



Pontificia Universidad  
Católica del Ecuador

**PONTIFICIA UNIVERSIDAD CÁTOLICA DEL ECUADOR**

**FACULTAD DE INGENIERÍA**

**CARRERA DE SISTEMAS Y COMPUTACIÓN**

**TRABAJO DE TITULACIÓN**

**“DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA LA  
GESTIÓN DE PARQUEADEROS DE LA PONTIFICIA UNIVERSIDAD  
CATÓLICA DEL ECUADOR”**

**NOMBRE**

**BRYAN LENIN ANGULO CAMUENDO**

**DIRECTOR**

**EDISON JAVIER GUAÑA MOYA**

**QUITO, 2021**

## Dedicatoria

A Dios por la salud de cada día, por la sabiduría e inteligencia en cada decisión tomada.

A mis padres Holger Angulo y María del Carmen Camuendo, y mi hermano Matías Angulo, quienes han sido el pilar fundamental de apoyo y entrega hacia mí. Además, por ser mi motor para cumplir todas las metas propuestas a lo largo de mi vida, por su trabajo y sacrificio para que yo pueda llegar hasta aquí.

A mis tíos y primos quienes también han sido parte de mi formación tanto académica como personal a lo largo de esta etapa.

A mis amigos por el apoyo, consejos y porque junto a ellos este camino ha sido más ameno, dejando huellas imborrables en mi mente y corazón.

A todas estas personas y más que han hecho que este trabajo sea posible, a pesar de sus complicaciones siempre estuvieron para mí.

## Agradecimiento

Agradezco a los docentes de la facultad de Ingeniería por brindarme el conocimiento para cumplir con esta meta, de igual manera que han aportado a mi formación académica y profesional.

A la Dirección de Informática que fueron quienes siempre estuvieron abiertos a que este trabajo se realizara y brindaron todas las facilidades para que se concluya.

Sin más agradecer a todas las personas que forman parte de mi vida ya que son pilares fundamentales para mi desarrollo tanto profesional como personal.

## Tabla de contenido

Justificación.....	12
Planteamiento del Problema .....	12
Objetivo General .....	13
Objetivo Específico.....	13
Introducción.....	15
1    Capítulo I: Marco Teórico .....	17
1.1    Aplicación Web .....	17
1.2    Arquitectura de Programación.....	18
1.2.1    Modelo.....	19
1.2.2    Vista.....	19
1.2.3    Controlador.....	19
1.3    Metodología .....	20
1.3.1    Análisis de requerimientos.....	21
1.3.2    Diseño .....	22
1.3.3    Implementación .....	22
1.3.4    Pruebas .....	22
1.3.5    Retroalimentación.....	22
1.4    CodeIgniter.....	23
1.5    ExtJS.....	24
1.6    MySQL .....	24
2    Capítulo II: Análisis de Requerimientos.....	26
2.1    Introducción.....	26
2.1.1    Propósito .....	26
2.1.2    Alcance.....	26
2.2    Descripción general.....	27
2.2.1    Perspectiva del producto .....	27

2.2.2	Funcionalidad del producto.....	27
2.2.3	Restricciones .....	27
2.2.4	Suposiciones y dependencias .....	28
2.3	Requisitos específicos.....	28
2.3.1	Requisitos comunes de las interfaces .....	28
2.3.2	Requisitos funcionales.....	28
2.3.3	Requisitos no funcionales .....	53
2.4	Otros .....	54
2.4.1	Diagrama Conceptual del Sistema .....	54
2.4.2	Plan de pruebas del sistema .....	54
3	Capítulo III: Diseño.....	59
3.1	Introducción.....	59
3.2	Estructura del producto .....	59
3.2.1	Diagrama de actividades .....	59
3.2.2	Diagrama de clases .....	60
3.3	Diagramas de Interacción .....	61
3.3.1	Administración de usuarios del sistema.....	61
3.3.2	Administración de parqueaderos .....	65
3.3.3	Administración de periodos .....	67
3.3.4	Administración de usuarios.....	70
3.3.5	Administración de bloqueos.....	73
3.3.6	Administración de solicitudes .....	75
3.3.7	Sorteo .....	77
3.3.8	Notificación .....	78
3.3.9	Envío de comprobantes.....	79
3.3.10	Reportes.....	80
3.4	Diagrama Entidad Relación.....	81

3.5	Plan de pruebas de integración.....	81
4	Capítulo IV: Desarrollo .....	82
5	Capítulo V: Pruebas .....	89
6	Capítulo VI: Manual de usuario .....	94
6.1	Manual de usuario.....	94
6.1.1	Introducción .....	94
6.1.2	Módulos/Componentes.....	94
6.2	Manual técnico .....	108
6.2.1	Administración .....	108
6.2.2	Usuarios .....	108
6.2.3	Solicitudes .....	109
6.2.4	Consulta .....	109
6.2.5	Sistema.....	109
6.3	Conclusiones y Recomendaciones .....	110
6.3.1	Conclusiones .....	110
6.3.2	Recomendaciones .....	111
	Bibliografía .....	112

## Tabla de ilustraciones

Figura 1: Ciclo de vida MVC Fuente: (Espitia, Armao, & Carbajo, 2016) .....	20
Figura 2: Modelo Cascada Fuente: (Romero, 2018) .....	21
Figura 3: Casos de uso general del sistema .....	29
Figura 4: Casos de uso Siguiete nivel F1 .....	30
Figura 5: Casos de uso A detalle nivel F1.1 .....	30
Figura 6: Casos de uso A detalle nivel F1.2 .....	31
Figura 7: Casos de uso A detalle nivel F1.3 .....	32
Figura 8: Casos de uso A detalle nivel F1.4 .....	32
Figura 9: Casos de uso Siguiete nivel F2 .....	33
Figura 10: Casos de uso A detalle nivel F2.1 .....	34
Figura 11: Casos de uso A detalle nivel F2.2 .....	34
Figura 12: Casos de uso A detalle nivel F2.3 .....	35
Figura 13: Casos de uso A detalle nivel F2.4 .....	36
Figura 14: Casos de uso Siguiete nivel F3 .....	37
Figura 15: Casos de uso A detalle nivel F3.1 .....	37
Figura 16: Casos de uso A detalle nivel F3.2 .....	38
Figura 17: Casos de uso A detalle nivel F3.3 .....	39
Figura 18: Casos de uso A detalle nivel F3.4 .....	40
Figura 19: Casos de uso Siguiete nivel F4 .....	41
Figura 20: Casos de uso A detalle nivel F4.1 .....	41
Figura 21: Casos de uso A detalle nivel F4.2 .....	42
Figura 22: Casos de uso A detalle nivel F4.3 .....	43
Figura 23: Casos de uso Siguiete nivel F5 .....	44
Figura 24: Casos de uso A detalle nivel F5.1 .....	44
Figura 25: Casos de uso A detalle nivel F5.2 .....	45
Figura 26: Casos de uso A detalle nivel F5.3 .....	46
Figura 27: Casos de uso A detalle nivel F5.4 .....	46
Figura 28: Casos de uso Siguiete nivel F6 .....	47
Figura 29: Casos de uso A detalle nivel F6.1 .....	48
Figura 30: Casos de uso A detalle nivel F6.2 .....	48
Figura 31: Casos de uso A detalle nivel F6.3 .....	49
Figura 32: Casos de uso A detalle nivel F6.4 .....	50

Figura 33: Casos de uso Siguiete nivel F7 .....	51
Figura 34: Casos de uso Siguiete nivel F8 .....	51
Figura 35: Casos de uso Siguiete nivel F9 .....	52
Figura 36: Casos de uso Siguiete nivel F10 .....	53
Figura 37: Diagrama Conceptual.....	54
Figura 38: Diagrama de actividades.....	60
Figura 39: Diagrama de clases.....	61
Figura 40: F1.1 Ingreso y asignación de rol .....	62
Figura 41: F1.2 Modificación .....	63
Figura 42: F1.3 Desactivado .....	64
Figura 43: F1.4 Consulta .....	64
Figura 44: F2.1 Ingreso .....	65
Figura 45: F2.2 Modificación .....	66
Figura 46: F2.4 Eliminación.....	66
Figura 47: F2.4 Consulta .....	67
Figura 48: F3.1 Ingreso .....	68
Figura 49: F3.2 Modificación .....	69
Figura 50: F3.3 Desactivado .....	69
Figura 51: F3.4 Consulta .....	70
Figura 52: F4.1 Ingreso .....	71
Figura 53: F4.2 Modificación .....	72
Figura 54: F4.3 Consulta .....	72
Figura 55: F5.1 Ingreso .....	73
Figura 56: F5.2 Modificación .....	74
Figura 57: F5.3 Eliminación.....	74
Figura 58: F5.4 Consulta .....	75
Figura 59: F6.1 Ingreso .....	76
Figura 60: F6.2 Modificación .....	76
Figura 61: F6.3 Consulta .....	77
Figura 62: F6.4 Ingreso Intranet .....	77
Figura 63: F7 Sorteo .....	78
Figura 64: F8 Notificación.....	79
Figura 65: F9 Envío de comprobantes .....	80
Figura 66: F10 Reportes .....	80

Figura 67: Diagrama entidad relación .....	81
Figura 68: Login .....	82
Figura 69: Usuarios del sistema .....	83
Figura 70: Periodos .....	83
Figura 71: Disponibilidad .....	84
Figura 72: Ingreso usuarios .....	84
Figura 73: Modificación usuarios .....	85
Figura 74: Ingreso solicitudes .....	85
Figura 75: modificación y consulta .....	86
Figura 76: Bloqueos .....	86
Figura 77: Sorteo .....	87
Figura 78: Notificaciones .....	87
Figura 79: Envío de comprobantes .....	87
Figura 80: Reportes .....	88
Figura 81 Login .....	89
Figura 82 Crear Usuario del Sistema .....	89
Figura 83 Creación Periodo .....	90
Figura 84 Creación Parqueadero .....	90
Figura 85 Creación Usuario .....	90
Figura 86 Creación Solicitud .....	91
Figura 87 Sorteo Exitoso .....	91
Figura 88 Correo .....	91
Figura 89 Envío de comprobantes .....	92
Figura 90: Pruebas .....	93
Figura 91: Login .....	94
Figura 92: Error login .....	94
Figura 93: Página principal .....	95
Figura 94: Submenú .....	95
Figura 95: Ventana .....	96
Figura 96: Crear usuario del sistema .....	97
Figura 97: Error ingreso usuario sistema .....	97
Figura 98: Modificar roles .....	98
Figura 99: Modificar usuario del sistema .....	98
Figura 100: Disponibilidad de parqueaderos .....	99

Figura 101: Crear un período .....	99
Figura 102: Modificar un período.....	100
Figura 103: Bloqueo .....	100
Figura 104: Quitar bloqueo.....	101
Figura 105: Ingresar usuario .....	101
Figura 106: Modificar usuario .....	102
Figura 107: Crear solicitud .....	103
Figura 108: Error crear solicitud .....	103
Figura 109: Modificar solicitud.....	104
Figura 110: Eliminar solicitud .....	104
Figura 111: Error eliminar solicitud.....	104
Figura 112: Consultar vehículo.....	105
Figura 113: Reportes solicitud.....	106
Figura 114: Reportes usuario .....	106
Figura 115: Sorteo.....	107
Figura 116: Correos .....	107
Figura 117: Comprobantes.....	108

## Índice de tablas

Tabla 1: Tarifas .....	15
Tabla 2: Casos de prueba F1 .....	55
Tabla 3: Casos de prueba F2 .....	55
Tabla 4: Casos de prueba F3 .....	56
Tabla 5: Casos de prueba F4 .....	56
Tabla 6: Casos de prueba F5 .....	56
Tabla 7: Casos de prueba F6 .....	57
Tabla 8: Casos de prueba F7 .....	57
Tabla 9: Casos de prueba F8 .....	58
Tabla 10: Casos de prueba F9 .....	58
Tabla 11: Casos de prueba F10 .....	58

## Justificación

En la última década el parque automotor en Ecuador creció en más 1.4 millones de vehículos lo que incrementa a más de 2,4 millones de unidades (INEC, 2020), de modo que se encuentra en constante crecimiento, por lo que una gran mayoría de familias constan con vehículos propios.

En Quito aproximadamente circulan 455000 vehículos (ANT, 2020), los cuales salen a diario por lo que es usual encontrarnos con sistemas automatizados de parqueaderos para solventar la gran demanda de vehículos.

La Pontificia Universidad Católica del Ecuador dispone de distintos parqueaderos los cuales son parte de las instalaciones que brinda a la comunidad universitaria. Consta con 3 parqueaderos físicos los cuales dos se encuentran dentro de la universidad y otro fuera.

El presente proyecto se enfoca en gestionar de las necesidades que se presentan a nivel de universidad como lo es brindar zonas de parqueo, que favorezcan las actividades laborales ejecutadas por la comunidad universitaria y simplificando el proceso que se realiza en la actualidad.

Lo cual es conveniente ya que así se puede tener la información organizada y centralizada, implementando el sistema bajo un modelo de procesos de desarrollo llamado cascada el cual permite dividir procesos de forma lineal en sucesivas fases (Alvear & Mazón , 2007), por lo que es altamente aplicable.

## Planteamiento del Problema

De las zonas de parqueo de la PUCE, existen diferentes tipos a los cuales se puede acceder ya sea como estudiantes o como trabajadores de la universidad los cuales son: Abierto, Girón, Semipresencial, Noche y Torres. Los cuales comprenden diferentes áreas y distintos horarios en los cuales se los puede hacer uso.

Sin embargo, cerca de 300 parqueaderos son destinados a los alumnos y en promedio se hacen 500 solicitudes (Dirección de Informática, 2020), por semestre de modo que necesita de procesos previos para la distribución de los mismos.

De acuerdo con la Dirección de Informática constan con un sistema para realizar el sorteo, sin embargo, se desea elaborar otro ya que es de difícil mantenimiento y adaptación a nuevos cambios porque implica de costos adicionales y es necesario mantener una licencia, por lo que se prefiere unificar el desarrollo de software, no cumple con todas las necesidades que presenta todo el proceso de los parqueaderos, no se tiene toda la información centralizada y para realizar cambios es necesario llamar a los técnicos que trabajan con este software los cuales son de una empresa ajena a la universidad, lo que complica el proceso de modificación de la misma y no se está aprovechando de las instalaciones y departamentos propios de la universidad.

De esto se puede identificar el siguiente problema principal:

- No se cuenta con un sistema que abarque todas las necesidades que la gestión de parqueaderos demanda para la Pontificia Universidad Católica del Ecuador en su sede Quito.

Y los siguientes problemas secundarios:

- No se ha realizado un análisis de la situación actual del manejo de parqueaderos en la Pontificia Universidad Católica del Ecuador.
- No se han diseñado los procesos necesarios para la automatización de la gestión de parqueaderos.
- No se ha desarrollado una aplicación web que permita soportar los procesos automatizados requeridos.
- No se ha realizado una implementación que se adapte a futuros requerimientos de gestión de parqueaderos.

## Objetivo General

- Realizar la implementación de un sistema web para la gestión de parqueaderos en la Pontificia Universidad Católica del Ecuador.

## Objetivo Específico

- Analizar la situación actual del manejo de parqueaderos para abarcar todas las implicaciones que comprende esta gestión.

- Diseñar los procesos necesarios para la automatización de la gestión de parqueaderos.
- Desarrollar una aplicación web que permita soportar los procesos de automatización requeridos y se adapte a futuros requerimientos de gestión de parqueaderos dentro de la universidad.

## Introducción

La carrera de Ingeniería en Sistemas y Computación tiene como objetivo fundamental resolver con técnicas computacionales problemas de la vida real, del mismo modo este proyecto se enfoca en automatizar ciertos procesos en la Pontificia Universidad Católica del Ecuador.

La universidad dispone de distintos parqueaderos, los cuales son alrededor de 400 aparcamientos que brinda a toda su comunidad universitaria (Almeida, 2020), y tienen el fin de facilitar el servicio de parqueo de vehículos durante los periodos de estudio y trabajo.

El campus consta con 3 parqueaderos físicos, dos se encuentran dentro de la universidad y el otro fuera, sin embargo, existen diferentes tipos de parqueaderos a los cuales se puede acceder como estudiantes o como trabajadores de la universidad y con diferentes tarifas (Ver Tabla 1).

*Tabla 1: Tarifas*

<b>Nombre</b>	<b>Descripción</b>	<b>Tarifa</b>
Abierto	Se encuentra dentro del campus universitario.	\$150
Girón	Está situado fuera del campus, junto a la funeraria Girón.	\$100
Semipresencial	Se destina específicamente para los estudiantes que estudian bajo la modalidad semipresencial.	\$60
Noche	Tiene acceso al parqueadero dentro del campus a partir de las 6 de la tarde.	\$60
Torres	Es únicamente para los trabajadores de la universidad.	No tiene tarifa.

Los parqueaderos destinados a los estudiantes de grado son 310, entre ellos en el parqueadero denominado Abierto existen 160, Girón 90, Semipresencial 30 y Noche 30. En promedio se hacen 500 solicitudes por semestre (Dirección de Informática, 2020), de modo que necesita de procesos previos para la distribución de los aparcamientos.

Por otro lado, los estudiantes de postgrado, personal administrativos y docentes poseen de este servicio sin necesidad de un sorteo ya que son un grupo que considerar al momento de la distribución de los estacionamientos.

# 1 Capítulo I: Marco Teórico

## 1.1 Aplicación Web

Los sistemas web con el paso de los años se han convertido cada vez más complejos, dando un apartado a la capacidad para sustituir a una gran variedad de aplicativos de escritorio (Sotomayor, 2017), es una tecnología que permiten acceder a una aplicación por medio del navegador, de modo que es más portable en comparación a la mencionada anteriormente, debido a que no se debe instalar el aplicativo en cada máquina, permitiendo ingresar desde cualquier dispositivo con conexión a internet y sin importar el sistema operativo con el que cuente el equipo.

Entre sus principales características se encuentra la facilidad de actualizar el sistema, sin necesidad de realizar modificaciones en los equipos que utiliza el usuario, ya que una vez que el servidor cuente con una nueva versión esta se ve reflejada en todos los dispositivos que se encuentran conectados al aplicativo web.

La interacción del usuario la hace directamente con el navegador, en consecuencia, de las acciones realizadas se envían peticiones al servidor donde se encuentra alojado el aplicativo (Castejón, 2004), mismo que se conecta con una base de datos que contiene la información pertinente. Una vez la petición sea ejecutada esta devuelve una respuesta que es presentada en el navegador para el uso y manejo del usuario.

HTTP (Hypertext Transfer Protocol), es el protocolo encargado de establecer la conexión y transferir recursos a través de internet, mismo que va de la mano con el protocolo de comunicaciones llamado TCP (Transport Control Protocol), el cual es el delegado de establecer un canal de comunicación de extremo a extremo es decir entre el cliente y el servidor.

También HTTP define como codificar el paso de parámetros entre páginas, el tunelizar las conexiones para sistemas de firewall, la existencia de servidores intermedios de cache, entre otros (Mateu, 2004). Además, existe una variación la cual es HTTPS que aumenta otra funcionalidad que es la seguridad bajo otro protocolo SSL (Secure Socket Layer), que se encarga de cifrar la información

enviada o recibida entre el usuario y el servidor. Dicho protocolo debe ser usado por el servidor para garantizar la confidencialidad de los datos.

Existen diferentes directivas de petición que son métodos para indicar acciones a realizar, estas son enviadas desde el cliente y receptadas por el servidor, también son conocidos como HTTP verbs. Cada una de las directivas de petición tiene un objetivo y uso distinto como se puede ver a continuación:

- GET, utilizada para recuperar datos en forma de un recurso específico o una entidad.
- POST, usado para enviar una entidad a un recurso específico, lo cual permite un cambio en el servidor.
- HEAD, es similar a una petición GET, pero con la diferencia de que no devuelve un cuerpo de respuesta.
- PUT, reemplaza la representación del elemento de destino.
- DELETE, es solicitado para pedirle al servidor que elimine un elemento determinado.
- TRACE, permite realizar pruebas de retorno, ayuda a la depuración y desarrollo.
- OPTIONS, describe las opciones de comunicación para el recurso de destino.
- CONNECT, establece un canal de comunicaciones para el recurso de destino.

## 1.2 Arquitectura de Programación

Modelo Vista Controlador es un patrón de arquitectura de software, este surge al necesitar de software más robusto que se adapte mejor a un ciclo de vida (Alvarez, DesarrolloWeb, 2020), es decir que sea de fácil mantenimiento, permita la reutilización de código y se pueda segmentar los distintos conceptos que intervienen, dando así al producto una mayor calidad al momento de realizar la construcción del aplicativo.

