

SISTEMA DE RESERVAS DE ESPACIOS GENÉRICO Y PARAMETRIZABLE

EDWIN EDUARDO CAMARGO MEDINA

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIA FISICOMECANICA
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA
BUCARAMANGA
2018**

SISTEMA DE RESERVAS DE ESPACIOS GENÉRICO Y PARAMETRIZABLE

EDWIN EDUARDO CAMARGO MEDINA

**TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE INGENIERO DE
SISTEMAS**

DIRECTOR

**FERNANDO ANTONIO ROJAS MORALES
M.Sc. INGENIERO DE SISTEMA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIA FISICOMECHANICAS
ESCUELA DE INGENIERIA DE SISTEMAS
BUCARAMANGA**

2018

DEDICATORIA

Esta meta alcanzada la dedico especialmente a Dios por llenarme de su luz, sabiduría, paciencia, amor y fortaleza para superar cada dificultad que se me presento a través de la carrera.

A mis padres Gerardo Camargo Pulido y Carmen Rosa Medina Herrera que con amor, esfuerzo y mucho sacrificio, hicieron que pudiera alcanzar este gran logro en mi vida.

A mis hermanas Edilma Camargo Medina y Angela Patricia Camargo Medina que con su ayuda, ánimo y consejos hicieron de mí una mejor persona.

Edwin Eduardo Camargo Medina

AGRADECIMIENTOS

Quiero agradecer especialmente a mi familia, que me brindo toda su ayuda, de todas las maneras posibles para poder estudiar y llegar a ser un profesional más en la familia.

A mi director de proyecto el profesor Fernando Antonio Rojas Morales por brindarme la oportunidad de trabajar con él, por estar siempre ahí para brindarme su orientación y ayuda en cada una de mis dudas y preguntas que tuve.

A la universidad Industrial de Santander por darme el privilegio de poder estudiar en esta gran alma mater.

A la escuela de Ingeniería de sistemas e informática y profesores que hicieron parte de mi carrera y me brindaron todo su conocimiento para lograr culminar con éxito esta etapa en mi vida.

También agradecer a mis compañeros que me ayudaron con su conocimiento para darme orientación en la realización de este proyecto.

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	15
1. DESCRIPCION DEL PROBLEMA	17
1.1 JUSTIFICACION Y PLANTEAMIENTO DEL PROBLEMA.....	17
2. OBJETIVOS	18
2.1. OBJETIVO GENERAL	18
2.2. OBJETIVOS ESPECÍFICOS.....	18
3. MARCO REFERENCIAL	19
3.1 CONTEXTUALIZACION DEL PROYECTO	19
3.2 ESTADO DEL ARTE.....	19
3.3 MARCO TEORICO	20
3.3.1. Sistemas de reserva.	20
3.3.2. BlueJ.....	20
3.3.3. NetBeans IDE.	21
3.3.4. PhpMyAdmin.....	21
3.3.5. Máquina virtual de Java.	22
3.3.6. Modelo-vista-controlador (MVC).	23
3.3.7. MySQL.....	23
4. METODOLOGIA	24
4.1. DESARROLLO ITERATIVO Y CRECIENTE.....	25
4.1.1 Inicialización.....	25
4.1.2. Iteración.	26

4.1.3 Lista de control del proyecto	26
5. ANALISIS DE CARACTERISTICAS DEL SISTEMA	27
5.1. DESCRIPCION DE LA APLICACION	27
5.2. ANALISIS DEL TEXTO	28
5.3. TABLA DE SUSTANTIVOS Y VERBOS	30
5.4. ESCENARIOS	31
5.5. TARJETAS CRC.....	32
5.6. MODELO DE OBJETOS DEL DOMINIO	34
5.7. LISTA DE CARACTERISTICAS DEL SISTEMA.....	35
5.8. PERSPECTIVA DEL SISTEMA	35
5.9. FUNCIONES DEL SISTEMA	35
5.9.1. Crear contenedor.	35
5.9.2. Modificar contenedor.	36
5.9.3. Eliminar contenedor.	36
5.9.4. Reservar plazas.	36
5.9.5. Buscar plazas.	36
5.9.6. Consultar reservas.....	36
5.9.7. Cancelar reserva.....	36
5.9.8. Eliminar reserva.	36
5.10. CARACTERISTICAS DEL USUARIO	37
5.11. RESTRICCIONES DEL SISTEMA.....	37
5.12. ACTORES DEL SISTEMA	38
5.13. DESCRIPCION DE ACTORES.....	38
5.14. DIAGRAMA DE CASOS DE USO.....	39
5.15. DESCRIPCION DE DIAGRAMAS DE CASOS DE USO.....	40
6. DISEÑO Y ESTRUCTURA DE DATOS	47
6.1. DISEÑO DE PROTOTIPOS.....	47
6.2. DISEÑO Y ESQUEMA DE LA BASE DE DATOS.....	50

6.3. DIAGRAMA ENTIDAD RELACION DE LA BASE DE DATOS (EER)	53
6.4. ORDEN DE LOS PAQUETES JAVA EN NETBEANS IDE	54
7. DESARROLLO DEL SISTEMA	59
7.1. RESULTADO DE PRUEBAS DEL SISTEMA CON USUARIOS FINALES....	67
8. CONCLUSIONES	69
9. RECOMENDACIONES.....	70
BIBLIOGRAFÍA	71

LISTA DE TABLAS

Pág.

Tabla 1. Sustantivos y verbos de la descripción.....	30
Tabla 2. Tarjeta CRC de la clase Contenedor	32
Tabla 3. Tarjeta CRC de la clase Reserva	32
Tabla 4. Tarjeta CRC de la clase Lugar.....	33
Tabla 5. Tarjeta CRC de la clase Plaza.....	33
Tabla 6. Tarjeta CRC de la clase Usuario	34
Tabla 7. Tarjeta CRC de la clase Cliente.....	34
Tabla 8. Descripción de Actores – Usuario.....	38
Tabla 9. Descripción de Actores – Cliente.....	39
Tabla 10. Descripción del caso de uso Crear Contenedor	40
Tabla 11. Descripción del caso de uso Actualizar Contenedor.....	41
Tabla 12. Descripción del caso de uso Eliminar Contenedor	42
Tabla 13. Descripción del caso de uso Realizar Reserva.....	43
Tabla 14. Descripción del caso de uso Consultar Reserva	44
Tabla 15. Descripción del caso de uso Cancelar Reserva	45
Tabla 16. Descripción del caso de uso Eliminar Reserva.....	46
Tabla 17. Descripción del caso de uso Utilizar Reserva.....	46
Tabla 18. Resultados de las pruebas con los usuarios finales	68

LISTA DE FIGURAS

	Pág.
Figura 1. Modelo de objetos	34
Figura 2. Actores del sistema	38
Figura 3. Diagrama de casos de uso	39
Figura 4. Pantalla principal del sistema.	47
Figura 5. Pantalla de creación del contenedor.	48
Figura 6. Pantalla de reservar plazas	49
Figura 7. Esquema de la base de datos - Tabla contenedor	50
Figura 8. Esquema de la base de datos-Tabla plazasblock	51
Figura 9. Esquema de la base de datos- Tabla usuario	51
Figura 10. Esquema de la base de datos- Tabla reserva	51
Figura 11. Esquema de la base de datos- Tabla plazareserva.....	52
Figura 12. Esquema de la base de datos- Tabla cliente.....	52
Figura 13. Diagrama entidad relación (EER)	53
Figura 14. Orden de paquetes Java en NetBeans IDE	54
Figura 15. Paquete BD	55
Figura 16. Paquete Modelo	55
Figura 17. Paquete Proyecto	56
Figura 18. Paquete Proyecto.Creacion.....	57
Figura 19. Paquete Reserva	57
Figura 20. Pantalla Principal.....	59
Figura 21. Pantalla de creación del Contenedor.....	60
Figura 22. Guardar datos de Usuario	61
Figura 23. Pantalla para seleccionar Contenedor.....	61
Figura 24. Pantalla de actualizar Contenedor.....	62

Figura 25. Pantalla para eliminar Contenedor	63
Figura 26. Pantalla para reservar Plazas.....	64
Figura 27. Pantalla para consultar las reservas de un Cliente.....	65
Figura 28. Pantalla para cancelar una Reserva realizada.	66
Figura 29. Pantalla para eliminar una Reserva.....	67

0

RESUMEN

TITULO: SISTEMA DE RESERVAS DE ESPACIOS GENÉRICO Y PARAMETRIZABLE.¹

AUTOR: EDWIN EDUARDO CAMARGO MEDINA²

PALABRAS CLAVES: JAVA, MYSQL, GENÉRICO, PARAMETRIZABLE, RESERVA.

DESCRIPCION:

La ciencia y la tecnología crece a cada instante sin darnos cuenta, día tras día se hace necesario el desarrollo de nuevos sistemas que sean innovadores y que brinden un mejor servicio a las personas que hacen uso de ellos. En este documento se presenta el desarrollo de un sistema de escritorio implementado con la tecnología Java 8 y como base de datos (MySQL).

A lo largo de este libro se va explicando paso por paso de donde nace la necesidad de desarrollar esta aplicación para la reserva de espacios, teniendo en cuenta que el reto más importante a cumplir es el de lograr que sea un sistema genérico y parametrizable lo cual lo hace un proyecto con alto nivel de complejidad durante su implementación, para ello se realizó el análisis de características que serán los requisitos del sistema y los diseños correspondientes tanto de prototipos del sistema, como el de la base de datos, actividades que permiten guiar y finalizar el desarrollo del proyecto.

Dentro del procedimiento de implementación de este sistema, se muestra los resultados obtenidos al poder tener una aplicación funcionando como se desea y al final poder realizar las pruebas necesarias para verificar el cumplimiento de los objetivos con los usuarios finales.

¹ Trabajo de grado

² Facultad de ingenierías físico-mecánicas. Escuela de ingeniería de sistemas e informática.

Director: M.Sc. Fernando Antonio Rojas Morales.

ABSTRACT

TITLE: GENERIC AND PARAMETERIZABLE SPACE RESERVATION SYSTEM³

AUTHOR: EDWIN EDUARDO CAMARGO MEDINA⁴

KEYWORDS: JAVA, MYSQL, GENERIC, PARAMETERIZABLE, RESERVATION.

DESCRIPTION:

Science and technology grows every moment without realizing it, day after day it becomes necessary to develop new systems that are innovative and that provide a better service to the people who make use of them. In this document presents the development of a desktop system implemented with Java 8 technology and as a database (MySQL).

Throughout this book is explained step by step from where the need to develop this application for the reservation of spaces, taking into account that the most important challenge to be met is to make it a generic and parameterizable system which a project with a high level of complexity during its implementation does this, for which purpose the analysis of characteristics that will be the requirements of the system and the corresponding designs of prototypes of the system, as well as of the database, activities that allow guiding and finalize the development of the project.

Within the implementation procedure of this system, the results obtained are shown by being able to have an application working as desired and in the end being able to perform the necessary tests to verify the fulfillment of the objectives with the end users.

³ Undergraduate Final Project

⁴ Physical- Mechanical Engineering Faculty. Systems and informatics Engineering School.

Project Director: M.Sc. Fernando Antonio Rojas Morales.

INTRODUCCIÓN

Actualmente las entidades dedicadas a la prestación de servicios de reservas se encuentran en una búsqueda por modernizar los sistemas y que le permita poder hacer cambios en la creación de nuevos espacios que se requieran aprovechar, esta búsqueda contribuye en gran manera a la mejora en la calidad del servicio que se le presta al usuario final.

Estos sistemas de prestación de servicios de reservas en Colombia tienen software dedicados a la reservación de espacios de un solo tipo, como pueden ser sistemas de reservas de cine, sistema de reserva de parqueadero, sistema de reserva de habitaciones entre muchos otros, en el cual un usuario puede reservar un espacio o varios de acuerdo a los que necesite utilizar en determinado evento. Estos lugares se comportan como matrices, que tienen un número de filas, por un número de columnas, en el cual un usuario queda ubicado al interior de ella después de realizar su respectiva reserva. Pero a la hora de querer diseñar nuevos espacios dentro del lugar por medio del sistema o que se requiera crear un espacio de tipo diferente no es posible, porque no hay ninguna opción en el software que permita modificar o configurar para así obtener un nuevo espacio.

Es importante y se hace necesario que se empiece a implementar nuevo software capaz de hacer de los sistemas de reserva algo más genérico que sea de utilidad para los clientes, además que sea también parametrizable para poder crear o adecuar espacios que pueden ser de beneficio para las empresas o clientes que utilicen estos servicios.

Por estas razones a lo largo de la exhibición de este documento se describe el procedimiento de la creación de un software que se encarga de dar solución a la

situación descrita anteriormente. Se mostrara el marco de referencia necesario para la elaboración y el entendimiento del software, la metodología empleada y el resultado obtenido al aplicar la ingeniería del software en cada etapa del desarrollo.

1. DESCRIPCION DEL PROBLEMA

1.1 JUSTIFICACION Y PLANTEAMIENTO DEL PROBLEMA

Con el crecimiento constante de la tecnología, se van encontrando y generando nuevas formas de implementar modernos sistemas que sean de mayor eficiencia y utilidad para las empresas o personas que prestan servicios de reservas de espacios de diferentes tipos.

Actualmente existen una cantidad de sistemas de reservas desarrollados que sirven para un solo tipo de servicio, ya sea para un parqueadero, un cinema, un hotel etc...Pero lastimosamente por lo nombrado anteriormente estos software solo sirven para un solo tipo de servicio de reserva en particular, ya que no se pueden crear nuevos espacios diferentes a los establecidos por el sistema, es decir, si tenemos un sistema de reserva de cines, solo puedo crear en el sistema salas de cine y no otro espacio diferente, ya sea un parqueadero u otro espacio semejante.

