a cabo la realización de este trabajo de investigación.

**Tabla 3.1 Categorías, variables e indicadores**

| <b>Categoría</b>       | <b>Variable</b>         | <b>Indicador</b>  |
|------------------------|-------------------------|---|
| Sistema de Control     | Análisis                | Análisis de sistemas de pago existentes en el mercado                       |
|                        | Diseño                  | Diseño del sistema electrónico e interfaces graficas                        |
|                        | Desarrollo              | Programación de la lógica e integración de los sistemas                     |
|                        | Validación              | Pruebas de funcionalidad al sistema   |
| Validación del Sistema | Usabilidad              | Facilidad para el usuario en el uso del sistema                             |
|                        | Funcionalidad           | Que tan adecuada es la respuesta del sistema ante una solicitud del usuario |
|                        | Experiencia del usuario | Nivel de comodidad alcanzado por los usuarios al hacer uso del sistema      |

**Fuente: elaboración propia**

### ***3.5 Población y muestra***

La población para este trabajo de investigación se encuentra constituida por el total de pruebas de funcionalidad necesarias para la validación del funcionamiento del sistema. En este mismo sentido, por la naturaleza de esta investigación, no se contará con una muestra.

### ***3.7 Instrumentos de recolección de datos***

En la Tabla 3.2 se presenta el formato de la prueba de funcionalidad utilizada para la validación del sistema propuesto.

**Tabla 3.2 Formato de la prueba de funcionalidad del sistema**

|                             |  |                         |            |
|-----------------------------|--|-------------------------|------------|
| <b>Organismo:</b>           | Universidad Autónoma de Tamaulipas                                       |                         |            |
| <b>Proyecto:</b>            | Desarrollo de un sistema para el pago de transporte publico mediante NFC |                         |            |
| <b>Autor:</b>               | Gustavo de Jesús Márquez Estrada   |                         |            |
| <b>Versión:</b>             | XXX  | <b>No Prueba:</b>       | XXX        |
| <b>Aprobado por:</b>        | Dr. Alejandro Humberto García Ruiz                                       | <b>Fecha Prueba:</b>    | DD/MM/AAAA |
|                             |  |                         |            |
| <b>Funcionalidad:</b>       | BUENA  | REGULAR                 | MALA       |
|                             | Observaciones:   |                         |            |
|                             |  |                         |            |
| <b>Tiempo de Respuesta:</b> |  | Observaciones:          |            |
|                             |  |                         |            |
| <b>Usabilidad:</b>          | BUENA  | REGULAR                 | MALA       |
|                             | Observaciones:   |                         |            |
|                             |  |                         |            |
| Observaciones Generales:    |  |                         |            |
|                             |  |                         |            |
| <b>REGISTRO DE CAMBIOS</b>  |  |                         |            |
| <b>No. Cambio</b>           | <b>Causa del Cambio</b>  | <b>Fecha del Cambio</b> |            |
| XX                          | Error de ...   | DD/MM/AAAA              |            |
|                             |  |                         |            |

**Fuente: Elaboración propia**

## Capítulo 4. Análisis de Resultados

### ***4.1 Análisis de sistemas de transporte existentes en el mercado***

En el mercado son diversos los sistemas que existen para facilitar el pago del transporte, sin embargo, a continuación, se presentará a los tres que resultaron de mayor relevancia para este trabajo de investigación debido a su estrecha relación con este.

- ***Movilidad Integrada (MI)***

Sistema en la Ciudad de México el cual tiene como función principal el optimizar y agilizar el acceso e interconexión entre los distintos sistemas de transporte público para los habitantes y visitantes que se encuentren en la ciudad por medio de un pago único, a través de la implementación de una tarjeta con la cual se podrá ingresar a los distintos sistemas de transporte público que existen dentro de la Ciudad de México.

Con este sistema los usuarios del sistema pueden ingresar a los transportes públicos tales como: Metro, Metrobús, trolebús, Cablebús, Tren Ligero y Ecobici<sup>1</sup>.

---

<sup>1</sup> <https://www.semovi.cdmx.gob.mx/movilidad-integrada/tarjeta-de-gratuidad>

La integración de este sistema dentro del transporte público hace posible que el uso de efectivo se reduzca y permite disminuir tiempos y agilizar el proceso de abordaje y descenso de los pasajeros en sus respectivas paradas, plataformas y andenes.

Los usuarios pueden adquirir la tarjeta de Movilidad Integrada en taquillas ubicadas en cualquier estación de metro de la Ciudad de México, por un costo de quince pesos y con una recarga mínima de seis pesos y un máximo de ciento veinte pesos.



**Figura 4.1 Diseño de la Tarjeta MI**

**Fuente:** <https://gobierno.cdmx.gob.mx/acciones/red-de-movilidad-integrada/>

Para evitar que las personas que cuenten con una tarjeta de Movilidad Integrada tengan alguna complicación a la hora de proceder con el pago correspondiente a su pasaje dentro del transporte público, existe un gran número de máquinas

automáticas capaces de realizar el cobro del costo del transporte, sin la necesidad de hacer contacto directo con los sensores de entrada de este.

Así mismo, el sistema cuenta con una sección dentro de la página oficial de la Ciudad de México en donde se presenta información relacionada con este, permitiendo así que los usuarios conozcan datos esenciales que les serán de apoyo para adaptarse correctamente al sistema. Dentro de la página los usuarios podrán obtener información acerca de los medios de transporte existentes dentro de la ciudad, visualizar a los puntos de recarga, visualizar el mapa de lo que abarca el transporte público, tener al alcance el manual MI, etc.

- ***Mi Pasaje - Mi movilidad***

Sistema de Jalisco el cual permite a los ciudadanos la oportunidad de pagar el costo de todo transporte público disponible en la Ciudad por el medio de una tarjeta llamada "Mi Movilidad". Esta tarjeta tiene como finalidad el hacer que el acceso al transporte público sea más ágil que el método convencional utilizando efectivo, así como también el tener un mayor control sobre los gastos que realizan las personas (Credencialización de Mi Pasaje en Jalisco, 2020).



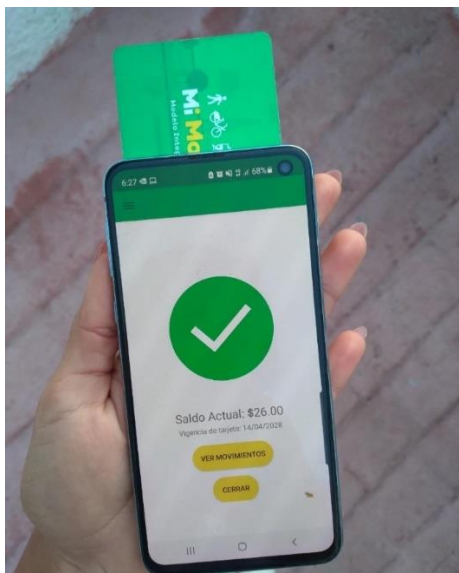


**Figura 4.2 Tarjeta Mi Movilidad**

**Fuente: [https://www.cronicajalisco.com/notas-arranco\\_sistema\\_de\\_citas\\_para\\_obtener\\_latarjeta\\_mi\\_pasaje-102428-2020](https://www.cronicajalisco.com/notas-arranco_sistema_de_citas_para_obtener_latarjeta_mi_pasaje-102428-2020)**

La recarga de saldo a esta tarjeta se requiere que los usuarios se dirijan a los puntos específicos dentro de la Ciudad en donde se encuentran las máquinas automáticas capaces de recibir dinero en efectivo, o si lo prefieren, recargar saldo dentro de las sucursales "OXXO" que ofrecen este servicio, permitiendo recargar un mínimo de cinco pesos hasta un máximo de doscientos cincuenta pesos.

Cada unidad de transporte público de la ciudad dispone con una terminal, la cual es la encargada de la detección de la tarjeta de los usuarios y permitir el acceso al transporte público de su preferencia. De igual manera, este sistema permite la integración de más de una tarjeta, es decir, personas de tercera edad, estudiantes, personal docente o personas discapacitadas, dispondrán de una tarjeta con tarifa preferencial.



**Figura 4.3 Aplicación Móvil "Mi Saldo"**

**Fuente:** <https://www.informador.mx/jalisco/Transporte-publico-Asi-puedes-revisar-el-saldo-de-tu-tarjeta-de-prepago-Mi-Movilidad-20220113-0119.html>

Así mismo, este sistema cuenta con una aplicación móvil para que las personas tengan a su alcance la oportunidad de saber el saldo con el que cuentan dentro de su tarjeta Mi Movilidad, la aplicación en cuestión se llama "Mi Saldo".

Dentro de la aplicación móvil es posible la recarga de saldo a la tarjeta con la que cuentan los usuarios. Para verificar con cuanto saldo cuenta una tarjeta, así como también el historial de viajes que se han realizado con esta, es necesario que el dispositivo móvil cuente con tecnología NFC<sup>2</sup>.

---

<sup>2</sup> <https://programamipasaje.jalisco.gob.mx/>

- **Qrobús**

Sistema de la ciudad de Querétaro que integra y fomenta el uso de una tarjeta de prepago llamada "Qrobús". Esta tarjeta tiene como finalidad el agilizar y, sobre todo, facilitar el acceso al transporte público por parte de los usuarios, así como también el evitar asaltos que las personas de la ciudad puedan llegar a sufrir (Flores, E. C. (s.f.)).



**Figura 4.4 Diseño de la tarjeta general**

**Fuente:** <https://super9.com.mx/2021/04/28/definen-iqt-qrobús-y-sindicato-de-operadores-acciones-para-fomentar-el-uso-de-la-tarjeta-de-prepago/>

Esta tarjeta cuenta con una variación llamada "Tarjeta Qrobús preferente", la cual dispone de la posibilidad de brindar un costo del transporte público con preferencia a la hora de abordar a los estudiantes, personas con discapacidad y adultos mayores. Estas tarjetas ayudan a tener una mayor diversidad

dentro del sistema de transporte público de la ciudad, permitiendo a las personas que cumplan con dichas preferencias mayor accesibilidad a la hora de abordar y pagar el transporte público.

El transporte público de la ciudad cuenta con terminales que permiten a los usuarios proceder con el pago del transporte público sin la necesidad de pagar con efectivo o de tener que esperar para abordar para hacer el pago, consiguiendo así el objetivo principal por el que el sistema fue creado, agilidad y facilidad al abordar.

Los usuarios tienen mucha facilidad de recargar el saldo de su tarjeta, pues en diferentes puntos de la ciudad se encuentran cajeros especiales que les permiten consultar o recargar su saldo.

Del mismo modo, los usuarios cuentan con la posibilidad de recargar saldo a su tarjeta Qrobús mediante el uso de una aplicación móvil, que, además, les permite realizar algunas otras acciones, como consultar su saldo o el historial de viajes realizados<sup>3</sup>.

---

<sup>3</sup> <https://www.iqt.gob.mx/>

## **4.2 Determinación de los módulos del sistema propuesto**

En este apartado, se presenta a los módulos que serán incorporados en el sistema propuesto para hacer posible la realización del pago del transporte público mediante el uso de la tecnología NFC.

### **Módulo de registro e inicio de sesión**

Modulo encargado de la verificación de los datos proporcionados por el usuario dentro de la aplicación móvil antes del acceso a la interfaz principal de la aplicación. Para el inicio de sesión en la aplicación, el usuario deberá ingresar las credenciales de su cuenta previamente registrada. Tras el ingreso de las credenciales del usuario y si estas son correctas, el usuario podrá acceder a la aplicación móvil para hacer usos de sus funciones.

Por su parte, para realizar el registro de una cuenta de usuario nueva, los interesados deberán ingresar en la aplicación datos como su nombre de usuario, correo electrónico y una contraseña, para que, de esta manera, de ser validos los datos ingresados y que estos no se encuentren asociados a algún usuario previamente registrado, se proceda con el registro de la cuenta.

## **Módulo de cobro**

Se encuentra integrado por el sistema electrónico que permitirá a los conductores del transporte público realizar el cobro del pasaje a los usuarios que requieran hacer uso de este. Es un componente imprescindible para que el sistema de pago funcione correctamente, pues sin el sistema electrónico no existirá alguna forma en la que se le pueda descontar el crédito a los usuarios.

Este módulo tiene como tarea principal que cada que se detecte una nueva entrada dentro del sistema electrónico, un programa en Python reciba los datos de esta transacción mediante el puerto serie, solicitando a su vez una consulta a la base de datos del sistema para verificar que exista registro del UID de la tarjeta en algún usuario y posteriormente realizar el cobro del pasaje correspondiente descontándolo del crédito del usuario asociado a la tarjeta.

Este subsistema será en encargado de solicitar al sistema principal el que se realice el almacenamiento en la base de datos de la fecha de cada transacción para tener un historial de cada ocasión que se haya realizado el cobro de crédito al usuario.

Del mismo modo, este subsistema será el encargado de solicitar la actualización del crédito de los usuarios por cada transacción realizada, para de esta manera tener en todo momento el crédito real con el que dispone el usuario.

### **Módulo de gestión de crédito e historial**

Los usuarios que cuenten con una sesión iniciada dentro de la aplicación móvil podrán verificar datos importantes dentro de la interfaz de usuario. Algunos de los datos que podrán consultar son: ID del usuario, nombre de usuario, crédito, entre otros. Cada que se presione un botón, se realizará una nueva consulta a la base de datos, la cual devolverá los datos correspondientes al usuario que realiza la solicitud.

### **Módulo de Agregar Tarjeta**

Por medio de este módulo los usuarios tendrán a su alcance la oportunidad de agregar una nueva tarjeta, en caso de que así lo deseen por gusto o porque la tarjeta que actualmente cuentan registrada haya sufrido algún daño o se les haya extraviado. Para esto se requerirá que el usuario ingrese el UID (Identificador Único) que disponga la nueva tarjeta que haya adquirido. Posteriormente, esta tarjeta y su UID serán enlazados a los datos del usuario dentro de la base de datos,

para que este pueda hacer uso de la tarjeta para realizar pagos.

Es importante hacer mención que una misma tarjeta no podrá ser usada por más de un usuario. Además, un usuario únicamente podrá contar con una tarjeta activa al mismo tiempo y el crédito de un usuario se encontrará asociado a su cuenta y no a su tarjeta, por lo que, aunque se extravíe la tarjeta no se perderá el crédito con el que el usuario cuente.

### ***4.3 Selección del Hardware y Software para el sistema***

En este apartado de la investigación se dará a conocer el hardware y el software que se utilizará para el correcto desarrollo de este trabajo de investigación. Del mismo modo, se describirá la razón que llevo a hacer uso en específico de cada una de estas herramientas.

#### ***4.3.1 Hardware***

##### **A) Arduino**

Arduino fue desarrollado por un equipo de estudiantes en Italia, liderado por Massimo Banzi. Es una plataforma de desarrollo de prototipos electrónicos de hardware libre. Permite a cualquier persona, ya sea principiante o profesional, el diseñar y construir proyectos electrónicos, esto debido a



que Arduino fue diseñado para ser fácil de usar y ser muy accesible. Arduino se compone por una placa de circuito impreso con un microcontrolador, un entorno de desarrollo integrado y una amplia variedad de componentes electrónicos que complementan de mejor manera la realización de los proyectos electrónicos (Desai. P, 2008).

Características técnicas:

- Arduino se basa en un microcontrolador, el cual permite controlar dispositivos y sensores de entrada y salida.
- Arduino puede ser programado mediante su entorno de desarrollo integrado de código abierto
- Arduino cuenta con diversas variantes, cada una cuenta con sus características específicas que se adaptan a distintos tipos de proyectos

En este trabajo de investigación se utilizará la placa de desarrollo Arduino Mega, pues esta cuenta con un gran número de pines de entrada/salida y memoria, esto con el fin de poder llevar a cabo proyectos de tamaño mediano a grande con una baja complejidad. En total, esta placa cuenta con 54 pines digitales de entrada/salida, además, cuenta con 4 puertos de comunicación serial (UART) y con una velocidad de reloj de 16 MHz.

## **B) Modulo MFRC522 (RFID)**

Es una tecnología diseñada para leer, identificar y capturar datos mediante el uso de radiofrecuencia. Constan de un transmisor, un receptor y de una antena, la cual se encargará de identificar la etiqueta y será capaz de la lectura del contenido de la etiqueta a distancia (Guo, Y., Fang, X., & Liu, J., 2020).