Se caracteriza por separar la lógica del negocio de las interfaces de usuario y de la información, por lo que se puede dividir a la aplicación en tres niveles siendo estos segmentados por el concepto al que están originados.

### 1.2.1 Modelo

Es aquel que trabaja directamente con los datos por lo que mediante diferentes mecanismos obtiene información o realiza cambios en la data. La información se encuentra en bases de datos por lo que es importante realizar una conexión previa para que los datos sean manejados sin complicaciones.

### 1.2.2 Vista

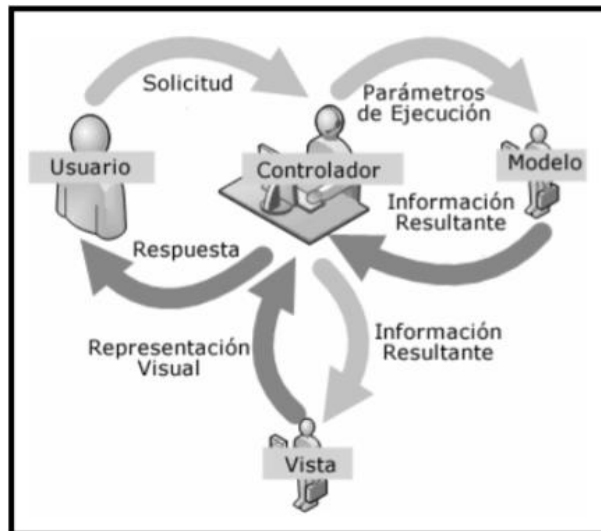
Contienen el código de la aplicación con el que interactúa directamente con el usuario (Alvarez, DesarrolloWeb, 2020), es la representación gráfica del modelo. En las aplicaciones web son páginas HTML que es desplegada por el navegador, el objetivo de esta es presentar de manera adecuada los datos y así satisfacer las necesidades del usuario.

### 1.2.3 Controlador

Es una capa que sirve como enlace entre las vistas y el modelo, es aquí donde se representan la lógica de los procesos reales para la automatización de estos. Se encarga del manejo de las solicitudes del usuario y el procesamiento de información (Espitia, Armao, & Carbajo, 2016), los cuales son enviados al modelo, de modo que no sea posible el acceso desde la vista al modelo directamente.

Existe un ciclo de vida del Modelo Vista Controlador (MVC), el cual es representado por las capas que conforman el mismo y el usuario (Ver Figura 1) como se describe a continuación:

- El usuario realiza la solicitud al controlador, donde contiene la información necesaria para que esta se realice de manera correcta.
- El controlador acepta la petición.
- El controlador decide a quien asignar esta tarea para lo cual hace la llamada al modelo.
- El modelo interactúa con la base de datos ya sea modificando u obteniendo información, de modo que responde a la llamada realizada por el controlador.
- El controlador recibe dicha información y es enviada a la vista.
- La vista procesa los datos y es presentado de una manera adecuada al usuario en respuesta a las acciones realizadas.



*Figura 1: Ciclo de vida MVC*  
 Fuente: (Espitia, Armao, & Carbajo, 2016)

Aunque originalmente fue desarrollado para aplicaciones de escritorio, actualmente ha sido adaptado de manera que se pueda implementar en aplicaciones web (Espitia, Armao, & Carbajo, 2016). Existe una gran cantidad de frameworks que utilizan este patrón, de manera que interpretan las relaciones que existen entre las diferentes capas.

### 1.3 Metodología

Las metodologías de software son marcos en los que se realiza el proceso de desarrollo de un sistema, de modo que se construya un sistema informático eficiente. Es usado para organizar, planificar y controlar la sucesión de tareas a realizar (Dominguez, 2020), lo que provee un enfoque metodológico que ordena estrictamente las fases a seguir.

El modelo Cascada como marco metodológico es una secuencia de procesos a seguir, específicamente en desarrollo de software, con etapas que se ejecutan una al cabo de terminar la anterior (Enríquez, y otros, 2017). Se denomina cascada por la manera en que están estructuradas de modo que, al estar de arriba hacia abajo las distintas etapas, por consecuencia crean un flujo vertical.

Este modelo nació en la industria y la construcción ya que se determinó que las modificaciones posteriores representan un gasto significativo. En el mundo del software se usaban otras metodologías (Dominguez, 2020), pero ocurría el mismo problema, por lo que se decidió adoptarla, generando así una

metodología para desarrollo de software que permitiera una buena estructura y marco para el desarrollo de aplicaciones. Existen diferentes fases que componen la metodología como se puede ver en la Figura 2.

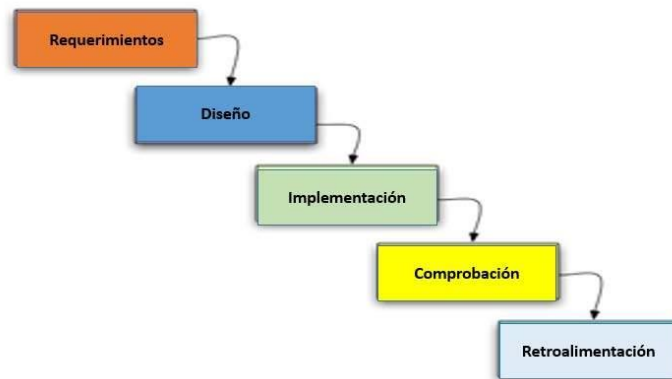


Figura 2: Modelo Cascada  
Fuente: (Romero, 2018)

### 1.3.1 Análisis de requerimientos

El propósito principal de esta etapa se centra en entender el problema que se va a resolver, creando un modelo conceptual o específico de los requerimientos, el cual servirá de base para todo el desarrollo del proyecto.

En esta fase se realiza un documento de especificación de requerimientos el cual tiene un doble propósito (De La Cruz, 2020), el primero abstraer las necesidades y expectativas del cliente y el segundo que sirva de base para las siguientes etapas. Por lo que el documento debe ser entendible, preciso y completo. Este consta de dos partes principales, las cuales son:

- Los requerimientos funcionales donde se especifica las funciones que el sistema va a realizar, es decir el comportamiento que este tendrá bajo las diferentes entradas y como respuesta las salidas obtenidas.
- Los requerimientos no funcionales son aspectos que no tienen una relación directa con el comportamiento del sistema (Quirola, 2020), las cuales son confiabilidad, desempeño, características de interfaces, restricciones operativas, restricciones físicas, portabilidad, entre otros.

De igual forma se realiza el plan de pruebas del sistema ya que en la fase de verificación se lo hace contra los requerimientos del proyecto.

### 1.3.2 Diseño

En esta etapa se describe la solución computacional, la cual será construida en dos niveles:

- La arquitectura que plantea la organización del software, la división en subsistemas y sus relaciones.
- La detallada que describe para cada módulo su objetivo, los algoritmos, estructura de datos y las características de la interfaz (De La Cruz, 2020).

En otras palabras, se establecen las políticas comunes que se usaran en la construcción del software.

Se realiza el plan de pruebas de integración, donde se probará que los módulos reaccionen de la manera adecuada en el funcionamiento en conjunto.

### 1.3.3 Implementación

El propósito de esta fase es desarrollar cada uno de los módulos propuestos a realizar, haciendo uso de las estructuras de datos diseñadas en la fase anterior (Dominguez, 2020). Programar es la implementación de algoritmos de modo que el sistema se comporte de la manera esperada.

El código escrito debe estar bajo estándares que maneje la compañía que incluyen: encabezados por módulo, convenciones de variables y subprogramas, máximo de líneas por componente, entre otros.

### 1.3.4 Pruebas

Se hace uso de los planes de prueba generados anteriormente, de modo que los componentes deben ser probados individual y en conjunto para verificar que se comporten de la manera esperada.

El objetivo de esta fase es obtener información sobre la calidad del software que se ha realizado, una vez termina esta fase el producto está listo para ser instalado y puesto en marcha.

### 1.3.5 Retroalimentación

Una vez instalado se debe comprobar que en el entorno de producción este funcione correctamente, se efectúan actividades después de que el software es

liberado al cliente (De La Cruz, 2020), lo que viene a ser mejoras, cambios o agregar características a la aplicación.

#### 1.4 CodeIgniter

Es un framework para el desarrollo de aplicaciones web que permite un rápido desarrollo a diferencia de construir desde cero, proporcionando un amplio catálogo de bibliotecas (ITCB, 2017), usa PHP como lenguaje de programación y además es de código libre para cualquier aplicación.

Este fue creado por la compañía norteamericana EllisLab, quien realizó la publicación de la primera versión en febrero de 2006, varios años después específicamente en el 2013 la compañía paso por un mal momento y no tenía recursos para continuar con el proyecto, el cual fue beneficiado por su compra por el British Columbian Institute of Technology (BCIT).

CodeIgniter propone una manera de implementar, la cual se debe seguir para darle el mejor uso al framework, marca una forma específica de desarrollar y de clasificar las diferentes estructuras de programación (Alvarez, DesarrolloWeb, 2009), de modo que se consigue un código bien organizado, el cual sea más fácil de realizar modificaciones o adaptaciones en un futuro.

La estructura con la que se maneja es Modelo Vista Controlador (MVC), es un estándar en la programación de aplicaciones, donde se recalca la estricta separación entre el código y las vistas.

Es capaz de trabajar en la mayoría de los entornos y servidores, hasta en sistemas de alojamiento compartido, en el cual hay que enviar los archivos bajo el protocolo File Transfer Protocol (FTP) al servidor ya que no hay acceso a la configuración y es necesaria solo la modificación de un archivo. Es compatible con PHP 4, lo que hace posible usar en servidores más antiguos y de igual manera con PHP 5 (Juanta Andalucía, 2020), de modo que es compatible con la mayoría de los servidores.

CodeIgniter posee una gestión de autocarga la cual proporciona la inicialización de librerías, asistentes y complementarios de forma automática, durante el arranque del sistema (Juanta Andalucía, 2020), lo que lo vuelve más eficaz ya que no necesita de la configuración previa, por otra parte, el manejo de errores

permite crear reporte de problemas en las aplicaciones, además permite guardar los mensajes de error en logs que son archivos de texto.

Para finalizar existen abundante documentación lo que permite realizar consultas y demás, de modo que se puede encontrar soluciones fácilmente.

### 1.5 ExtJS

Es un marco de JavaScript para el desarrollo de aplicaciones web y tiene características que se alinean al manual de estándares denominado ADA (Sencha INC, 2020), los mismos que define el lenguaje mediante la ISO-8652: 1995.

Se origino como una extensión de la biblioteca YUI y esta se pone en el mercado el 1 de abril del 2007, sus creadores fueron: Jack Slocum, Brian Moeskau, Aaron Conran y Rich Waters.

Entre sus principales características se puede realizar aplicaciones complejas utilizando componentes predefinidos. Es totalmente portable (Marin, 2019), ya que es compatible con todos los navegadores, a diferencia de otros marcos de desarrollo con los cuales no están totalmente optimizados, por lo que se indica que la aplicación es mejor si se la usa en alguno en específico.

Con el aplicativo Secha Architect permite realizar vistas en HTML5 de una manera más fácil ya que permite realizar las interfaces de forma visual, es decir arrastrar y soltar componentes ya precodificados (Sencha INC, 2020), lo que produce un beneficio al momento de programar, ya que el tiempo que toma realizar una interfaz es menor al realizarlo de esta manera.

La generación automática de código acelera el desarrollo de aplicaciones, con modificar las propiedades y configuraciones de los componentes elimina el error que las personas pueden cometer al realizar de manera manual (Sencha INC, 2020), volviéndolo más intuitivo y con una fácil depuración de errores al momento de la programación.

### 1.6 MySQL

Es un sistema de administración de bases de datos relacionales rápido, sólido y flexible (Pérez & Cobo, 2005). Entre sus principales características es de código abierto lo que permite consulta y modificar la información de su código fuente.

MySQL fue desarrollado por una compañía sueca que se llamaba MySQLAB en 1995, sus creadores fueron: Michael Widenius, David Axmark y Allan Larson. Posteriormente sería comprada por Sun Microsystems y luego por Oracle Corporation en 2010.

Se encuentra desarrollado en C y C++, consta con un licenciamiento GPL lo que garantiza que es software libre, también existe la licencia comercial que es para quien no desea que sus desarrollos sean de código abierto.

Utiliza el lenguaje SQL (Structured Query Language), el cual es el lenguaje más común en bases de datos relacionadas. Permite utilizar mediante líneas de comando sin necesidad de la instalación de un entorno gráfico. Además, existe una gran cantidad de soporte y software realizado sobre MySQL por lo que es de fácil acceso a información o ayudas de solución de errores.

Existe una gran cantidad de empresas que utilizan a MySQL como motor de base de datos, como lo son: Yahoo!, Google, CISCO, Motorola, Silicon Graphics, entre otros (Pérez & Cobo, 2005).

MySQL tiene una gran compatibilidad con PHP por lo que el trabajo conjuntamente de estas herramientas es de las más comunes (Dominguez, 2020), de igual manera es portable ya que está disponible en más de veinte plataformas como lo son las distribuciones más usadas: Mac, Unix, Windows.

Todas estas características han hecho de MySQL uno de los gestores de bases de datos más utilizados por desarrolladores lo que le ha dado la robustez necesaria para ponerse en competencia con otros gestores.

## 2 Capítulo II: Análisis de Requerimientos

### 2.1 Introducción

La presente Especificación de Requerimientos de Software (SRS) del sistema a construir surge de un conjunto de información con la cual ayuda a entender los requisitos que el cliente desea, de esta forma conocer lo que realmente se quiere obtener como producto final, teniendo un documento que sirva como base para el desarrollo del software, mismo que contendrá en forma detallada las interfaces de usuario y del hardware, así como de los requerimientos del cliente, atributos del sistema, entre otros.

#### 2.1.1 Propósito

Se describe el funcionamiento general del proyecto acerca de la gestión de parqueaderos en la Pontificia Universidad Católica del Ecuador, definiendo las especificaciones funcionales y no funcionales del sistema para su implementación, que será utilizado por el personal a cargo de la gestión de estos procesos.

#### 2.1.2 Alcance

Dicho proyecto será una aplicación la cual funcionará en un entorno web, que permitirá administrar y consultar información referente al proceso que cumple, con su documentación basada en la metodología aplicada lo que comprende los siguientes módulos:

- Gestión de usuarios del sistema
- Gestión de parqueaderos
- Gestión de periodos (semestres)
- Gestión de bloqueos
- Gestión de usuarios (comunidad universitaria)
- Gestión de solicitudes
- Sorteo
- Notificación (favorecidos en el sorteo)
- Envío de comprobantes (envío de información para posterior generación de comprobantes de pago)
- Reportes

## 2.2 Descripción general

Se presenta las funcionalidades que van a desarrollarse, también las características de los usuarios que van a manejar el sistema.

Existen dependencias entre las funcionalidades, por lo que, es importante establecer un orden ya para al momento de la implementación ciertos procesos dependen de la implementación de otros.

### 2.2.1 Perspectiva del producto

El producto se desarrollará en un solo ciclo por lo que al terminar el mismo se presentará el aplicativo final.

### 2.2.2 Funcionalidad del producto

Se hace referencia a la administración de una funcionalidad, en otras palabras, el manejo o gestión del mismo, lo que comprende el ingreso, modificación, eliminación o desactivación y consulta de una funcionalidad dentro del sistema. Sin embargo, existen funciones del proceso las cuales no serán nombradas administración ya que comprenden funcionalidades complementarias. A continuación, las funcionalidades que serán realizadas:

- F0. Login.
- F1. Administración de usuarios del sistema.
- F2. Administración de parqueaderos.
- F3. Administración de periodos.
- F4. Administración de usuarios.
- F5. Administración de bloqueos.
- F6. Administración de solicitudes.
- F7. Sorteo.
- F8. Notificación.
- F9. Envío de comprobantes.
- F10. Reportes.

### 2.2.3 Restricciones

El sistema será desarrollado en base a la arquitectura de modelo vista controlador, haciendo uso del lenguaje ExtJs para el frontend con un entorno de

desarrollo llamado Sencha Architect, bajo el framework CodeIgniter para el backend y base de datos MySQL.

#### 2.2.4 Suposiciones y dependencias

Existen varias dependencias al momento de implementar las funcionalidades entre las cuales se pueden detectar las siguientes:

- Bloqueos depende de Usuarios.
- Solicitudes depende de Parquero, Periodo y Usuarios.
- Sorteo depende de Solicitudes.
- Notificación depende de Solicitudes.
- Envío de comprobantes depende de Solicitudes.

### 2.3 Requisitos específicos

Se detalla los requerimientos específicos que servirán como guía para el desarrollo, se habla de las interfaces necesarias para el funcionamiento adecuado del sistema. Los requisitos funcionales que es la descripción de lo que permite realizar el sistema y los no funcionales que son el comportamiento esperado en seguridad, fiabilidad, disponibilidad, mantenimiento y portabilidad.

#### 2.3.1 Requisitos comunes de las interfaces

##### 2.3.1.1 Interfaces de usuario

Las interfaces serán realizadas de manera que estén acorde a los sistemas realizados en la Dirección de Informática de modo que tengan una estructura uniforme, además contendrán componentes que faciliten el uso del producto.

##### 2.3.1.2 Interfaces de hardware

El sistema interacciona entre las pantallas del computador, las cuales se verán a través del navegador, también el uso se realizará mediante el ratón y teclado ya que son los medios de comunicación entre el usuario y el sistema.

#### 2.3.2 Requisitos funcionales

Los casos de uso son la descripción de los distintos escenarios con los que los distintos usuarios interactúan con el sistema. Se puede evidenciar las distintas gestiones (Ver Figura 3) donde los eclipses representan los distintos casos de uso y a los actores con figuras humanas.

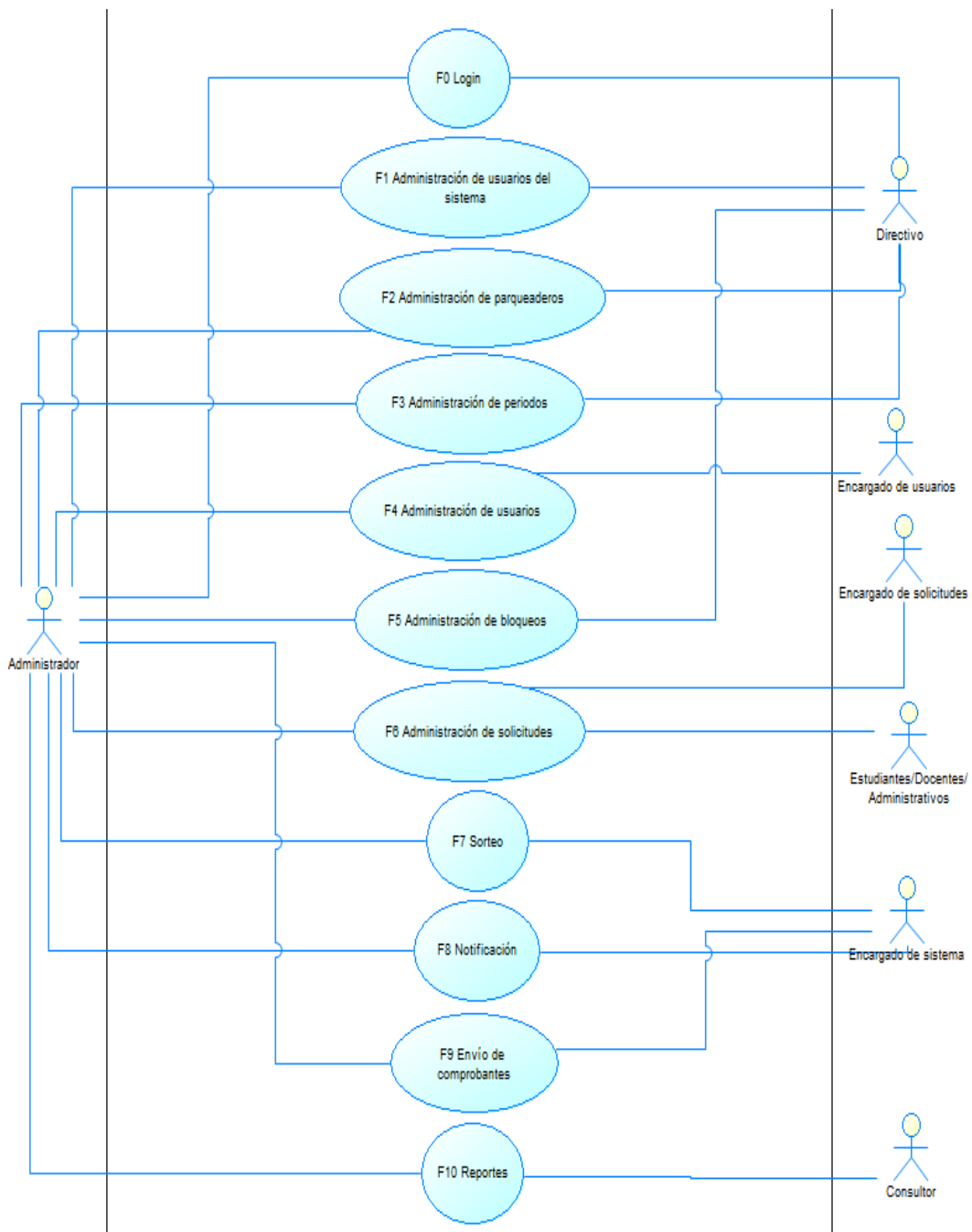


Figura 3: Casos de uso general del sistema

### 2.3.2.1 F1. Administración de usuarios del sistema

La administración de usuarios del sistema comprende al proceso de ingresar y asignar roles, modificar, desactivar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 4).