A partir de esta necesidad se planteó este proyecto para el diseño de un sistema de reservas que permitiese crear en un solo software varios tipos de espacios (Salas de cine, parqueaderos, habitaciones de hotel, entre otras) los cuales pudieran ser reservados, esto produce un sistema tanto genérico como parametrizable y que da solución al problema encontrado.

2. OBJETIVOS

2.1. OBJETIVO GENERAL

Analizar, diseñar, codificar y probar un sistema de reservas, que permita a un usuario configurar diferentes espacios de acuerdo a su necesidad.

2.2. OBJETIVOS ESPECÍFICOS

- Definir y analizar las principales características del sistema, identificando los diferentes actores y escenarios.
- Diseñar los prototipos de interfaz de usuario que permitan la interacción entre los usuarios y el sistema.
- Desarrollar un sistema que satisfaga los requisitos estipulados y evaluar el funcionamiento.
- Realizar pruebas con un grupo piloto de usuarios finales para realizar ajustes a la aplicación que la hagan útil.

3. MARCO REFERENCIAL

3.1 CONTEXTUALIZACION DEL PROYECTO

En esta parte del libro se expondrán las bases teóricas en las que se apoya cada una de las etapas de desarrollo del proyecto.

3.2 ESTADO DEL ARTE

En Colombia existen varias formas de realizar reservaciones, comenzando por las de persona a persona, telefónicas, correo electrónico, hasta llegar a los sistemas automatizados de reservaciones. Las reservas en línea es una práctica que ha tenido mucha acogida y ha mejorado junto a las tecnologías en que son desarrolladas. Estos sistemas pueden utilizarse para reservar boletos de cine, boletos de avión, teatros, citas médicas, hoteles, carros, restaurantes, habitaciones, eventos mundiales y locales, servicios a domicilio y mucho más.

En la actualidad, existen varios sistemas de reservas en los que se pueden destacar los desarrollados por las empresas de Cinecolombia o Cinemark, los sistemas de reservación de vuelos de Avianca, sistemas de reservación de hoteles o muchos otros los cuales son sistemas bien estructurados, que brindan un buen servicio y comodidad a los usuarios al momento de reservar un espacio en alguno de los diferentes tipos de servicio, pero cada uno de estos sistemas se ven limitados a solo cumplir esa función específica y hasta el momento no se ha encontrado algún sistema de reservas de espacios, que permita hacer algo similar a lo que se pretende con este trabajo de investigación.

Este proyecto lo que lo hace diferente a los demás, es que no solo es un sistema de reserva de espacios como cualquier otro, sino lo que se pretende lograr, es que sea tanto genérico como parametrizable y que ayude a los usuarios en la creación o modificación de nuevos espacios de acuerdo a la necesidad que tengan, esta herramienta será de gran ayuda para darle solución a dicho problema.

3.3 MARCO TEORICO

3.3.1. Sistemas de reserva. Un sistema de reservas es un sistema informático computarizado que se utiliza para guardar y recuperar información de diferentes tipos relacionadas con salas de cine, parqueaderos, hoteles, viajes aéreos, alquiler de automóviles u otros servicios.

Los sistemas de distribución global (GDS) reservan y venden boletos para múltiples aerolíneas. La mayoría de las compañías aéreas han subcontratado sus CRS a empresas GDS, que también permiten el acceso de los consumidores a través de pasarelas de Internet. Los GDS modernos generalmente permiten a los usuarios reservar habitaciones de hotel, alquiler de automóviles, boletos de avión y otras actividades y tours. También brindan acceso a reservas ferroviarias y reservas de autobuses en algunos mercados, aunque estos no siempre están integrados con el sistema principal. Estos también se utilizan para transmitir información computarizada a los usuarios de la industria hotelera, hacer la reserva y garantizar que el hotel no esté sobrevendido.

3.3.2. BlueJ. BlueJ es un entorno integrado de desarrollo para el lenguaje de programación Java, desarrollado principalmente con propósitos educativos, pero también es adecuado para el desarrollo de software a pequeña escala.

BlueJ fue desarrollado para apoyar la enseñanza y el aprendizaje de la programación orientada a objetos, y como resultado, su diseño difiere de otros entornos de desarrollo. La pantalla principal muestra gráficamente la estructura de

clases de una aplicación en desarrollo (en un diagrama muy parecido a UML), y los objetos pueden ser creados y probados interactivamente. Combinado con una interfaz de usuario simple, esta facilidad de interacción permite experimentar de manera fácil con los objetos en desarrollo. Los conceptos de la orientación a objetos (clases, objetos, comunicación a través de llamadas a métodos) son representados visualmente en el diseño de interacción en la interfaz⁵.

3.3.3. NetBeans IDE. NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso. La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software. El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles)⁶.

3.3.4. PhpMyAdmin. PhpMyAdmin es una herramienta de software gratuita escrita en PHP, diseñada para manejar la administración de MySQL a través de la Web. PhpMyAdmin admite una amplia gama de operaciones en MySQL y MariaDB. Las operaciones de uso frecuente (gestión de bases de datos, tablas, columnas, relaciones, índices, usuarios, permisos, etc.) se pueden realizar a través

⁵ Wikipedia. Bluej. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/BlueJ>

⁶ Wikipedia. Netbeans IDE. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/NetBeans>

de la interfaz de usuario, mientras que todavía tiene la capacidad de ejecutar directamente cualquier instrucción SQL.

PhpMyAdmin es un proyecto maduro con una base de código estable y flexible; Puede obtener más información sobre el proyecto y su historia y los premios que obtuvo. Cuando el proyecto cumplió 15 años, publicamos una página de celebración. El proyecto phpMyAdmin es miembro de Software Freedom Conservancy. SFC es una organización sin fines de lucro que ayuda a promover, mejorar, desarrollar y defender proyectos de software libre, libre y de código abierto (FLOSS).

3.3.5. Máquina virtual de Java. Una máquina virtual Java (en inglés Java Virtual Machine, JVM) es una máquina virtual de proceso nativo, es decir, ejecutable en una plataforma específica, capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial (el bytecode Java), el cual es generado por el compilador del lenguaje Java.

El código binario de Java no es un lenguaje de alto nivel, sino un verdadero código máquina de bajo nivel, viable incluso como lenguaje de entrada para un microprocesador físico. Como todas las piezas del rompecabezas Java, fue desarrollado originalmente por Sun.

La JVM es una de las piezas fundamentales de la plataforma Java. Básicamente se sitúa en un nivel superior al hardware del sistema sobre el que se pretende ejecutar la aplicación, y este actúa como un *punte* que entiende tanto el bytecode como el sistema sobre el que se pretende ejecutar. Así, cuando se escribe una aplicación Java, se hace pensando que será ejecutada en una máquina virtual Ja-

va en concreto, siendo ésta la que en última instancia convierte de código bytecode a código nativo del dispositivo final.

3.3.6. Modelo-vista-controlador (MVC). Es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento

3.3.7. MySQL. MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

4. METODOLOGIA

En el desarrollo de software actualmente se encuentra una gran cantidad de metodologías que ayudan en la elaboración del proyecto, estas herramientas sirven como guía para tener un marco de referencia de cómo encaminar el proyecto desde un comienzo hasta obtener un resultado final, se puede decir que son mecanismos que permiten encontrar una solución al problema propuesto, la planificación del proyecto, la variación de los requisitos durante el proyecto, el aseguramiento de la calidad del software, la creación de la documentación, entre otras.

Este proyecto consiste en desarrollar una aplicación que será usada por una gran cantidad de usuarios, por tanto, habrá momentos en donde se manejará una cantidad considerable de carga y la calidad de la herramienta dependerá de la preparación que se tenga para afrontar estos factores.

Un software que se crea para prestar un servicio a toda una comunidad requiere de una excelente preparación y de una buena acogida por parte de los usuarios, que en últimas, de la percepción de ellos dependerá el éxito del proyecto. Por esto se ha decidido realizar el proyecto basándose en la metodología de desarrollo iterativo, ya que es un modelo que se enfatiza en la implementación extrema iterativa y en la usabilidad. Al finalizar todo el proceso y luego de un seguimiento muy riguroso, se tendrá como resultado un producto muy completo que cumplirá tanto con los requerimientos técnicos de los desarrolladores como con los requerimientos de usabilidad del usuario.

4.1. DESARROLLO ITERATIVO Y CRECIENTE

Es un modelo de desarrollo que consiste en un conjunto de tareas agrupadas en pequeñas etapas repetitivas, ha sido creado para corregir errores del modelo de cascada. Es muy usado en diversas metodologías ya que es un modelo que se enfatiza mucho en la programación extrema y en la calidad de usabilidad. Se compone por distintas etapas de desarrollo en cada incremento, se inicia con el análisis y se termina con la aprobación del producto.

Cada iteración se compone de un determinado proceso de trabajo, que da un resultado completo para el producto final de la etapa. Cada requerimiento debe tener una implementación muy completa con documentación y pruebas de rendimiento antes de finalizar la iteración. La idea central del modelo es que en cada repetición se logre evolucionar el producto con respecto a los productos completados en anteriores repeticiones, siempre agregando más funcionalidades que cumplan requisitos y que mejoren cada vez más su calidad. La metodología se ha creado basándose en dos premisas:

Los usuarios nunca saben qué es lo que necesitan para satisfacer sus necesidades. En la implementación, los procesos tienden a cambiar.

4.1.1 Inicialización. En esta etapa se realiza la primera versión de la herramienta. Esta primera parte debe ser una implementación simple de los requerimientos del sistema. La meta de esta etapa es tener como resultado una aplicación que se le pueda dar al usuario para que este pueda interactuar con sus interface y poder retroalimentar el proceso. Se tendrá por parte del usuario algunos aspectos claves del problema y una posible solución para que esta pueda ser tomada en cuenta para la implementación de la siguiente iteración.

4.1.2. Iteración. Esta etapa es la más importante del modelo ya que involucra el rediseño e implementación de una tarea contenida en la lista de control de proyecto y el análisis y revisión de la versión más reciente del programa. El objetivo de todas las modificaciones que se hagan es que sean simples y directas para así garantizar el éxito en el desarrollo de la siguiente iteración. La retroalimentación por parte del usuario y la revisión de funcionalidades disponibles para la herramienta, son muy importantes a la hora de realizar el análisis de una iteración. En la etapa de análisis se involucra todo lo que tiene que ver con la estructura, la modularidad, confiabilidad, eficiencia y eficacia de toda la documentación de la herramienta.

4.1.3 Lista de control del proyecto. Es usada para guiar el proceso de iteración del modelo. Contiene un historial de todas las tareas que están por realizar, como nuevas funcionalidades y aspectos de rediseño. Todo esto para que se pueda tener en cuenta en la implementación de la siguiente iteración del modelo. Esta lista es revisada y analizada constantemente para así llevar un control del avance del proyecto, cuando es conveniente se actualiza siempre con respecto a los resultados que se tengan de los análisis realizados a anteriores iteraciones.

5. ANALISIS DE CARACTERISTICAS DEL SISTEMA

En esta parte del libro se muestran las características que fueron creadas durante el desarrollo del proyecto.

5.1. DESCRIPCION DE LA APLICACION

El sistema genérico y parametrizable de reservas es un sistema que permite crear espacios con una cantidad de plazas que pueden ser reservadas y consultadas; se desea que el sistema de reservas sea una aplicación de escritorio.

El sistema ofrece las siguientes funcionalidades:

- Crear espacios.
- Gestionar plazas
 - Reservar plazas
 - Consultar reserva
 - Cancelar reserva

Crear espacios permite a un usuario crear nuevos espacios genéricamente los cuales van organizados por filas, cada fila tendrá una cantidad de plazas disponibles para reservar para clientes particulares. Es posible crear espacios sobre otros; esto permite el manejo de niveles. También es posible ver el nuevo espacio creado gráficamente por medio de una interface.

Gestionar plazas permite realizar las siguientes actividades:

- Reservar plazas facilita al usuario realizar una reserva, especificando la plaza a reservar con sus respectivos: fecha, horario y tarifa determinada. Es posible reservar múltiples plazas consecutivamente. Hay que tener en

cuenta que el usuario es quien crea los espacios y realiza las reservas, mientras que el cliente es a quien se le hace las reservas.

La consulta de plazas se puede hacer por:

Horario de disponibilidad.

La consulta de disponibilidad según horario muestra los horarios disponibles de las plazas según el servicio que se esté ofreciendo, además muestra el estado de las plazas (Disponible/Ocupado).

- Consultar reserva facilita al usuario averiguar las plazas reservadas y la hora de la reserva en caso de que se le solicite dar esa información.
- Cancelar reserva facilita al usuario anular la reserva en caso de que el cliente no pueda hacer uso de ella.

Adicionalmente el sistema permite al usuario poder eliminar las reservas.

5.2. ANALISIS DEL TEXTO

El sistema genérico y parametrizable de reservas es un sistema que permite crear espacios con una cantidad de plazas que pueden ser reservadas y consultadas; se desea que el sistema de reservas sea una aplicación de escritorio.

El sistema ofrece las siguientes funcionalidades:

- Crear espacios.
- Gestionar plazas
- Reservar plazas
- Consultar reserva

- Cancelar reserva

Crear espacios permite a un usuario crear nuevos espacios genéricamente los cuales van organizados por filas, cada fila tendrá una cantidad de plazas disponibles para reservar para clientes particulares. Es posible crear espacios sobre otros; esto permite el manejo de niveles. También es posible ver el nuevo espacio creado gráficamente por medio de una interface.

Gestionar plazas permite realizar las siguientes actividades:

- Reservar plazas facilita al usuario realizar una reserva, especificando la plaza a reservar con sus respectivos: fecha, horario y tarifa determinada. Es posible reservar múltiples plazas consecutivamente. Hay que tener en cuenta que el usuario es quien crea los espacios y realiza las reservas, mientras que el cliente es a quien se le hace las reservas.