En este proyecto de investigación, este módulo será utilizado para la identificación y autenticación de los usuarios en el sistema electrónico. Del mismo modo, con este dispositivo, será posible habilitar al sistema para realizar el cobro del pasaje a los pasajeros y la actualización del nuevo crédito a estos.

## **C) LED**

Es un diodo emisor de luz que se utiliza comúnmente como un indicador visual dentro de sistemas electrónicos. Es una manera simple y eficiente de mostrar si una salida digital esta activada o desactivada (Según Pedrera, A. C., 2017).

Su propósito en este trabajo de investigación es que el LED se encienda cada que el sistema electrónico propuesto reciba una señal UID desde el NFC en la aplicación móvil, para de esta manera tanto el operario como el usuario comprueben visualmente que la tarjeta fue reconocida correctamente por el sistema.

### **4.3.2      Software**

#### **A) XAMPP**

Desarrollado por Apache Friends, es un paquete de software gratuito y de código abierto, el cual proporciona los atributos necesarios para la creación y gestión de servidores web (En Bou, R. C., 2019).

XAMPP resulta sumamente útil para el desarrollo de webs que necesiten la creación de aplicaciones y sitios web en su entorno local antes de la delegación en un servidor en vivo. Dentro de la interfaz gráfica de XAMPP se permite iniciar y detener el servidor web, gestionar bases de datos y configurar la seguridad del servidor.

#### **B) Sistema de gestión de base de datos (MySQL)**

MySQL es un software de gestión de base de datos de código abierto que se dio a conocer a finales de los 90's. Este permite almacenar, organizar y gestionar la información en forma de tablas relacionadas entre sí. Utiliza el lenguaje de consulta estructurado (SQL) para la interacción con bases de datos, permitiendo la realización de operaciones, tales como la inserción, actualización, eliminación y consulta de datos.

De igual manera, permite la creación de índices para la aceleración de la búsqueda de datos y la creación de vistas para proporcionar una representación personalizada de los datos.

En Combaudon, S. (2018) se hace mención que MySQL es el sistema de gestión de base de datos de código abierto más popular del mundo, esto debido a que tiene la fama de ser conocido por su buen rendimiento y fiabilidad, ganando cada vez más popularidad por parte de los grandes protagonistas de la web, siendo que la mayoría de los sitios web más demandados en la actualidad, como lo son las redes sociales o portales comunitarios, han alcanzado un alto nivel de rendimiento por medio del uso de MySQL.

### **C) Entorno de Desarrollo de Integrado (IDE)**

Se trata de un software que se utiliza para la escritura, depuración y la compilación de código de programación en un solo lugar, permitiendo a los programadores el desarrollar códigos de software de una manera más eficiente y productiva (DataScientest, 2023).

A continuación, se describirá a los distintos IDE que fueron utilizados para el desarrollo de este trabajo de investigación:

- **Android Studio**

Es un entorno de desarrollo integrado (IDE), su utilización se basa en la creación de aplicación para teléfonos celulares con el sistema operativo Android. Incluye herramientas que permiten a los programadores la creación de interfaces de usuario, la codificación de las aplicaciones, la depuración y la prueba de estas aplicaciones para su correcto funcionamiento (Castillo, J. D., 2019).

De igual manera, este IDE cuenta con la característica de contar con un emulador de un teléfono celular para que se puedan probar las aplicaciones sin la necesidad de conectar nuestros dispositivos en todo momento en la computadora, obteniendo así la oportunidad de probar las aplicaciones en tiempo real.

Android Studio también ofrece herramientas avanzadas para el diseño de interfaces de usuario y la optimización del rendimiento de la aplicación.

Estas herramientas incluyen el editor de diseño visual, analizador de rendimiento, el depurador y la integración con GitHub y otras herramientas de control de versiones. Una ventaja de Android Studio es que actualmente cuenta con una comunidad activa de desarrolladores, lo cual proporciona una amplia variedad de recursos.

Sera el software principal de este trabajo de investigación, mediante este IDE se realizará la aplicación que les permitirá a los usuarios el registrarse o iniciar sesión en la aplicación móvil, así como, visualizar los datos importantes, como lo son los datos de los usuarios y el crédito con el que estos cuentan, entre algunas otras funcionalidades del sistema propuesto.

- **Arduino IDE**

Entorno de desarrollo integrado que se utiliza para la creación o carga de código en las placas Arduino. De igual manera, este IDE puede utilizarse para tener la posibilidad de cargar programas en las placas de desarrollo de otros proveedores (Peña, C., 2020).

Arduino IDE utiliza un lenguaje de programación que se basa en C/C++, el cual es fácil de aprender y de desarrollar para la mayoría de los desarrolladores que trabajan con esta herramienta.

Este IDE se utilizará en este trabajo de investigación para la programación del sistema electrónico.

- **Pycharm**

Este es el IDE más popular para el desarrollo y programación en Python, desarrollado por JetBrains en 2010. Pycharm se convirtió en el IDE más popular debido a su sencillez, otorgando a los programadores de software en Python una mayor eficacia y rapidez al programar sus aplicaciones, pues cuenta con módulos y paquetes que facilitan todo esto (DataScientest, 2022).

En este trabajo de investigación, el IDE Pycharm será utilizado para la programación del lenguaje de programación Python, el cual permitirá la conexión del sistema electrónico con la base de datos del sistema propuesto.

- **Visual Studio Code**

Visual Studio Code fue desarrollado por Microsoft, se trata de un editor de código fuente optimizado, proporcionando facilidad a los programadores el editar, depurar y la prueba de código. Permitiéndole a los programadores obtener un mayor grado de eficacia, se caracteriza por su flexibilidad, su enfoque ligero y la amplia gama de extensiones para personalizar y adaptar el software a las necesidades del desarrollador (Flores, F., 2023).

En este software se programará todo lo relacionado a la programación de PHP, lo cual permitirá la conexión entre la aplicación y sus acciones dentro de ella con la base de datos del sistema, permitiendo realizar acciones como, por ejemplo, el insertar un nuevo usuario u obtener los datos de un usuario en particular.

#### **D) Lenguajes de programación**

Descripción de los lenguajes de programación empleados en este trabajo de investigación para el desarrollo del sistema propuesto.



- **PHP**

Según Arias, M. A. (2013) Fue creado en 1994 por Rasmus Lerdorf. Es un lenguaje de programación de código abierto, el cual está diseñado específicamente para el desarrollo de aplicaciones web dinámicas y para sitios web.

PHP cuenta con las características de facilidad, escalabilidad, capacidad para trabajar con múltiples plataformas, la gran cantidad de librerías y Frameworks disponibles para el desarrollo de aplicaciones. De igual manera, PHP es un lenguaje interpretado, lo que quiere decir que no es necesario que el código sea compilado antes de ser ejecutado, ocasionando así que sea más fácil de aprender y utilizar para muchos desarrolladores.

PHP también cuenta con la capacidad para interactuar con bases de datos relacionales, permitiendo a los desarrolladores la posibilidad de crear aplicaciones web dinámicas y escalables que puedan adaptarse a diferentes necesidades y requisitos.

- **Python**

Lenguaje de programación de alto nivel, creado y desarrollado en 1989 por Guido van Rossum. En la actualidad, es considerado uno de los lenguajes de programación más populares y utilizados, conocido por su sintaxis simple y legible, fácil de aprender para principiantes y de cierta forma, más atractivo para desarrolladores experimentados (Montoro, A. F., 2013).

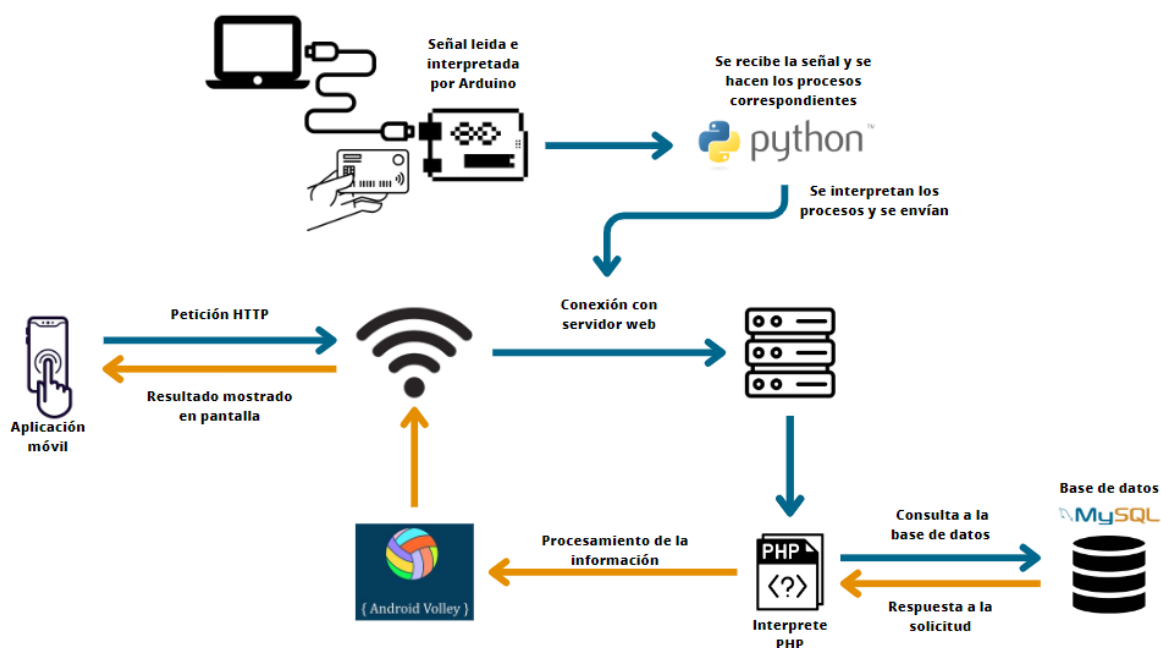
Python tiene como principales características que es interpretado y multiplataforma, siendo que no es necesario compilar el código para su ejecución, teniendo un intérprete encargado de leer el fichero fuente y ejecutarlo. Esta característica permite a los desarrolladores la posibilidad de ejecutar el código en distintas plataformas y sistemas operativos sin tener la necesidad de modificar el código fuente.

Este lenguaje es utilizado en una amplia variedad de aplicaciones, que van desde la creación de scripts y herramientas de línea de comandos hasta el desarrollo de aplicaciones web. De igual manera, Python cuenta con una amplia biblioteca estándar, esto significa que dispone de una gran cantidad de módulos y librerías que hacen que el desarrollo de aplicaciones y herramientas sea aún más fácil y rápido.

Cabe hacer mención que Python es un lenguaje de programación orientado a objetos, lo que significa que se basa en el concepto de clases y objetos, permitiendo la programación modular y el mantener una buena organización del código.

### 4.3 Esquema general del sistema propuesto

A través del esquema del sistema propuesto (Figura 4.5) se busca que el lector tenga una visión más clara del funcionamiento de la aplicación y el cómo esta se comunica con el sistema electrónico para que se pueda llevar a cabo la tarea que se le está demandando, siendo en este caso la de facilitar el cobro del pasaje a los usuarios con base en el crédito con el que estos cuentan en sus cuentas.



**Figura 4.5 Esquema del funcionamiento del sistema**  
Fuente: Elaboración propia

#### **4.3.3      *Funcionalidad del sistema***

La funcionalidad del sistema se puede describir en cuatro puntos principales que permiten se logró el cometido del sistema propuesto. Estos puntos son presentados a continuación:

1. **Pago del transporte.** Logrado a través de un sistema electrónico constituido por una placa Arduino Mega con un módulo RFID para la lectura de etiquetas, así como, leds para identificar los distintos estados del sistema.
2. **Acceso a la aplicación móvil.** El usuario accederá a la aplicación móvil, ya sea iniciando sesión si es que ya posee una cuenta, o creando una nueva a través de su registro desde la misma plataforma. Por medio de la aplicación los usuarios podrán realizar distintas acciones relacionadas a su cuenta.
3. **Gestión de cuentas.** Una vez iniciada la sesión del usuario, este podrá visualizar en la pantalla principal de la aplicación datos importantes, como lo son: el nombre de usuario, su ID, el crédito disponible, entre otros datos. El usuario también podrá asociar a su cuenta una tarjeta con un UID, que le servirá como un identificador para realizar movimientos y conocer su historial de operaciones en el sistema.

4. **Actualización de crédito.** Existen dos formas de actualizar el crédito con el que cuenta un usuario, la primera de ellas ocurre cuando el sistema electrónico detecte una entrada UID, en la que se determine que el código se encuentra asociado a algún usuario que cuente con el crédito suficiente para realizar la transacción. De ser así, se procederá a realizar el cobro del pasaje y se descontará el crédito al usuario correspondiente. La segunda forma en que se actualizará el crédito de un usuario ocurre si este realiza la adquisición de crédito extra desde la aplicación móvil. El crédito que sea adquirido desde la aplicación móvil podrá ser usado de manera inmediata una vez que se haya procesado la transacción de compra.

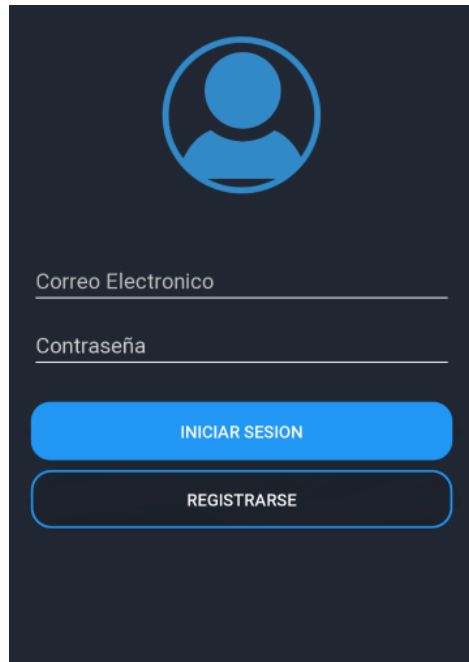
#### ***4.5 Diseño del sistema electrónico y de las interfaces gráficas***

En este apartado del trabajo de investigación se presentará a las interfaces graficas con las que el usuario interactuará para poder llevar a cabo el proceso de inicio o registro de sesión, así como, para la visualización de sus datos en la aplicación móvil y realización del resto de funcionalidades de la aplicación móvil. De igual manera, se verá el diseño que dispondrá el sistema electrónico para poder trabajar con el módulo RFID y con los leds.

#### **4.5.1 Diseño de las interfaces graficas**

El objetivo principal de las interfaces graficas es el garantizar un diseño amigable y fácil de utilizar para los usuarios. Se estableció a la pantalla de inicio de sesión como la primera pantalla que visualizará el usuario al entrar a la aplicación. En esta pantalla al ingresar sus datos el usuario contará con acceso a la aplicación móvil y podrá acceder a las funcionalidades del resto de pantallas que integran a la aplicación.

En la Figura 4.6 se muestra la interfaz gráfica en la cual el usuario podrá iniciar sesión. La interfaz es simple, cuenta con dos campos de texto que el usuario deberá rellenar con sus datos para proceder con al acceso a la aplicación móvil. Una vez ingresados los datos del usuario y dar clic en el botón "Iniciar Sesión" se verificará si la información del usuario existe dentro de la base de datos y de ser así se procederá a pasar a la pantalla principal de la aplicación.



**Figura 4.6 Interfaz de inicio de sesión**  
**Fuente: Elaboración propia**

Si un usuario intenta iniciar sesión sin colocar alguno de los datos dentro de los campos de texto, le aparecerá un mensaje de error, indicándole que deberá ingresar un correo electrónico o una contraseña, según el campo de texto que el usuario haya dejado vacío.

De igual manera, si un usuario intenta iniciar sesión con los datos de una cuenta que no haya sido registrada previamente en el sistema, se le notificara con un mensaje emergente, para que de esta manera el usuario rectifique sus datos, si es que ha cometido algún error al ingresar los datos, o en su defecto, que proceda a realizar el registro de su cuenta en la aplicación para que pueda comenzar a usar el sistema.