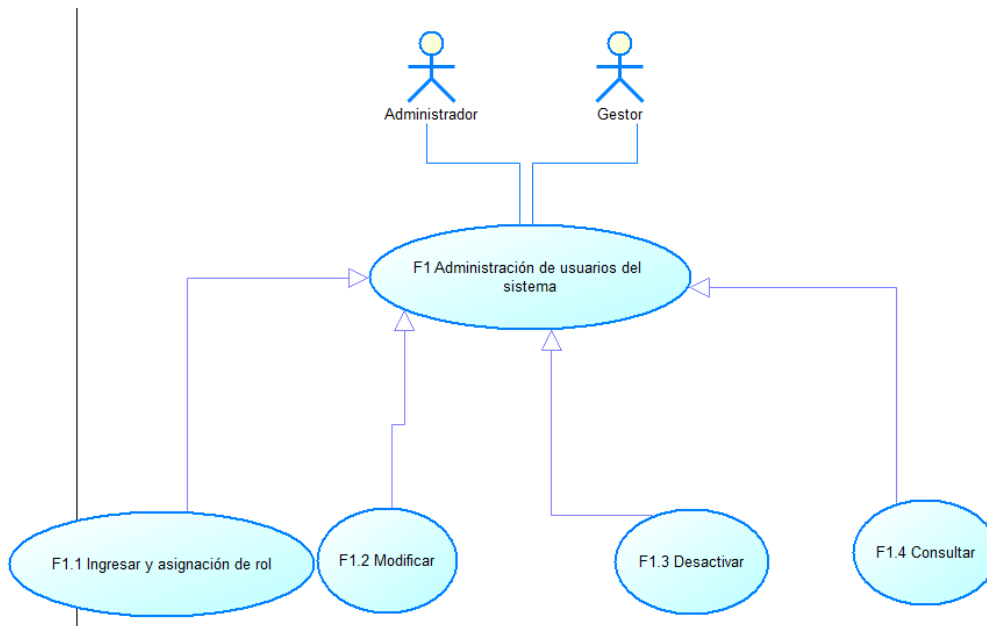


Figura 4: Casos de uso Siguiete nivel F1

#### 2.3.2.1.1 F1.1 Ingresar

El sistema permite ingresar usuarios con sus respectivos roles, los que tendrán acceso a ejecutar los distintos procesos que comprende el sistema (Ver Figura 5).

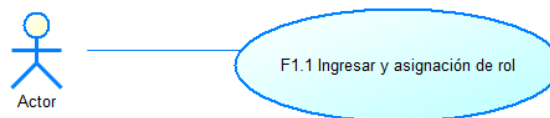


Figura 5: Casos de uso A detalle nivel F1.1

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Usuarios sistema.
3. El sistema muestra por pantalla la ventana de Usuarios sistema.
4. El sistema carga usuarios. (E1)
5. El actor presiona en el botón Nuevo.
6. El actor ingresa datos del usuario.
7. El actor presiona Guardar
8. El sistema verifica el usuario. (E1)
9. El sistema almacena los datos en la base de datos. (E2, E1)
10. El actor selecciona usuario.
11. El sistema carga roles.

12. El actor selecciona los roles.

13. El sistema guarda roles.

### Flujo Alternativo

8. Ver CU F1.2 y F1.3

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.1.2 F1.2 Modificar

Se permite la modificación de la información de los usuarios (Ver Figura 6).

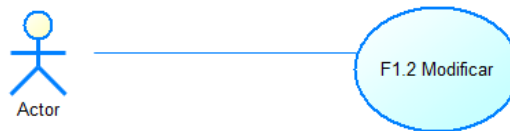


Figura 6: Casos de uso A detalle nivel F1.2

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Usuarios sistema
3. El sistema muestra por pantalla la ventana de Usuarios sistema.
4. El sistema carga usuarios. (E1)
5. El actor selecciona el usuario a modificar.
6. El actor ingresa datos del usuario.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alternativo

5. Ver CU F1.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.1.3 F1.3 Desactivar

Se puede desactivar a los usuarios del sistema (Ver Figura 7).



Figura 7: Casos de uso A detalle nivel F1.3

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Usuarios sistema
3. El sistema muestra por pantalla la ventana de Usuarios sistema.
4. El sistema carga usuarios. (E1)
5. El actor selecciona el usuario a modificar.
6. El actor cambia el estado a inactivo.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alterno

5. Ver CU F1.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.1.4 F1.4 Consultar

Se consulta la información de los usuarios del sistema (Ver Figura 8).

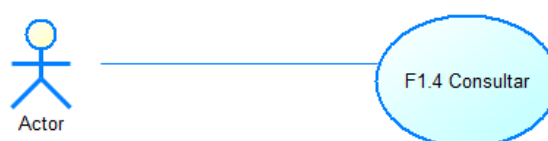


Figura 8: Casos de uso A detalle nivel F1.4

## Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Usuarios sistema
3. El sistema muestra por pantalla la ventana de Usuarios sistema.
4. El sistema carga usuarios. (E1)

## Flujo Alternativo

5. Ver CU F1.1

## Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.2 F2. Administración de parqueaderos

La administración de parqueaderos del sistema comprende al proceso de ingresar, modificar, eliminar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 9).

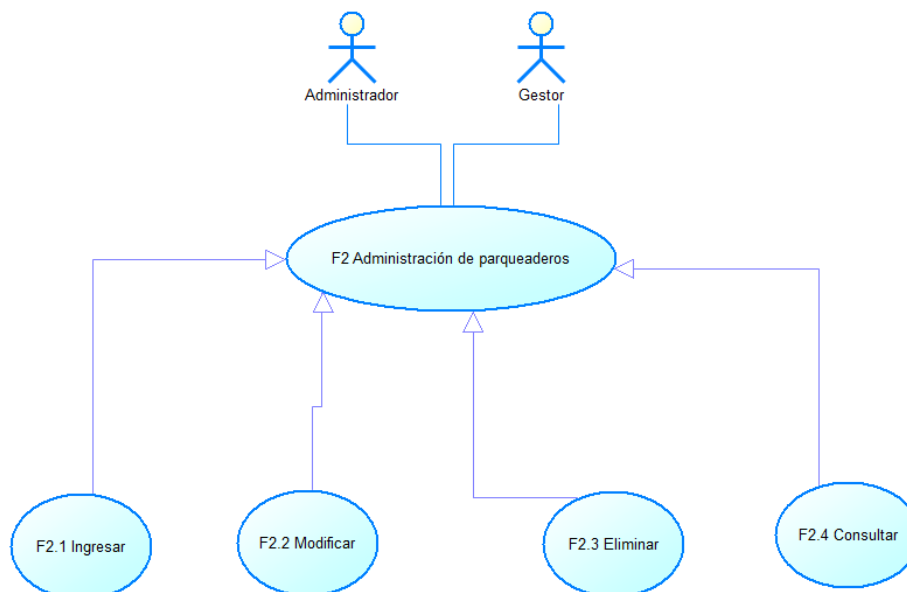


Figura 9: Casos de uso Siguiendo nivel F2

#### 2.3.2.2.1 F2.1 Ingresar

Se ingresa la información de los parqueaderos (Ver Figura 10).

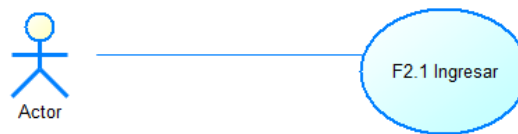


Figura 10: Casos de uso A detalle nivel F2.1

#### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Parqueaderos.
3. El sistema muestra por pantalla la ventana de Parqueaderos.
4. El sistema carga parqueaderos. (E1)
5. El actor presiona en el botón Nuevo.
6. El actor ingresa datos del parqueadero.
7. El actor presiona Guardar
8. El sistema verifica el parqueadero. (E1)
9. El sistema almacena los datos en la base de datos. (E2, E1)

#### Flujo Alternativo

8. Ver CU F2.2 y F2.3

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.2.2 F2.2 Modificar

Se modifica la información de los parqueaderos (Ver Figura 11).

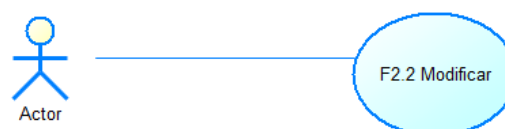


Figura 11: Casos de uso A detalle nivel F2.2

## Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Parqueaderos
3. El sistema muestra por pantalla la ventana de Parqueaderos.
4. El sistema carga parqueaderos. (E1)
5. El actor selecciona el parqueadero a modificar.
6. El actor ingresa datos del parqueadero.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

## Flujo Alterno

5. Ver CU F2.1

## Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.2.3 F2.3 Eliminar

Se permite eliminar la información de los usuarios del sistema (Ver Figura 12).



Figura 12: Casos de uso A detalle nivel F2.3

## Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Parqueaderos.
3. El sistema muestra por pantalla la ventana de Parqueaderos.
4. El sistema carga parqueaderos. (E1)
5. El actor selecciona el parqueadero a eliminar.
6. El actor presiona botón Eliminar.

7. El sistema elimina los datos en la base de datos. (E2, E1)

### Flujo Alternativo

5. Ver CU F2.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.2.4 F2.4 Consultar

Se consulta la información de los parqueaderos del sistema (Ver Figura 13).

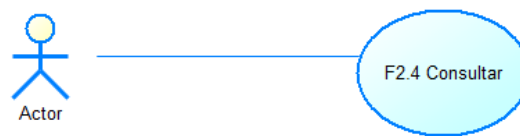


Figura 13: Casos de uso A detalle nivel F2.4

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Parqueaderos.
3. El sistema muestra por pantalla la ventana de Parqueaderos.
4. El sistema carga parqueaderos. (E1)

### Flujo Alternativo

5. Ver CU F2.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.3 F3. Administración de periodos

La administración de periodos del sistema comprende al proceso de ingresar, modificar, desactivar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 14).

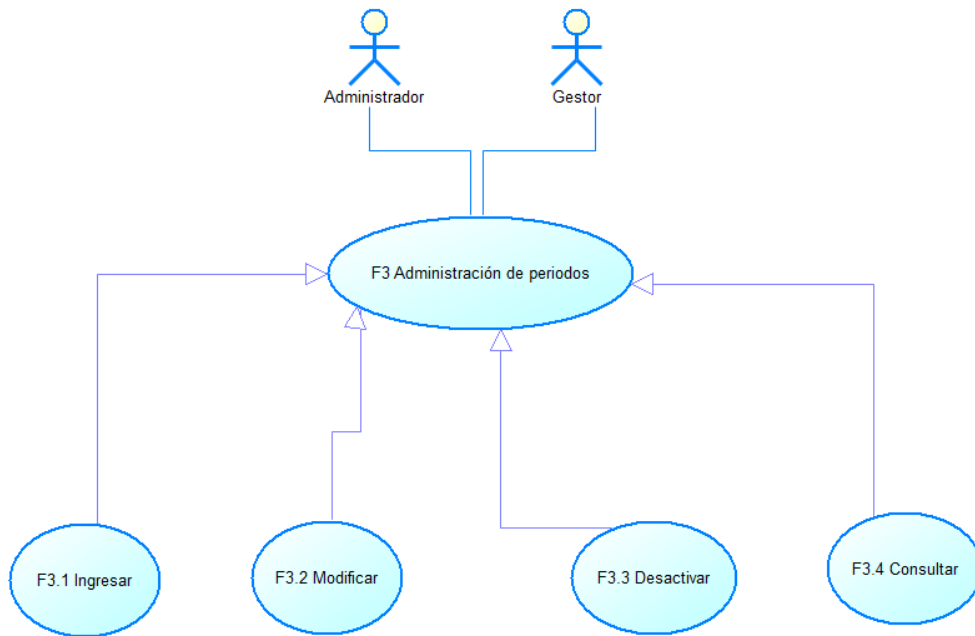


Figura 14: Casos de uso Siguiete nivel F3

#### 2.3.2.3.1 F3.1 Ingresar

El sistema permite ingresar información de los periodos (Ver Figura 15).

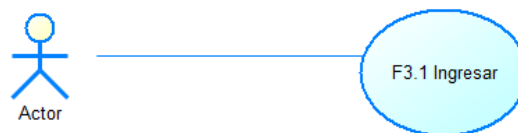


Figura 15: Casos de uso A detalle nivel F3.1

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Periodos.
3. El sistema muestra por pantalla la ventana de Periodos.
4. El sistema carga periodos. (E1)
5. El actor presiona en el botón Nuevo.
6. El actor ingresa datos del periodo.
7. El actor presiona Guardar

8. El sistema verifica el periodo. (E1)
9. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alterno

8. Ver CU F3.2 y F3.3

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.3.2 F3.2 Modificar

Se puede modificar la información de los periodos (Ver Figura 16).



Figura 16: Casos de uso A detalle nivel F3.2

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Periodos.
3. El sistema muestra por pantalla la ventana de Periodos.
4. El sistema carga periodos. (E1)
5. El actor selecciona el periodo a modificar.
6. El actor ingresa datos del periodo.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alterno

5. Ver CU F3.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.3.3 F3.3 Desactivar

El sistema permite desactivar los periodos (Ver Figura 17).

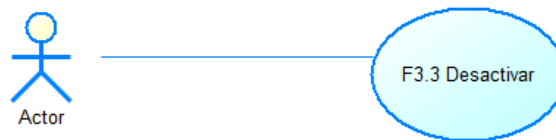


Figura 17: Casos de uso A detalle nivel F3.3

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Periodos.
3. El sistema muestra por pantalla la ventana de Periodos.
4. El sistema carga periodos. (E1)
5. El actor selecciona el periodo a poner activo.
6. El actor cambia el estado.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alternativo

5. Ver CU F3.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.3.4 F3.4 Consultar

Se puede consultar la información de los periodos (Ver Figura 18).

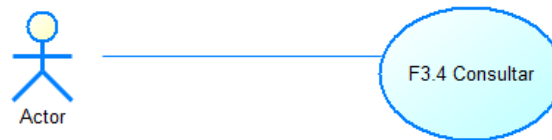


Figura 18: Casos de uso A detalle nivel F3.4

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Periodos.
3. El sistema muestra por pantalla la ventana de Periodos.
4. El sistema carga periodos. (E1)

### Flujo Alternativo

5. Ver CU F3.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.4 F4. Administración de usuarios

La administración de usuarios comprende al proceso de ingresar, modificar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 19).

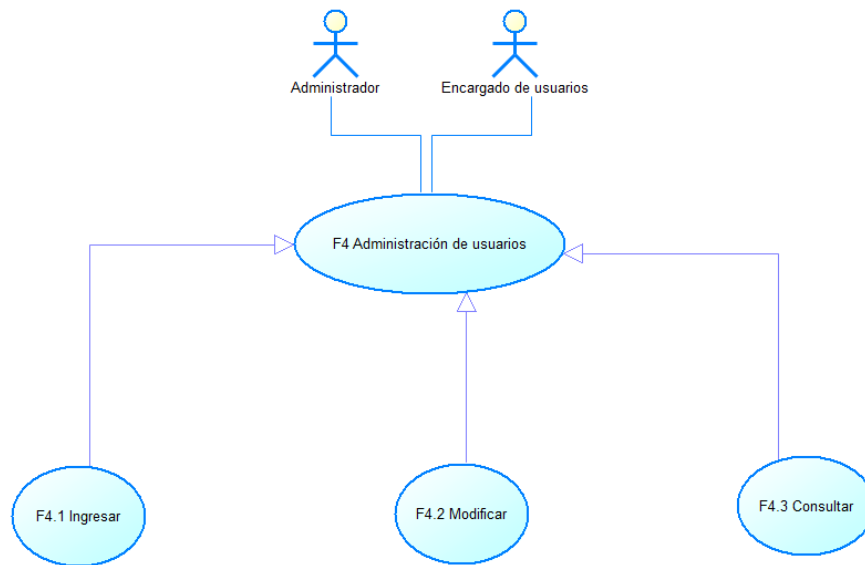


Figura 19: Casos de uso Siguiete nivel F4

#### 2.3.2.4.1 F4.1 Ingresar

El sistema permite ingresar la información de los usuarios (Ver Figura 20).

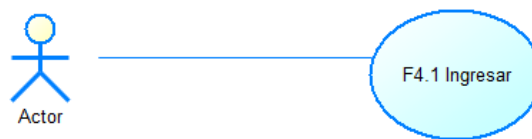


Figura 20: Casos de uso A detalle nivel F4.1

#### Flujo Principal

1. El actor selecciona del menú la opción Usuarios.
2. El actor selecciona Ingresar usuarios.
3. El sistema muestra por pantalla la ventana de Ingresar usuarios.
4. El actor ingresa cédula.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El actor ingresa datos del usuario y vehículo.
8. El actor presiona Guardar
9. El sistema almacena los datos en la base de datos. (E2, E1)

#### Flujo Alterno

7. Ver CU F4.2

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.4.2 F4.2 Modificar

Se modifica la información de los usuarios (Ver Figura 21).

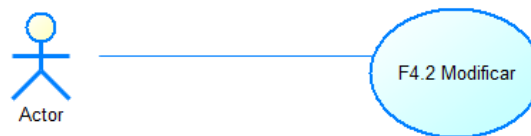


Figura 21: Casos de uso A detalle nivel F4.2

### Flujo Principal

1. El actor selecciona del menú la opción Usuarios.
2. El actor selecciona Modificar usuarios.
3. El sistema muestra por pantalla la ventana de Modificar usuarios.
4. El actor ingresa cédula.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El actor modifica datos del usuario.
8. El actor presiona Guardar.
9. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alternativo

7. Ver CU F4.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.4.3 F4.3 Consultar

Se consulta la información de los usuarios (Ver Figura 22).

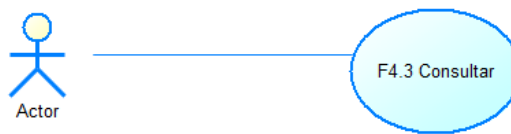


Figura 22: Casos de uso A detalle nivel F4.3

#### Flujo Principal

1. El actor selecciona del menú la opción Consulta.
2. El actor selecciona Buscar.
3. El sistema muestra por pantalla la ventana de Consultar usuarios.
4. El actor ingresa cédula.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El sistema carga parqueaderos.

#### Flujo Alternativo

7. Ver CU F4.1

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada

#### 2.3.2.5 F5. Administración de bloqueos

La administración de bloqueos comprende al proceso de ingresar, modificar, eliminar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 23).