La consulta de plazas se puede hacer por:

Horario de disponibilidad.

La consulta de disponibilidad según horario muestra los horarios disponibles de las plazas según el servicio que se esté ofreciendo, además muestra el estado de las plazas (Disponible/Ocupado).

- Consultar reserva facilita al usuario averiguar las plazas reservadas y la hora de la reserva en caso de que se le solicite dar esa información.
- Cancelar reserva facilita al usuario anular la reserva en caso de que el cliente no pueda hacer uso de ella.

Adicionalmente el sistema permite al usuario poder eliminar las reservas.

5.3. TABLA DE SUSTANTIVOS Y VERBOS

Sustantivos	Verbos
Sistema genérico	
Sistema parametrizable	
Reserva de espacios	
Sistema	
Espacio	<i>Crear, ver</i>
Plaza	<i>Crear , consultar, reservar</i>
Sistema de reservas	
Sistema de escritorio	
Pantalla principal	
Interface	
Cliente	<i>Registrar, consultar, modificar y archivar.</i>
Actividad	
Reserva	<i>Crear, realizar, cancelar,</i>
Fila	
Usuario	<i>Registrar, consular, modificar y archivar.</i>
Horario de disponibilidad	
Estado de las plazas	
Horario	
Servicio	
Estado	
Información	
Tarifa	
Fecha	

Tabla 1.Sustantivos y verbos de la descripción.

La Tabla 1 muestra los sustantivos candidatos a clases.

- Reserva de espacios = Sistema de reservas = **SistemaDeReserva**
- Reserva
- Sistema (Eliminada)
- Espacio= Contenedor
- Plaza
- Pantalla principal= Interface (**Eliminada**)
- Cliente
- Fila = **Lugar**
- Usuario

5.4. ESCENARIOS

- El usuario quiere crear un espacio cualquiera (Salas de cine, parqueaderos, Restaurantes etc.)
- El usuario crea un lugar y quiere agregar un número indeterminado de plazas al lugar.
- El usuario ha creado un espacio y necesita eliminar alguna plaza.
- El usuario ha guardado un espacio y necesita modificar para añadir o eliminar plazas.
- El usuario quiere reservar una plaza en determinado lugar.
- El usuario quiere reservar varias plazas consecutivas en un lugar.
- El usuario quiere ver la disponibilidad de plazas que hay en un determinado espacio.
- El usuario quiere mirar los horarios de disponibilidad de las plazas a reservar.
- El cliente no puede llegar y requiere cancelar la reserva realizada.
- El cliente olvido la información de su reserva y quiere consultarla en el sistema.

- El cliente ya adquirió una reserva y llama para mirar si puede reservar otra plaza cercana a la que ya tiene.

5.5. TARJETAS CRC

Contenedor	Colaboradores
<i>Responsabilidades</i>	
<i>Modificar un espacio</i>	Lugar
<i>Acepta la solicitud de reserva</i>	Plaza
<i>Almacena plazas</i>	Usuario

Tabla 2. Tarjeta CRC de la clase Contenedor

Reserva	Colaboradores
<i>Responsabilidades</i>	
<i>Realizar reserva de plazas</i>	Plaza
<i>Confirmar la reserva</i>	Cliente
<i>Cancelar la reserva</i>	
<i>Eliminar la reserva</i>	
<i>Buscar la reserva realizada</i>	

Tabla 3. Tarjeta CRC de la clase Reserva

Lugar	Colaboradores
<i>Responsabilidades</i> <i>Encontrar plazas consecutivas.</i> <i>Mostrar las plazas</i>	Plaza Contenedor

Tabla 4. Tarjeta CRC de la clase Lugar

Plaza	Colaboradores
<i>Responsabilidades</i> <i>Crea las plazas</i> <i>Obtiene las plazas</i> <i>Guarda las plazas</i> <i>Almacena una colección de plazas.</i> <i>Agrega varias plazas.</i> <i>Guarda las plazas agregadas</i> <i>Elimina las plazas que se requiera</i> <i>Acepta reservas</i> <i>Almacena el estado de la reserva(Disponible/Reservado)</i>	Reserva Contenedor Lugar Usuario

Tabla 5. Tarjeta CRC de la clase Plaza

Usuario	Colaboradores
<i>Responsabilidades</i>	Contenedor
<i>Hacer reservas</i>	
<i>Crear los espacios genéricos en el sistema.</i>	

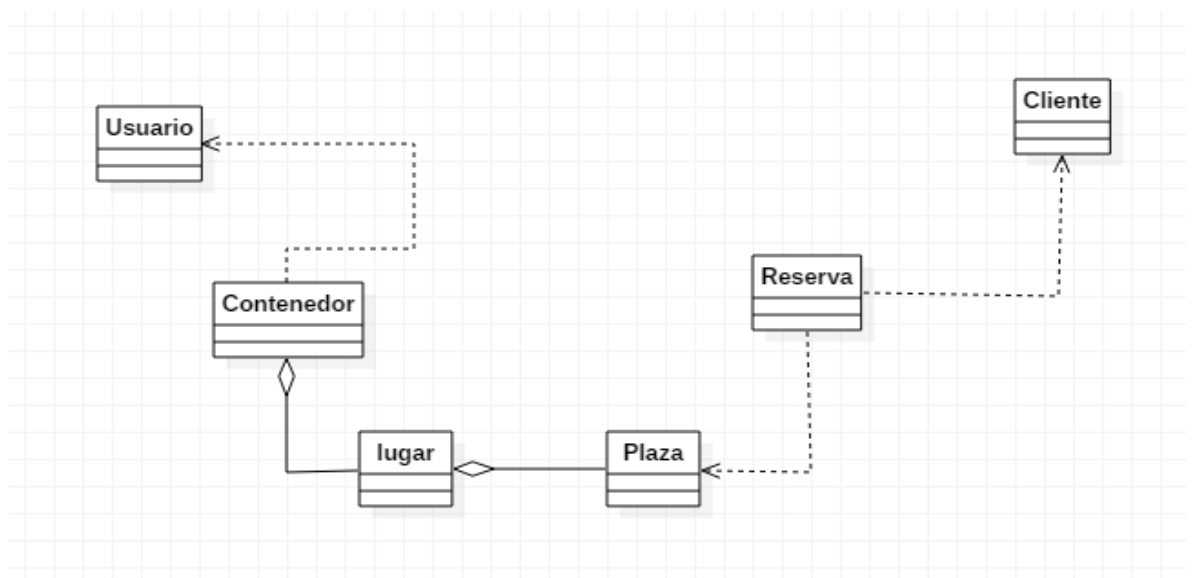
Tabla 6. Tarjeta CRC de la clase Usuario

Cliente	Colaboradores
<i>Responsabilidades</i>	Reserva
<i>Asistir a la reserva</i>	

Tabla 7. Tarjeta CRC de la clase Cliente

5.6. MODELO DE OBJETOS DEL DOMINIO

Figura 1. Modelo de objetos



En la imagen anterior podemos observar cómo queda conformado el modelo de objetos de dominio para la realización del proyecto.

5.7. LISTA DE CARACTERISTICAS DEL SISTEMA

1. Crear espacios ya sea uno o varios de diferente género (Salas de cine, restaurantes, parqueaderos, habitaciones de hotel, etc...)
2. Ordenar por filas las plazas
3. Guardar, modificar y eliminar plazas.
4. Ver gráficamente los espacios
5. Modificar los espacios creados.
6. Consultar disponibilidad de plazas por fecha y hora
7. Reservar plazas
8. Marcar y desmarcar plazas a reservar
9. Guardar, consultar, cancelar y eliminar una reserva.
10. Guardar información del cliente para una reserva.

5.8. PERSPECTIVA DEL SISTEMA

Este proyecto cumplirá con el desarrollo de una aplicación de escritorio para la creación de cualquier tipo de espacio y la posterior reservación de algún servicio según el tipo de espacio que haya sido configurado. Además la aplicación debe tener un entorno amigable donde el usuario pueda hacer uso de todas las herramientas necesarias que le permitan diseñar los espacios y reservar para el beneficio de los clientes.

5.9. FUNCIONES DEL SISTEMA

5.9.1. Crear contenedor. La aplicación permite crear un contenedor con algunos detalles que lo identifican y lo diferencian de los demás, además de guardar la información del usuario.

5.9.2. Modificar contenedor. Cuando se ha guardado un contenedor la aplicación permite actualizarlo, cambiando todos los detalles que desee a excepción del identificador.

5.9.3. Eliminar contenedor. Después de guardado un contenedor y si se requiere, la aplicación permite eliminarlo totalmente del sistema para la creación de uno nuevo o simplemente borrarlo de la base de datos.

5.9.4. Reservar plazas. Cuando se ha creado un contenedor la aplicación permite realizar reserva de plazas, guardando los detalles de reserva y la información del cliente que hará uso de ella.

5.9.5. Buscar plazas. Después de realizar reservas podrá buscar plazas disponibles haciendo un filtro por la fecha y la hora.

5.9.6. Consultar reservas. Cuando en la aplicación se han realizado reservas, se podrá consultar la información haciendo filtros por la Cedula de Ciudadanía del cliente, la fecha y hora de la reserva o la plaza reservada en una fecha específica.

5.9.7. Cancelar reserva. Si por algún motivo se requiere cancelar la reserva, la aplicación le permite tener la opción de realizarlo.

5.9.8. Eliminar reserva. Cuando se han realizado reservas la aplicación permite eliminarla del sistema para la actualización de una nueva que necesite el cliente.

5.10. CARACTERISTICAS DEL USUARIO

La aplicación de escritorio permitirá su uso exclusivamente al rol del usuario. El usuario será el responsable de la creación, modificación y realización de las reservas, además de la interpretación y visualización de los datos obtenidos por el sistema.

5.11. RESTRICCIONES DEL SISTEMA

- Uso de base de datos relacional (MySQL)
- Lenguaje y tecnologías en JAVA
- Desarrollado en NetBeans IDE
- Es necesario un servidor donde se instancie la base de datos.

5.12. ACTORES DEL SISTEMA

Figura 2. Actores del sistema



5.13. DESCRIPCION DE ACTORES

Actor	Usuario
Caso de uso	Crear contenedor, actualizar contenedor, eliminar contenedor, realizar reserva, consultar reserva, cancelar reserva y eliminar reserva.
Descripción	El usuario no tiene restricciones en las tareas que realice en el sistema, tiene acceso completo al sistema incluyendo tareas a la base de datos.
Tipo	Primario

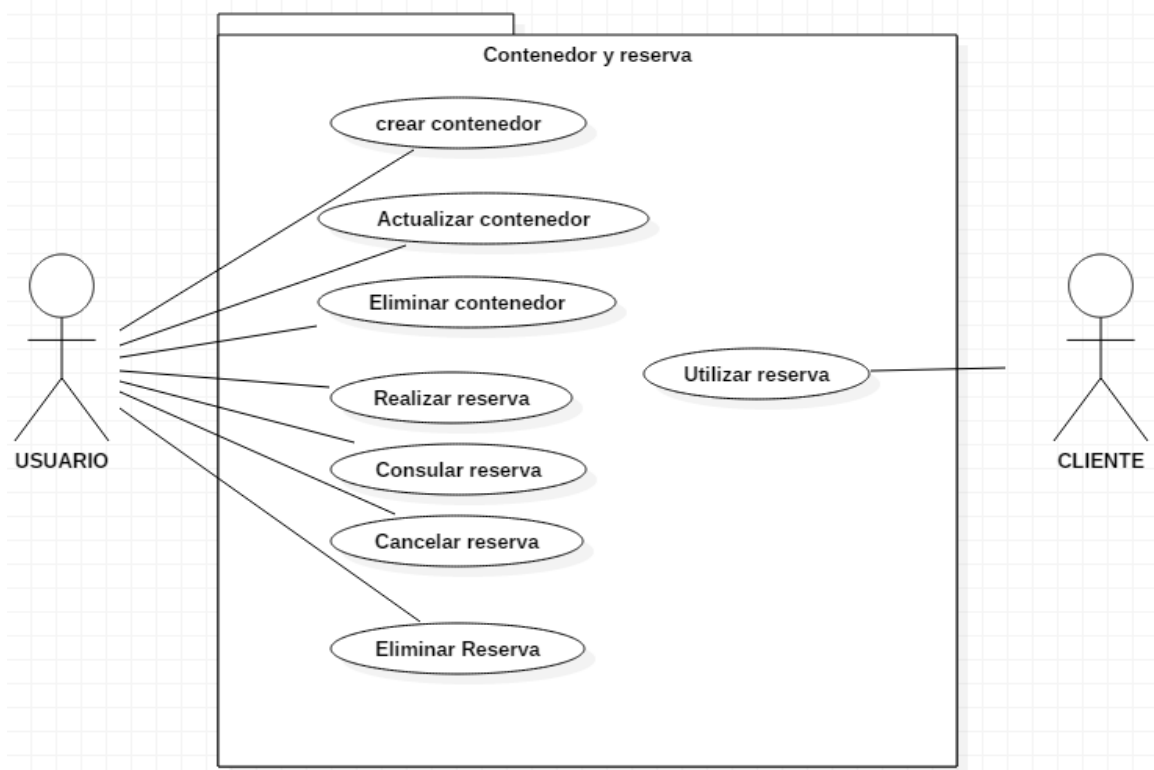
Tabla 8. Descripción de Actores – Usuario

Actor	Cliente
Caso de uso	Utilizar reserva
Descripción	El cliente solo tiene la tarea de utilizar la reserva que le realizo el usuario en el sistema.
Tipo	Secundario

Tabla 9. Descripción de Actores – Cliente

5.14. DIAGRAMA DE CASOS DE USO

Figura 3. Diagrama de casos de uso



5.15. DESCRIPCION DE DIAGRAMAS DE CASOS DE USO

Título del Caso de Uso	Crear Contenedor	ID del Caso de Uso	CU 001
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción de "Creación"			
Entidades/Actores Involucrados			
Usuario			
Condiciones de Entrada			
Ninguna, solo seleccionar la opción			
Flujo Normal de Eventos			
	Usuario	Sistema	
1.	Selecciona la opción "Creación"	Presenta en pantalla una ventana con unos campos para ingresar la información del nuevo contenedor, pudiendo crear las plazas que desee y además ingresando los datos del usuario quien lo crea.	
1.1	Selecciona la opción "Guardar"	Guarda toda la información del contenedor, de las plazas bloqueadas y además la información del usuario.	
Condiciones de Salida			
Puede salir sin guardar nada o apenas guarde el nuevo contenedor.			
Caminos de Excepción/Flujos alternativos			
Ninguno.			