Para registrar una cuenta de usuario, los interesados deberán dar clic en el botón "Registrarse", el cual llevará a los usuarios a la interfaz correspondiente para realizar este procedimiento (Figura 4.7), donde deberán ingresar los datos solicitados para realizar su registro.

La imagen muestra una interfaz de usuario para el registro de una cuenta. El fondo es de color gris oscuro. En la parte superior, hay un ícono de una persona con una cruz blanca superpuesta. Debajo del ícono, hay cinco campos de texto con etiquetas grises: "Nombre", "Apellidos", "Correo electronico", "Contraseña" y "Confirmar Contraseña". Cada campo tiene una línea horizontal blanca para ingresar los datos. En la parte inferior, hay un botón rectangular de color naranja con el texto "CREAR CUENTA" en letras blancas mayúsculas.

**Figura 4.7 Interfaz de registro de sesión**  
**Fuente: Elaboración propia**

Al igual que en el inicio de sesión, si el usuario deja algún campo vacío al momento de ingresar los datos necesarios para el registro de sesión, se mandará un mensaje de error indicando que es necesario ingresar datos en todos los campos.



De igual manera, si el usuario intenta registrar una nueva cuenta con un correo electrónico ya existente, se le notificara con un mensaje emergente.

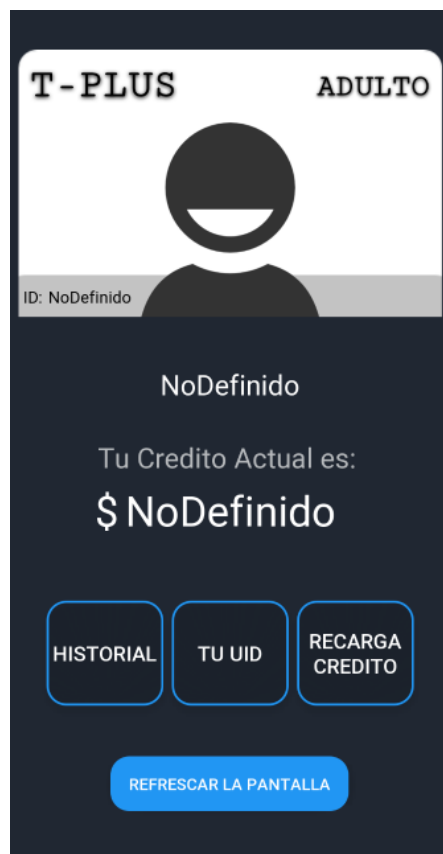
Si el usuario ingresa un correo electrónico valido y que no se encuentre previamente registrado, se procederá realizar una petición a la base de datos para almacenar la nueva cuenta. Una vez validada la conexión con la base de datos, se abrirá nuevamente la interfaz de inicio de sesión, mostrándole mediante un mensaje emergente al usuario que el registro se ha completado con éxito.

Cuando sea ingresado un correo electrónico y una contraseña que sean válidos y que existan registrados en la base de datos, mediante el inicio de sesión, se dará paso a la apertura de la interfaz gráfica principal del usuario (Figura 4.8), en la cual, se podrán visualizar a algunos datos obtenidos desde la base de datos, tales como lo son: ID del usuario, el nombre que proporcione y el crédito disponible en su cuenta.

Además, habrá cuatros botones que servirán para realizar acciones dentro de la aplicación móvil, como lo son: visualizar el historial de uso de la tarjeta, recargar crédito y visualizar el UID de la tarjeta vinculada a la cuenta, en donde

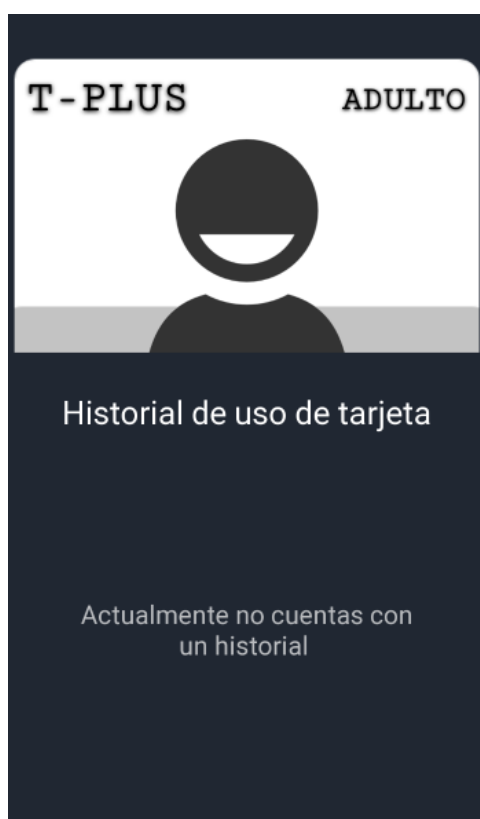
además de revisarlo, el usuario podrá cambiarlo por el de otra tarjeta o eliminarlo si ya no desea seguir haciendo uso del sistema.

Finalmente, el cuarto botón, permitirá al usuario el refrescar la pantalla, para que de esta manera los usuarios puedan visualizar sus datos actualizados en el momento en que así lo deseen.



**Figura 4.8** Interfaz gráfica principal de usuario  
**Fuente:** Elaboración propia

Al dar clic en el botón de "historial" el usuario podrá visualizar el historial de movimientos que su cuenta haya realizado con relación al pago del transporte público. Permitiendo así que se pueda visualizar el año, día, mes y la hora en la que se realizó un cobro exitoso a su cuenta (Figura 4.9) .



**Figura 4.9 Historial de usos de la tarjeta**  
**Fuente: Elaboración Propia**

La visualización del historial de un usuario dependerá si este ha realizado o no algún pago con su tarjeta dentro del sistema electrónico, de no ser así, se le mostrara al usuario un cuadro de texto indicándole que no cuenta con un historial.

La interfaz gráfica de "Tu UID" (Figura 4.10), servirá para que el usuario que ha iniciado sesión dentro de la aplicación móvil pueda visualizar, actualizar o eliminar el UID que esta enlazado a su cuenta. El UID será el encargado de que el usuario tenga enlazada una tarjeta para que este pueda pagar el transporte público.

Si el usuario ya cuenta con un UID, este lo visualizará dentro de la caja de texto correspondiente, sustituyendo a la etiqueta "NoDefinido" por el UID asociado a la cuenta del usuario. Además, podrá actualizar o eliminar el UID. Los campos de texto "UID" y "Confirmar UID" se harán invisibles y se deshabilitarán, a excepción de que el usuario de clic en "Actualizar UID", haciendo que los campos de texto se hagan visibles y el cuadro de texto en el que aparecería el UID se vuelva invisible.

Para el caso de que el usuario no cuente con UID enlazado, entonces este podrá ingresar uno, para esto, los campos de texto "UID" y "Confirmar UID" se harán visibles y se habilitaran, además, el cuadro de texto correspondiente a mostrar el UID al usuario se volverá invisible. El usuario deberá colocar el UID y confirmarlo en los campos de texto, una vez hecho esto, podrá actualizar o eliminar el UID.

Al igual que en las interfaces anteriores, si el usuario deja campos vacíos, se le notificará con un error dentro de los campos de texto. El dato de UID se actualizará en la base de datos del usuario con respecto a la acción realizada.



**Figura 4.10 UID de usuario**  
**Fuente: Elaboración Propia**

La interfaz gráfica para la recarga de crédito (Figura 4.11), hará posible a los usuarios el realizar una recarga al crédito de sus cuentas, permitiendo ingresar el crédito que más se les acomode entre las 6 opciones que tienen (10, 20, 30, 50, 200 o 500 pesos).

Debido a las limitaciones de este trabajo de investigación, el crédito se actualizará directamente a la base de datos con respecto a la interfaz gráfica presentada para la recarga de crédito sin que se haya realizado en algún momento el cargo en la cuenta bancaria de los usuarios. Esto debido a que actualmente no se cuenta con una pasarela de pagos para hacer posible la integración de pagos con algún banco o con algún proveedor como Mercado pago o PayPal.

Una vez el usuario coloque la cantidad a recargar, se mandará esa cantidad a la base de datos y se le actualizara el crédito disponible, sumándole el crédito que el usuario selecciono dentro de la interfaz a los datos de su cuenta.

RECARGA DE CREDITO

\$ 0

\$10 \$20 \$50

\$100 \$200 \$500

Añade una tarjeta de credito

Numero de la Tarjeta

MM/AAAA CVV

CONFIRMAR

**Figura 4.11 Recarga de crédito**  
**Fuente: Elaboración Propia**

#### **4.5.2 Diseño del sistema electrónico**

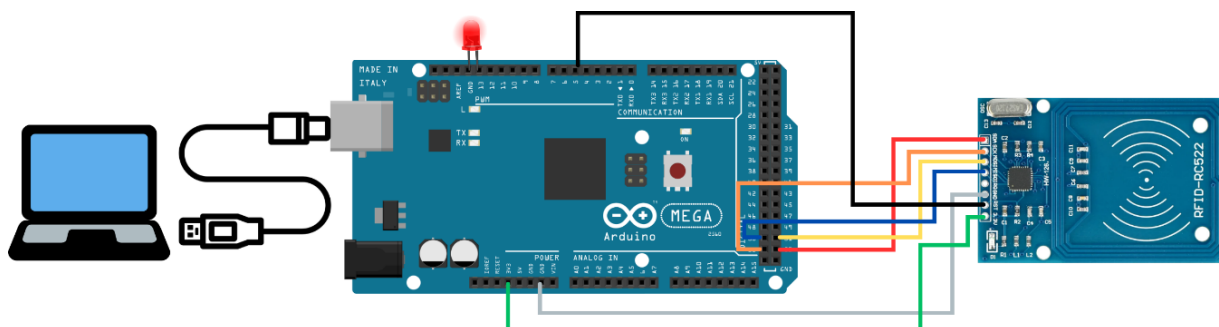
A continuación, en la Tabla 4.1 se presenta a la disposición con la que contara cada PIN del Arduino Mega para su correcta conexión con el módulo MFRC522, indicando de igual manera, el color del cable con el que fue ensamblado para un mejor entendimiento del esquema.

**Tabla 4.1 Disposición de la conexión del módulo MFRC522 con la placa de Arduino Mega**

| <b>Modulo</b> | <b>Arduino Mega</b> | <b>Color cable</b> |
|---------------|---------------------|--------------------|
| SDA           | 53                  | rojo               |
| SCK           | 52                  | naranja            |
| MOSI          | 51                  | amarillo           |
| MISO          | 50                  | azul               |
| GND           | GND                 | blanco             |
| RST           | 5                   | negro              |
| 3.3V          | 3.3V                | verde              |

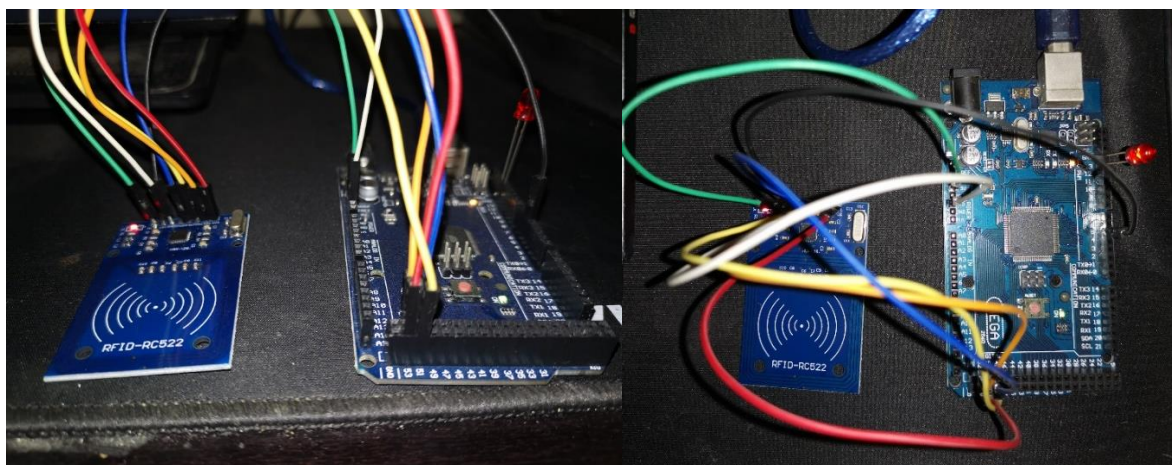
**Fuente: Elaboración propia**

Posteriormente, una vez indicada la disposición con la que contarán los pines del Arduino, en las Figura 4.12 se muestra el diseño del sistema electrónico con el que contará este trabajo de investigación, el cual será el encargado de la detección y lectura de cada UID de la tarjeta entrante.



**Figura 4.12 Diseño del sistema electrónico**  
**Fuente: Elaboración propia**

Finalmente, en la Figura 4.13, se presenta al sistema electrónico físico que fue armado para este trabajo.



**Figura 4.13 Sistema electrónico**  
**Fuente: Elaboración propia**

#### **4.6 Programación del sistema propuesto**

Para este trabajo de investigación, se consideró pertinente el uso de dos tablas, esto debido a que estas dos cumplen con lo necesario para completar los objetivos de este trabajo de investigación.




La primera tabla se encargará de almacenar los datos básicos de los usuarios y la otra administrará el uso de las tarjetas de los usuarios, permitiendo así la generación del historial de uso. Para contener a las tablas utilizadas en este trabajo de investigación se hará uso de una base de datos en MySQL utilizando XAMPP.

La primera tabla de la base de datos lleva de nombre "usuarios" (Figura 4.14), y esta se encuentra compuesta por siete campos, los cuales serán los encargados de almacenar la información de los usuarios. Los campos considerados son:

- Id: identificador único de usuario, auto incrementable.
- Nombre: nombre del usuario.
- Apellido: apellidos del usuario.
- Correo: correo electrónico del usuario.
- Contraseña: contraseña de inicio de sesión.
- Crédito: crédito que dispone cada usuario en su cuenta.
- UID: Correspondiente al identificador único de la tarjeta de usuario.

Se considero que al momento de registrar una nueva cuenta esta de manera predeterminada tendrá un UID con el valor "0", haciendo esto alusión a que dicha cuenta aún no posee un UID valido para comenzar a realizar pagos con el sistema. Dicho UID inicial podrá ser cambiado en cualquier momento por el usuario para ingresar el UID correspondiente a la tarjeta que usará para realizar sus pagos en el transporte público.

Del mismo modo y con fines de poner a prueba al sistema propuesto, se consideró establecer de manera predeterminada al crédito de los usuarios nuevos en 1000, de tal manera que esto permita realizar un gran número de pruebas con el sistema a través de las cuales se busca validar la funcionalidad de este.

| # | Nombre   | Tipo        | Cotejamiento       | Atributos | Nulo | Predeterminado | Comentarios | Extra          |
|---|--|-------------|--------------------|-----------|------|----------------|-------------|----------------|
| 1 | id  | int(15)     |                    |           | No   | Ninguna        |             | AUTO_INCREMENT |
| 2 | nombre   | varchar(50) | utf8mb4_general_ci |           | No   | Ninguna        |             |                |
| 3 | apellido   | varchar(50) | utf8mb4_general_ci |           | No   | Ninguna        |             |                |
| 4 | correo   | varchar(50) | utf8mb4_general_ci |           | No   | Ninguna        |             |                |
| 5 | contraseña   | varchar(15) | utf8mb4_general_ci |           | No   | Ninguna        |             |                |
| 6 | credito  | int(255)    |                    |           | No   | 1000           |             |                |
| 7 | uid  | varchar(20) | utf8mb4_general_ci |           | Sí   | 0              |             |                |

**Figura 4.14 Estructura de la tabla "usuarios"**  
**Fuente: Elaboración propia**

La segunda tabla de la base de datos lleva de nombre "Historial" (Figura 4.15), y esta será la encargada de almacenar el historial de uso de la tarjeta a la hora de realizar el cobro del transporte público por parte de los usuarios. Esta tabla contiene tres campos:

- ID: Identificador para conocer el número de la entrada de datos
- ID Usuario: Identificador único de usuario al que se encuentra asociado el UID de la transacción realizada
- UID: Correspondiente al identificador único de la tarjeta entrante
- Fecha: Fecha en la que fue registrada la entrada de la tarjeta a la base de datos

| # | Nombre   | Tipo         | Cotejamiento       | Atributos | Nulo | Predeterminado      | Comentarios | Extra          |
|---|--|--------------|--------------------|-----------|------|---------------------|-------------|----------------|
| 1 | id  | int(11)      |                    |           | No   | Ninguna             |             | AUTO_INCREMENT |
| 2 | id_usuario   | int(50)      |                    |           | No   | Ninguna             |             |                |
| 3 | uid  | varchar(255) | utf8mb4_general_ci |           | No   | Ninguna             |             |                |
| 4 | fecha  | timestamp    |                    |           | No   | current_timestamp() |             |                |

**Figura 4.15 Estructura de la tabla "historial"**  
**Fuente: Elaboración propia**

Las tablas "historial" y "usuarios" se encuentran relacionadas entre sí a través del campo "id\_usuario".