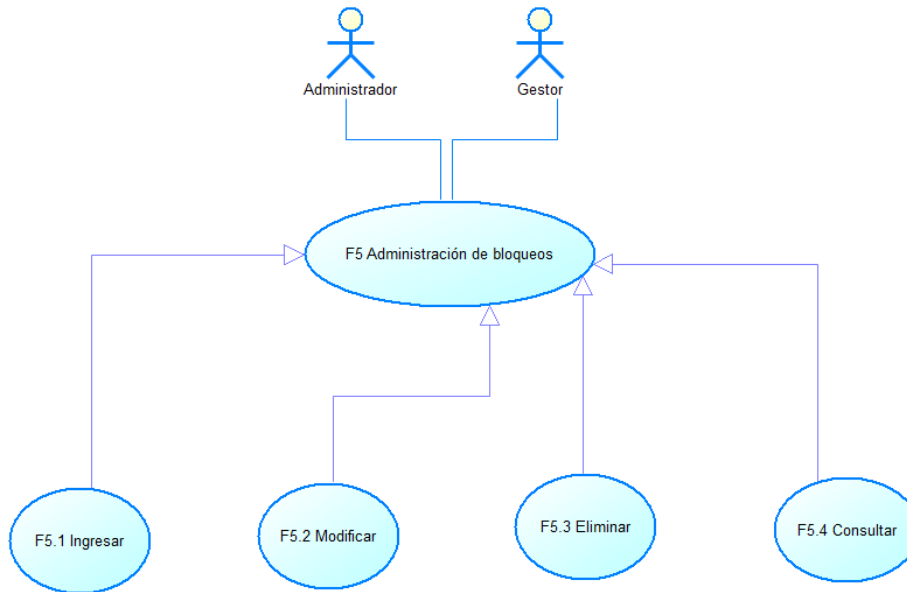


Figura 23: Casos de uso Siguiete nivel F5

#### 2.3.2.5.1 F5.1 Ingresar

Se ingresa la información del bloqueo con su respectivo usuario (Ver Figura 24).

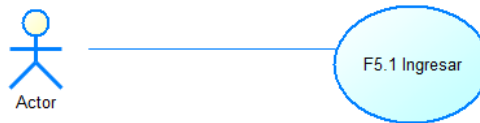


Figura 24: Casos de uso A detalle nivel F5.1

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Bloqueos.
3. El sistema muestra por pantalla la ventana de Bloqueos.
4. El sistema carga bloqueos. (E1)
5. El actor presiona en el botón Nuevo.
6. El actor ingresa cédula.
7. El actor presiona Buscar.
8. El sistema carga información. (E1)
9. El actor ingresa datos del bloqueado.
10. El actor presiona Guardar
11. El sistema verifica al bloqueado. (E1)
12. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alternativo

12. Ver CU F5.2 y F5.3

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.5.2 F5.2 Modificar

Se modifica la información de los bloqueos (Ver Figura 25).

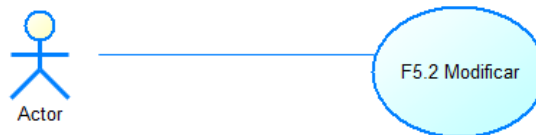


Figura 25: Casos de uso A detalle nivel F5.2

### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Bloqueos.
3. El sistema muestra por pantalla la ventana de Bloqueos.
4. El sistema carga bloqueos. (E1)
5. El actor selecciona el bloqueo a modificar.
6. El actor ingresa datos del bloqueo.
7. El actor presiona Actualizar.
8. El sistema almacena los datos en la base de datos. (E2, E1)

### Flujo Alternativo

5. Ver CU F5.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.5.3 F5.3 Eliminar

Se puede eliminar los bloqueos (Ver Figura 26).

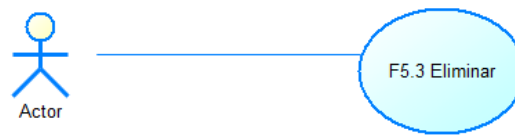


Figura 26: Casos de uso A detalle nivel F5.3

#### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Bloqueos.
3. El sistema muestra por pantalla la ventana de Bloqueos.
4. El sistema carga bloqueos. (E1)
5. El actor selecciona el bloqueo a eliminar.
6. El actor presiona botón Eliminar.
7. El sistema elimina los datos en la base de datos. (E2, E1)

#### Flujo Alternativo

5. Ver CU F5.1

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.5.4 F5.4 Consultar

Se puede consultar la información de los bloqueos (Ver Figura 27).



Figura 27: Casos de uso A detalle nivel F5.4

#### Flujo Principal

1. El actor selecciona del menú la opción Administración.
2. El actor selecciona Bloqueos.
3. El sistema muestra por pantalla la ventana de Bloqueos.
4. El sistema carga bloqueos. (E1)

### Flujo Alternativo

5. Ver CU F5.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.6 F6. Administración de solicitudes

La administración de solicitudes comprende al proceso de ingresar, modificar y consultar dependiendo de los actores que se implican en esta gestión (Ver Figura 28).

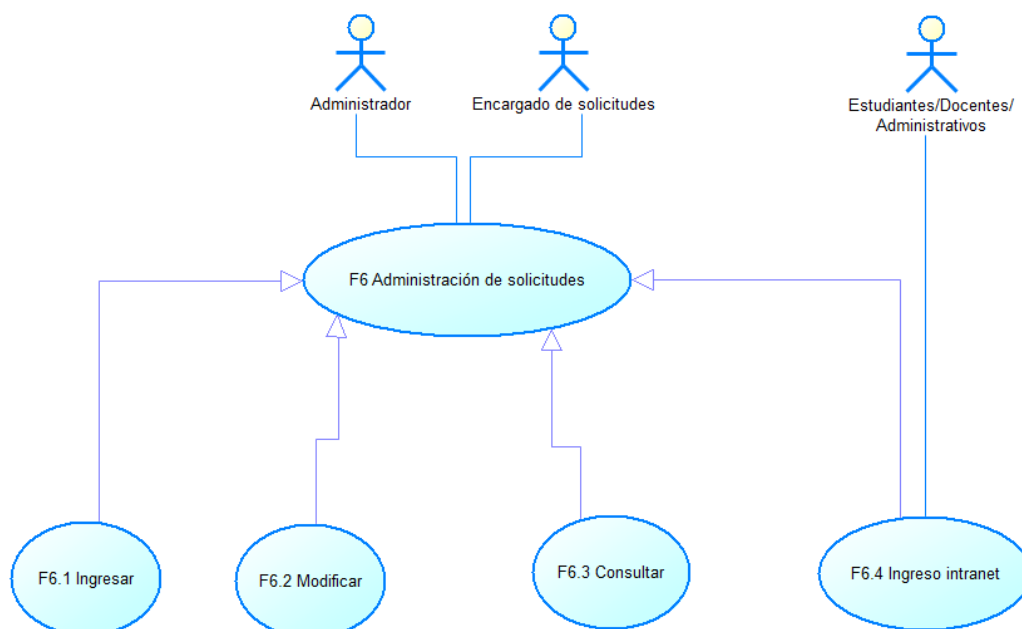


Figura 28: Casos de uso Siguiendo nivel F6

### 2.3.2.6.1 F6.1 Ingresar

Se ingresa la información de las solicitudes (Ver Figura 29).

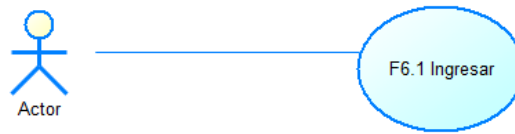


Figura 29: Casos de uso A detalle nivel F6.1

#### Flujo Principal

1. El actor selecciona del menú la opción Solicitudes.
2. El actor selecciona Ingresar solicitud.
3. El sistema muestra por pantalla la ventana de Ingresar solicitudes.
4. El actor ingresa cédula.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El actor ingresa datos de la solicitud.
8. El actor presiona Guardar
9. El sistema almacena los datos en la base de datos. (E2, E1)

#### Flujo Alternativo

7. Ver CU F6.2

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.6.2 F6.2 Modificar

Se puede modificar la información de las solicitudes (Ver Figura 30).

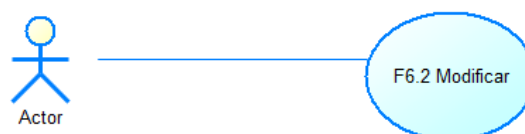


Figura 30: Casos de uso A detalle nivel F6.2

## Flujo Principal

1. El actor selecciona del menú la opción Solicitudes.
2. El actor selecciona Modificar solicitudes.
3. El sistema muestra por pantalla la ventana de Modificar solicitudes.
4. El actor ingresa cédula.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El actor selecciona solicitud a modificar.
8. El actor modifica datos del usuario.
9. El actor presiona Actualizar.
10. El sistema almacena los datos en la base de datos. (E2, E1)

## Flujo Alterno

7. Ver CU F6.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.6.3 F6.3 Consultar

Se consulta la información de las solicitudes (Ver Figura 31)

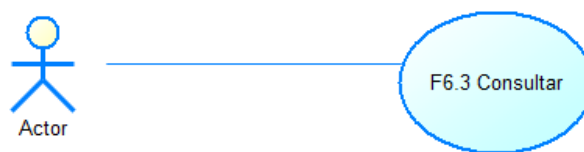


Figura 31: Casos de uso A detalle nivel F6.3

## Flujo Principal

1. El actor selecciona del menú la opción Consulta.
2. El actor selecciona Reportes.
3. El sistema muestra por pantalla la ventana de Reportes.
4. El actor ingresa parámetros.
5. El actor presiona Buscar.

6. El sistema verifica existencia y carga datos. (E1)
7. El sistema carga solicitudes.

### Flujo Alternativo

7. Ver CU F6.1

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

#### 2.3.2.6.4 F6.4 Ingreso intranet

Se ingresa la información de las solicitudes con acceso a toda la comunidad universitaria (Ver Figura 32).

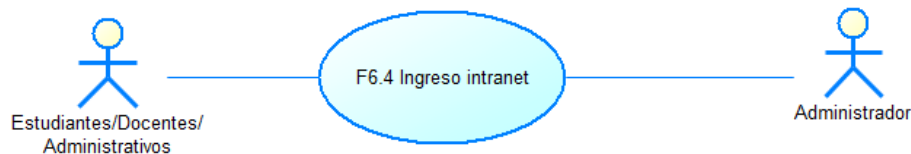


Figura 32: Casos de uso A detalle nivel F6.4

### Flujo Principal

1. El actor selecciona Parqueo.
2. El sistema verifica si ya ha solicitado.
3. El actor ingresa datos.
4. El actor presiona Guardar.
5. El sistema almacena los datos en la base de datos. (E2, E1)

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

**E2:** Datos incompletos.

Posible solución: Desplegar mensaje solicitando al actor completar la información de manera adecuada.

### 2.3.2.7 F7. Sorteo

El sorteo permite asignar parqueaderos a las solicitudes en el periodo actual (Ver Figura 33).



Figura 33: Casos de uso Siguiete nivel F7

#### Flujo Principal

1. El actor selecciona del menú la opción Sistema.
2. El actor selecciona Sorteo.
3. El sistema muestra por pantalla la ventana de Sorteo.
4. El sistema carga datos sobre el sorteo.
5. El actor presiona Sortear.
6. El sistema verifica que no se haya sorteado en el periodo actual. (E1)
7. El sistema actualiza las solicitudes en la base de datos. (E1)

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

### 2.3.2.8 F8. Notificación

Las notificaciones permiten avisar a los favorecidos del sorteo (Ver Figura 34).

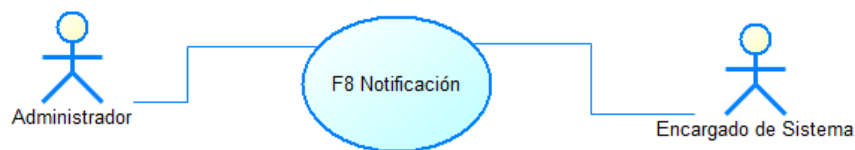


Figura 34: Casos de uso Siguiete nivel F8

#### Flujo Principal

1. El actor selecciona del menú la opción Sistema.
2. El actor selecciona Enviar correos.
3. El sistema muestra por pantalla la ventana de Enviar correos.
4. El sistema carga datos sobre la notificación.

5. El actor presiona Enviar correos.
6. El sistema verifica que personas faltan por notificar en el periodo actual.  
(E1)
7. El sistema envía correos. (E1)
8. El sistema actualiza las solicitudes notificadas en la base de datos. (E1)

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

#### 2.3.2.9 F9. Envío de comprobantes

Se realiza en envío de comprobantes de las solicitudes favorecidas en el sorteo (Ver Figura 35).

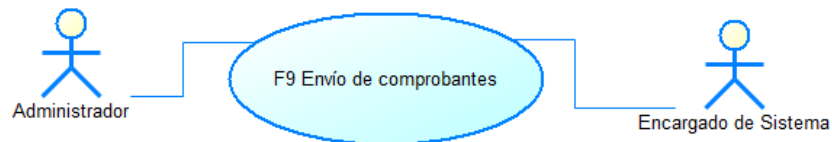


Figura 35: Casos de uso Siguiete nivel F9

### Flujo Principal

1. El actor selecciona del menú la opción Sistema.
2. El actor selecciona Comprobantes.
3. El sistema muestra por pantalla la ventana de Comprobantes.
4. El sistema carga datos sobre los comprobantes enviados y pendientes.
5. El actor presiona Enviar comprobantes.
6. El sistema verifica que solicitudes faltan por enviar en el periodo actual.  
(E1)
7. El sistema envía comprobantes. (E1)
8. El sistema actualiza las solicitudes notificadas en la base de datos. (E1)

### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

### 2.3.2.10 F10.Reportes

Se general reportes para el análisis de los mismos dependiendo de los parámetros que determine el usuario (Ver Figura 36).



Figura 36: Casos de uso Siguiete nivel F10

#### Flujo Principal

1. El actor selecciona del menú la opción Consulta.
2. El actor selecciona Reportes.
3. El sistema muestra por pantalla la ventana de Reportes.
4. El actor ingresa parámetros.
5. El actor presiona Buscar.
6. El sistema verifica existencia y carga datos. (E1)
7. El sistema carga reportes.

#### Descripción de excepciones

**E1:** Se ha producido un error.

Posible solución: Contactarse con el administrador de la base de datos.

### 2.3.3 Requisitos no funcionales

#### 2.3.3.1 Requisitos de rendimiento

Representa el desempeño que el sistema tendrá bajo ciertas condiciones de recursos utilizados.

#### 2.3.3.2 Seguridad

Es importante la protección de la información y los datos de manera que solo las personas autorizadas puedan ver la o modificar dicha información.

#### 2.3.3.3 Fiabilidad

Posibilidad de un sistema a comportarse de manera adecuada bajo ciertas condiciones y periodos de tiempo determinados.

#### 2.3.3.4 Disponibilidad

Capacidad de estar operativo, funcional y accesible al sistema siempre que se lo necesite o requiriera.

#### 2.3.3.5 Mantenibilidad

Es la facultad a ser evolutivo ya que el producto puede tener modificaciones y estas deben ser hechas de manera efectiva y eficientes, correctivas o perfectivas.

#### 2.3.3.6 Portabilidad

Es ser compatible bajo diferentes entornos en los que el usuario hará uso del sistema el cual debe tener un correcto comportamiento.

### 2.4 Otros

#### 2.4.1 Diagrama Conceptual del Sistema

El diagrama conceptual del sistema representa las tablas y relaciones con las que se cuenta, misma que sirve de base para el desarrollo de la base de datos (Ver Figura 37).

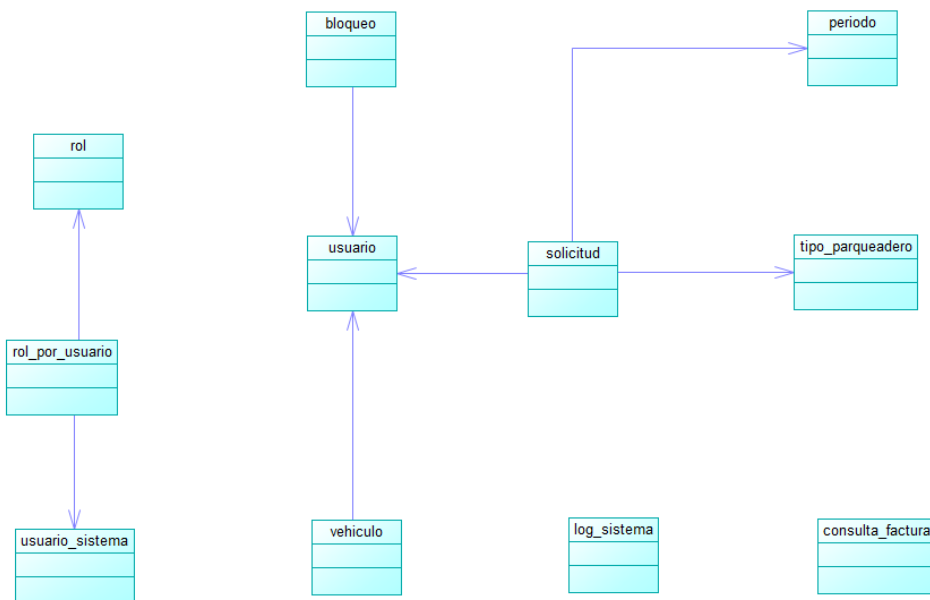


Figura 37: Diagrama Conceptual

#### 2.4.2 Plan de pruebas del sistema

##### 2.4.2.1 F1: Administración de usuarios del sistema

#### Precondiciones:

Las precondiciones para probar cada funcionalidad de administración de usuarios del sistema (Ver Tabla 2).

*Tabla 2: Casos de prueba F1*

<b>Funcionalidad</b>	<b>Dato entrante</b>	<b>Resultados Esperados</b>
F1.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F1.4	Selecciona opción de usuarios y despliega los mismos	Despliegue de la pantalla de consultas
F1.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error
F1.3	Selecciona opción de inactivo un registro	Registro desactivado

**Postcondiciones:** Repetir F1.1 10 veces

#### *2.4.2.2 F2: Administración de parqueaderos*

#### **Precondiciones:**

Las precondiciones para probar cada funcionalidad de administración de parqueaderos (Ver Tabla 3).

*Tabla 3: Casos de prueba F2*

<b>Funcionalidad</b>	<b>Dato entrante</b>	<b>Resultados Esperados</b>
F2.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F2.4	Selecciona opción de parqueaderos y despliega los mismos	Despliegue de la pantalla de consultas
F2.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error
F2.3	Selecciona opción de eliminar un registro	Registro eliminado

**Postcondiciones:** Repetir F2.1 10 veces

#### *2.4.2.3 F3: Administración de periodos*

#### **Precondiciones:**

Las precondiciones para probar cada funcionalidad de administración de periodos (Ver Tabla 4).

Tabla 4: Casos de prueba F3

Funcionalidad	Dato entrante	Resultados Esperados
F3.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F3.4	Selecciona opción de periodos y despliega los mismos	Despliegue de la pantalla de consultas
F3.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error
F3.3	Activar otro periodo en el registro	Registro desactivado

**Postcondiciones:** Repetir F3.1 10 veces

#### 2.4.2.4 F4: Administración de usuarios

##### **Precondiciones:**

Las precondiciones para probar cada funcionalidad de administración de usuarios (Ver Tabla 5).

Tabla 5: Casos de prueba F4

Funcionalidad	Dato entrante	Resultados Esperados
F4.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F4.3	Selecciona consulta y buscar y se despliega los mismos	Despliegue de la pantalla de consultas
F4.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error

**Postcondiciones:** Repetir F4.1 10 veces

#### 2.4.2.5 F5: Administración de bloqueos

##### **Precondiciones:**

Las precondiciones para probar cada funcionalidad de administración de bloqueos (Ver Tabla 6).

Tabla 6: Casos de prueba F5

Funcionalidad	Dato entrante	Resultados Esperados
F5.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error

F5.4	Selecciona opción de bloqueos y despliega los mismos	Despliegue de la pantalla de consultas
F5.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error
F5.3	Selecciona opción de eliminar un registro	Registro eliminado

**Postcondiciones:** Repetir F5.1 10 veces

#### 2.4.2.6 F6: Administración de solicitudes

##### **Precondiciones:**

Las precondiciones para probar cada funcionalidad de administración de solicitudes (Ver Tabla 7).

*Tabla 7: Casos de prueba F6*

Funcionalidad	Dato entrante	Resultados Esperados
F6.1	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F6.4	Datos válidos	Información almacenada
	Datos no válidos	Mensaje de error
F6.3	Selecciona opción de reportes y consulta, luego despliega los mismos	Despliegue de la pantalla de consultas
F6.2	Datos válidos	Registro modificado
	Datos no válidos	Mensaje de error

**Postcondiciones:** Repetir F6.1 10 veces

#### 2.4.2.7 F7: Sorteo

##### **Precondiciones:**

La precondición para probar cada funcionalidad del sorteo (Ver Tabla 8).

*Tabla 8: Casos de prueba F7*

Funcionalidad	Dato entrante	Resultados Esperados
F7	Realizar sorteo	El sorteo se ha realizado correctamente

**Postcondiciones:** Repetir F7 10 veces

#### 2.4.2.8 F8: Notificación

##### **Precondiciones:**

La precondition para probar cada funcionalidad de las notificaciones (Ver Tabla 9).