Tabla 10. Descripción del caso de uso Crear Contenedor

Título del Caso de Uso	Actualizar Contenedor	ID del Caso de Uso	CU 002
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Edición"			
Entidades/Actores Involucrados			
Usuario			
Condiciones de Entrada			
Tener contenedores creados para actualizar			
Flujo Normal de Eventos			
	Usuario	Sistema	
1.	Seleccionar la opción "Edición"	Presenta en pantalla la opción de seleccionar el contenedor que desea actualizar para cargarlo en la ventana.	
1.1	Seleccionar la opción "Actualizar"	Actualiza los cambios hechos en el contenedor a excepción de los datos del usuario que quedan bloqueados.	
Condiciones de Salida			
Puede salir sin actualizar o apenas se actualice el contenedor.			
Caminos de Excepción/Flujos alternativos			
Ninguno			

Tabla 11. Descripción del caso de uso Actualizar Contenedor

Título del Caso de Uso	Eliminar Contenedor	ID del Caso de Uso	CU 003
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Edición"			
Entidades/Actores Involucrados			
Usuario			
Condiciones de Entrada			
Tener contenedores creados para Eliminar			
Flujo Normal de Eventos			
	Usuario	Sistema	
1.	Seleccionar la opción "Edición"	Presenta en pantalla la opción de seleccionar el contenedor que desea eliminar para cargarlo en la ventana.	
1.1	Seleccionar la opción "Eliminar"	Elimina totalmente el contenedor del sistema.	
Condiciones de Salida			
Puede salir sin eliminar o apenas elimine el contenedor.			
Caminos de Excepción/Flujos alternativos			
Ninguno			

Tabla 12. Descripción del caso de uso Eliminar Contenedor

Título del Caso de Uso	Realizar Reserva	ID del Caso de Uso	CU 004
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Reservas"			
Entidades/Actores Involucrados			
Usuario			

Condiciones de Entrada		
Tener contenedores creados para reservar		
Flujo Normal de Eventos		
	Usuario	Sistema
1.	Seleccionar la opción "Reservas"	Presenta en pantalla la opción de seleccionar el contenedor en el cual desea reservar para cargarlo en la ventana, después seleccionara las plazas que quiere reservar para darle en la opción reservas la cual despliega un formulario en el cual ingresara la información de la reserva como la del cliente.
1.1	Seleccionar la opción "Guardar"	Guarda la información de la reserva y el cliente además de la plaza o plazas que fueron reservadas en el sistema.
Condiciones de Salida		
Ninguna.		
Caminos de Excepción/Flujos alternativos		
Ninguno		

Tabla 13. Descripción del caso de uso Realizar Reserva

Título del Caso de Uso	Consultar Reserva	ID del Caso de Uso	CU 005
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Detalles"			
Entidades/Actores Involucrados			

Usuario		
Condiciones de Entrada		
Tener reservas realizadas en el contenedor.		
Flujo Normal de Eventos		
	Usuario	Sistema
1.	Seleccionar la opción "Detalles"	Carga la ventana de consultas, las cuales se pueden realizar por la identificación del usuario, la fecha y hora de reserva o por la fecha y plaza reservada.
1.1	Seleccionar la opción "Buscar"	Busca los detalles de la reserva como los datos de la reserva, los datos del cliente y si desea con el Id de la reserva puede consultar las plazas reservadas.
Condiciones de Salida		
Ninguna.		
Caminos de Excepción/Flujos alternativos		
Ninguno		

Tabla 14. Descripción del caso de uso Consultar Reserva

Título del Caso de Uso	Cancelar Reserva	ID del Caso de Uso	CU 006
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Detalles"			
Entidades/Actores Involucrados			
Usuario			
Condiciones de Entrada			

Tener búsquedas de reservas realizadas.		
Flujo Normal de Eventos		
	Usuario	Sistema
1.	Seleccionar la opción "Buscar"	Presenta en pantalla todas las reservas según por la opción de búsqueda realizada
1.1	Seleccionar dentro de un registro de reserva para que aparezca la opción de "Cancelar"	Cancela la reserva realizada pasando su estado de activo ha cancelado.
Condiciones de Salida		
Ninguna.		
Caminos de Excepción/Flujos alternativos		
Ninguno		

Tabla 15. Descripción del caso de uso Cancelar Reserva

Título del Caso de Uso	Eliminar Reserva	ID del Caso de Uso	CU 007
Descripción General del Caso de Uso o Requisito			
Es iniciado por el usuario al seleccionar la opción "Detalles"			
Entidades/Actores Involucrados			
Usuario			
Condiciones de Entrada			
Tener búsquedas de reservas realizadas.			
Flujo Normal de Eventos			
	Usuario	Sistema	
1.	Seleccionar la opción "Buscar"	Presenta en pantalla todas las reservas según por la opción de búsqueda realizada	

1.1	Seleccionar dentro de un registro de reserva para que aparezca la opción de “Eliminar”	Elimina la reserva realizada del sistema.
Condiciones de Salida		
Ninguna.		
Caminos de Excepción/Flujos alternativos		
Ninguno		

Tabla 16. Descripción del caso de uso Eliminar Reserva

Título del Caso de Uso	Utilizar Reserva	ID del Caso de Uso	CU 008
Descripción General del Caso de Uso o Requisito			
Es iniciado por el Cliente al momento de utilizar su plaza reservada.			
Entidades/Actores Involucrados			
Cliente			
Condiciones de Entrada			
Dirigirse al lugar de reserva			
Flujo Normal de Eventos			
	Cliente	Sistema	
1.	Utilizar la reserva que le realizó el usuario.	Ninguna.	
Condiciones de Salida			
Ninguna.			
Caminos de Excepción/Flujos alternativos			
Ninguno			

Tabla 17. Descripción del caso de uso Utilizar Reserva

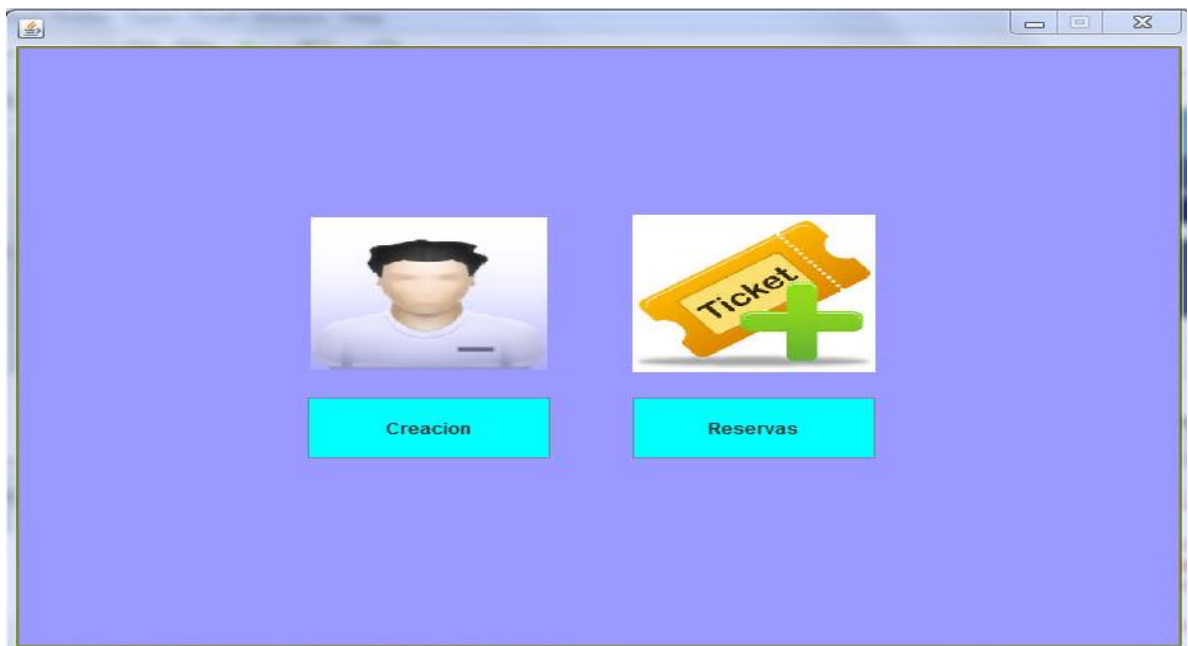
6. DISEÑO Y ESTRUCTURA DE DATOS

Finalizando la parte del documento donde se exponen todas las características necesarias para la realización del sistema, se procede a la parte de visualización de los prototipos de diseño y la estructura de datos que se realizó para el desarrollo del proyecto.

6.1. DISEÑO DE PROTOTIPOS

Con base a lo nombrado en la parte de análisis de características se crearon tres ventanas como prototipos para mostrar un diseño semejante a las ventanas que serán implementadas en la parte del desarrollo donde los usuarios pudieron interactuar con el sistema.

Figura 4. Pantalla principal del sistema.



En la Figura 4. Se tiene el prototipo de la pantalla principal, la cual estará conformada por dos botones:

- **Creación:** Es el botón que redirige a la pantalla de creación del contenedor, donde el usuario podrá crear todos los espacios genéricos y parametrizables que desee.
- **Reservas:** Es el botón que redirige a la pantalla de reservar plazas en un contenedor previamente creado.

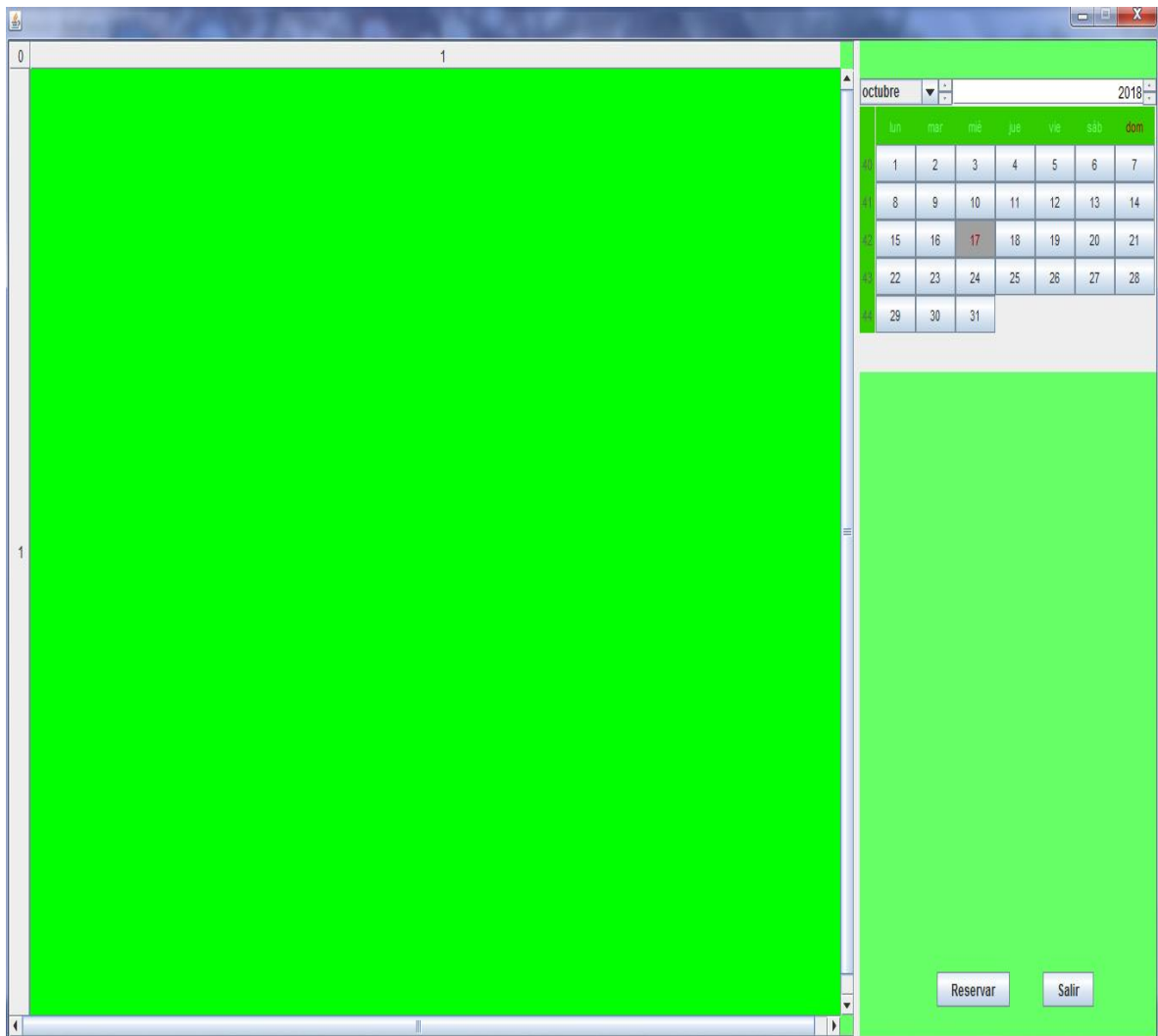
Figura 5. Pantalla de creación del contenedor.