En los siguientes apartados se mostrará a los fragmentos de código más relevantes para la correcta funcionalidad del sistema propuesto en este trabajo de investigación.

#### **A. Registro de una nueva cuenta**

Para que un usuario pueda registrarse en la aplicación es necesario que se ingrese un nombre de usuario, un correo electrónico y una contraseña. Además, se estableció que el nombre de usuario y el correo electrónico deben ser únicos para proceder con el registro, es decir, en la base de datos no debe haber un usuario repetido.

#### **Códigos en los distintos IDE**

##### **1) Android Studio**

En este apartado se logra obtener los valores ingresados por el usuario en el cuadro de texto para obtener su nombre, apellido, correo electrónico, contraseña y confirmar contraseña, mediante estos datos, se valida que ninguno de los campos de texto este vacío. Mediante el uso de "isEmpty" se corrobora si está vacío o no, de ser vacío se manda un mensaje de error en el campo correspondiente. De igual manera, se corrobora que las contraseña coincidan entre sí (Figura 4.16).

```

private void insertarDatos(){
    final String nombre = mUsuario.getText().toString().trim();
    final String apellido = mApellido.getText().toString().trim();
    final String correo = mCorreo.getText().toString().trim();
    final String contraseña = mContraseña.getText().toString().trim();
    final String confContra = mConfirmarContraseña.getText().toString().trim();

    if(nombre.isEmpty()) {
        mUsuario.setError("Introduce tu nombre");
        return;
    }else if (apellido.isEmpty()){
        mApellido.setError("Introduce tus apellidos");
        return;
    }else if(correo.isEmpty()){
        mCorreo.setError("Introduce tu correo electronico");
        return;
    }else if(contraseña.isEmpty()){
        mContraseña.setError("Introduce una contraseña");
        return;
    }else if(confContra.isEmpty()){
        mConfirmarContraseña.setError("Confirma tu contraseña");
        return;
    }else if(!contraseña.equals(confContra)){
        mConfirmarContraseña.setError("Las contraseñas no coinciden");
        return;
    }
}

```

**Figura 4.16 Código para validar datos en el registro de cuentas nuevas**

**Fuente: Elaboración Propia**

Posteriormente, después de comprobar que ningún campo de texto este vacío, mediante el uso de la biblioteca "Volley" se crea una solicitud "POST" mediante la URL correspondiente para la inserción de los datos en la base de datos. Los métodos "onResponse" y "onErrorResponse" son los encargados de manejar la solicitud al servidor (Figura 4.17).

```

        String request = new StringRequest(Request.Method.POST,
        "http://192.168.1.68:8080/tesis_/register.php", new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                if (response.equalsIgnoreCase("El registro se ha completado
        Exito")) {
                    Toast.makeText(Registro.this, response,
        Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(Registro.this, MainActivity.class);
                    startActivity(intent);
                } else {
                    Toast.makeText(Registro.this, response,
        Toast.LENGTH_SHORT).show();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Toast.makeText(Registro.this, error.toString(),
        Toast.LENGTH_SHORT).show();
            }
        })
    }
}

```

**Figura 4.17 Código para realizar una petición POST de registro de cuentas**

**Fuente: Elaboración Propia**

Finalmente, una vez establecida la conexión con la base de datos, un objeto "Map" será el encargado de almacenar los datos ingresados por el usuario y enviarlos para que la base de datos pueda almacenarles (Figura 4.18).

```

    @Override
    protected Map<String, String> getParams() throws AuthFailureError {
        Map<String, String> param = new HashMap<String, String>();
        param.put("nombre", nombre);
        param.put("apellido", apellido);
        param.put("correo", correo);
        param.put("contraseña", contraseña);
        return param;
    }
};
RequestQueue requestQueue = Volley.newRequestQueue(Registro.this);
requestQueue.add(request);
}
}

```

**Figura 4.18 Encapsulado de datos para registrar una cuenta**

**Fuente: Elaboración Propia**

## 2) PHP

Se comienza con el código necesario para realizar una conexión con la base de datos en MySQL, para lo que se requiere establecer el nombre de la base a datos, el del servidor, el usuario y la contraseña. Con estos datos se procede a realizar la conexión con la base de datos y de ocurrir algún error, se manda un mensaje advirtiéndolo al usuario de esto (Figura 4.19).

```
<?php

$servername = 'localhost';
$username = 'root';
$password = '';
$dbname = 'bd_tesis';

$conexion = new mysqli($servername, $username, $password, $dbname);

. if($conexion -> connect_errno){
.     echo "Ha ocurrido un error en la conexion";
. }
```

**Figura 4.19 Código en PHP para establecer una conexión con la base de datos**

**Fuente: Elaboración Propia**

Posteriormente, se realiza la solicitud a la base de datos con respecto a los datos obtenidos desde Android Studio, primero el archivo PHP hace una solicitud "SELECT" para verificar que no exista ya un usuario registrado previamente con el mismo correo electrónico. De no encontrar un registro previo con el mismo usuario, entonces se realiza un "INSERT", el cual almacenará los datos de nombre, apellido, correo y contraseña a la base de datos (Figura 4.20).

```

$nombre = $_POST['nombre'];
$apellido = $_POST['apellido'];
$correo = $_POST['correo'];
$contraseña = $_POST['contraseña'];

$query = "SELECT * FROM usuario WHERE correo = '$correo'";
$resultado = mysqli_query($conexion, $query);

if($resultado -> num_rows > 0){
    echo "El correo o el usuario ingresado ya fue registrado";
}else{
    $query = "INSERT INTO usuario
(nombre,apellido,correo,contraseña) values
('$nombre','$apellido','$correo','$contraseña')";

    if(mysqli_query($conexion, $query)){
        echo "El registro se ha completado Exitosamente";
    }else{
        echo "Ha ocurrido un Error al registrar";
    }
}
?>

```

**Figura 4.20 Código en PHP para el registro de una cuenta**  
**Fuente: Elaboración Propia**

## **B. Inicio de sesión**

Un usuario podrá iniciar sesión en la aplicación móvil siempre y cuando este disponga de un registro previo. Para poder iniciar sesión es necesario un correo electrónico o nombre de usuario junto con su contraseña. De lo contrario, el proceso no podrá continuar debido a la falta de uno de estos campos, si no se cuenta con una cuenta, el usuario podrá registrarse.



**Códigos en los distintos IDE para lograr esta funcionalidad:**

### **1) Android Studio**

La parte de la programación para el inicio de sesión utilizó una variante del código para el registro de un nuevo usuario, con la excepción de que dentro del método "onResponse" se agregó un arreglo para almacenar datos obtenidos con la conexión con el servidor. Esto para obtener el nombre, el id y el UID para trabajar más fácil con los datos al momento de intercambiar información entre las distintas pantallas de la aplicación (Figura 4.21).

```
public void onResponse(String response) {  
    if (response.contains("Se ha iniciado sesión correctamente"))  
    {  
        String[] datos = response.split(",");  
        String nombre = datos[1];  
        String id = datos[2];  
        String uid = datos[3];  
        Intent intent = new Intent(getApplicationContext(),  
Inicio.class);  
        intent.putExtra("nombre", nombre);  
        intent.putExtra("id", id);  
        intent.putExtra("uid", uid);  
        startActivity(intent);  
    }  
}
```

**Figura 4.21 Validación para determinar si el inicio de sesión fue correcto**

**Fuente: Elaboración Propia**

Al igual que en el registro de una cuenta, se utilizó un objeto "Map" para mandar los parámetros "login" y "contraseña", que corresponden al correo electrónico y a la contraseña respectivamente (Figura 4.22).

```

        @Nullable
        @Override
        protected Map<String, String> getParams() throws AuthFailureError
    {

        Map<String, String> param = new HashMap<>();
        param.put("login", login);
        param.put("contraseña", contraseña);
        return param;
    }
};
RequestQueue requestQueue = Volley.newRequestQueue(MainActivity.this);
requestQueue.add(stringRequest);

```

**Figura 4.22 Mapeo de los datos para validar inicio de sesión**  
**Fuente: Elaboración Propia**

## 2) PHP

De igual manera, el código de PHP que está a continuación tiene la misma funcionalidad que el código PHP anterior, crear una conexión con la base de datos para ejecutar una consulta, con la diferencia que este código verifica si los campos ingresados por el usuario son válidos, es decir, verificar si existe algún registro en la base de datos que coincide con lo ingresado por el usuario desde la aplicación móvil, de ser así entonces se extraen los datos "nombre", "id" e "uid" concatenados en una cadena de texto separada por comas (Figura 4.23).

```

$login = $_POST['login'];
$contraseña = $_POST['contraseña'];

$query = "SELECT * FROM usuario WHERE correo = '$login' AND
contraseña = '$contraseña'";
$resultado = mysqli_query($conexion, $query);

if($resultado -> num_rows > 0){
    $usuario = mysqli_fetch_assoc($resultado);
    $nombre = $usuario['nombre'];
    $id = $usuario['id'];
    $uid = $usuario['uid'];
    echo "Se ha iniciado sesión correctamente," . $nombre . "," .
$id . "," . $uid;
}else{
    echo "Uno de los campos no es correcto";
}

?>

```

**Figura 4.23 Código para comprobar los datos de inicio de sesión**

**Fuente: Elaboración Propia**

### **C. Pantalla principal**

Es la pantalla principal de la aplicación, en donde los usuarios podrán gestionar su cuenta y visualizar sus datos en la misma, como lo son el nombre de usuario, el crédito y la ID.

#### **Código para lograr esta funcionalidad:**

Se utilizó una variante de los códigos anteriores en Android Studio para la actualización del crédito del usuario. La recuperación de los datos "nombre", "id" y "uid" se ven reflejados en esta nueva actividad, mismos que serán recuperados a través del método "getExtras()" y se almacenarán

en un objeto "Bundle", de tal forma que si el objeto no resulta ser nulo, entonces esos datos se almacenaran en variables y posteriormente se colocan en unos cajas de texto ("textView") (Figura 4.24).

```
Bundle extras = getIntent().getExtras();
if (extras != null) {
    nombre = extras.getString("nombre");
    id = extras.getString("id");
    uid = extras.getString("uid");
    nom.setText(nombre);
    idd.setText(id);
    uidd.setText(uid);
}
```

**Figura 4.24 Obtención de los datos generales de los usuarios**  
**Fuente: Elaboración Propia**

Mediante un botón el usuario podrá ejecutar la actualización de la pantalla principal. Al dar clic en dicho botón se llamará al método responsable de obtener el crédito del usuario, esto para tener siempre la posibilidad de visualizar los datos actualizados de la cuenta del usuario.

De igual manera, al pasar a las distintas pantallas del sistema (Historial, Tu UID y Recarga crédito), se utilizará el código presentado en la Figura 4.25, el cual además de hacer posible el cambio de pantallas, mandará a estas los datos correspondientes al ID y el UID del usuario, para que de esta manera las pantallas puedan realizar su función correctamente.

```

public void cambioPantalla (){
    String idValue = idd.getText().toString();
    String uidValue = uidd.getText().toString();
    Intent intent = new Intent(getApplicationContext(), Historial.class);
    intent.putExtra("idValue", idValue);
    intent.putExtra("uidValue", uidValue);
    startActivity(intent);
}

```

**Figura 4.25 Método para el cambio entre pantallas**  
**Fuente: Elaboración Propia**

#### **D. Historial**

Para el historial se hizo uso de una tabla específicamente destinada a almacenar la información relacionada con este en la base de datos. En esta tabla se añadirá un registro cada vez que se haga uso de la tarjeta y el sistema electrónico.

**Códigos en los distintos IDE para lograr esta funcionalidad:**

##### **1) Python**

En el código en Python, dentro del "While" se realizan las acciones necesarias para la inserción de los distintos movimientos de pago efectuados por los usuarios. Para esto, primeramente, se realiza una consulta a la tabla "usuario" con la finalidad de obtener el ID del usuario del que se registrarán los movimientos en el historial. Finalmente, se procede al registro de los movimientos en la base de datos (Figura 4.26).

```

while True:
    datos = ard.readline().decode().strip()
    if datos:
        print(datos)

        sql = f"SELECT id FROM usuario WHERE uid='{datos}'"
        mycursor.execute(sql)
        resultado = mycursor.fetchone()

        if resultado:
            id_user = resultado[0]

            sql_h = "INSERT INTO historial (uid, id_usuario)
VALUES (%s,%s)"
            val = (datos,id_user)
            mycursor.execute(sql_h, val)
            mydb.commit()

```

**Figura 4.26 Código para la inserción de registros asociados al historial de movimientos**  
**Fuente: Elaboración Propia**

## 2) PHP

Se requiere que se proporcione el ID del que se consultará el historial. De esta forma, este procedimiento recuperará a todas las fechas de los movimientos que se encuentren asociados al ID del usuario que desea hacer la consulta, mostrando al usuario de manera general los movimientos que ha tenido su cuenta. De esta forma, se incluye a los movimientos de todas las tarjetas que el usuario haya asociado alguna vez en su cuenta (Figura 4.27). Para un mejor entendimiento y análisis de los datos devueltos, las fechas se encontrarán compuestas por el día, mes, año y registro de hora completo en que se realizó cada movimiento.

Este código recuperará a los datos de las fechas en las que se ha realizado algún movimiento con respecto al ID único del usuario. La información recuperada será almacenada en un arreglo y se devolverá como un objeto JSON.

```
$id_usuario = $_POST['id_usuario'];

$query = "SELECT fecha FROM historial WHERE id_usuario = '$id_usuario'";
$resultado = mysqli_query($conexion, $query);

if ($resultado->num_rows > 0) {
    $dates = array();
    while ($row = $resultado->fetch_assoc()) {
        $dates[] = $row['fecha'];
    }
    echo json_encode($dates);
} else {
    echo "No se encontraron filas con el ID de usuario proporcionado.";
}
```

**Figura 4.27 Recuperación del historial asociado a un usuario**  
**Fuente: Elaboración Propia**

Las fechas del usuario correspondiente a su UID serán almacenadas dentro de un arreglo, esto para la comodidad de su procesamiento a la hora de mostrar el historial al usuario dentro de la aplicación.

Dentro del "While", se itera y se obtendrá la información del resultado como un arreglo asociativo, siendo las claves el nombre de la columna y los valores son los valores correspondientes a cada fila.

### 3) Android Studio

Mediante el método "obtenerHistorial" se hace posible la obtención del historial de usuario con base a su ID. Dentro de este método, se realiza una solicitud a la URL asignada al proceso de obtención de los datos del usuario. Posteriormente, el método se encarga de procesar la respuesta obtenida de la solicitud para mostrarla al usuario por medio de la pantalla "Historial (Figura 4.28).

```
private void obtenerHistorial (final String idValue) {  
  
    StringRequest stringRequest = new  
StringRequest(Request.Method.POST,  
"http://192.168.1.68:8080/tesis_/historial.php", new  
Response.Listener<String>() {  
        @Override  
        public void onResponse(String response) {  
            try {  
                JSONArray jsonArray = new JSONArray(response);  
                ArrayList<String> fechas = new ArrayList<>();  
  
                for (int i = 0; i < jsonArray.length(); i++) {  
                    fechas.add(jsonArray.getString(i));  
                }  
  
                ArrayAdapter<String> adapter = new  
ArrayAdapter<>(Historial.this, android.R.layout.simple_list_item_1,  
fechas);  
  
                listView.setAdapter(adapter);  
            }  
        }  
    }  
}
```

**Figura 4.28** Obtención del historial de los usuarios  
**Fuente:** Elaboración propia



Este método será llamado y ejecutado cada que el usuario ingrese a la pantalla "Historial", mostrándole de manera general, todo el historial de uso que su tarjeta o tarjetas han tenido desde la creación de la cuenta.