Tabla 9: Casos de prueba F8

Funcionalidad	Dato entrante	Resultados Esperados
F8	Enviar correos	Los correos han sido enviados correctamente

**Postcondiciones:** Repetir F8 10 veces

#### 2.4.2.9 F9: Envió de comprobantes

**Precondiciones:**

La precondition para probar cada funcionalidad del envío de comprobantes (Ver Tabla 10).

Tabla 10: Casos de prueba F9

Funcionalidad	Dato entrante	Resultados Esperados
F9	Enviar comprobantes	Los comprobantes han sido enviados correctamente

**Postcondiciones:** Repetir F9 10 veces

#### 2.4.2.10 F10: Reportes

**Precondiciones:**

La precondition para probar cada funcionalidad de los reportes (Ver Tabla 11).

Tabla 11: Casos de prueba F10

Funcionalidad	Dato entrante	Resultados Esperados
F10	Consultar reportes	La pantalla presenta los reportes requeridos.

**Postcondiciones:** Repetir F10 10 veces

## 3 Capítulo III: Diseño

### 3.1 Introducción

Es este apartado se detalla todos los procesos del producto. Además, se inspecciona a detalle el diseño y se realiza el plan de pruebas de integración. Aquí se especifica como se va a desenvolver y manejar el producto a desarrollar.

### 3.2 Estructura del producto

#### 3.2.1 Diagrama de actividades

A través de dicho diagrama se presenta el flujo de control entre las actividades a lo largo del ciclo de vida del proceso que comprende el proceso actual, es decir desde el momento que se crean un nuevo periodo en el sistema anterior, hasta que los guardias reciben el listado de los estudiantes con acceso a un parqueadero.

El proceso comienza al crear un nuevo periodo, por la general los estudiantes de grado realizan una solicitud por medio del intranet, mientras que los de postgrado se acercan a la dirección general para solicitar su parqueadero, una vez ingresadas las solicitudes se procede a asignar parqueaderos a los estudiantes de postgrado, y por otro lado a sortear a los de grado por medio del sistema anterior.

Una vez se haya completado el proceso de sorteo se envía un listado a recursos humanos con los favorecidos para que se adjunte a la lista de administrativos y docentes.

Para finalizar, se envía dicho reporte al departamento de seguridad donde se entrega los stickers y se toma los datos de las personas que solicitaron el servicio de parqueaderos. Se puede evidenciar este proceso en la Figura 38.

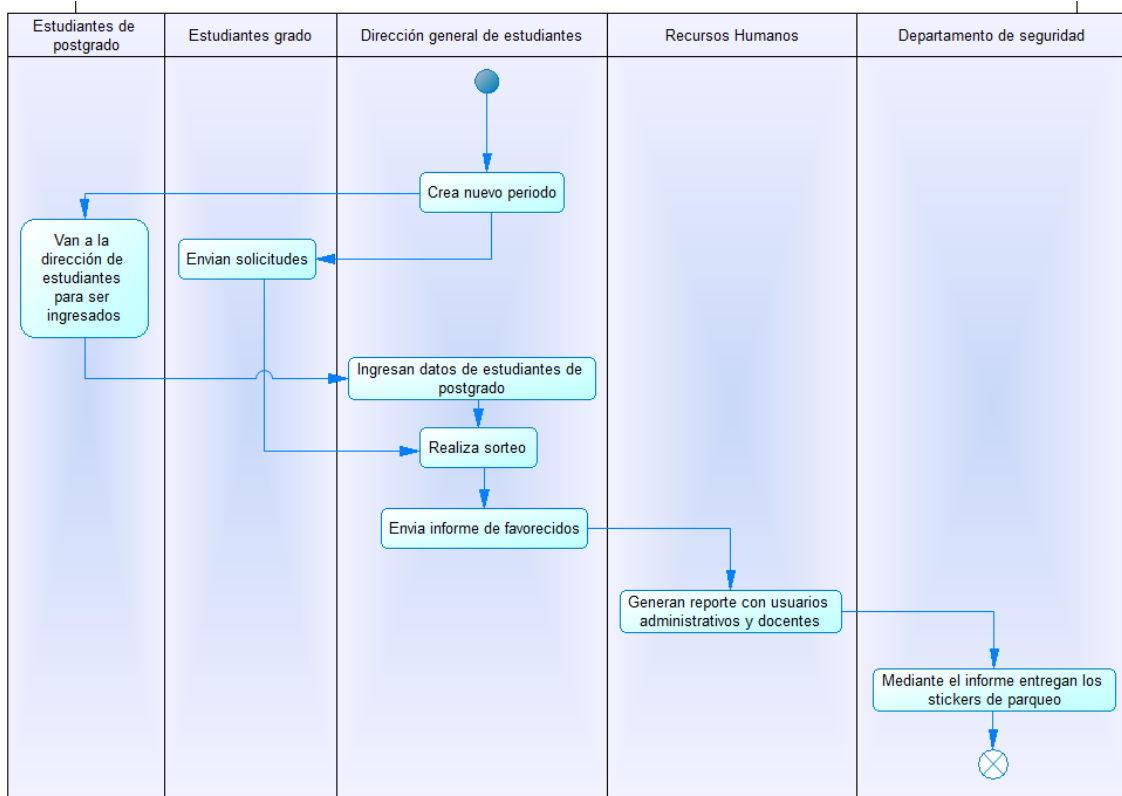


Figura 38: Diagrama de actividades

### 3.2.2 Diagrama de clases

Se puede identificar a continuación la estructura que tiene el sistema donde se visualiza los atributos y operaciones, de las distintas entidades que forman parte del sistema (Ver Figura 39).

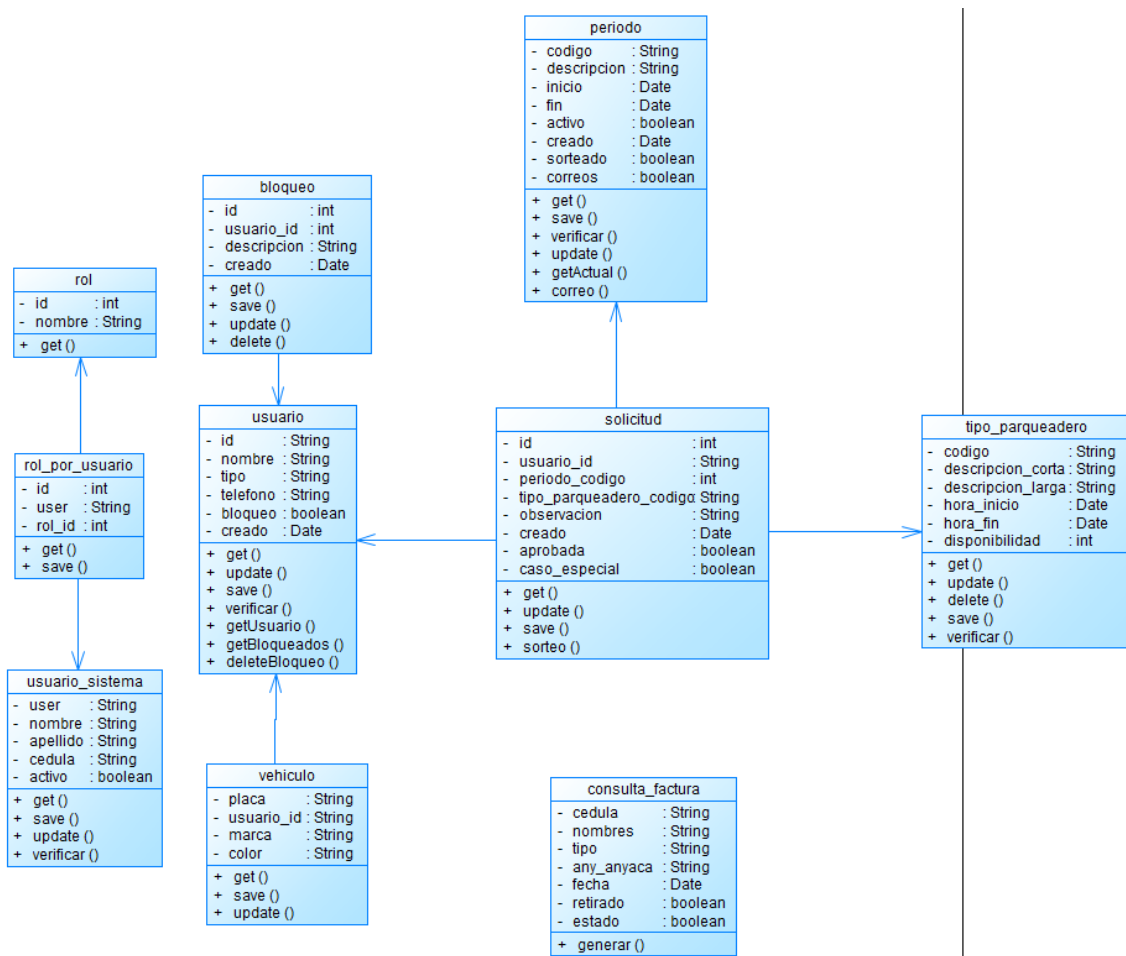


Figura 39: Diagrama de clases

### 3.3 Diagramas de Interacción

También conocidos como diagramas de secuencia, y son aquellos que muestran la correlación de objetos del sistema a través del tiempo y contiene especificaciones del desarrollo.

#### 3.3.1 Administración de usuarios del sistema

##### 3.3.1.1 Ingreso y asignación de rol

El diagrama de secuencia del ingreso de usuarios del sistema indica la interacción entre los componentes para que el proceso sea efectuado, dando paso desde las acciones que realiza el actor, pasando por las distintas capas que comprende el sistema hasta las respuestas hacia quien lo solicito.

Como se presenta en la Figura 40 el flujo comienza cuando el usuario selecciona la opción de administración en la ventana principal y además escoge usuario sistema, es aquí donde la ventana por medio del constructor carga la pantalla indicada, misma que al ejecutarse realiza una petición para traer los usuarios y

presentarlos. Posteriormente el actor presiona el botón de nuevo y se presenta ejecuta la pantalla de nuevo usuario del sistema, una vez el usuario llene los campos presiona guardar y se ejecutan las funciones para guardar los datos, donde primero se realizan verificaciones de los datos, los almacena y presenta por pantalla. Luego el actor selecciona los roles y presiona actualizar de modo que se asignan al usuario dichos accesos.

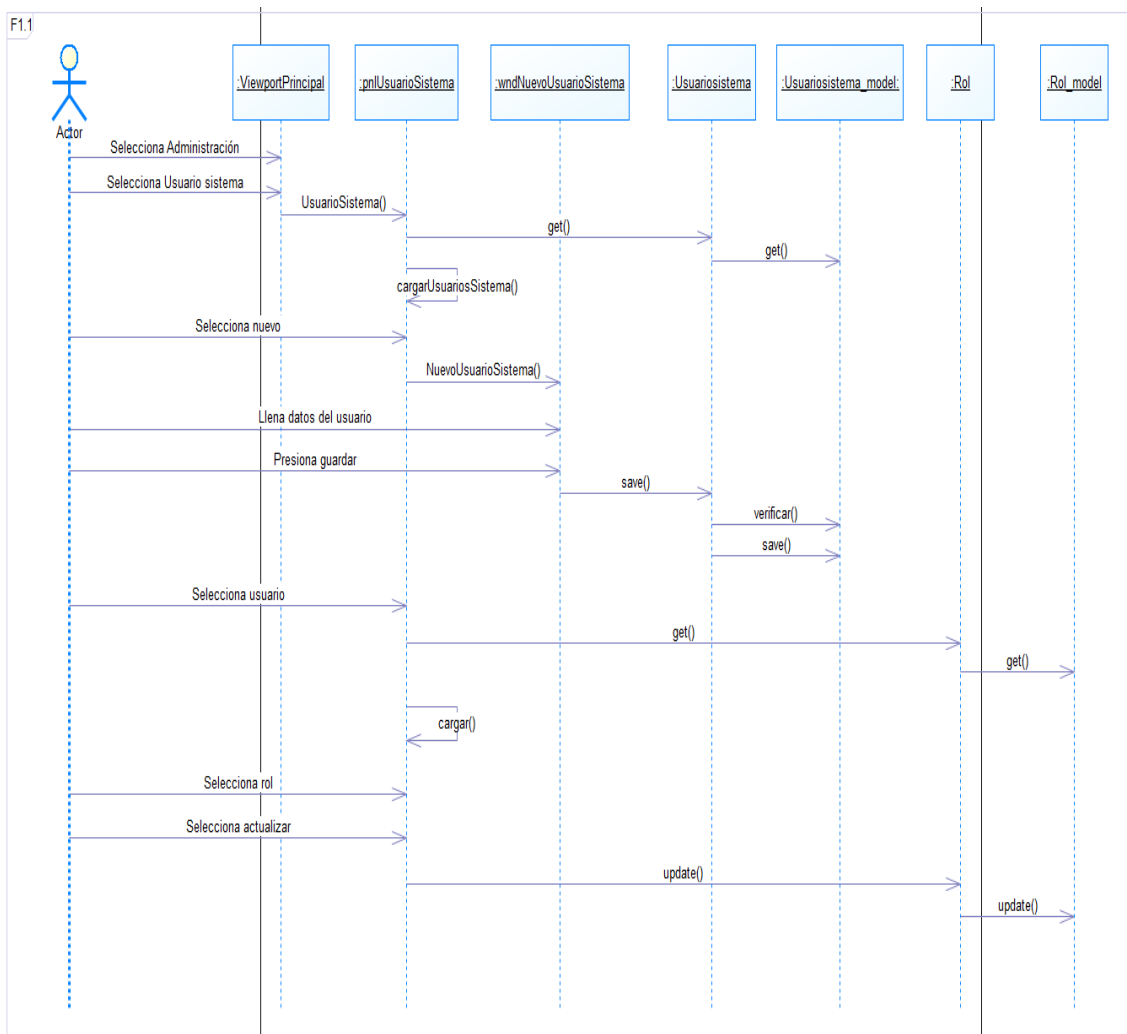


Figura 40: F1.1 Ingreso y asignación de rol

### 3.3.1.2 Modificación

En la modificación describe las acciones cuando se desea actualizar o modificar el usuario del sistema.

Según la Figura 41 una vez que el actor se sitúe en la pantalla de usuarios del sistema, selecciona el registro a modificar e inmediatamente se cargan los roles, donde el actor puede realizar cambios ya sea en información del usuario o la

asignación de accesos y presiona actualizar, el sistema ejecuta las funciones para modificar y estas son almacenadas en la base de datos.

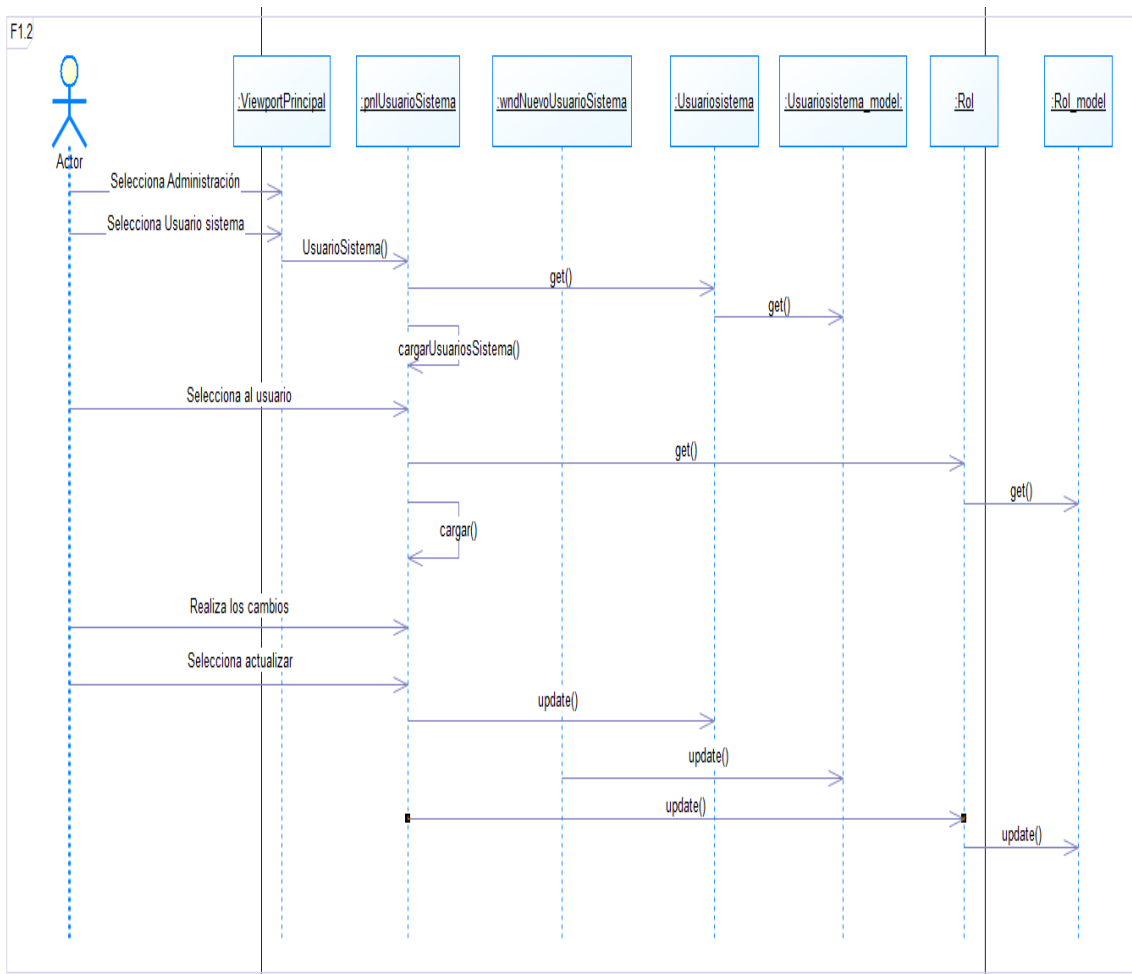


Figura 41: F1.2 Modificación

### 3.3.1.3 Desactivado

La desactivación de usuarios permite tener un registro completo de los usuarios ya que no se permite eliminar físicamente, de modo que sea solo lógico.

El actor al encontrarse en la pantalla indicada selecciona el usuario a desactivar y en uno de sus campos cambia la opción de activo y presiona actualizar y el sistema modifica el registro de modo que ya no tiene acceso al sistema (Ver Figura 42).

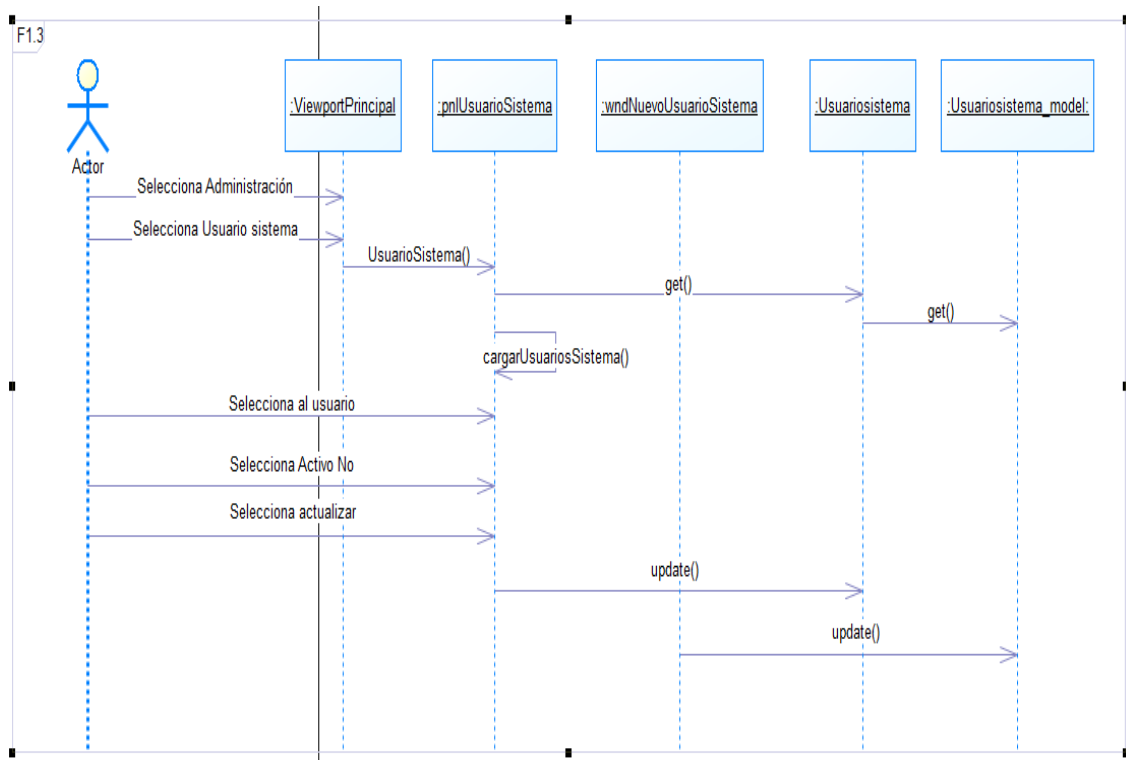


Figura 42: F1.3 Desactivado

### 3.3.1.4 Consulta

Mediante la consulta permite ver los usuarios del sistema que se encuentran ingresados en el sistema.