The screenshot shows a software window titled "Características" and "Usuario". The main area is a large green rectangle. On the right side, there are several input fields and buttons:

- Descripcion:** A section containing three input fields: "Titulo:", "Tipo:", and "Descripcion:".
- Propiedades:** A section containing two spinners for "Nº Filas" and "Nº Columna", and an "Asignar" button.
- Selection:** A table with two columns: "Inhabilitar" and "Habilitar". The rows are "Seleccion", "Columna", and "Fila". Each cell contains a radio button.
- Buttons:** "Salir" and "Siguiente" buttons are located at the bottom right of the window.

En la Figura 5. Se tiene el prototipo de la pantalla de creación del contenedor, la cual tiene los datos que tendrá el contenedor y a la parte izquierda de la pantalla se puede observar el espacio donde serán creadas las plazas según las filas y columnas ingresadas.

Figura 6. Pantalla de reservar plazas



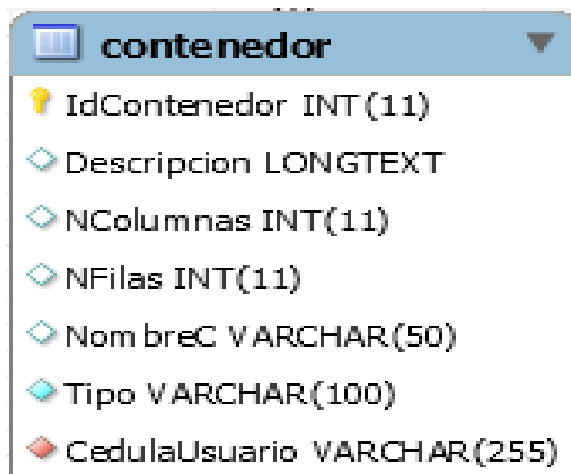
En la Figura 6. Se tiene el prototipo de la pantalla de reservar plazas, la cual cargara en su pantalla los contenedores creados para reservar la plaza o plazas que desee el cliente, además tendrá una fecha para indicar el día que se realizó la reserva.

6.2. DISEÑO Y ESQUEMA DE LA BASE DE DATOS

Para la creación de la base de datos para la aplicación de escritorio desarrollada, se utilizó una base de datos relacional (MySQL).

Las tablas presentadas a continuación son todas aquellas que son importantes en el sistema para el funcionamiento de la aplicación de escritorio, en las siguientes figuras se muestran las tablas que están conformadas en la parte superior por el nombre de la tablas, en la parte izquierda de la figura se tendrá el nombre de los campos que contiene cada tabla, y en la parte derecha de la figura se tiene el tipo de datos utilizado en cada campo, acompañado del tamaño máximo que tiene cada campo.

Figura 7. Esquema de la base de datos - Tabla contenedor



contenedor	
IdContenedor	INT(11)
Descripcion	LONGTEXT
NColumnas	INT(11)
NFilas	INT(11)
NombreC	VARCHAR(50)
Tipo	VARCHAR(100)
CedulaUsuario	VARCHAR(255)

Figura 8. Esquema de la base de datos-Tabla plazasblock



Figura 9. Esquema de la base de datos- Tabla usuario

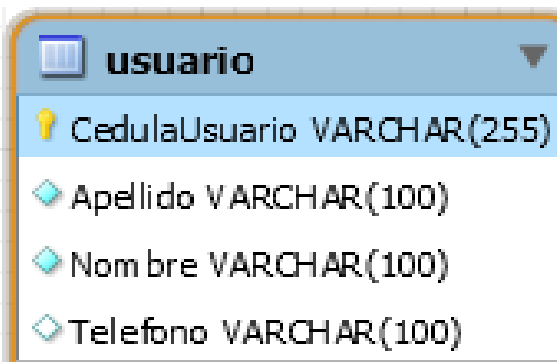


Figura 10. Esquema de la base de datos- Tabla reserva

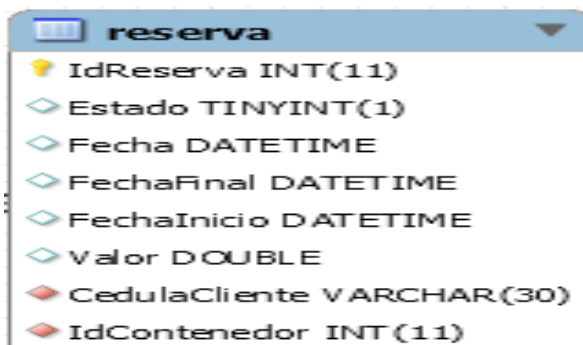
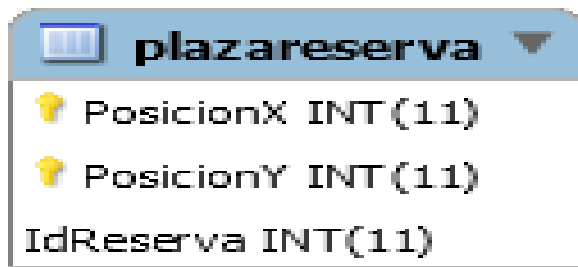
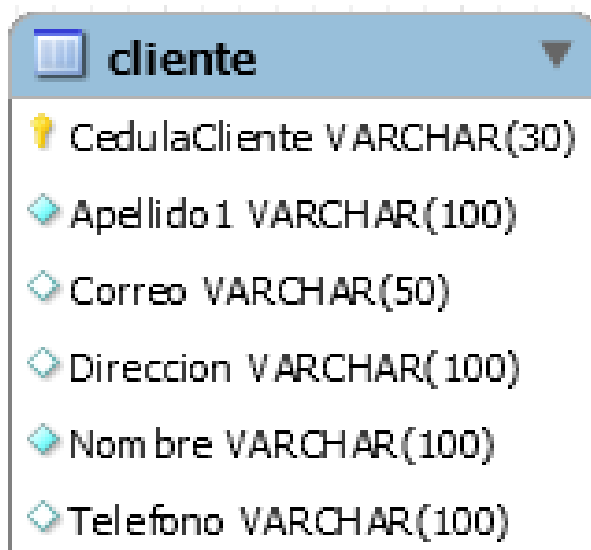


Figura 11. Esquema de la base de datos- Tabla plazareserva



plazareserva	
PosicionX	INT(11)
PosicionY	INT(11)
IdReserva	INT(11)

Figura 12. Esquema de la base de datos- Tabla cliente



cliente	
CedulaCliente	VARCHAR(30)
Apellido1	VARCHAR(100)
Correo	VARCHAR(50)
Direccion	VARCHAR(100)
Nombre	VARCHAR(100)
Telefono	VARCHAR(100)

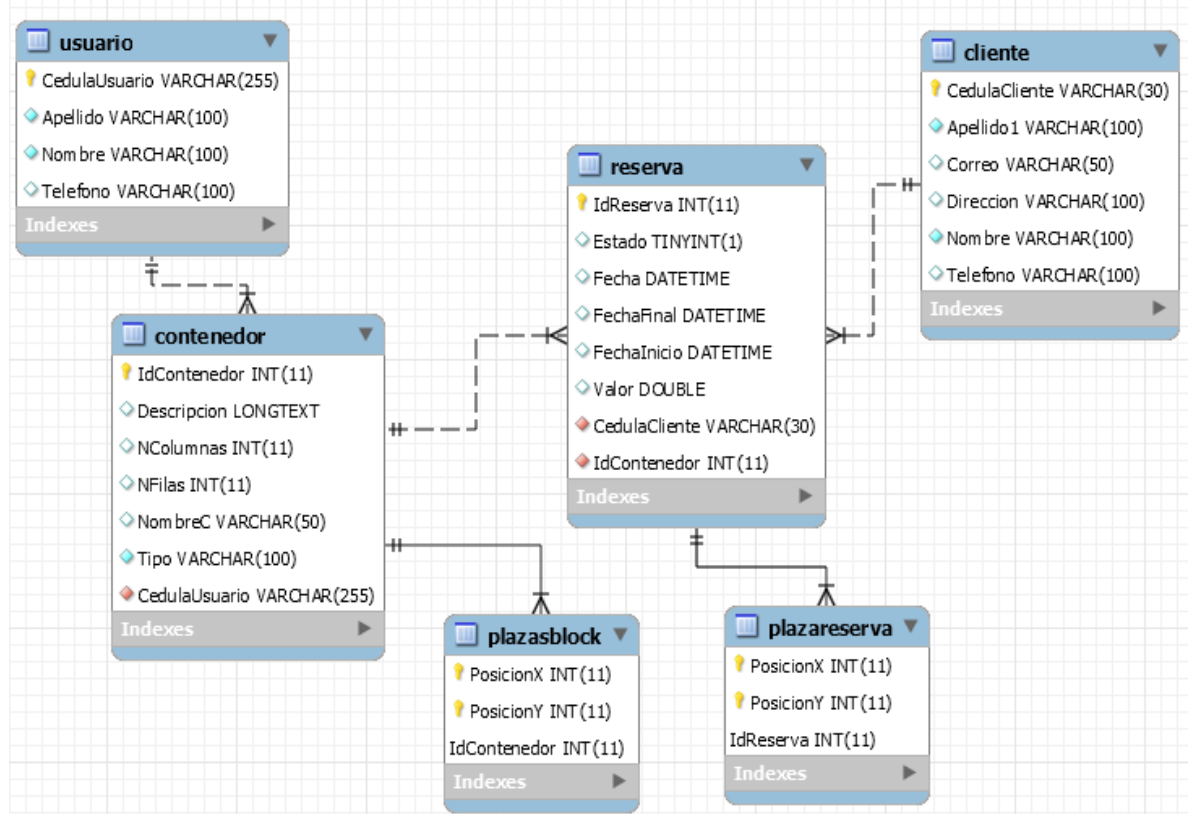
Las tablas presentadas en las figuras anteriores permiten el funcionamiento de la base de datos con la aplicación de escritorio de la siguiente manera:

- **Contenedor:** Guarda toda la información del contenedor cuando se crea un nuevo espacio en el sistema.

- **Plazasblock:** Guarda en una lista todas las plazas bloqueadas en el momento de crear un nuevo espacio en el sistema.
- **Usuario:** Guarda los datos del usuario quien crea el contenedor en el sistema.
- **Reserva:** Guarda todos los detalles de la reserva que realiza el usuario para el cliente.
- **Plazareserva:** Guarda en una lista todas las plazas reservadas en cada una de las reservas realizadas.
- **Cliente:** Guarda toda la información del cliente que pide la reserva para determinadas plazas en determinado contenedor.

6.3. DIAGRAMA ENTIDAD RELACION DE LA BASE DE DATOS (EER)

Figura 13. Diagrama entidad relación (EER)



En la Figura 13. Se tiene el diagrama entidad relación de las tablas donde se tiene relaciones uno a muchos (1: N) y muchos a uno (N: 1) que indican la cardinalidad del diagrama de la base de datos.

6.4. ORDEN DE LOS PAQUETES JAVA EN NETBEANS IDE

Los paquetes java quedan ordenados en el proyecto en NetBeans de la siguiente manera:

Figura 14. Orden de paquetes Java en NetBeans IDE

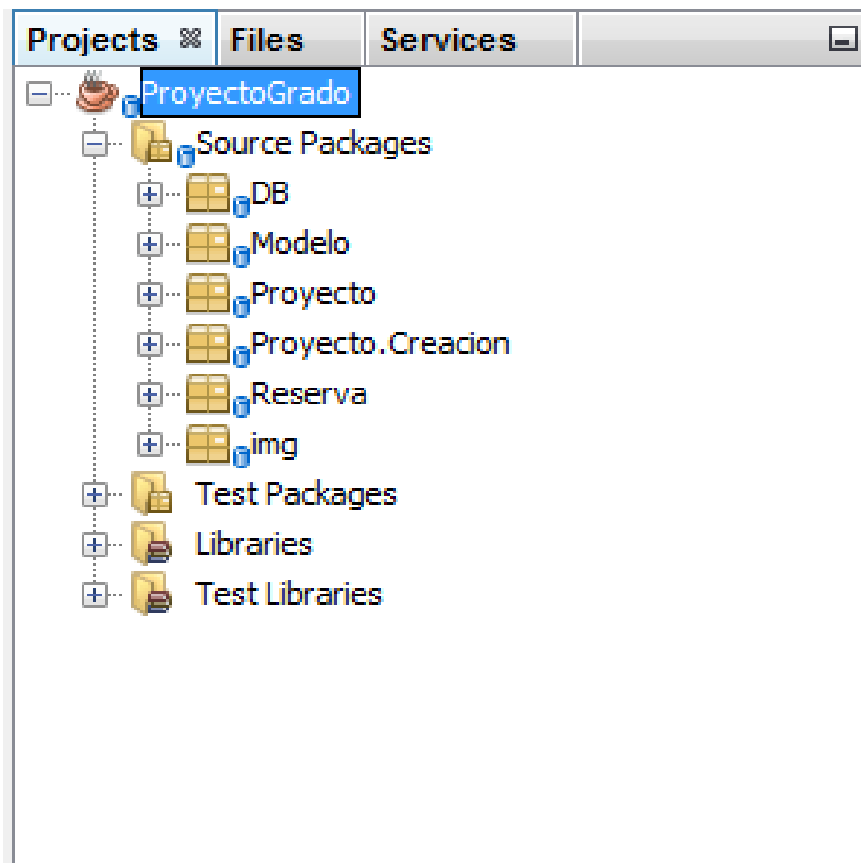
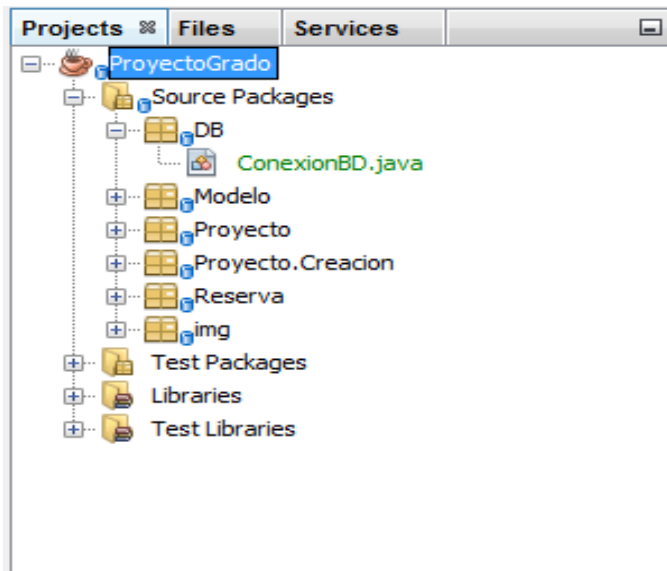
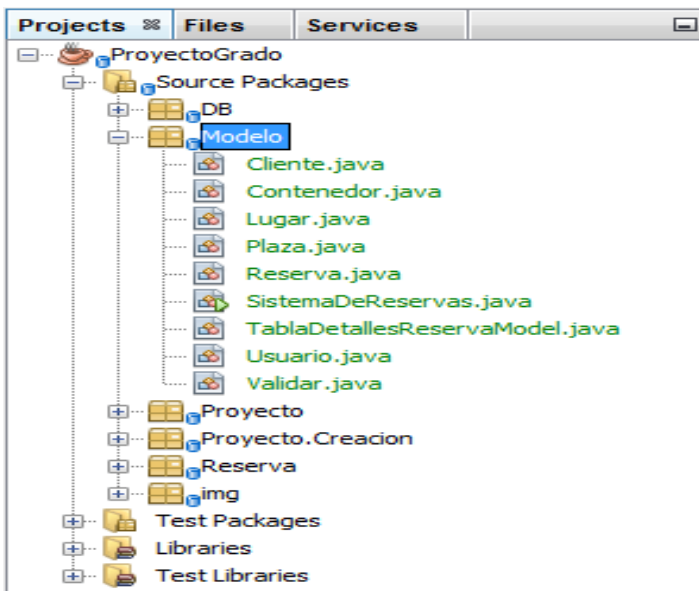