#### **E. Identificación y cobro de crédito mediante UID**

Este apartado del código es el encargado de leer el UID de una tarjeta, utilizando un módulo RFID y del cobro de crédito.

#### **Códigos en los distintos IDE para lograr esta funcionalidad:**

##### **1) Arduino IDE**

Este código fue elaborado con el objetivo de hacer posible reconocer al identificador único (UID) asociado a una tarjeta de identificación por radiofrecuencia (RFID) de algún usuario. Para esto, cuando el módulo MFRC522 detecte alguna tarjeta, el programa leerá cada byte del UID mediante un bucle y concatenará a estos en una cadena de texto llamada "uid". Los bytes leídos se encontrarán en formato hexadecimal.

La cadena obtenida como resultado será enviada a través de comunicación serial a la computadora en la que se encontrará el programa en Python destinado a procesar la cadena para el resto de funcionalidad del sistema. Cuando esto ocurra se

cambiará el estado del led empleado como medio de notificación visual.

Mientras que el módulo MFRC522 no detecte alguna tarjeta, este se encontrará en modo de suspensión para evitar que el código de Arduino se ejecute de manera indeterminada (Figura 4.29).

```
void loop() {  
  
    if (mfrc522.PICC_IsNewCardPresent() &&  
        mfrc522.PICC_ReadCardSerial()) {  
        // Lee el UID de la tarjeta  
        String uid = "";  
        for (byte i = 0; i < mfrc522.uid.size; i++) {  
            uid.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? "0" : ""));  
            uid.concat(String(mfrc522.uid.uidByte[i], HEX));  
        }  
        Serial.println(uid);  
        mfrc522.PICC_HaltA();  
  
        digitalWrite(LED_PIN, HIGH);  
        delay(1000);  
        digitalWrite(LED_PIN, LOW);  
    }  
}
```

**Figura 4.29. Código en Arduino para la lectura de las tarjetas de identificación de los usuarios**

**Fuente: Elaboración Propia**

## **2) Python**

El código en Python tiene como función principal el monitorear al sistema electrónico e identificar cuando exista información enviada por este a la computadora.

Para realizar el monitoreo y lectura de información de Arduino con Python se hará uso de un hilo de ejecución independiente e indeterminado, que constantemente se encontrará validando al sistema electrónico.

Para que el código en Python funcione correctamente es necesario establecer al puerto COM que será utilizado por el sistema electrónico. Del mismo modo, es necesario asegurarse que el servidor de base de datos se encuentre en línea y accesible.

En este sentido, cuando Python reciba algún UID desde el sistema electrónico, procederá a realizar una consulta a la base de datos para comprobar si existe algún usuario al que corresponda el UID identificado y posteriormente validará si este corresponde efectivamente a algún usuario válido, el que además cuente con el crédito suficiente, de cumplirse todas las condiciones se procederá a realizar el cobro del pasaje correspondiente, de tal manera que esto descontará al valor del pasaje al crédito con el que cuente el usuario (Figura 4.30).

```

def realizar_cobro(id_usuario, nombre_usuario, nuevo_credito):
    fecha = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    consulta = f"UPDATE usuario SET credito = {nuevo_credito},
    fecha = '{fecha}' WHERE id = {id_usuario}"
    cursor.execute(consulta)
    conexion.commit()
    print(f"Se realizo el cobro al usuario {nombre_usuario} con ID
    {id_usuario}, credito actual: {nuevo_credito}, fecha: {fecha}")

while True:
    datos = puerto.readline().decode().strip()
    ,
    if datos:
    ,     consulta = f"SELECT id, nombre, credito FROM usuario WHERE
    uid = '{datos}'"
    ,     cursor.execute(consulta)
    ,     resultado = cursor.fetchone()
    ,
    ,     if resultado:
    ,         id_usuario, nombre_usuario, credito = resultado
    ,         nuevo_credito = credito - 9
    ,         realizar_cobro(id_usuario, nombre_usuario,
    nuevo_credito)
    ,     else:
    ,         print("No se encontró al usuario en la base de datos")
    ,

```

**Figura 4.30. Código en Python para el cobro del pasaje del transporte público**  
**Fuente: Elaboración propia**

## **F. Administración de tarjeta enlazada**

Apartado encargado de mostrar el código clave para el agregar, actualizar o eliminar una nueva tarjeta mediante el UID.

**Códigos en los distintos IDE para lograr esta funcionalidad:**

### **1) Android Studio**

Para el código en Android Studio se utilizaron las variantes de los códigos anteriores con excepción de algunas secciones de Código que serán mostradas a continuación. En este, para saber qué es lo que vera el usuario en la interfaz gráfica, es necesario saber si cuenta con un UID, como se dijo

anteriormente en este trabajo de investigación, los usuarios recién registrados tendrán un "0" de manera predeterminada en la sección de su UID, entonces, primero se verifica si los datos son "0", de ser así entonces se llama al método "habilitarEdicion", el cual mostrara los campos de texto correspondientes par que el usuario ingrese un UID. De no ser así, entonces se llama al método "deshabilitarEdicion", el cual ocultara los campos de texto y únicamente mostrara el UID del usuario (Figura 4.31).

```
if (uidValue.equals("0")) {  
    confirmar.setText("Ingresar UID");  
    habilitarEdicion();  
  
} else {  
    confirmar.setText("Actualizar uid");  
    uidVal.setText(uidValue);  
  
    deshabilitarEdicion();  
    isEditMode = false;  
}
```

**Figura 4.31 Activa o desactiva controles con base en el estado del UID**

**Fuente: Elaboración Propia**

De igual manera, el botón "confirmar" será el encargado de que el usuario pueda ingresar un UID o actualizarlo si es que "isEditMode" es igual a "true", llamando al método "agregarUID", y si es presionado teniendo ya un UID se llamara

al método "habilitarEdicion", permitiendo al usuario editar su UID (Figura 4.32).

```
confirmar.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        if(isEditMode){  
            agregarUID();  
        }else{  
            habilitarEdicion();  
            confirmar.setText("Actualizar UID");  
        }  
    }  
});
```

**Figura 4.32** Código a ejecutarse cuando se da clic al botón "Confirmar"

**Fuente:** Elaboración Propia

El método "habilitarEdicion" tiene como propósito, como su nombre lo indica, habilitar al usuario las opciones y los campos para insertar o actualizar su UID (Figura 4.33).

```
private void habilitarEdicion(){  
    isEditMode = true;  
    uidd.setEnabled(true);  
    cUID.setEnabled(true);  
    uidd.setVisibility(View.VISIBLE);  
    cUID.setVisibility(View.VISIBLE);  
    uidVal.setVisibility(View.INVISIBLE);  
    eliminar.setVisibility(View.INVISIBLE);  
    eliminar.setEnabled(false);  
    hola.setText("Introduce los datos siguientes para vincular un UID:");  
}
```

**Figura 4.33** Código para habilitar controles de edición de UID

**Fuente:** Elaboración Propia

Para el metodo "desHabilitarEdicion" se utilizó una variante del método de la Figura 4.36, con la diferencia de que los campos se invierten, es decir, "uidd.setEnabled" y "cUID.setEnabled" se cambian a false, los campos visibles se vuelven invisibles y viceversa. También, se encuentra el botón "eliminar" el cual cumple la función de eliminar el UID que el usuario tenía dentro de su información en la base de datos. Finalmente, se utilizó una variante del método "agregarUID", pero esta vez, mediante el uso de la URL del PHP eliminar un UID.

## **2) PHP**

Para el código en PHP se utilizó una variante de los códigos anteriores a la que se le realizó una modificación para permitir cambiar el UID de un usuario registrado, de tal forma que a través de este código sea posible registrar, cambiar o eliminar al UID de los usuarios en cualquier momento que estos así lo deseen.

Cuando se realice la eliminación del UID de algún usuario se pondrá un "0" en el campo correspondiente al identificador, de tal forma que este pueda ser usado en la aplicación como medio de control para identificar a los usuarios sin UID (Figura 4.34).

```
$uid = $_POST['uid'];  
$id = $_POST['id'];  
  
$query = "UPDATE usuario SET uid = '$uid' WHERE id = '$id'";  
$resultado = mysqli_query($conexion, $query);
```

**Figura 4.34 Código en PHP para modificar el UID de los usuarios**

**Fuente: Elaboración Propia**

#### **G. Recarga de crédito**

Código clave para mostrar el cómo se les permite a los usuarios el realizar una recarga a su crédito. Este procedimiento actualmente solo será programado a modo de usarlo para probar la funcionalidad del sistema.

**Códigos en los distintos IDE para lograr esta funcionalidad:**

##### **1) Android Studio**

En este apartado, se presenta el código que fue desarrollado para dar funcionalidad a los botones a través de los cuales los usuarios podrán seleccionar la cantidad de crédito a recargar en sus cuentas. Cada botón permitirá realizar la recarga de una cantidad diferente de crédito, según sea requerido por los usuarios (Figura 4.35). Cada botón dentro de la interfaz gráfica servirá como medio para actualizar un cuadro de texto el cual ayudará al usuario el rectificar la cantidad de crédito que desea recargar a su cuenta, llamando cada vez al método "actualizarCantidadRecarga".



Una vez se dé clic a algún botón no se podrá dar clic en el mismo botón hasta que se cliquee otro, esto quiere decir que la cantidad de recarga no se sumará si el usuario presiona varias veces el mismo botón.

```
for(final Button boton : botones){
    boton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if(botonPrevioClicado != null){
                botonPrevioClicado.setEnabled(true);
            }

            boton.setEnabled(false);
            botonPrevioClicado = boton;

            int cantidad = 0;
            switch (boton.getId()){
                case R.id.btn_10:
                    cantidad = 10;
                    break;
                case R.id.btn_20:
                    cantidad = 20;
                    break;
                case R.id.btn_50:
                    cantidad = 50;
                    break;
                case R.id.btn_100:
                    cantidad = 100;
                    break;
                case R.id.btn_200:
                    cantidad = 200;
                    break;
                case R.id.btn_500:
                    cantidad = 500;
                    break;
            }
            actualizarCantidadRecarga(cantidad);
        }
    });
}
```

**Figura 4.35 Código para dar funcionalidad a los botones de recarga de crédito**

**Fuente: Elaboración Propia**

El método "actualizarCantidadRecarga" (Figura 4.36), tiene la tarea de realizar la actualización del crédito del usuario con base a la cantidad de crédito que desea recargar. Esta actualización consiste en la suma del crédito con la que actualmente cuenta el usuario con la cantidad de recarga deseada. Además, en todo momento muestra al usuario cual sería el nuevo crédito de su cuenta de aplicarse la recarga de crédito.

```
private void actualizarCantidadRecarga(int cantidad) {  
    cred.setText(String.valueOf(cantidad));  
    int intCred = Integer.parseInt(credVal.getText().toString());  
    intCred -= recargaActual;  
    int recarga = intCred + cantidad;  
    recargaActual = cantidad;  
    credVal.setText(String.valueOf(recarga));  
}
```

**Figura 4.36 Suma de crédito con la cantidad de recarga actual**  
**Fuente: Elaboración Propia**

Una vez decidida la cantidad de crédito a recargar, el botón "confirmar" funge como medio para actualizar el crédito de los usuarios en la base de datos. Este botón llama al método "recargaCredito", el cual es el responsable de solicitar la actualización de la información en la base de datos del sistema (Figura 4.37).

```

confirmar.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        recargaCredito();
    }
});
}

private void recargaCredito(){

    final String nTrajeta = tarjeta.getText().toString().trim();
    final String expireDate = expira.getText().toString().trim();
    final String nCVV = cvv.getText().toString().trim();
    final String id = idd.getText().toString().trim();
    final String credito = credVal.getText().toString().trim();

    if(nTrajeta.isEmpty()) {
        tarjeta.setError("Introduce tu numero de tarjeta");
        return;
    }else if(expireDate.isEmpty()) {
        expira.setError("Introduce la fecha de expiracion de la tarjeta");
        return;
    }else if(nCVV.isEmpty()){
        cvv.setError("Introduce el CVV de la tarjeta");
        return;
    }else{
        StringRequest stringRequest = new StringRequest(Request.Method.POST,
        "http://192.168.1.68:8080/tesis_/update.php", new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                if(response.equalsIgnoreCase("La recarga se ha realizado
correctamente")){
                    Toast.makeText(Recarga.this, response,
                    Toast.LENGTH_SHORT).show();
                    finish();
                }else{
                    Toast.makeText(Recarga.this, response,
                    Toast.LENGTH_SHORT).show();
                }
            }
        })
    }
}

```

**Figura 4.37. Funcionalidad del botón para confirmar la recarga de crédito**

**Fuente: Elaboración Propia**

## 2) PHP

En este caso el código en PHP únicamente se encuentra destinado a permitir la actualización del crédito de un usuario con base en su ID (Figura 4.38).

Es importante hacer mención que una vez que el crédito de los usuarios es actualizado este puede comenzar a ser usado de manera inmediata.

```
14. $id = $_POST['id'];
15. $credito = $_POST['credito'];
16.
17. $query = "UPDATE usuario SET credito = '$credito' WHERE id =
    '$id'";
18. $resutaldo = mysqli_query($conexion, $query);
```

**Figura 4.38. Código en PHP para la actualización del crédito de los usuarios**

**Fuente: Elaboración Propia**

#### ***4.7 Validación del sistema***

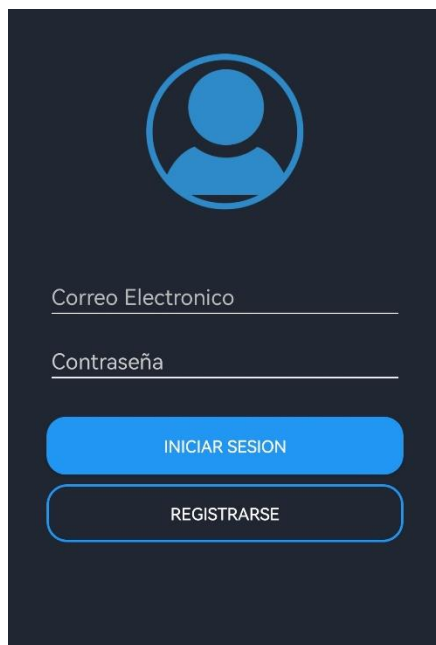
En este apartado se mostrará el proceso de validación que se llevó a cabo a la hora de realizar las pruebas requeridas para verificar el correcto funcionamiento del sistema presentado en este trabajo de investigación.

La validación consistirá en primera instancia en probar a cada una de las funcionalidades del sistema propuesto, para de esta manera asegurar que no existan errores de conexión, visualización o de programación, y en caso de encontrarse algún error permitir corregirle antes de liberar al prototipo del sistema.

Adicionalmente, como parte de la validación se pretende comprobar que se haya logrado satisfactoriamente el desarrollo de un sistema que sea amigable para los usuarios, de tal manera que su uso no represente alguna inconformidad debido a una compleja operación o menús complicados de entender. Del mismo modo, se comprobará que se mantenga en todo momento la integridad de los datos ingresados y generados por el sistema

#### **4.7.1 Registro de una cuenta**

Una vez abierta la aplicación móvil, se podrá observar como primera pantalla a la interfaz gráfica de inicio/registro de sesión, permitiendo el ingreso con una sesión si es que ya cuenta con una o el registrar una cuenta nueva (Figura 4.39).

La imagen muestra una interfaz de usuario para el inicio de sesión o registro. En la parte superior, hay un ícono de un usuario dentro de un círculo. Debajo de esto, hay dos campos de entrada de texto: 'Correo Electronico' y 'Contraseña'. En la parte inferior, hay dos botones: 'INICIAR SESION' (en azul) y 'REGISTRARSE' (en blanco con un borde azul).