El actor al ejecutar en el componente de usuario en el sistema realiza la petición para traer los registros correspondientes y presentarlos por pantalla como se representa en la Figura 43.

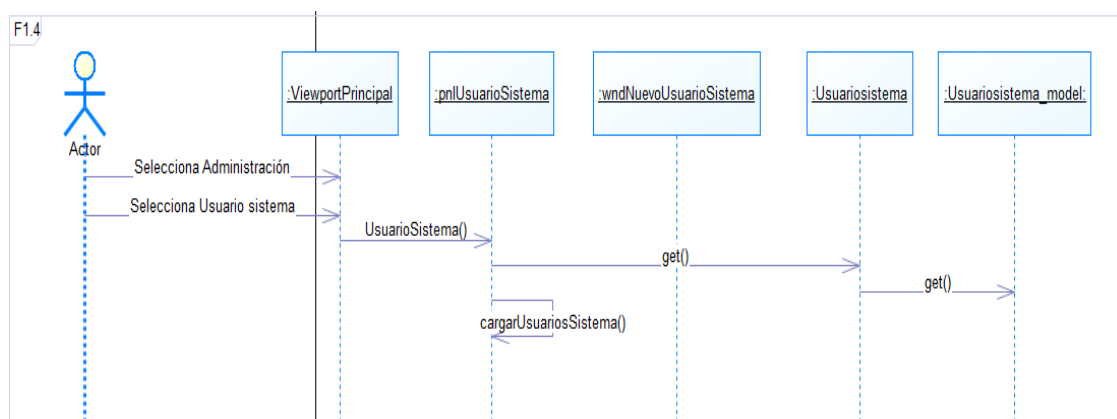


Figura 43: F1.4 Consulta

### 3.3.2 Administración de parqueaderos

#### 3.3.2.1 Ingreso

Permite ingresar nuevos parqueaderos y se compone de las siguientes acciones a nivel de las distintas capas de programación.

Representado en la Figura 44 el usuario selecciona la administración, disponibilidad y el sistema presenta la pantalla de parqueaderos donde se ejecuta la función que hace la petición al backend y carga los registros en pantalla. Luego el usuario presiona nuevo y se presenta el formulario donde el actor llena los campos requeridos y presiona el botón guardar que ejecuta las funciones de verificación y guardado.

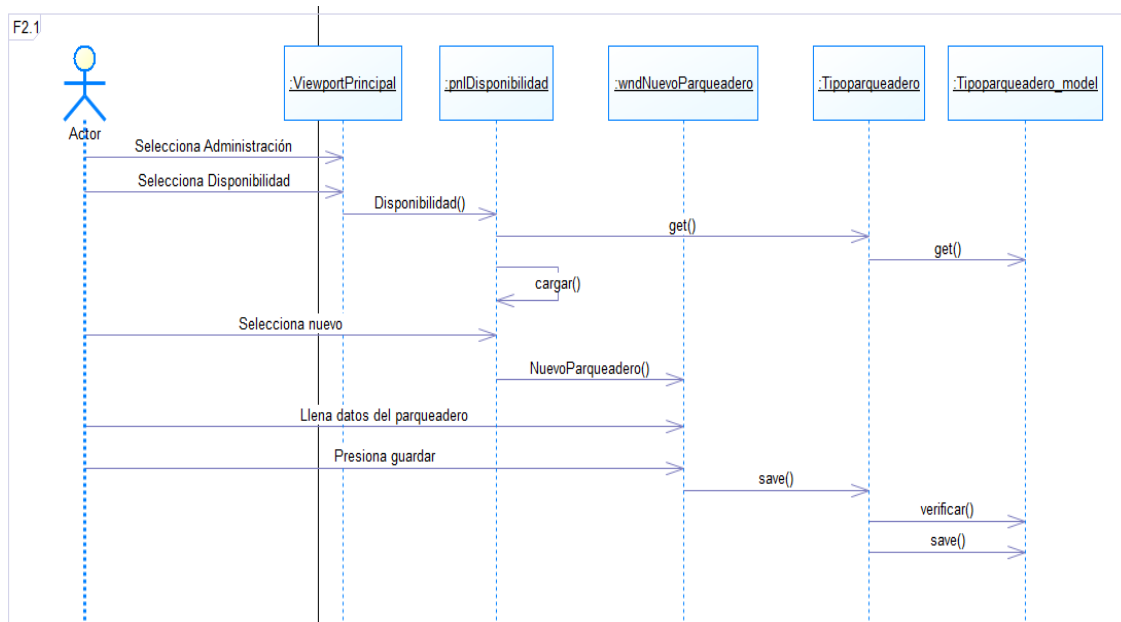


Figura 44: F2.1 Ingreso

#### 3.3.2.2 Modificación

Mediante la modificación se realizan la actualización de registros existentes en la base de datos, ya que puede haber datos que varíen en el tiempo.

El actor al situarse en la pantalla de disponibilidad según la Figura 45 se selecciona el parqueadero a ser actualizado y cambia los datos de este, luego presiona actualizar y se ejecuta la función de actualizado que modifica en la base.

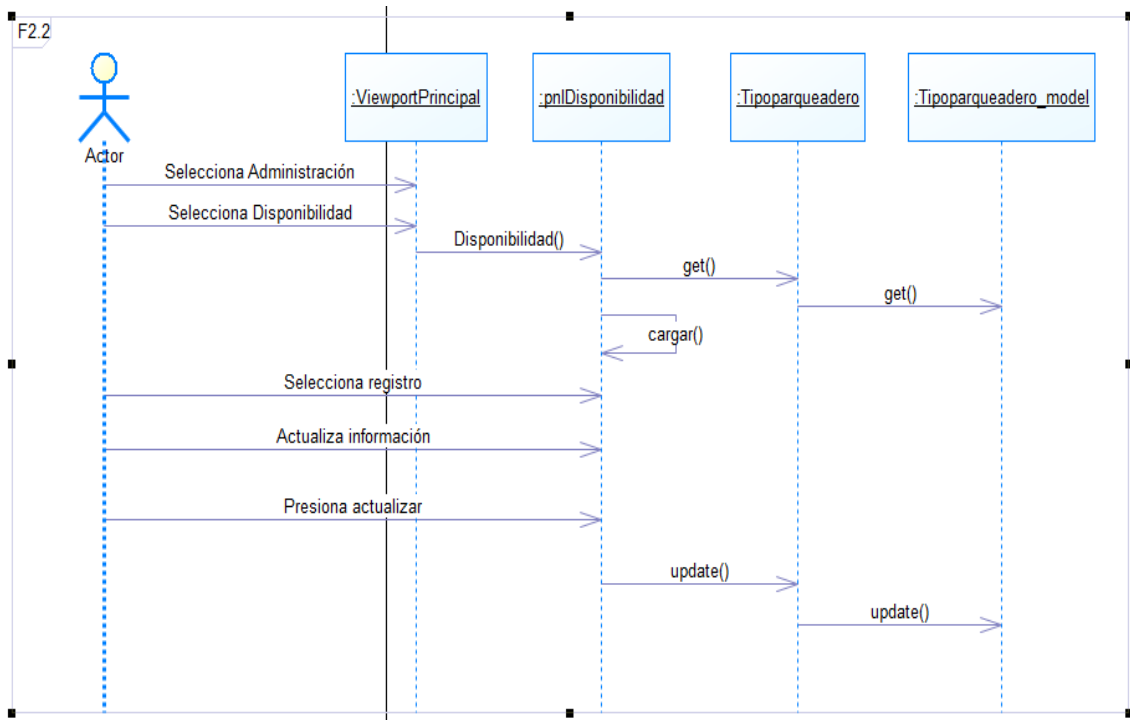


Figura 45: F2.2 Modificación

### 3.3.2.3 Eliminación

En la eliminación el actor puede desechar aquellos registros que no sirvan para este catálogo de parqueaderos.

El flujo comienza cuando el actor ingresa en la pantalla de disponibilidad y selecciona un parqueadero, posteriormente presiona el botón eliminar que ejecuta las funciones de eliminado donde se borra el registro de la base de datos como lo indica la Figura 46.

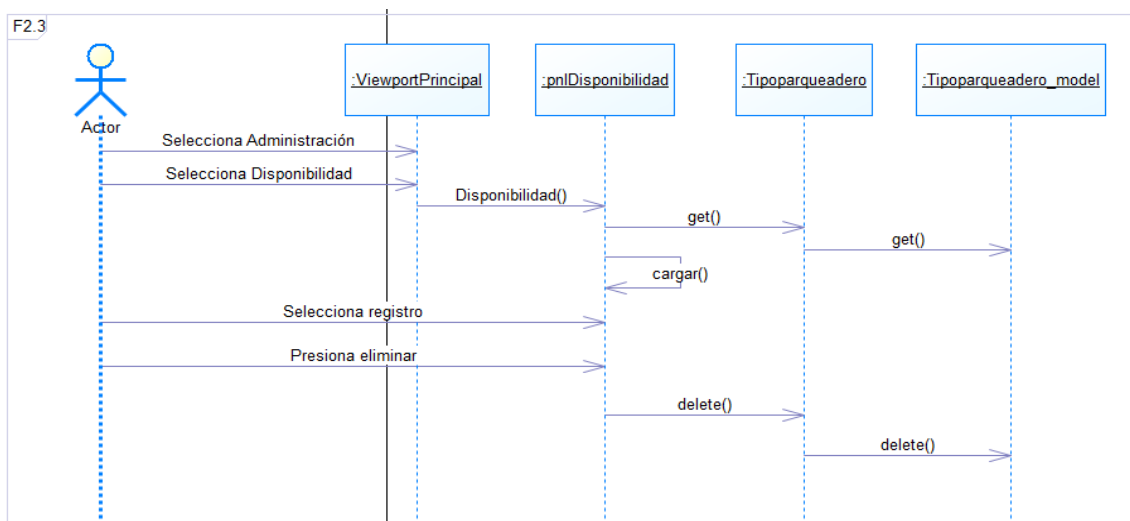


Figura 46: F2.4 Eliminación

### 3.3.2.4 Consulta

Se puede consultar los distintos tipos de parqueaderos que se encuentran disponibles, por tal razón, el actor debe ingresar en la pantalla de disponibilidad y se realiza una petición para traer los parqueaderos y presentarlos en pantalla (Ver Figura 47).

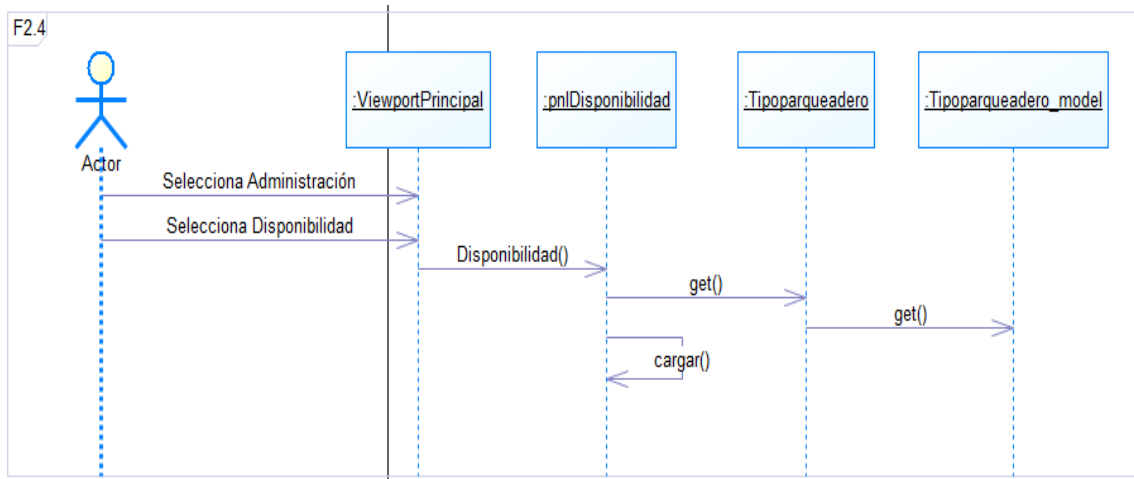


Figura 47: F2.4 Consulta

## 3.3.3 Administración de periodos

### 3.3.3.1 Ingreso

Los periodos son ingresados para cada nuevo semestre por lo que esta actividad comprende de distintas acciones representadas en la Figura 48.

El actor ingresa a la pantalla de periodos y se ejecuta la función para cargar los registros en pantalla, además al presionar el botón nuevo se presenta el formulario de periodos, una vez se encuentren llenos los campos requeridos el usuario presiona guardar, se verifica y guarda el nuevo periodo.

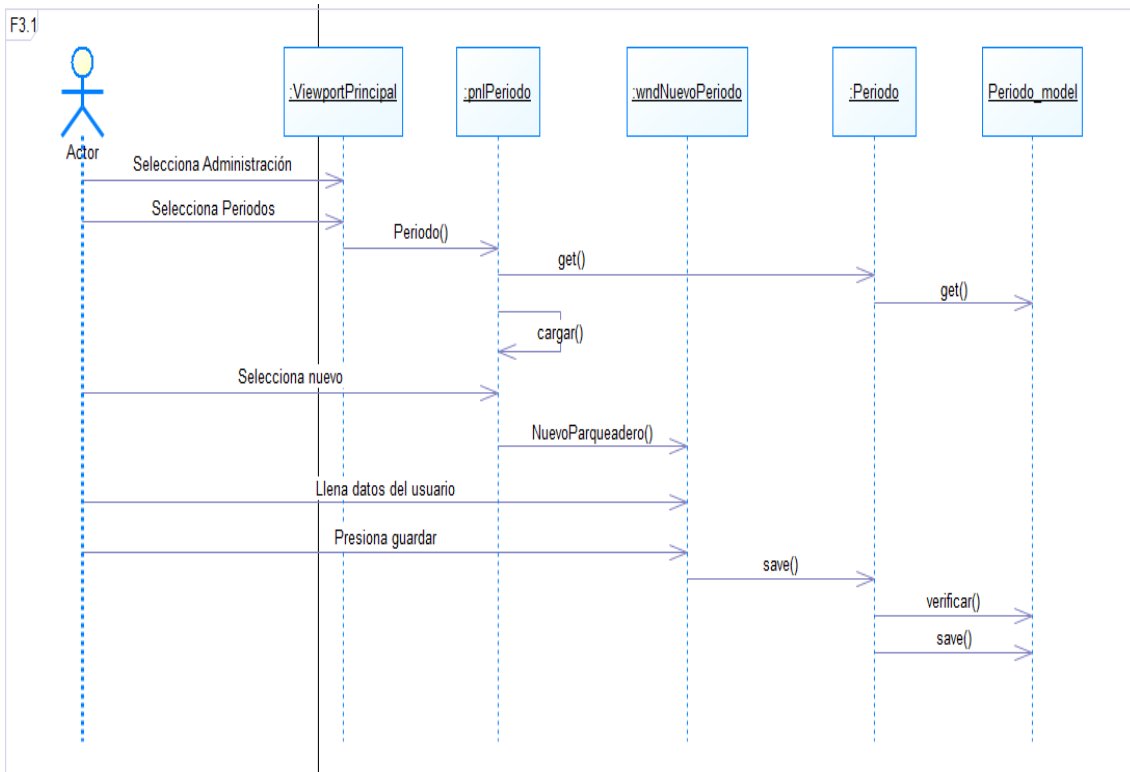


Figura 48: F3.1 Ingreso

### 3.3.3.2 Modificación

La modificación de periodos permite ingresar datos a registros existentes, de modo que no sean estáticos, por lo que es necesario la interacción entre las distintas capas del sistema.

En este proceso, el actor al situarse en la pantalla de periodos selecciona el registro a editar y se habilita la edición al mismo, una vez haya modificado los campos presiona actualizar y se ejecutan las funciones de modificación que cambian la información del registro como se puede visualizaren la Figura 49.

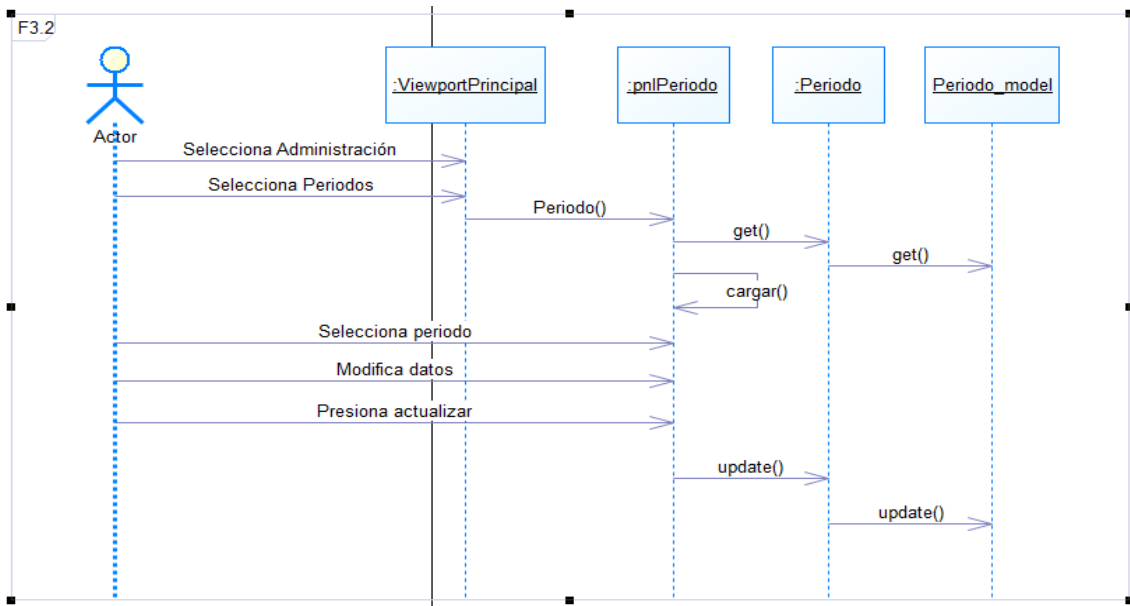


Figura 49: F3.2 Modificación

### 3.3.3.3 Desactivado

El desactivado permite que no se haga un eliminado físico de la base de datos, sino solo un desactivado, de modo que los datos ingresados se mantengan históricamente. Sin embargo, solo puede haber un periodo activo el cual se actualiza semestralmente.

Una vez el actor se encuentre en la pantalla de periodos selecciona el registro a desactivar y cambia el campo de activo y presiona actualizar y se ejecuta la función que realiza la acción en la base de datos (Ver Figura 50).

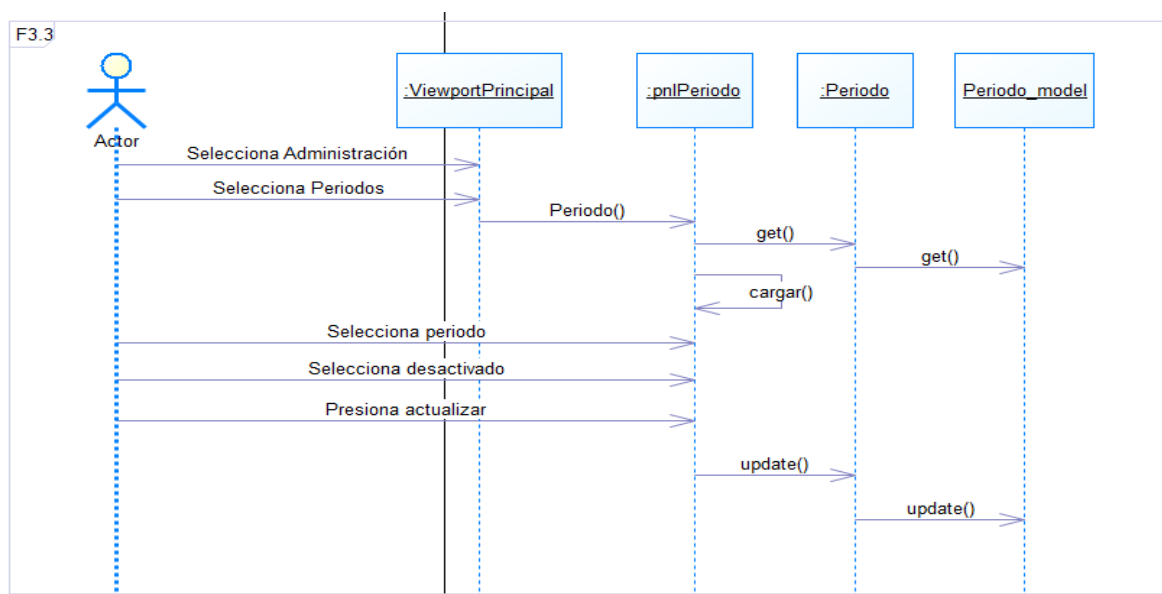


Figura 50: F3.3 Desactivado

### 3.3.3.4 Consulta

Es posible consultar los periodos existentes, lo cual se hace de la siguiente manera.