Figura 15. Paquete BD



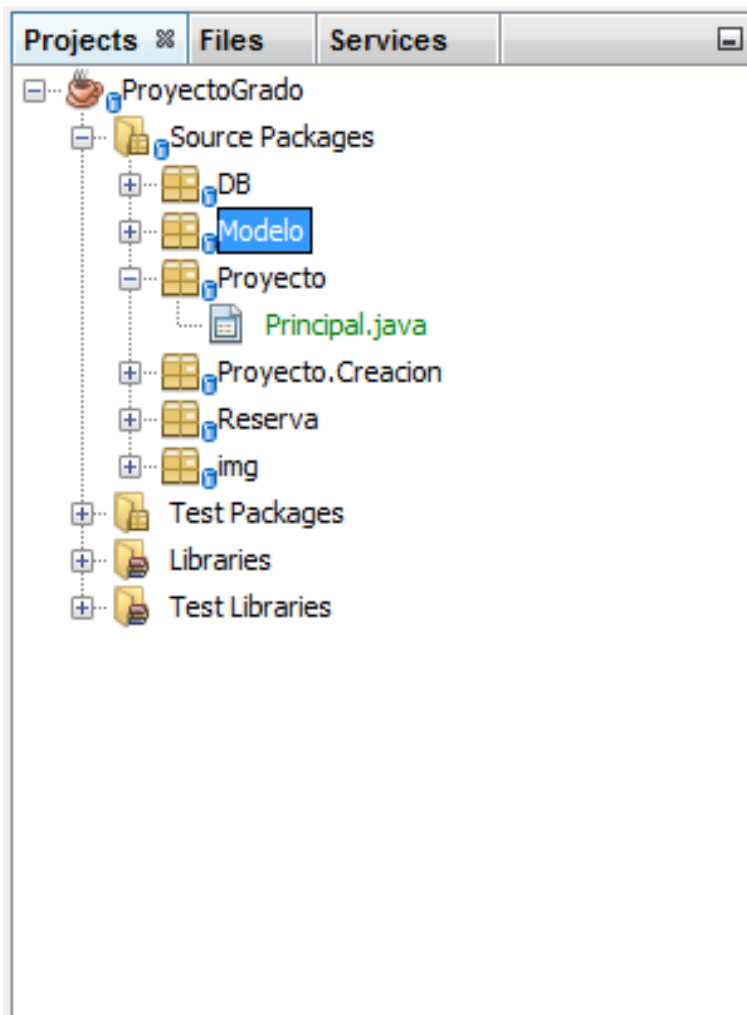
La Figura 15. Contiene la clase que hace la conexión con la base de datos MySQL.

Figura 16. Paquete Modelo



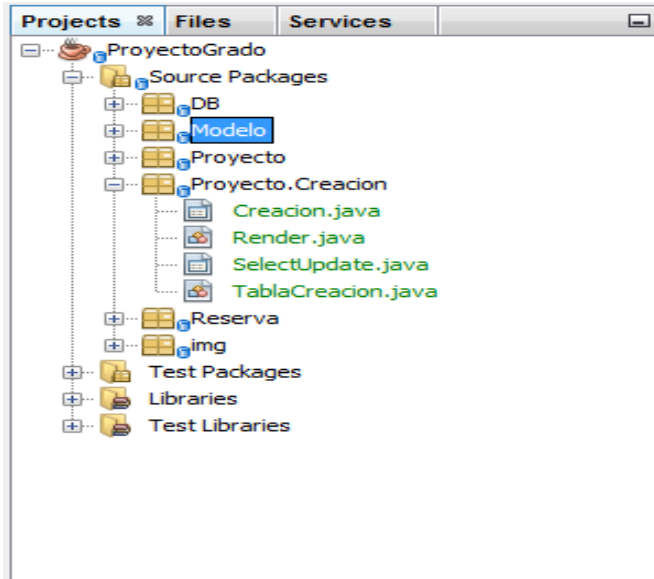
La Figura 16. Contiene todas las clases que se encontraron en el análisis de características realizado en la parte de la descripción del proyecto, la clase Main donde se ejecuta la aplicación y además de tener una clase que sirve para validar números y letras en los campos del sistema.

Figura 17. Paquete Proyecto



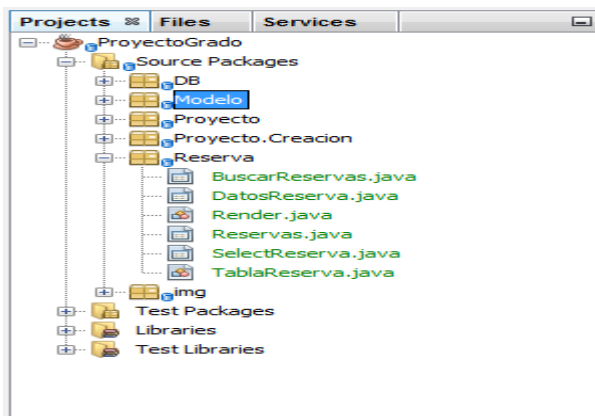
La Figura 17. Contiene la clase que presenta la ventana principal de la aplicación.

Figura 18. Paquete Proyecto.Creacion



La Figura 18. Contiene todas las clases que permiten la creación, actualización y eliminación del contenedor, también la clase que abre la ventana de creación y además la clase TablaCreacion que permite el manejo de las plazas en la aplicación.

Figura 19. Paquete Reserva



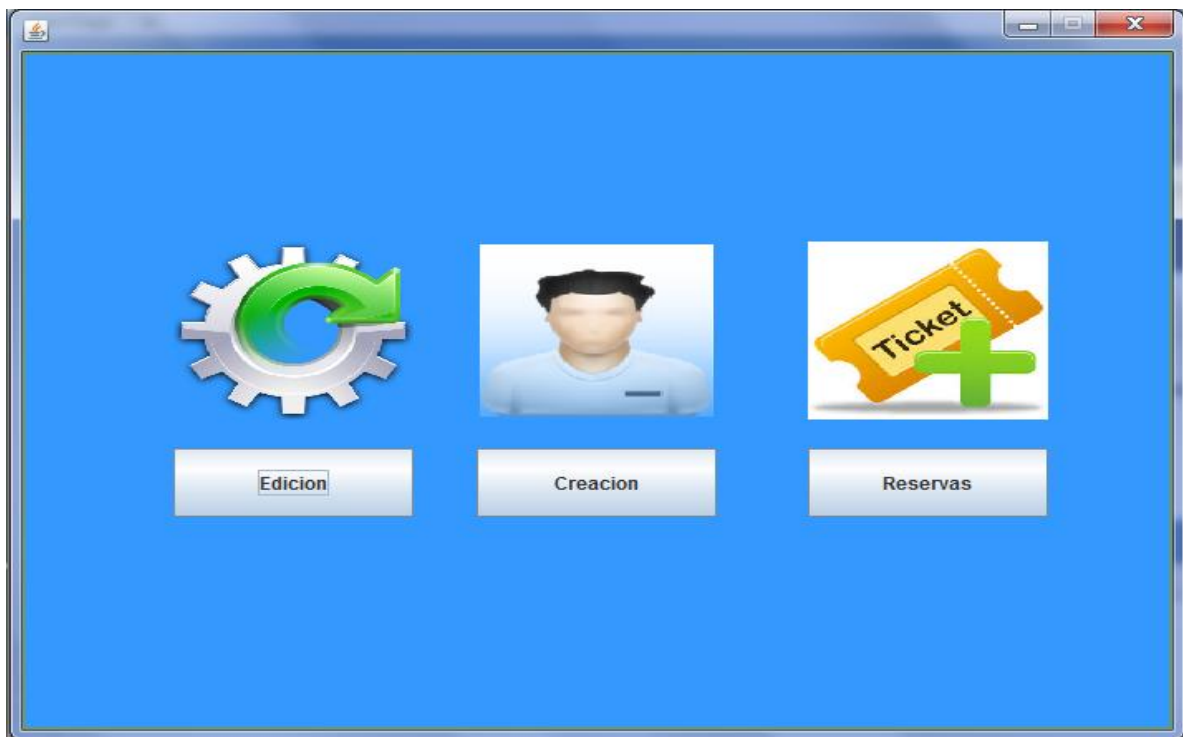
La Figura 19. Contiene todas las clases que permiten la reserva y búsqueda de reservas, también la clase que presenta la pantalla de reservas y consultas, además la clase TablaReserva que permite el manejo de las plazas en esta parte del sistema.

7. DESARROLLO DEL SISTEMA

Se desarrolló un sistema de reservas de espacios genérico y parametrizable el cual brinda acceso al usuario para hacer uso de las diferentes funciones que cumple el software implementado, el sistema se realizó por iteraciones en cada una de las pantallas mostradas a continuación.

Para comenzar con la presentación de la aplicación se tiene como primera pantalla la ventana de inicio del sistema, que presenta las opciones que tiene el usuario dentro del sistema, es importante mencionar que al ingresar por primera vez a la aplicación solo se podrá acceder a la ventana de “Creación”, porque para poder ingresar a “Edición” o “Reservas” es necesario tener un contenedor ya creado y guardado en la base de datos.

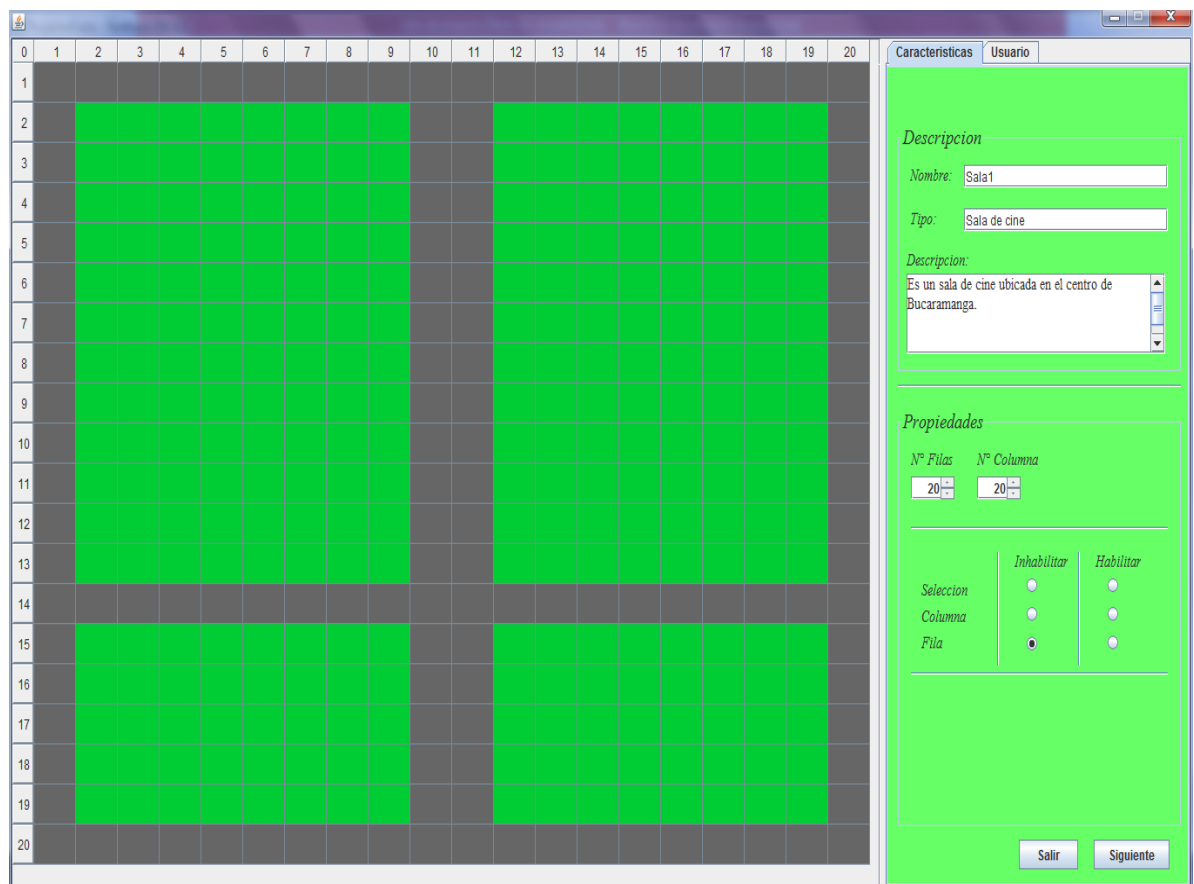
Figura 20. Pantalla Principal



A continuación se explicara más a fondo todas las opciones que tiene el sistema al momento de acceder a cada una de las opciones mostradas.