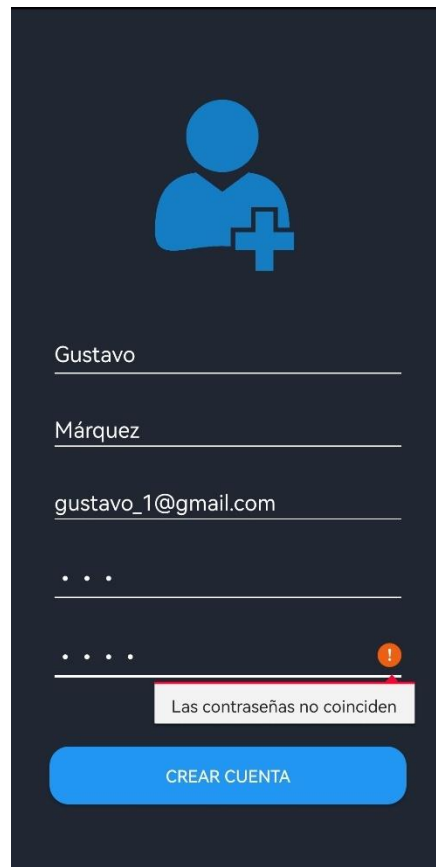
**Figura 4.39 Pantalla inicio/registro de sesión**  
**Fuente: Elaboración Propia**

Al dar clic al botón "registrarse" se pasará al usuario a la interfaz gráfica en la que se procederá a realizar el proceso de registro. Dentro de la interfaz de registro existirá un apartado el cual se encargará de que, si el usuario deja campos vacíos al momento de registrar una cuenta nueva, entonces se mostrará un mensaje de error dependiendo del campo o campos que el usuario haya dejado vacíos (Figura 4.40).

The image shows two identical registration forms side-by-side on a dark blue background. Each form features a blue user icon at the top. Below the icon are five input fields: 'Nombre', 'Apellidos', 'Correo electronico', 'Contraseña', and 'Confirmar Contraseña'. At the bottom of each form is a blue button labeled 'CREAR CUENTA'. The left form has error messages: 'Introduce tu nombre' next to the 'Nombre' field and 'Introduce tus apellidos' next to the 'Apellidos' field. The right form has error messages: 'Introduce tus apellidos' next to the 'Apellidos' field and 'Introduce tus apellidos' next to the 'Correo electronico' field. Each error message is accompanied by a small orange circle with an exclamation mark.

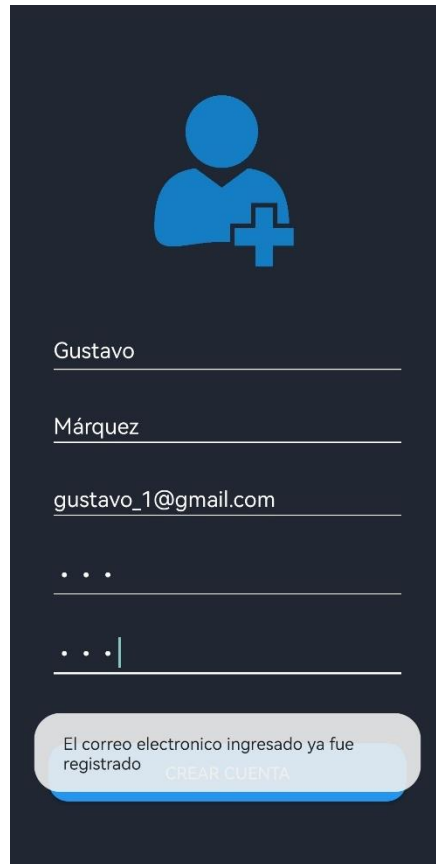
**Figura 4.40 Mensaje de error en registro de cuenta**  
**Fuente: Elaboración Propia**

Si al registrarse la cuenta nueva la contraseña y su confirmación no coinciden, se mostrará un mensaje en pantalla indicando del error (Figura 4.41).



**Figura 4.41 Las contraseñas no coinciden**  
**Fuente: Elaboración propia**

De igual manera, si el usuario intenta crear una cuenta nueva utilizando a un usuario asociado a un correo electrónico que ya fue previamente registrado en la aplicación, aparecerá un mensaje emergente indicándole que el correo electrónico ya ha sido previamente registrado, por lo que no será posible proceder con el registro de la cuenta (Figura 4.42).



**Figura 4.42 Correo electrónico ya registrado**  
**Fuente: Elaboración Propia**

Cuando los datos de registró hayan sido colocados correctamente y si no existe alguna cuenta con el mismo usuario, se procederá a efectuar la creación de la cuenta nueva, mostrándole al usuario un mensaje emergente indicándole que el registro se ha completado con éxito y finalmente, se le regresará a la pantalla principal de inicio de sesión para que pueda acceder a la aplicación con su cuenta nueva (Figura 4.43).



**Figura 4.43 Registro exitoso**  
**Fuente: Elaboración propia**

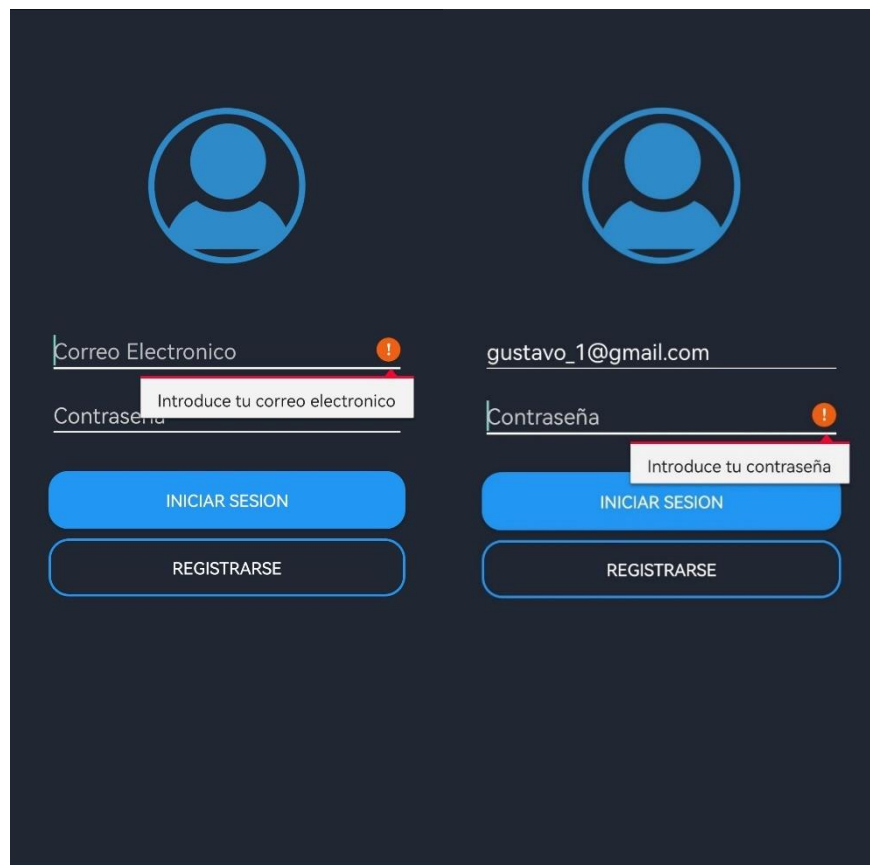
Como parte del proceso de registro de una cuenta nueva, en la base de datos se insertará un registro asociado a los datos de la cuenta (Figura 4.44).

| id | nombre  | apellido | correo              | contraseña | credito | uid |
|----|---------|----------|---------------------|------------|---------|-----|
| 1  | Gustavo | Márquez  | gustavo_1@gmail.com | 123        | 1000    | 0   |

**Figura 4.44 Base de datos actualizada**  
**Fuente: Elaboración Propia**

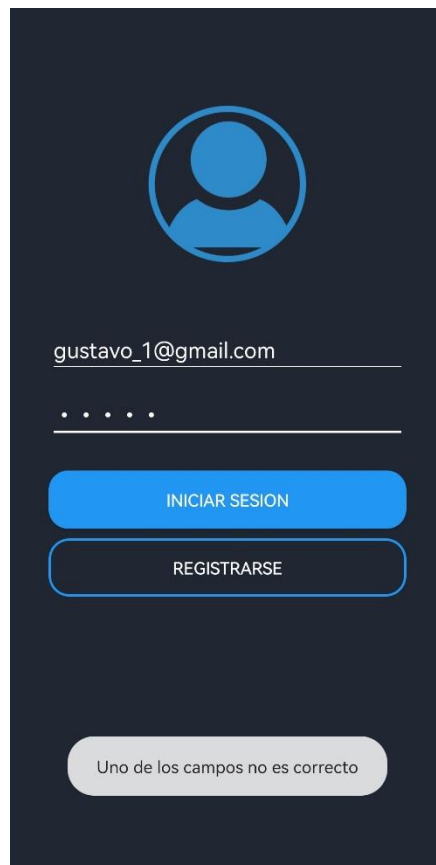
#### 4.7.2 Inicio de sesión en la aplicación

En la interfaz de inicio de sesión, al igual que en la de registro de cuentas, se realiza una validación para comprobar que no se hayan dejado campos vacíos, mostrando un mensaje de error en caso de encontrarse con algún campo en esta situación (Figura 4.45).



**Figura 4.45 Mensajes de error inicio de sesión**  
**Fuente: Elaboración propia**

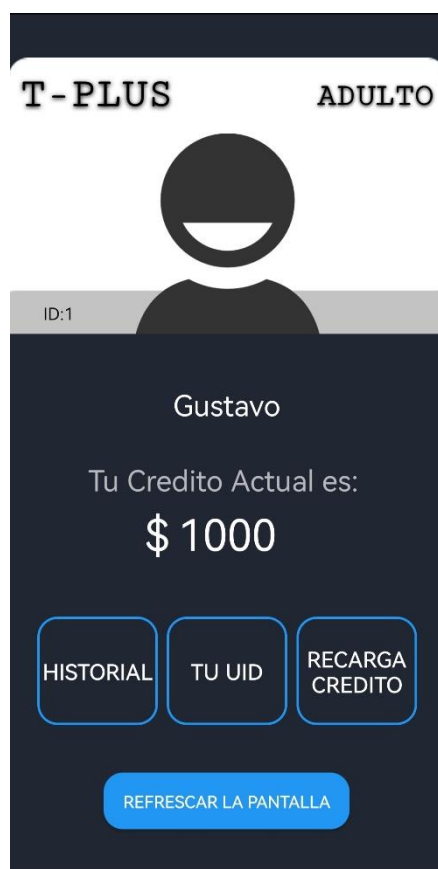
Por su parte, en la Figura 4.46 se presenta al mensaje de error que recibirá un usuario cuando los datos de usuario y contraseña que haya ingresado no correspondan a los de una cuenta previamente registrada o si alguno de los datos no es correcto.



**Figura 4.46 Campo erróneo**  
**Fuente: Elaboración propia**

Cuando sean ingresados datos correctos de inicio de sesión, se le mostrará al usuario la pantalla principal de la aplicación. En esta pantalla se podrá visualizar el nombre del usuario y su crédito actual.

De igual manera, se tendrá a disposición del usuario a los distintos botones que le permitirán acceder a las funcionalidades de la aplicación, siendo estas: consultar el historial de uso de la tarjeta del usuario, visualizar o cambiar su tarjeta, actualizar sus datos y recargar crédito en su cuenta (Figura 4.47).



**Figura 4.47 Pantalla principal de usuario**  
**Fuente: Elaboración Propia**

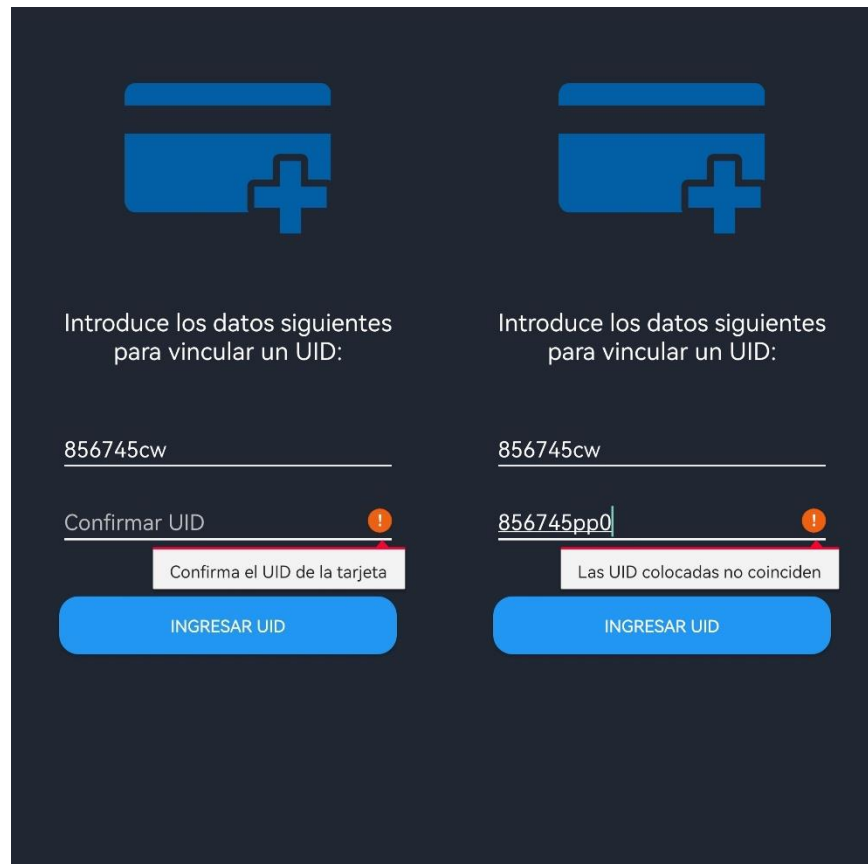
### **4.7.3 Administración de tarjeta mediante el UID**

La primera vez que un usuario acceda a esta pantalla deberá registrar el UID de su tarjeta para que de esta forma pueda comenzar a hacer uso del sistema de pago. Para esto es necesario ingresar su UID dos veces, para asegurarse de que no cometa errores durante el registro del UID (Figura 4.48).



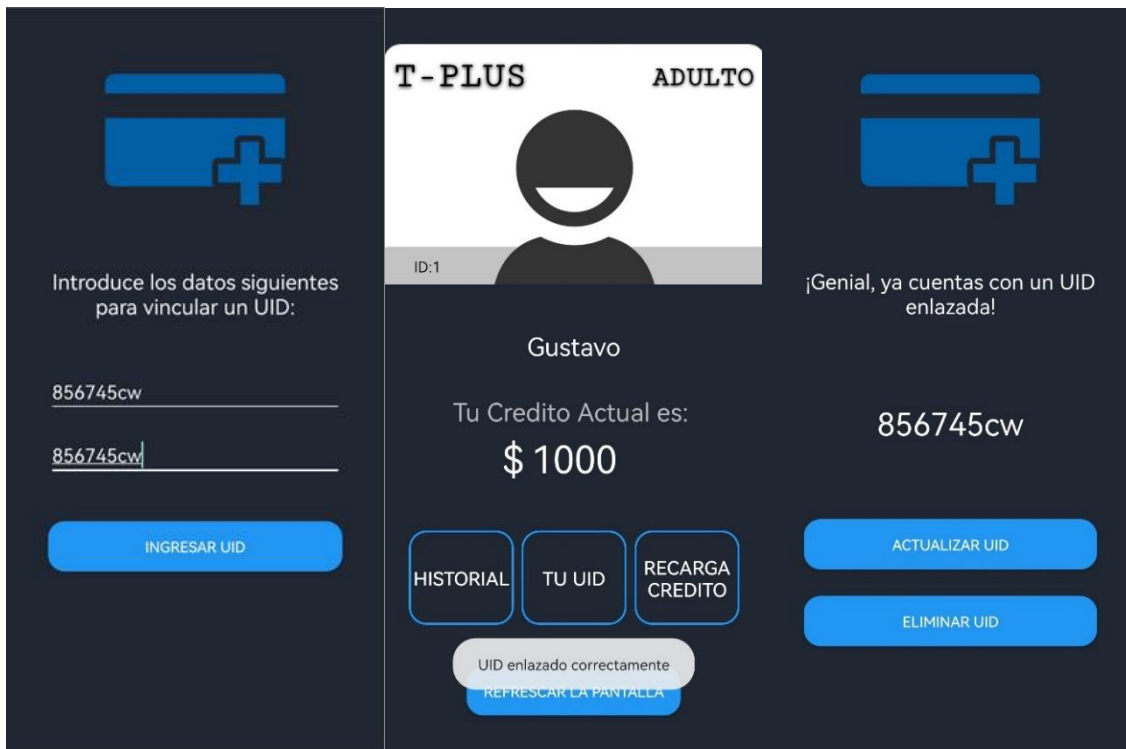
**Figura 4.48 Ingresar UID**  
**Fuente: Elaboración propia**

Esta interfaz al igual que las anteriores, mostrará mensajes de error en caso de que se hayan dejado campos vacíos o que las UID no coincidan, evitando así que se ingresen datos erróneos al sistema (Figura 4.49).