Según la Figura 51 el flujo comienza cuando el actor se sitúa en la pantalla principal y se dirige hacia periodos, misma que al cargarse realiza una petición para traer los registros correspondientes.

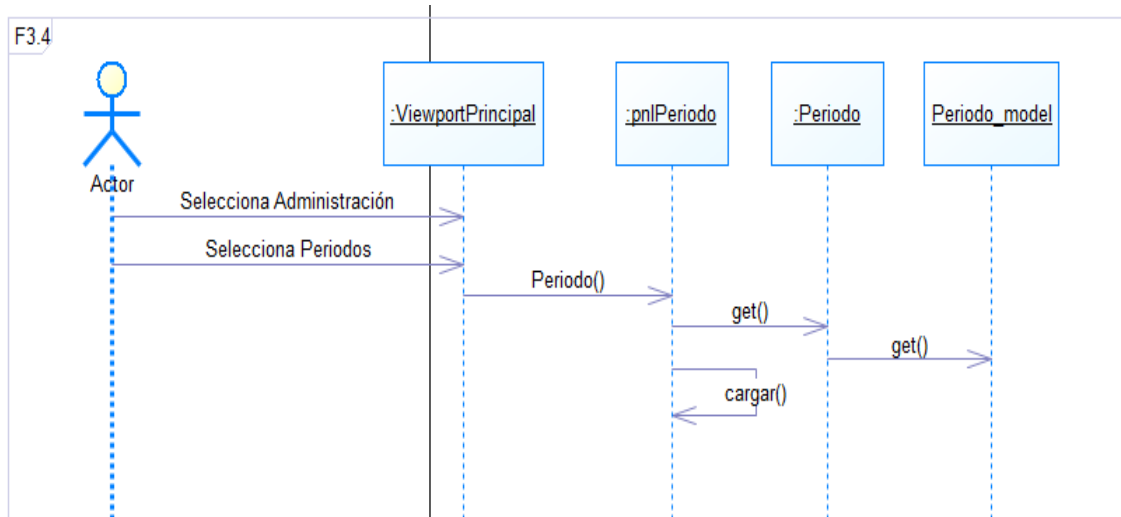


Figura 51: F3.4 Consulta

## 3.3.4 Administración de usuarios

### 3.3.4.1 Ingreso

El ingreso de usuarios es todos aquellos postulantes a los parqueaderos por lo que constantemente existen nuevos.

En primer lugar, el actor se encuentra en la pantalla principal, se dirige a la opción de usuarios y selecciona ingresar usuario lo que ejecuta la función para presentar un formulario con los campos correspondientes al usuario. Una vez se hayan llenado los campos requeridos el actor presiona guardar y se realiza la verificación y el guardado de los datos lo que se encuentra representado en la Figura 52.

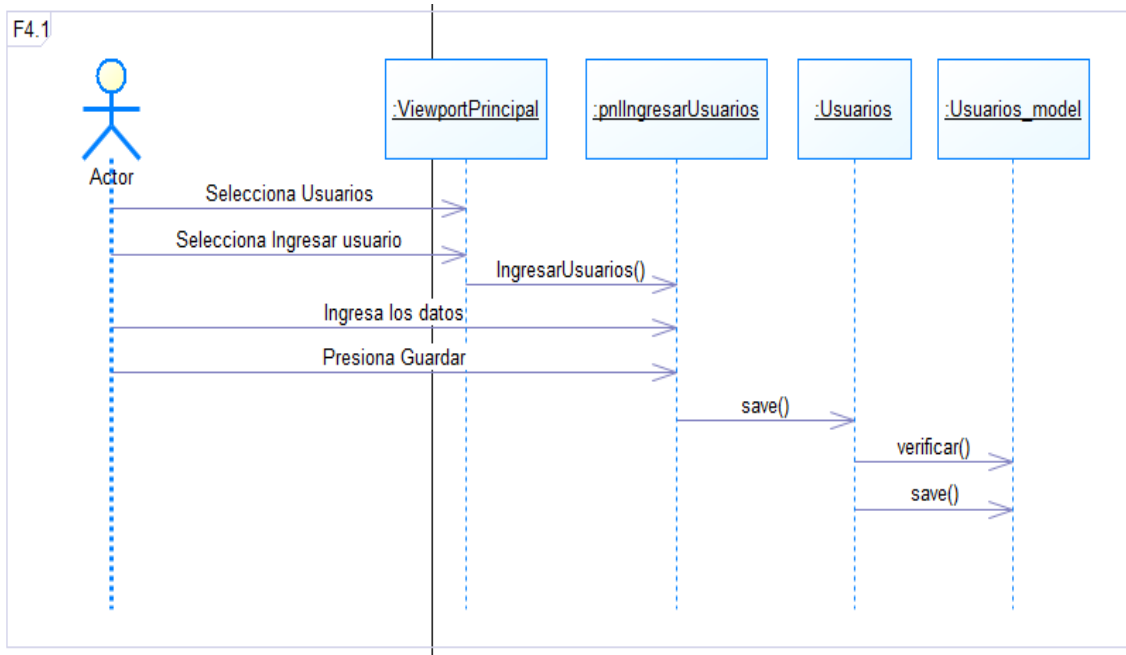


Figura 52: F4.1 Ingreso

### 3.3.4.2 Modificación

Mediante la modificación de usuarios es posible actualizar datos en su registro, para lo cual el usuario debe dirigirse a la opción de usuarios y seleccionar modificar usuario, lo que permite ejecutar la pantalla para realizar la actualización.

En este apartado, el actor ingresa la cedula de identidad del usuario al que desea modificar y presiona buscar, el sistema realiza una búsqueda por usuario, trae los datos y los presenta por pantalla, mismos que pueden ser editados. Cuando el actor presione guardar se ejecutan las funciones de guardado como se puede visualizar en la Figura 53.

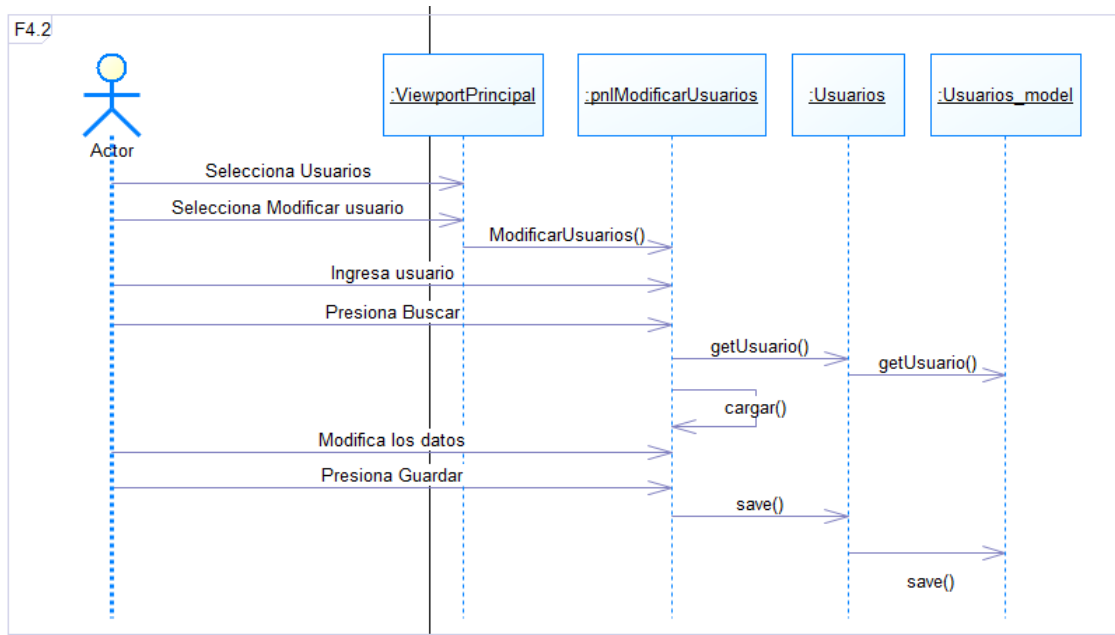


Figura 53: F4.2 Modificación

### 3.3.4.3 Consulta

Para la consulta de usuarios se representa en la Figura 54 que el actor debe situarse en la pantalla principal y dirigirse a la consulta de reportes, donde al cargar la pantalla ingresa el usuario a consultar y presiona buscar, lo que ejecuta funciones para traer las coincidencias y presentarlas por pantalla.

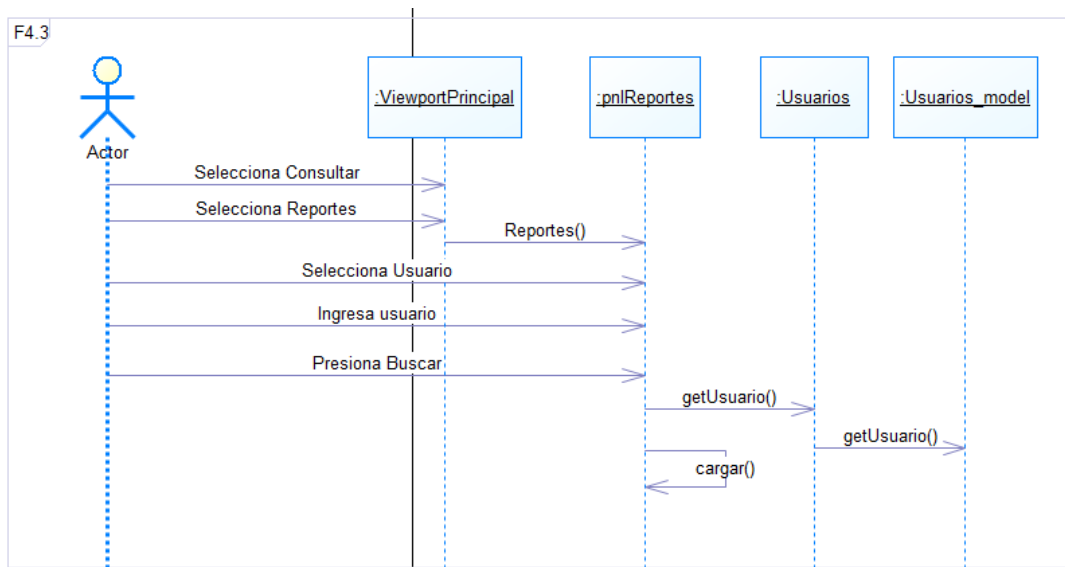


Figura 54: F4.3 Consulta

### 3.3.5 Administración de bloqueos

#### 3.3.5.1 Ingreso

El ingreso de bloqueos es importante cuando los usuarios han hecho un mal uso de las instalaciones o recursos que brinda el servicio de parqueaderos, por tal razón, el actor se dirige a la pantalla de bloqueos y se ejecuta la función que despliega el componente de bloqueos, además se realiza acciones para presentar los registros ya ingresados. Posteriormente el actor presiona nuevo y se presenta el formulario para ingresar un nuevo bloqueo. Una vez llenos los campos se presiona guardar y se realiza las verificaciones de datos y el guardado de los mismos (Ver Figura 55).

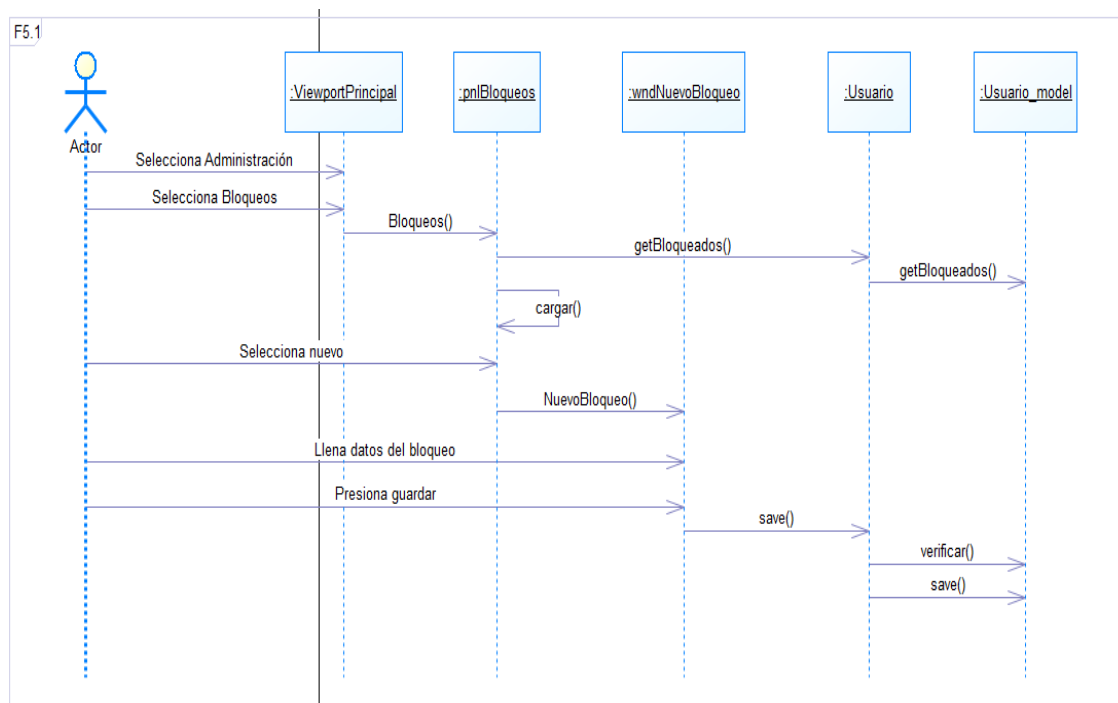


Figura 55: F5.1 Ingreso

#### 3.3.5.2 Modificación

Para la edición de un bloqueo se debe dirigir hacia la pantalla de bloqueos, misma que al renderizarse carga de la base de datos los bloqueos ya ingresados. Una vez el actor seleccione un bloqueo se habilita la edición que para finalizar presiona actualizar y se ejecutan las funciones de actualizado que se encuentra representado en la Figura 56.

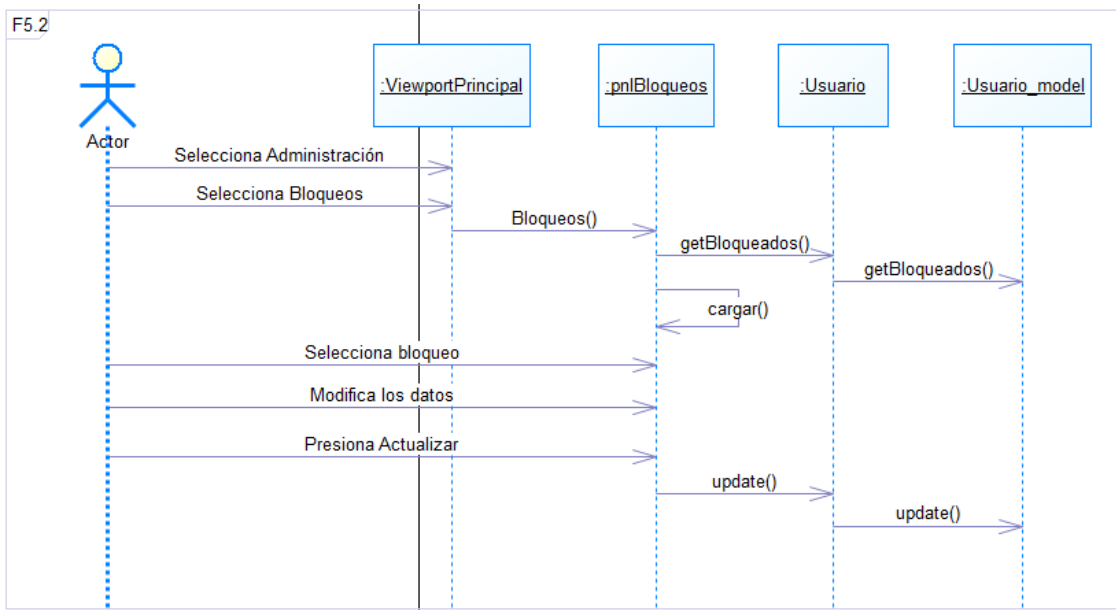


Figura 56: F5.2 Modificación

### 3.3.5.3 Eliminación

Según la Figura 57 la eliminación de un bloqueo se permite que el usuario al cual se le asignó dicho bloqueo pueda volver a participar en el sorteo, es por esto que, el actor se dirige a la pantalla de bloqueo, la cual carga los bloqueos existentes y permite que el actor seleccione el registro a retirar y presiona eliminar, lo que ejecuta la función de eliminado.

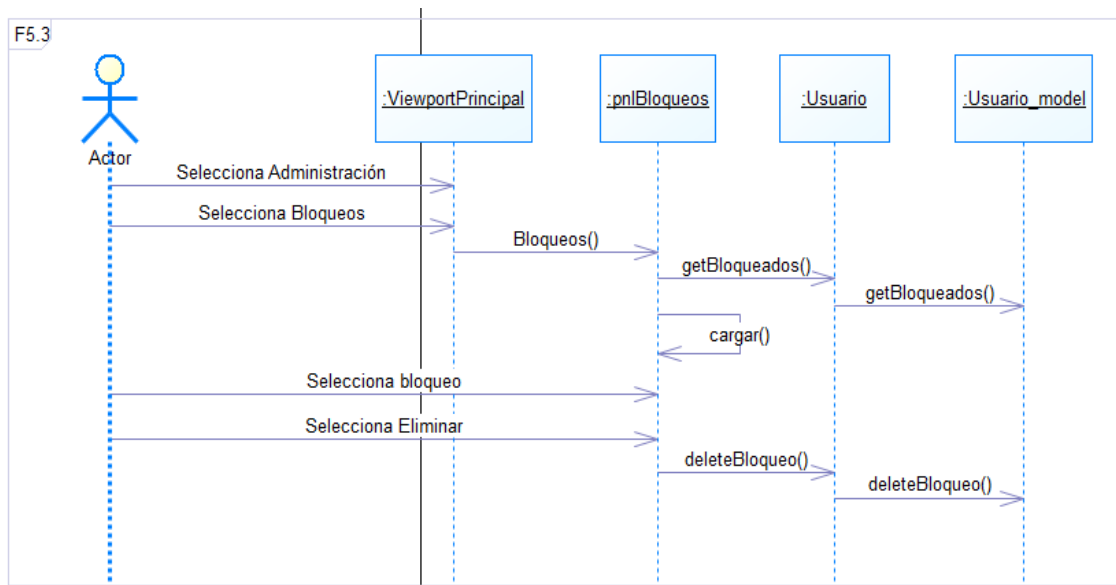


Figura 57: F5.3 Eliminación

### 3.3.5.4 Consulta

Se permite visualizar los bloqueos existentes según el flujo de la Figura 58, para lo cual el actor al abrir la pantalla de bloqueos ejecuta la función de carga que permite traer los registros de la base de datos y presentarlos en pantalla.