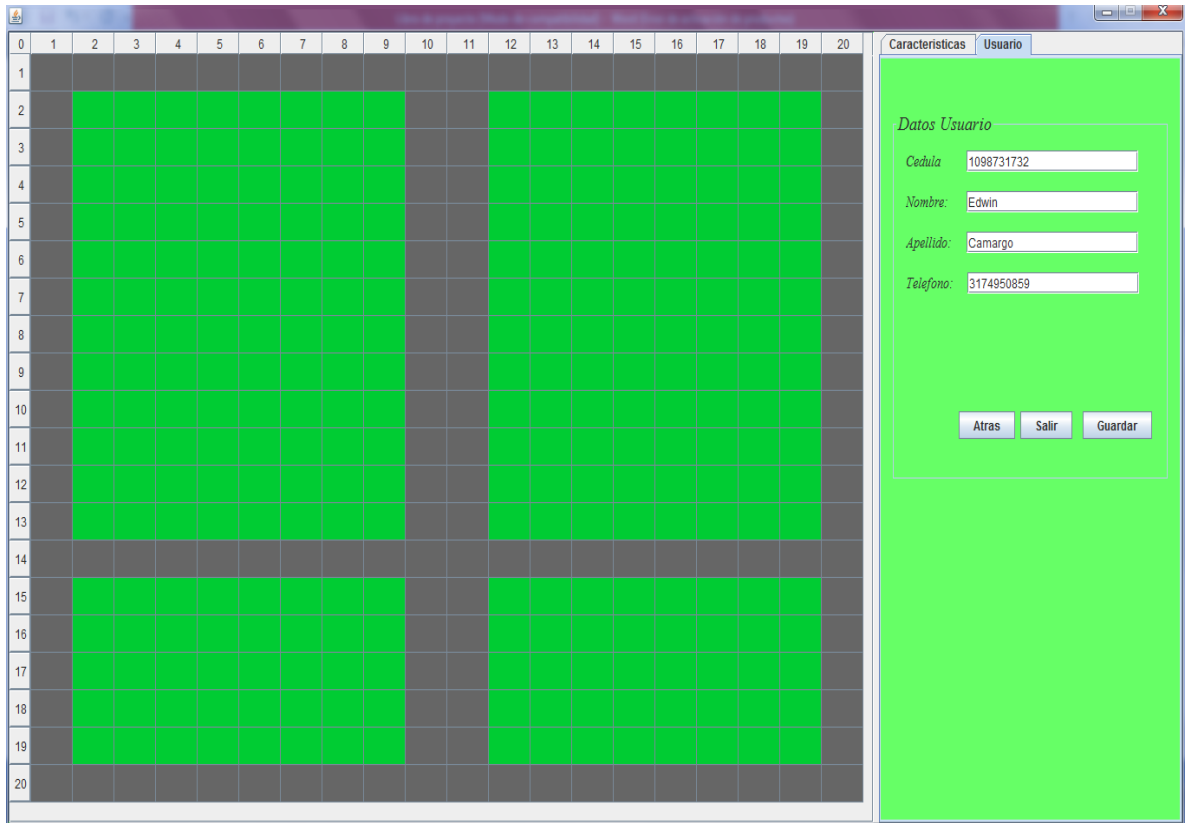
- **Crear Contenedor:** El usuario ingresa a la pantalla de creación donde podrá crear el contenedor que desee, ya sea una sala de cine, un parqueadero, un teatro, un auditorio, etc... ingresando todos los detalles que tiene ese contenedor, el número de plazas a crear y las plazas bloqueadas que significan que serán plazas que no podrán ser reservadas en el sistema.

Figura 21. Pantalla de creación del Contenedor.



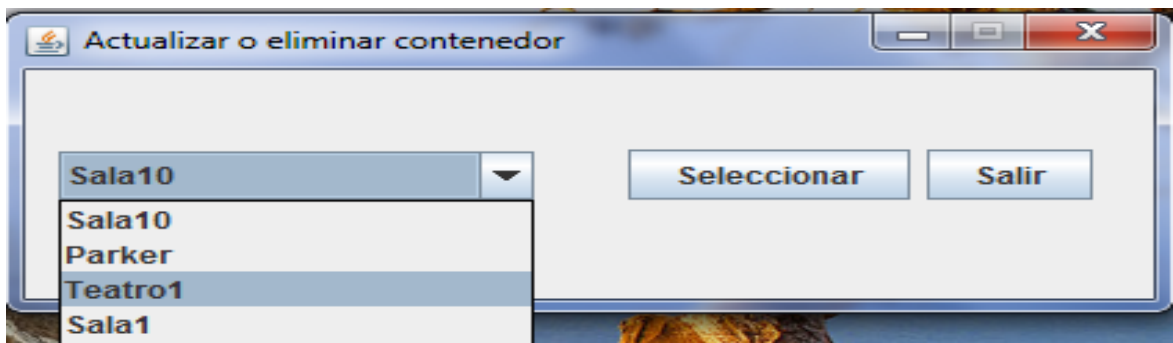
- **Guardar Usuario:** Guarda los datos del usuario quien crea el contenedor.

Figura 22. Guardar datos de Usuario



- **Seleccionar contenedor (Edición):** Seleccionar contenedor al cual quiere ingresar.

Figura 23. Pantalla para seleccionar Contenedor.



- **Actualizar Contenedor:** Actualiza contenedor con los detalles que quiere cambiar y las plazas a eliminar o agregar.

Figura 24. Pantalla de actualizar Contenedor

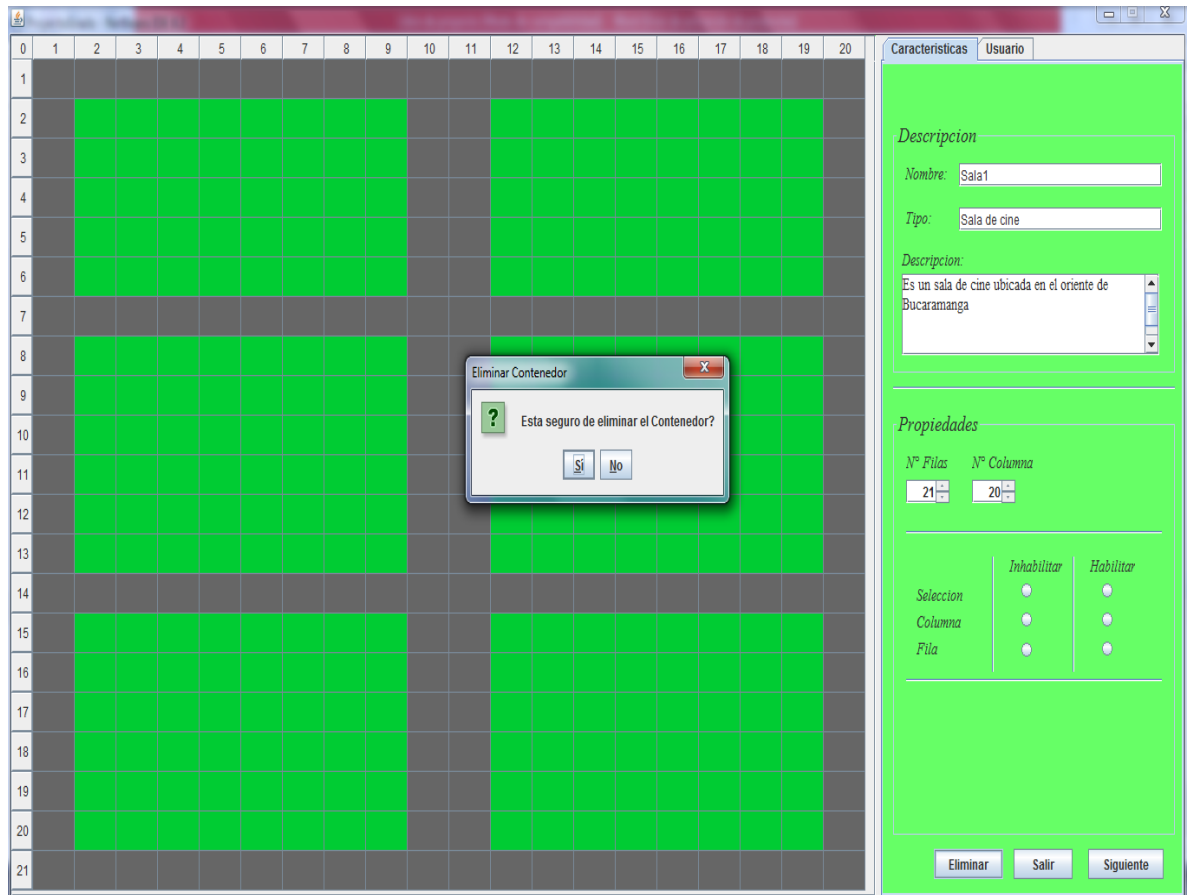
The screenshot shows a software interface for updating a container. On the left, there is a grid with columns labeled 0 through 20 and rows labeled 1 through 21. The grid cells are colored green or grey. On the right, there is a 'Datos Usuario' (User Data) form with the following fields:

- Cedula: 1098731732
- Nombre: Edwin
- Apellido: Camargo
- Telefono: 3174950859

Below the form are three buttons: 'Atras', 'Salir', and 'Actualizar'.

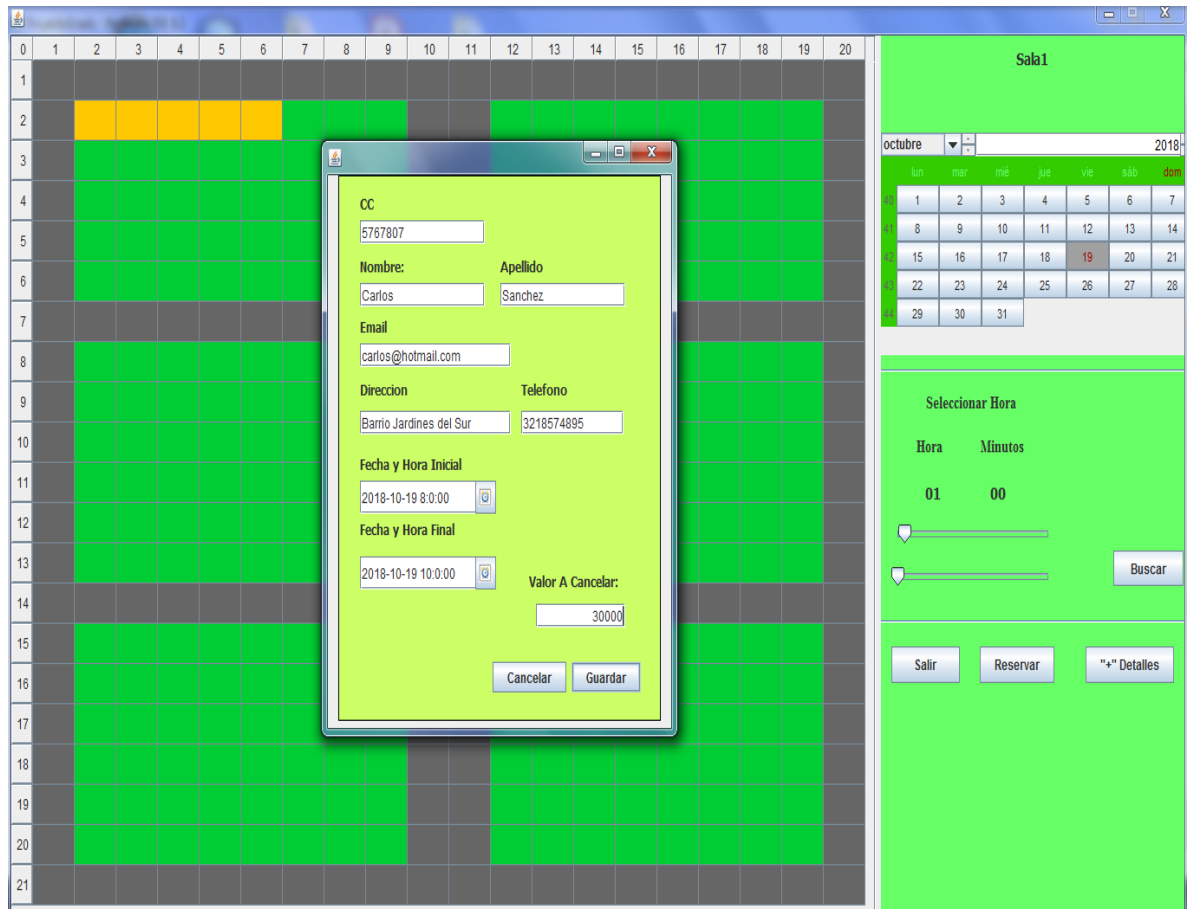
- **Eliminar Contenedor:** Elimina completamente el contenedor del sistema pero antes de eliminarlo le sale una pantalla con un mensaje de advertencia de si está seguro de seguir con esa acción.