**Figura 4.49 Mensajes de error al enlazar UID**  
**Fuente: Elaboración Propia**

Cuando los datos del UID sean ingresados correctamente, se regresará al usuario a la pantalla principal y se le notificará mediante un mensaje emergente que el UID ha sido enlazado correctamente. De esta forma, una vez se ingrese nuevamente a "Tu UID" se mostrará el UID que el usuario haya registrado, y los campos de texto "UID" y "Confirmar UID" no serán visibles, debido a que ya se cuenta con un UID valido (Figura 4.50).



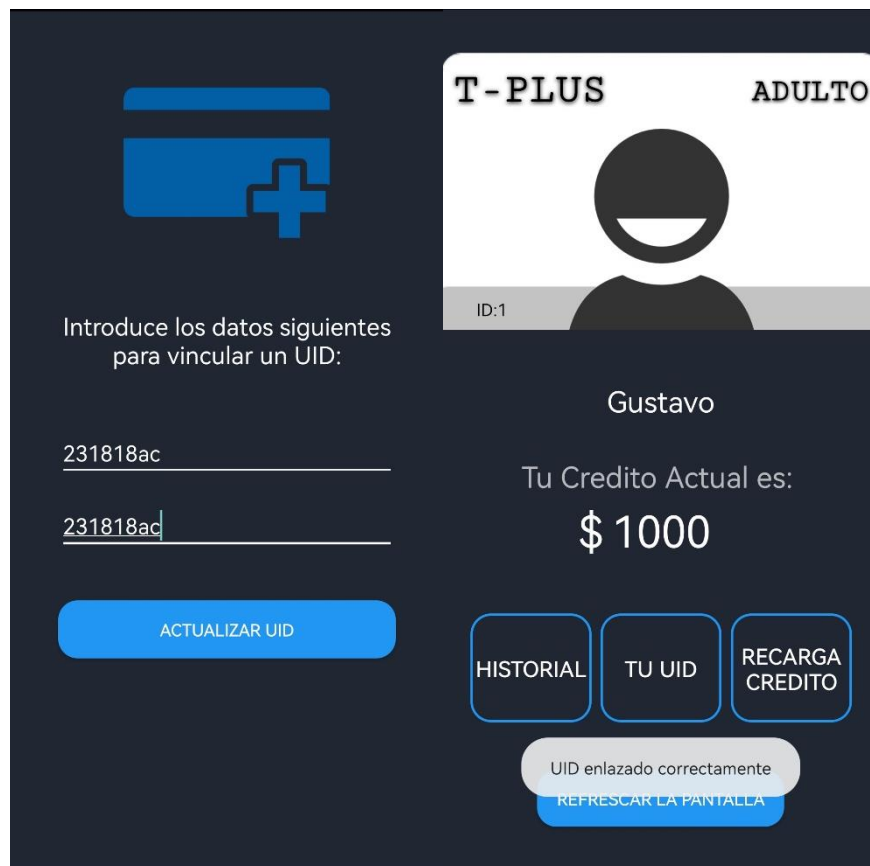
**Figura 4.50 Ingresar UID**  
**Fuente: Elaboración Propia**

La Figura 4.51 muestra a la base de datos actualizada con el UID registrado por el usuario.

| id | nombre  | apellido | correo              | contraseña | credito | uid      |
|----|---------|----------|---------------------|------------|---------|----------|
| 1  | Gustavo | Márquez  | gustavo_1@gmail.com | 123        | 1000    | 856745cw |

**Figura 4.51 Base de datos con UID asignada**  
**Fuente: Elaboración Propia**

En el caso de que un usuario desee cambiar de tarjeta, será necesario que de clic en el botón "actualizar UID", para que de esta forma registre al UID de la nueva tarjeta y pueda continuar usando al sistema de pago (Figura 4.52).



**Figura 4.52 Actualizar UID**  
**Fuente: Elaboración Propia**

En la Figura 4.53, se muestra el nuevo UID del usuario actualizado en la base de datos, mismo que será reconocido por el sistema de pago de manera inmediata, dejando al anterior sin validez para el usuario.

| id | nombre  | apellido | correo              | contraseña | credito | uid      |
|----|---------|----------|---------------------|------------|---------|----------|
| 1  | Gustavo | Márquez  | gustavo_1@gmail.com | 123        | 1000    | 231818ac |

**Figura 4.53 Base de datos con UID actualizado**  
**Fuente: Elaboración Propia**



En cualquier momento el usuario será capaz de eliminar al UID asociado a su cuenta. Para esto únicamente será necesario que de clic al botón "Eliminar UID". Después de eliminar el UID del usuario se le regresará a este a la pantalla principal y se le notificará de la eliminación a través de un mensaje (Figura 4.54). Esta opción es de especial interés en caso de que al usuario le hayan robado su tarjeta o que la haya extraviado. Es importante hacer mención que el crédito con el que contaba no sufrirá ningún cambio, pues este se encuentra asociado a su cuenta y no a la tarjeta.



**Figura 4.54 UID eliminado**  
**Fuente: Elaboración Propia**

#### 4.7.4 Recarga de crédito

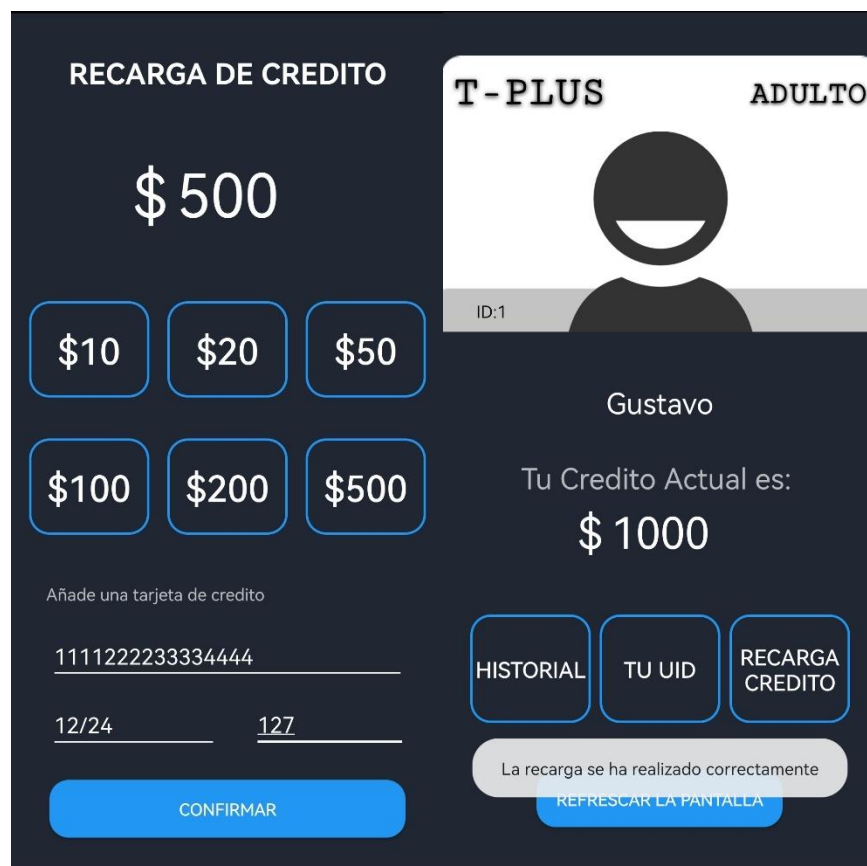
Al igual que en las interfaces anteriores, los campos de texto no podrán estar vacíos. Si se desea realizar una recarga de crédito, se deberá rellenar primero los campos de texto para proceder con la recarga de crédito (Figura 4.55).

The figure displays three sequential screenshots of a credit top-up interface, each titled "RECARGA DE CREDITO". Each screen shows a selection of top-up amounts (\$10, \$20, \$50, \$100, \$200, \$500) and a form to add a credit card. The first screenshot shows the "Numero de la Tarjeta" field with a red error message "Introduce tu numero de tarjeta". The second screenshot shows the "MM/AAAA" and "CVV" fields with red error messages "Introduce la fecha de expiracion de la tarjeta" and "Introduce el CVV de la tarjeta" respectively. The third screenshot shows the "CONFIRMAR" button.

**Figura 4.55 Campos vacíos en recarga de crédito**  
**Fuente: Elaboración Propia**

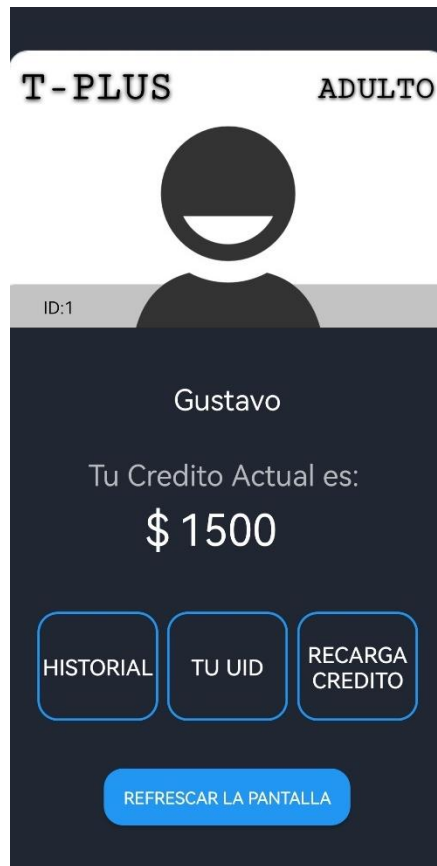
Una vez llenados todos los campos de texto, seleccionada una cantidad a recargar y presionado el botón "Confirmar" se procederá a realizar la recarga directamente a la base de datos, sumando el crédito seleccionado con el crédito que ya

se tenía disponible dentro de la cuenta y, además, se le regresará al usuario a la pantalla principal en la que se mostrará un mensaje emergente indicándole que la recarga se ha realizado correctamente (Figura 4.56).



**Figura 4.56 Recarga de crédito exitoso**  
**Fuente: Elaboración Propia**

Para visualizar al nuevo crédito será necesario pulsar el botón "Refrescar la pantalla", el cual se encarga de obtener el crédito del usuario desde la base de datos y colocarlo en la pantalla principal para su visualización (Figura 4.57).



**Figura 4.57 Refrescar la pantalla**  
**Fuente: Elaboración propia**

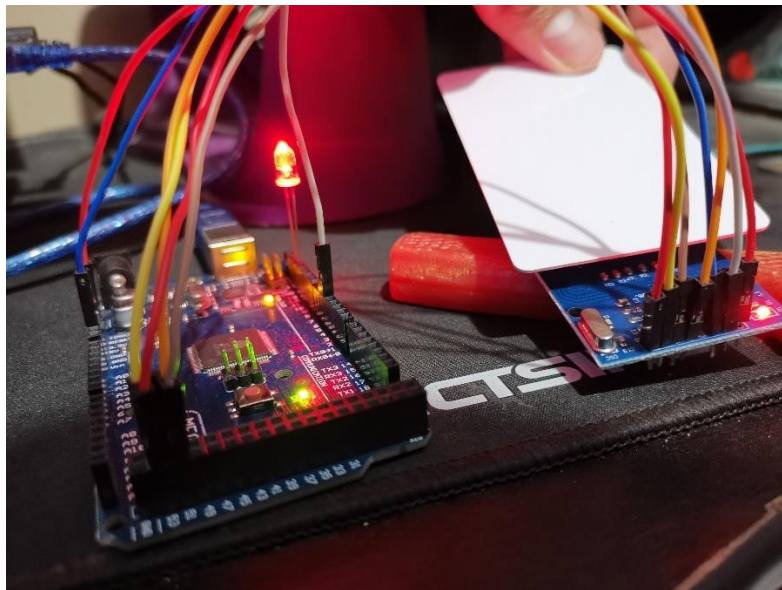
La Figura 4.58 muestra a la base de datos con el crédito del usuario actualizado después de haber realizado el proceso de recarga de crédito correctamente.

| id | nombre  | apellido | correo              | contraseña | credito | uid      |
|----|---------|----------|---------------------|------------|---------|----------|
| 1  | Gustavo | Márquez  | gustavo_1@gmail.com | 123        | 1500    | 231818ac |

**Figura 4.58 Base de datos con crédito actualizado**  
**Fuente: Elaboración Propia**

#### **4.7.5 Cobro de crédito**

En la Figura 4.59 se observa como el sistema electrónico es capaz de detectar correctamente a una tarjeta para la obtención de su UID. Esto es visualmente comprobable por los usuarios a través del encendido de un LED de estado, el cual únicamente enciende cuando el sistema electrónico ha detectado una tarjeta y se apaga pasados algunos segundos de haberla detectado.



**Figura 4.59 Identificación de etiqueta**  
**Fuente: Elaboración propia**

Posteriormente a la detección del UID de la tarjeta, este es transferido a la computadora a la que se encuentra conectado el sistema electrónico y en donde través del programa desarrollado en Python será validado y en su caso procesado para realizar el cobro del transporte al usuario (Figura 4.60).

```
Se realizo el cobro al usuario Gustavo con ID 1, credito actual: 1491, fecha: 2023-07-28 16:28:06
Se realizo el cobro al usuario Gustavo con ID 1, credito actual: 1482, fecha: 2023-07-28 16:28:23
Se realizo el cobro al usuario Gustavo con ID 1, credito actual: 1473, fecha: 2023-07-28 16:28:26
Se realizo el cobro al usuario Gustavo con ID 1, credito actual: 1464, fecha: 2023-07-28 16:28:28
```

**Figura 4.60 Impresión de cobro en consola**  
**Fuente: Elaboración propia**

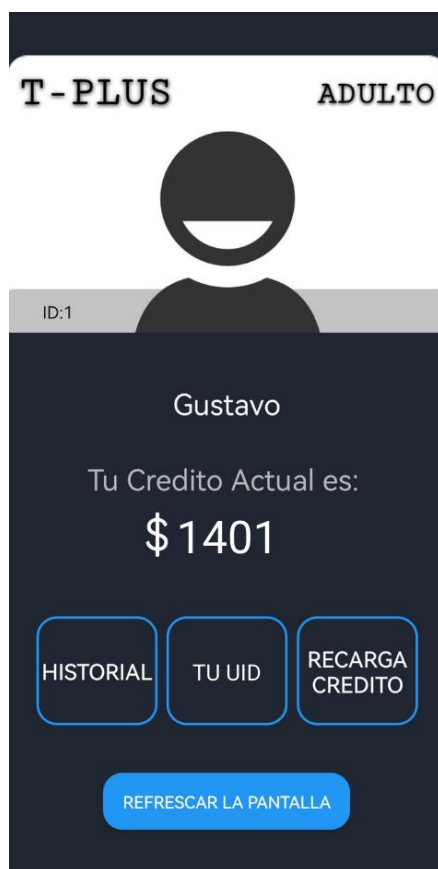
En este mismo proceso, cada que se efectúe el cobro de crédito a un usuario, los datos del movimiento se almacenarán en la tabla "Historial". Siendo los datos que se almacenarán: el ID del usuario, el UID de la tarjeta y la fecha en la que se registro la entrada de la tarjeta (Figura 4.61).

| id | id_usuario | uid      | fecha               |
|----|------------|----------|---------------------|
| 1  | 1          | 231818ac | 2023-07-28 16:28:06 |
| 2  | 1          | 231818ac | 2023-07-28 16:28:23 |
| 3  | 1          | 231818ac | 2023-07-28 16:28:26 |
| 4  | 1          | 231818ac | 2023-07-28 16:28:28 |
| 5  | 1          | 231818ac | 2023-07-28 16:28:29 |
| 6  | 1          | 231818ac | 2023-07-28 16:28:31 |
| 7  | 1          | 231818ac | 2023-07-28 16:28:33 |
| 8  | 1          | 231818ac | 2023-07-28 16:28:34 |
| 9  | 1          | 231818ac | 2023-07-28 16:28:38 |
| 10 | 1          | 231818ac | 2023-07-28 16:28:39 |
| 11 | 1          | 231818ac | 2023-07-28 16:28:41 |

**Figura 4.61 Almacenamiento de registros en la tabla "historial"**

**Fuente: Elaboración propia**

En la Figura 4.62 se muestra la visualización que el usuario tiene a la hora de ingresar a la aplicación después de proceder con los pagos correspondientes al transporte público, reflejando el crédito que le quedo disponible.



**Figura 4.62 Cobro de crédito reflejado en la aplicación**  
**Fuente: Elaboración propia**

#### ***4.7.6 Pantalla de historial***

El historial tiene como objetivo principal el mostrar al usuario los movimientos de pago que ha realizado con sus tarjetas, en los que además se mostrará el año, mes, día y hora en que se realizaron los distintos movimientos en su cuenta.

Esto para que los usuarios puedan tener un mejor control del uso que dan a su cuenta y que puedan fácilmente identificar cuando y cuantas veces han realizado algún movimiento de pago del transporte público (Figura 4.63).



**Figura 4.63 Pantalla de historial**  
**Fuente: Elaboración Propia**

En esta interfaz los usuarios únicamente podrán visualizar su historial, de tal manera que no podrán modificar ni eliminar ningún dato dentro de la interfaz.



## **Capitulo 5. Conclusiones y recomendaciones**

### ***5.1 Respuesta al problema de investigación***

Como resultado del trabajo de investigación, se contó con un sistema orientado al público en general para realizar el pago del transporte público a través de tecnología NFC. El sistema que fue desarrollado se encontró conformado por dos grandes componentes destinados a trabajar en conjunto para hacer posible el logro del objetivo general establecido en este trabajo de investigación, siendo estos componentes: una aplicación móvil y un sistema electrónico.

En cuanto a la aplicación móvil, esta fue destinada a la creación, administración y gestión de las cuentas de los usuarios. Dicha aplicación fue desarrollada en Android Studio, permitiendo así ser ejecutada en cualquier dispositivo celular que cuente con Android como sistema operativo.

Por su parte, el sistema electrónico fue desarrollado basado en tecnología Arduino y este se encontró integrado principalmente por un Arduino mega, un led y un módulo MFRC522 destinado a la lectura de tarjetas RFID. Con el sistema electrónico propuesto fue posible el hacer que los usuarios realicen el pago del transporte público mediante el uso de tarjetas RFID.

## **5.1 Cumplimiento de los objetivos específicos**

### **5.1.1 Analizar sistemas de pago de transporte publico existentes**

Con la finalidad de realizar un adecuado diseño y desarrollo de un sistema de pago del transporte público, se realizó un análisis de algunos de los principales sistemas de pago que actualmente se encuentran implementados en distintas zonas de la república mexicana.

Derivado del análisis de los sistemas de pago considerados, así como, de la información obtenida de otras fuentes de información como libros, revistas, noticias, entre otras, se procedió a definir a las funcionalidades con las que el sistema propuesto en este trabajo de investigación contaría, siendo estas listadas a continuación:

- Registro de un nuevo usuario
- Registro, actualización y eliminación de tarjetas para realizar el pago del transporte
- Pago del transporte público a través de tecnología NFC
- Inicio de sesión para el ingreso a la interfaz de usuario principal de la aplicación

- Visualización del nombre de usuario, la ID y el crédito disponible
- Actualización de la interfaz de usuario principal para la visualización a tiempo real del cobro de crédito
- Consulta del historial de movimientos de la cuenta

#### ***5.1.2 Diseñar el sistema electrónico y el software de control del sistema de pago***

Una vez realizada la definición de las funcionalidades con las que contaría el sistema de pago se procedió a realizar el diseño del sistema electrónico, de tal manera que a través de este fuera posible la detección de las tarjetas de los usuarios a través del módulo MFRC522, para posteriormente obtener el UID asociado a las tarjetas que fue sean leídas, y finalmente, proceder con el cobro del pasaje del transporte público al usuario al que pertenece cada tarjeta, notificando a este último de que el proceso se ejecutó correctamente a través del encendido de un led de estado.

Del mismo modo, en esta etapa se realizó el diseño de las distintas pantallas que compondrían a la aplicación móvil asociada al sistema de pago. Durante el diseño de las interfaces gráficas se tuvo en cuenta en todo momento el contar con un diseño sencillo e intuitivo que permitirá a los usuarios

acceder a las distintas funcionalidades de la aplicación de una manera sencilla y sin tener que acceder a complicados menús o realizar largos procedimientos. Así mismo, se tomó en cuenta el uso de una paleta de colores y componentes visuales que no resultarán desagradables para los usuarios, y que les permitiera hacer uso de la aplicación sin ninguna clase de complicación.

### ***5.1.3 Realizar la programación de lógica del sistema de pago***

Durante esta etapa se realizó la definición de la lógica que sería necesario programar para lograr la adecuada ejecución de cada una de las funcionalidades del sistema de pago del transporte público.

Como parte de este objetivo se tuvo que definir la manera correcta de lograr la integración de los distintos componentes de Software que constituyen al sistema en su totalidad, y para los que se utilizaron distintos lenguajes y entornos de programación.

Cada una de las funciones y métodos que fueron programados se validaron de manera independiente para garantizar su integración de forma transparente para el usuario, de tal

manera que no resultarán complicados de comprender para los usuarios al momento de requerir hacer uso de estos.

Además de las funcionalidades propias del sistema, en esta etapa se procedió a añadir mecanismos de validación para asegurarse de mantener en todo momento la integridad de los datos y del propio sistema.

#### ***5.1.4 Validar la funcionalidad del sistema desarrollado***

Después de haber realizado el desarrollo del sistema de pago propuesto en este trabajo de investigación, se procedió a realizar la validación integral de cada una de las funciones del sistema. Para esto se procedió a analizar la totalidad del sistema, como si de ejecuciones cotidianas se tratase.

Los resultados del proceso de validación integral demostraron que el sistema propuesto es capaz de cumplir satisfactoriamente con los objetivos trazados en este trabajo de investigación, por lo que se concluye que es apto para una futura implementación. Del mismo modo, a través de la validación se identificaron algunas recomendaciones para hacer un mejor uso del sistema, así como, algunas líneas de trabajo futuro que permitirán mejorar su funcionalidad.

## **5.2 Aceptación del supuesto**

Se argumenta que el trabajo de investigación cumple con el supuesto planteado, debido a que al término del trabajo fue posible contar con un prototipo a nivel de laboratorio capaz de cumplir satisfactoriamente con los objetivos trazados en este trabajo de investigación, es decir, permitir el pago del transporte público por medio del uso de tecnología NFC.

## **5.3 Recomendaciones para el trabajo**

Con la finalidad de obtener el mayor beneficio con el sistema propuesto en este trabajo de investigación, se recomienda seguir a las recomendaciones listadas a continuación, para que de esta manera se disminuya la probabilidad de que se presente alguna falla o que el sistema no funcione de la manera esperada:

- Hacer uso de la aplicación móvil en teléfonos inteligentes que cuenten con una versión actualizada del sistema operativo Android
- Programar respaldos y depuraciones frecuentes de la base de datos, para evitar perder información y asegurar la integridad de los datos.

- Mantener en buen estado a las tarjetas de los usuarios, para evitar que éstas se vean dañadas o que se interfiera en su funcionalidad
- Dar mantenimiento al sistema electrónico, para que de esta manera sus componentes no sufran averías y se encuentren siempre en óptimas condiciones.

#### ***5.4 Recomendaciones para investigaciones futuras***

Se recomienda a los investigadores a continuar proponiendo soluciones que coadyuven a disminuir la escasez de sistemas que actualmente existe con relación al transporte público. Derivado de lo anterior, se invita a extender a este trabajo de investigación enfocándose en alguno de los siguientes aspectos:

- Llevar al sistema propuesto a su implementación, incorporando al sistema electrónico en las unidades de transporte de toda la zona y entregando a la aplicación a los usuarios.
- Extender al sistema y su funcionalidad fuera de los dispositivos móviles, permitiendo su utilización y administración en sitios Web

- Enlazar la tecnología NFC del dispositivo móvil con la base de datos para que la UID generada sirva como señal temporal y completar el pago.
- Incorporar un método de recarga de crédito mediante transacciones
- Integrar nuevas funcionalidades al sistema que permitan que sean más llamativo para los usuarios
- Concientizar a los usuarios y operarios de la importancia de integrar sistemas tecnológicos al transporte público para extender y mejorar sus tareas y funcionalidades



## BIBLIOGRAFIA

Acevedo Mosquera, M. E., Acevedo Mosquera, M. A., Neri Espinoza, K. A., Chávez Álvarez, A. (2015), recuperado de:

<http://pistaseducativas.celaya.tecnm.mx/index.php/pistas/article/view/453/440#>

Aguilar, L. J., Azuela, M. F., & Baena, L. R. (1988). Fundamentos de programación. McGraw Hill.

Albornoz, M. C., Berón, M., & Montejano, G. A. (2017). Interfaz gráfica de usuario: el usuario como protagonista del diseño. Recuperado de [http://sedici.unlp.edu.ar/bitstream/handle/10915/62078/Documento\\_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/62078/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y)

Anaya-Cantellán, A., & López-Martínez, I. (2014). La tecnología NFC en teléfonos celulares, sus retos y aplicaciones. *Research in Computing Science*, 77, 97-107.

Arias, M. A. (2013). Introducción a PHP. IT Campus Academy.

Báez, M., Borrego, Á., Cordero, J., Cruz, L., González, M., Hernández, F., ... & Zapata, Á. (2019). Introducción a android.

Bou, R. C. (2019). Usando xampp con bootstrap y wordpress. Mercedes Gómez Alcalá.

Bueno Delgado, M. V., Pavón Mariño, P., & Gea García, A. D. D. (2011). La tecnología NFC y sus aplicaciones en un entorno universitario.

Castillo, J. D. L. (2019). Desarrollo de aplicaciones Android con Android Studio: Conoce android studio. José Dimas Luján Castillo.

Combaudon, S. (2018). MySQL 5.7: administración y optimización. Ediciones Eni.

Chavarría, D. (2011). Tecnología de comunicación de campo cercano (NFC) y sus aplicaciones. Trabajo de Grado. Universidad de Costa Rica.

DataScientest (2023). IDE : ¿Qué es un Entorno de Desarrollo Integrado?. Recuperado de: <https://datascientest.com/es/ide-que-es>

DataScientest. (2022). PyCharm: Todo sobre el IDE de Python más popular. Formation Data Science Recuperado de: <https://datascientest.com/es/pycharm>

Desai, P. (2018). Arduino Programming with Python. Packt Publishing.

Dorf, R., & Svoboda, J. (2015). Circuitos eléctricos. Alpha Editorial.

Elmasri, R., Navathe, S. B., Castillo, V. C., Pérez, G. Z., & Espiga, B. G. (2007). Fundamentos de sistemas de bases de datos (No. QA76. 9D3 E553 2007.). Pearson educación.

Esteban Nieto, N. (2018). Tipos de Investigación.  
<http://190.117.99.173/bitstream/USDG/34/1/Tipos-de-Investigacion.pdf>

FACUA (2007). El transporte público. Recuperado de  
<https://www.facua.org/es/guia.php?Id=77>

FLEXBOT (2019). Puerto de serie en Arduino. Recuperado de:  
[https://www.flexbot.es/puerto-de-serie-en-arduino/#:~:text=El%20puerto%20de%20serie%20o,tambi%C3%A9n%20como%20UART%20o%20USART\).](https://www.flexbot.es/puerto-de-serie-en-arduino/#:~:text=El%20puerto%20de%20serie%20o,tambi%C3%A9n%20como%20UART%20o%20USART).)

Flores, F. (2023). Qué es Visual Studio Code y qué ventajas ofrece. OpenWebinars.net. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>

García-Peñalvo, F. J. (2016). Dispositivos móviles en educación. Ediciones Universidad de Salamanca.

Gironés, J. T. (2019). El gran libro de Android. Alpha Editorial.

Goggin, G. (2017). Mobile Technology and Place. Routledge.

Gomez, M. D. (2013). Bases de datos. MB Manuela Mejia, Entrevistador.

Guo, Y., Fang, X., & Liu, J. (2020). Design and Implementation of an RFID-based Library Management System. Journal of Physics: Conference Series, 1603(3), 032053.  
<https://doi.org/10.1088/1742-6596/1603/3/032053>

Herazo, L. (2019). ¿QUÉ ES UNA APLICACIÓN MÓVIL?. Anincubator.  
 Recuperado de: <https://anincubator.com/que-es-una-aplicacion-movil/>

Kroenke, D. M., & Auer, D. J. (2016). Database processing: Fundamentals, design, and implementation. Pearson.

La importancia del transporte público en las ciudades - Imbric. (n.d.). Recuperado de <https://www.imbric.com/la-importancia-del-transporte-publico-en-las-ciudades/>

López Bermeo, S. P. (2019). Máquina dispensadora de snacks con pago por NFC (bachelor's thesis, quito).

Llama, L. (2016). Lecturas de tarjetas RFID con Arduino y lector MIFARE RC522. <https://www.luisllamas.es/arduino-rfid-mifare-rc522/#:~:text=%C2%BFQu%C3%A9%20es%20el%20RFID%3F,aut%C3%B3mata%20o%20procesador%20como%20Arduino.>

Marqués, M. (2011). Bases de datos. Castelló de la Plana: Publicacions de la Universitat Jaume I. Servei de Comunicació i Publicacions, 2011

Mentor, A., & Programaci, S. (2014). Desarrollo de aplicaciones para Android I. D. Robledo, Desarrollo de aplicaciones para android I.

Montoro, A. F. (2013). Python 3 al descubierto. RC libros.

Mora Cerezo, R. D., y Santos Bajaña, C. A. (2019), recuperado de:

<http://repositorio.ug.edu.ec/bitstream/redug/45009/1/B-CINT-PTG-N.464%20Mora%20Cerezo%20Richard%20Daniel%20.%20Santos%20Baja%C3%B1a%20Christian%20Andr%C3%A9s.pdf>

Moreno, J. S. (2013). Aplicaciones prácticas de NFC. 3 c TIC: cuadernos de desarrollo aplicados a las TIC, 2(1), 3.

Mottershead, G. (2015). Electronics: A First Course. CRC Press.

Pedrera, A. C. (2017). Arduino para Principiantes: 2ª Edición. IT Campus Academy.

Peña, C. (2020). Arduino IDE: Domina la programación y controla la placa. RedUsers.

Portilla Peñafiel, J. J. (2016). Análisis de comunicaciones y seguridades en la implementación del cobro de servicios de transporte público mediante tecnología NFC basada en la plataforma de dinero electrónico del Ecuador

Rodríguez Ruiz, A. I., y Parra Alvira, H. A. (2018). Diseño de una aplicación prototipo para dispositivos ´ móviles Android que permita recargas y acceso al sistema de transporte Transmilenio mediante tecnología NFC

Sacristán, C. R., & Fernández, D. R. (2012). Programación en Android. Ministerio de Educación.

Sanchez, D. (2015). Desarrollo de aplicaciones para dispositivos móviles. Universidad Nacional de Colombia.

- Silberschatz, A., Galvin, P. B., & Gagne, G. (2014).  
Fundamentos de sistemas operativos. John Wiley & Sons.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002).  
Fundamentos de bases de datos (6a ed.). McGraw-Hill.
- Stopka, U., Schäfer, G., Kreisel, A. (2017). Recuperado de  
[https://link.springer.com/chapter/10.1007/978-3-319-58077-7\\_46](https://link.springer.com/chapter/10.1007/978-3-319-58077-7_46)
- Tanenbaum, A. S., & Bos, H. (2015). Sistemas operativos  
modernos. Pearson Educación.
- Tanzi, T., & Valero, M. (2012). Arquitectura de computadores:  
un enfoque cuantitativo. Editorial Reverté.