Figura 25. Pantalla para eliminar Contenedor



- **Reservar plazas:** Se selecciona la plaza o plazas a reservar para que cuando se oprima en el botón "Reservar" despliegue un formulario para llenar con los datos del cliente, fecha y hora de la reserva.

Figura 26. Pantalla para reservar Plazas



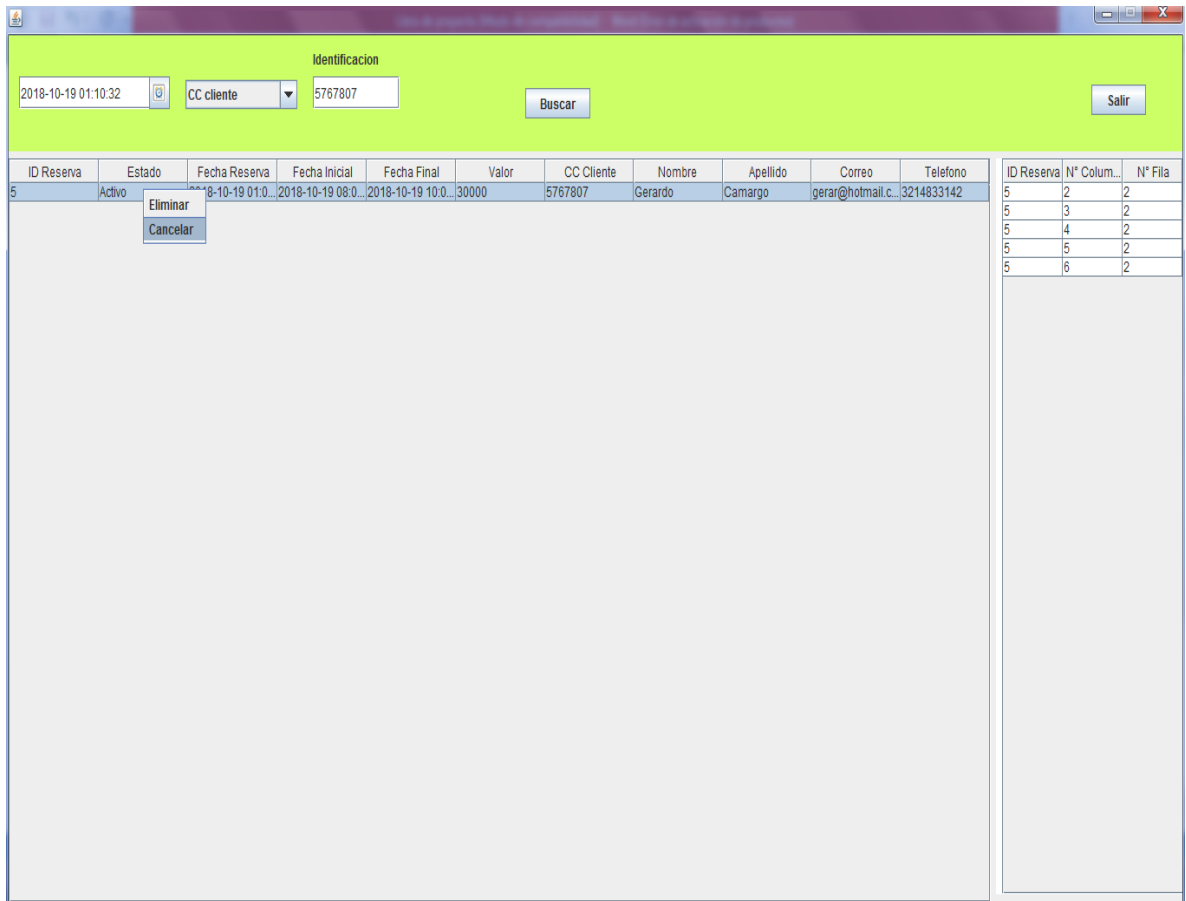
- **Consultar Reserva:** La reserva puede ser consultada de tres maneras diferentes en el sistema:
 1. CC cliente: Se ingresa el número de identificación del cliente para realizar la búsqueda.
 2. Fecha hora: Se ingresa la fecha y hora en la que está reservada la plaza o plazas y se realiza la búsqueda.
 3. Plaza: Se ingresa la columna, la fila, la fecha y la hora en que de la reserva y se realiza la búsqueda.

Figura 27. Pantalla para consultar las reservas de un Cliente.

ID Reserva	Estado	Fecha Inicial	Fecha Final	Valor	CC Cliente	Nombre	Apellido	Correo	Telefono	ID Reserva	N° Colum...	N° Fila
5	Activo	2018-10-19 01:0...	2018-10-19 08:0...	30000	5767807	Gerardo	Camargo	gerar@hotmail.c...	3214833142	5	2	2
										5	3	2
										5	4	2
										5	5	2
										5	6	2

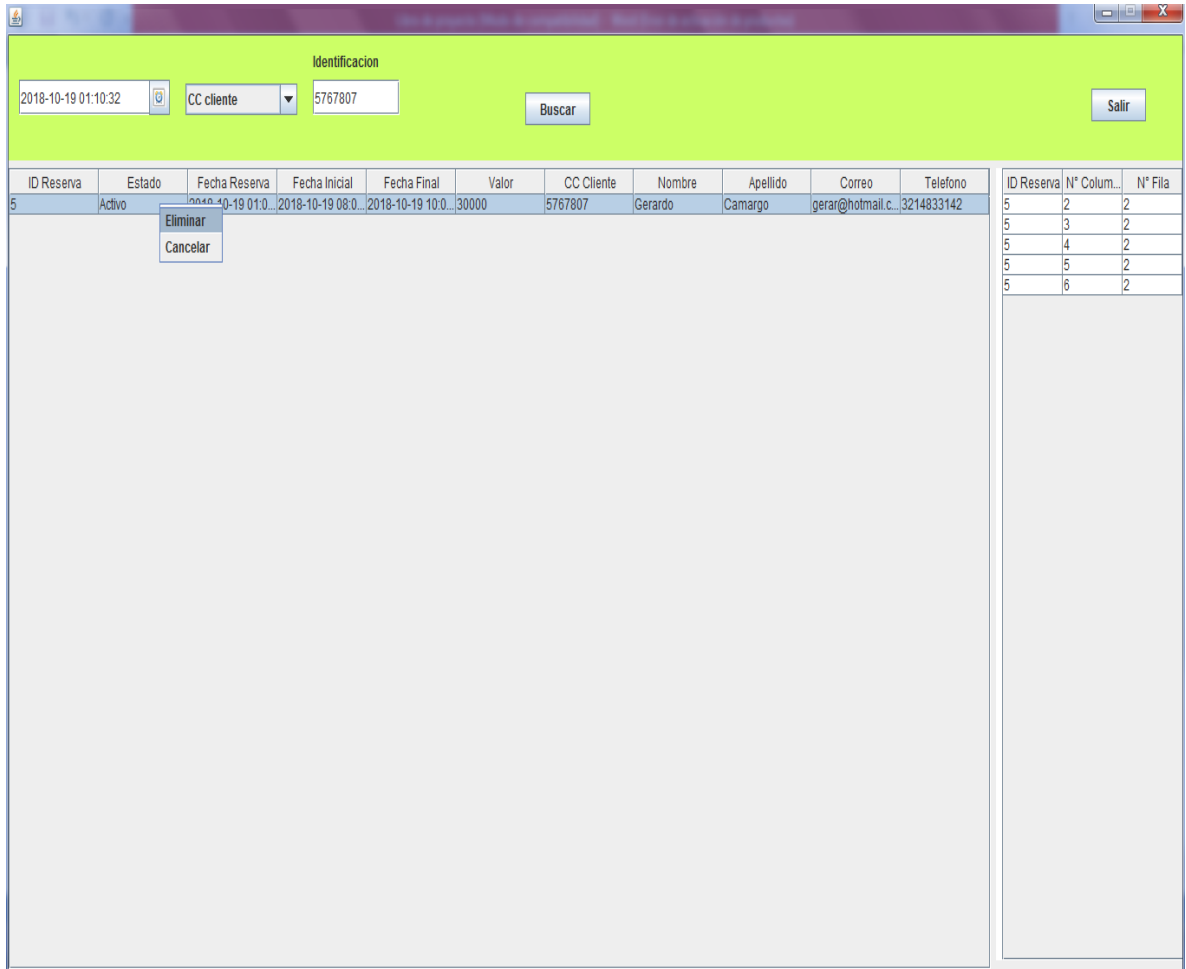
- **Cancelar Reserva:** Para cancelar una reserva solo se tiene que tener un registro de alguna reserva y sobre ese registro presionar clic izquierdo en el mouse para seleccionar la opción de cancelar reserva.

Figura 28. Pantalla para cancelar una Reserva realizada.



- **Eliminar Reserva:** Para eliminar una reserva solo se tiene que tener un registro de alguna reserva y sobre ese registro presionar clic izquierdo en el mouse para seleccionar la opción de eliminar reserva, la cual la borrara del sistema completamente.

Figura 29. Pantalla para eliminar una Reserva.



7.1. RESULTADO DE PRUEBAS DEL SISTEMA CON USUARIOS FINALES

Se realizaron una serie de pruebas finales al sistema de reservas de espacios genérico y parametrizable, para llevar a cabo las pruebas se tomó un grupo de personas de la comunidad UIS y otras de la comunidad donde resido para que interactuaran con la aplicación, dieran su punto de vista, y se verificara que se cumplía con las características planteadas en un comienzo del desarrollo del pro-

yecto, para esto se tomó el promedio entre las respuestas de todos los usuarios que iba entre una escala de uno a cinco (1-5) que se divide de esta manera:

1. Pésimo
2. Malo
3. Regular
4. Bueno
5. Excelente

Los usuarios respondieron según algunas características del sistema y que se guiaba por dos criterios importantes que son la funcionalidad y la usabilidad, los resultados de la prueba se pueden ver en la Tabla 18 que está a continuación.

Característica	Funcionalidad	Usabilidad
Crear Contenedor	4.8	4.7
Actualizar Contenedor	4.7	4.8
Eliminar contenedor	4.7	4.7
Realizar Reservas	4.8	4.8
Consultar Reservas	4.6	4.6
Cancelar Reservas	4.8	4.5
Eliminar Reservas	4.5	4.4
Resultado del Sistema en General	4.7	4.6

Tabla 18. Resultados de las pruebas con los usuarios finales

En la Tabla 18 podemos observar que el sistema tuvo una muy alta aceptación entre los usuarios finales, lo cual permite confirmar que se cumplieron con cada una de las características propuestas a desarrollar en este proyecto de grado.

8. CONCLUSIONES

- Se desarrolló e implementó un sistema de reservas de espacios genérico y parametrizable que cumple con los requisitos pedidos para dar solución a la problemática expuesta en el documento.
- Se utilizó tecnología Java y algunas herramientas como BlueJ y Netbeans IDE para el diseño y desarrollo del sistema respectivamente.
- Se investigó la manera más eficiente para crear los espacios en el sistema de una forma más rápida y sencilla para los usuarios.
- Se innovó al lograr desarrollar un sistema que pueda crear diferentes espacios para diferentes necesidades de los usuarios que utilizan el sistema, la innovación fue posible al poner en práctica los conocimientos aprendidos en el transcurso de la carrera.
- Se realizaron pruebas con datos reales para verificar el debido funcionamiento de la aplicación.
- Se realizaron pruebas con usuarios finales que permiten concluir que se obtuvo como resultado el proyecto deseado desde un comienzo.

9. RECOMENDACIONES

- Desarrollar nuevas versiones del sistema para añadir más funcionalidades a cada uno de los contenedores creados que permiten de este un software más robusto y de mayor utilidad para los usuarios que lo usen y los clientes que lo reserven.
- Pasar de un sistema de escritorio a un sistema web que permita ser utilizado en cualquier navegador web para la realización de las reservas por parte de los clientes.
- Se recomienda utilizar hibernate como framework, que facilite el mapeo de atributos entre la base de datos relacional y el modelo de objetos de la aplicación, conservando las características de la programación orientada a objetos.

BIBLIOGRAFÍA

BARNES, David J y KÖLLING, Michael. Programación orientada a objetos con java. Una introducción práctica usando BlueJ. 3ra Edición, PEARSON EDUCACION S.A. Madrid, 2007.

DEITEL, Harvey M y DEITEL, Paul J. Como programar en C/C++ y Java, 4ta Edición. Tomo II. PEARSON EDUCACION S.A.

PRESSMAN, Roger. Software Engineering A. Practitioner's Approach. 7ma Edición, McGraw-Hill Interamericana Editores S.A de C.V. Marzo 4 2010.

WEITZENFELD, Alfredo. Ingeniería de Software orientada a objetos con UML. JAVA E INTERNET.2005

Wikipedia. Bluej. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/BlueJ>

Wikipedia. Computer Reservation System. [En línea]. Disponible en: https://en.wikipedia.org/wiki/Computer_reservation_system

Wikipedia. Netbeans IDE. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/NetBeans>

Wikipedia. Máquina virtual de java. [En línea]. Disponible en: https://es.wikipedia.org/wiki/M%C3%A1quina_virtual_Java

Wikipedia. Modelo-vista-controlador [En línea]. Disponible en:
<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>
Wikipedia. MySQL. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/MySQL>

Wikipedia. PhpMyAdmin. [En línea]. Disponible en:
<https://es.wikipedia.org/wiki/PhpMyAdmin>