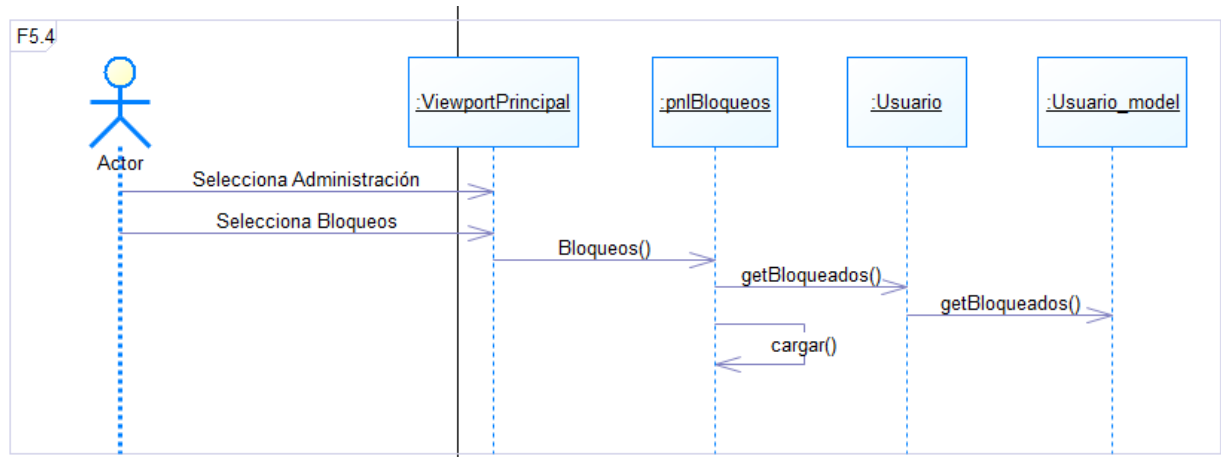


Figura 58: F5.4 Consulta

### 3.3.6 Administración de solicitudes

#### 3.3.6.1 Ingreso

Las solicitudes son aquellas que contienen la información sobre la petición de un parqueadero, las mismas que son ingresadas a través del siguiente proceso.

En primer lugar, el usuario se dirige a la pantalla de nueva solicitud, donde ingresa la cédula del estudiante a quien se le asignará la solicitud, y presiona crear, lo que ejecuta una función para traer la información del usuario. Luego llena el formulario y presiona guardar lo que conlleva a la verificación y el guardado de los datos representados en la Figura 59.

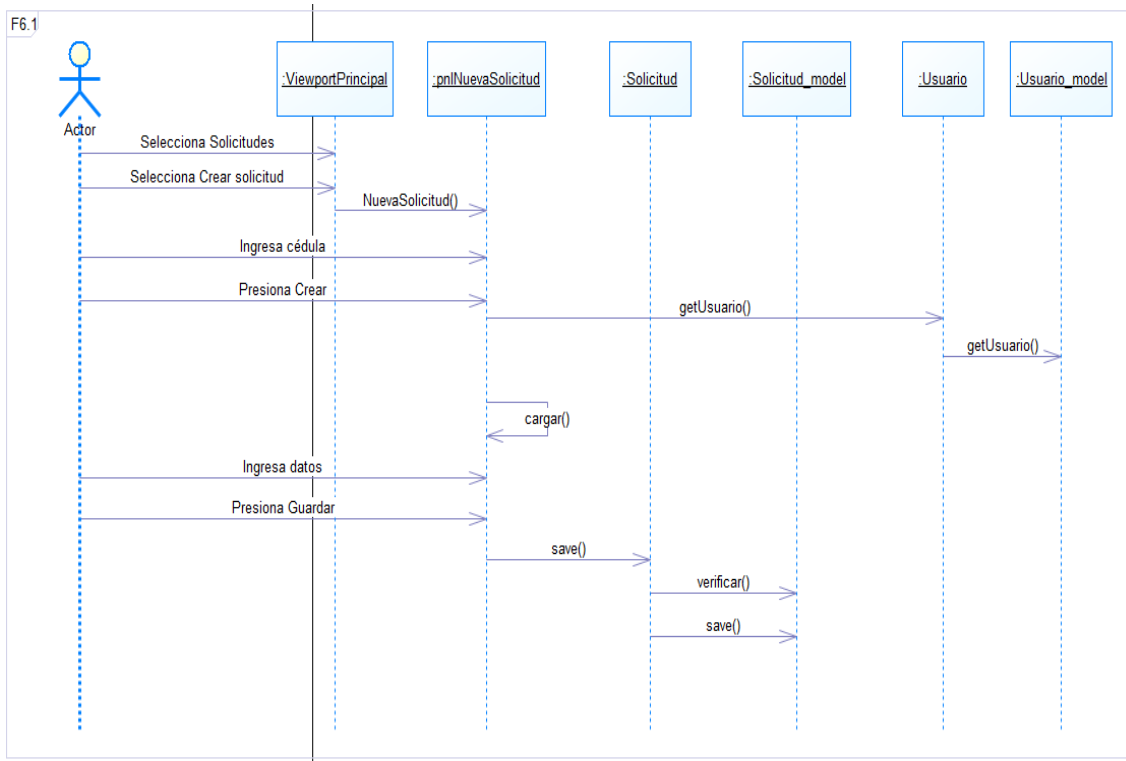


Figura 59: F6.1 Ingreso

### 3.3.6.2 Modificación

Mediante las modificaciones se puede actualizar datos de una solicitud existente, para lo cual el flujo comienza con el ingreso del usuario a la pantalla de solicitud, donde se ingresa la cédula y se ejecuta funciones para traer las coincidencias y presentar estos registros. Al seleccionar la solicitud se habilita la edición, para terminar, se presiona actualizar y se realiza la modificación de los datos (Ver Figura 60).

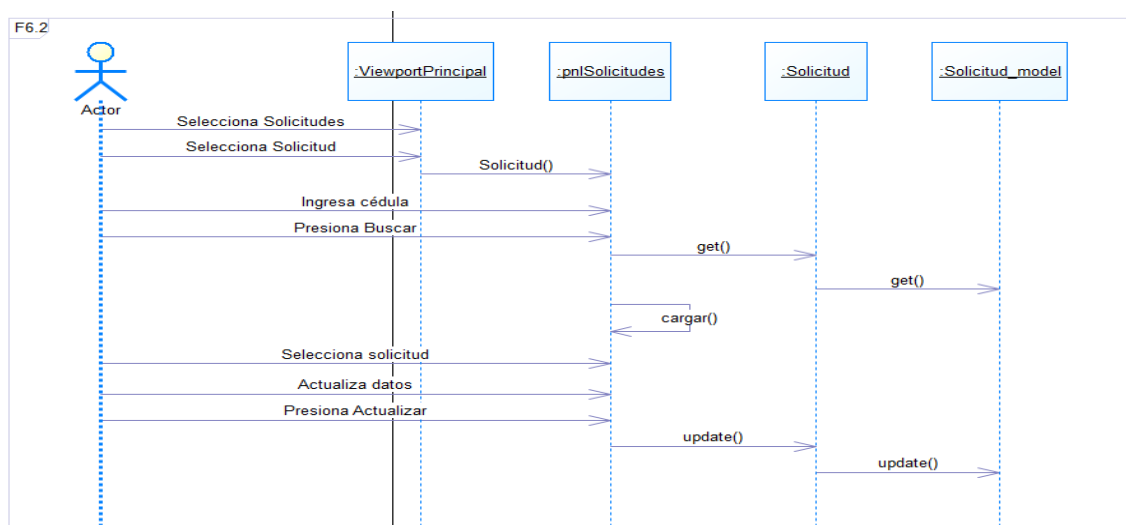


Figura 60: F6.2 Modificación

### 3.3.6.3 Consulta

Según la Figura 61 es posible visualizar las solicitudes el actor se dirige a la pantalla de solicitud, luego se ingresa la cédula y presiona buscar, lo que ejecuta la función para traer los registros y presentarlos en pantalla.

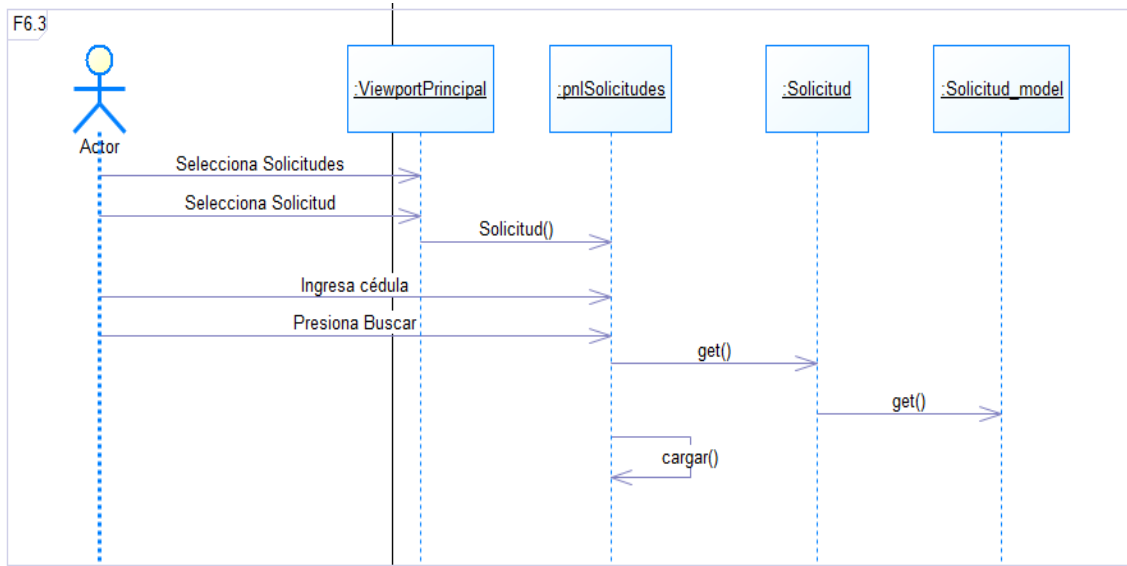


Figura 61: F6.3 Consulta

### 3.3.6.4 Ingreso Intranet

El ingreso mediante el Intranet es el que más frecuenta la comunidad universitaria. Una vez situado en la opción de parqueaderos se ingresa los datos y se ejecuta la función de guardado representado en la Figura 62.

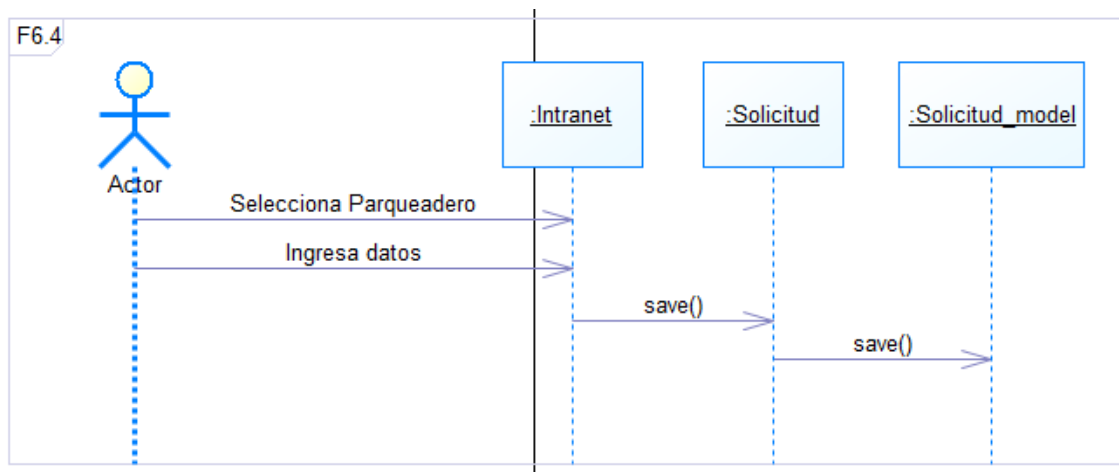


Figura 62: F6.4 Ingreso Intranet

### 3.3.7 Sorteo

El sorteo de las solicitudes se realiza mediante un proceso de actualización de solicitudes, por tal razón, el actor al ingresar a la pantalla de sorteo se ejecuta

una función para traer el periodo actual, de modo que se conozca para que semestre se está realizando el sorteo. Posteriormente se presiona sortear y se confirma dicha acción lo que ejecuta las funciones de sorteo que se pueden visualizar en la Figura 63.

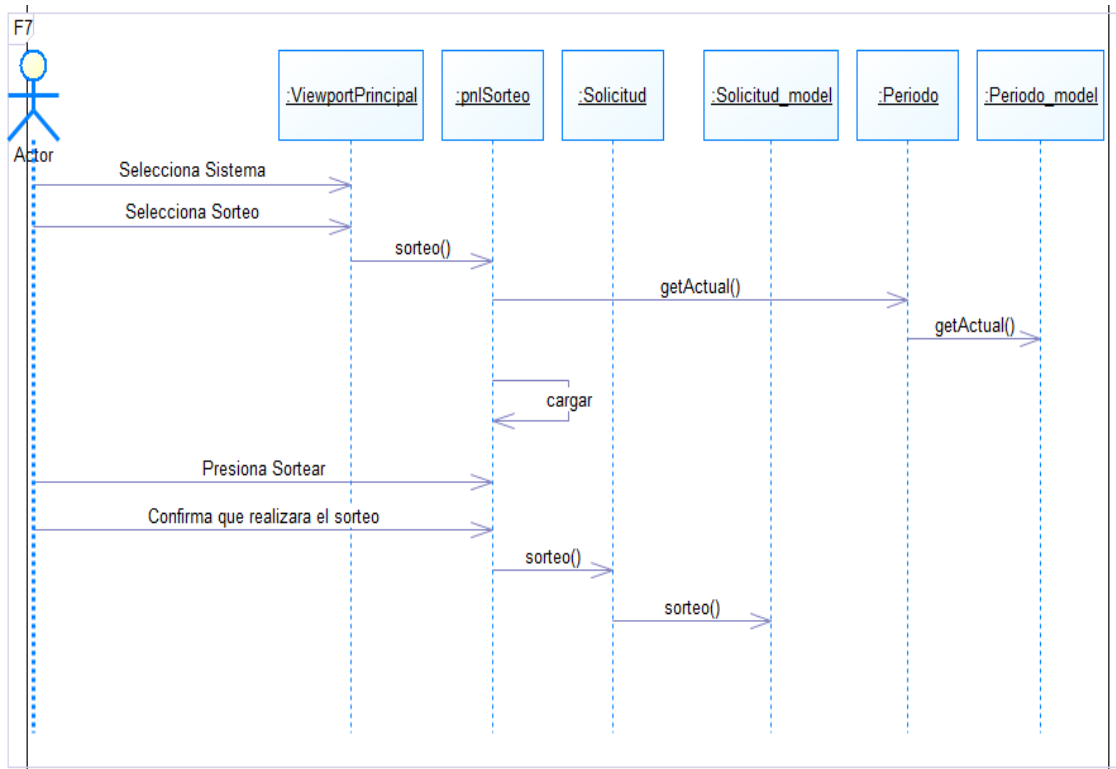


Figura 63: F7 Sorteo

### 3.3.8 Notificación

Mediante las notificaciones el sistema envía correos a los estudiantes que resultaron favorecidos del sorteo.

En primer lugar, el actor se dirige a la pantalla de notificaciones y carga el periodo actual, una vez se presione el botón enviar notificaciones es presentado un mensaje de confirmación que al aceptar ejecuta las funciones de envío de correos (Ver Figura 64).

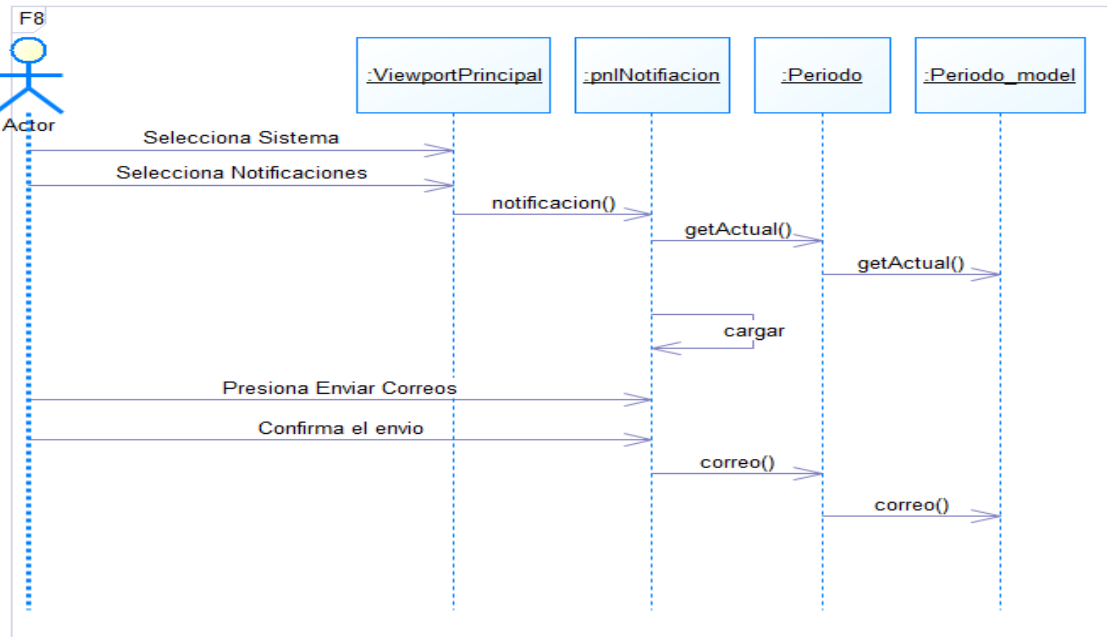


Figura 64: F8 Notificación

### 3.3.9 Envío de comprobantes

El envío de comprobantes inserta la información de las solicitudes en una tabla con la estructura que lee el sistema para generar los comprobantes de pago para los estudiantes.

El actor al situarse en la pantalla de enviar comprobantes se carga el periodo para el cual se está realizando el proceso, donde presiona enviar comprobantes y confirma el envío. Estas acciones están representadas en la Figura 65 y ejecuta las funciones que van a mover la información a la tabla consulta factura.

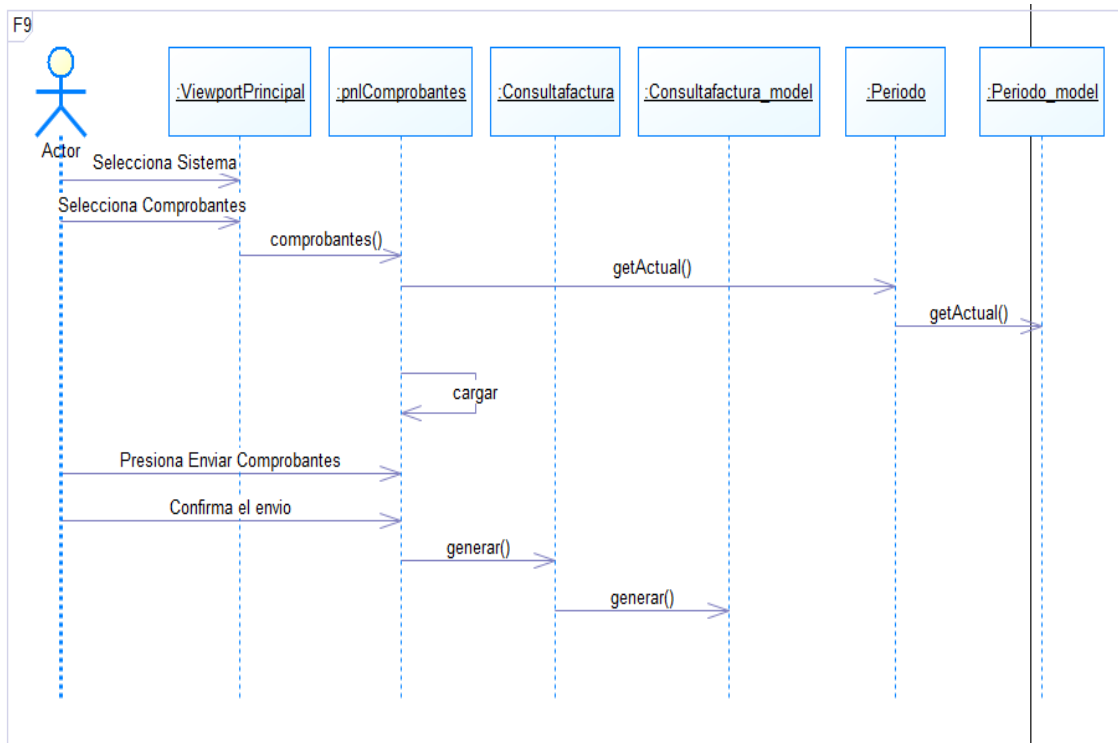


Figura 65: F9 Envío de comprobantes

### 3.3.10 Reportes

Mediante la reportería se puede descargar documentos Excel con la información que se presenta en pantalla, misma que se llena dependiendo de parámetros elegidos. A continuación, en la Figura 66 se encuentra la información descrita.

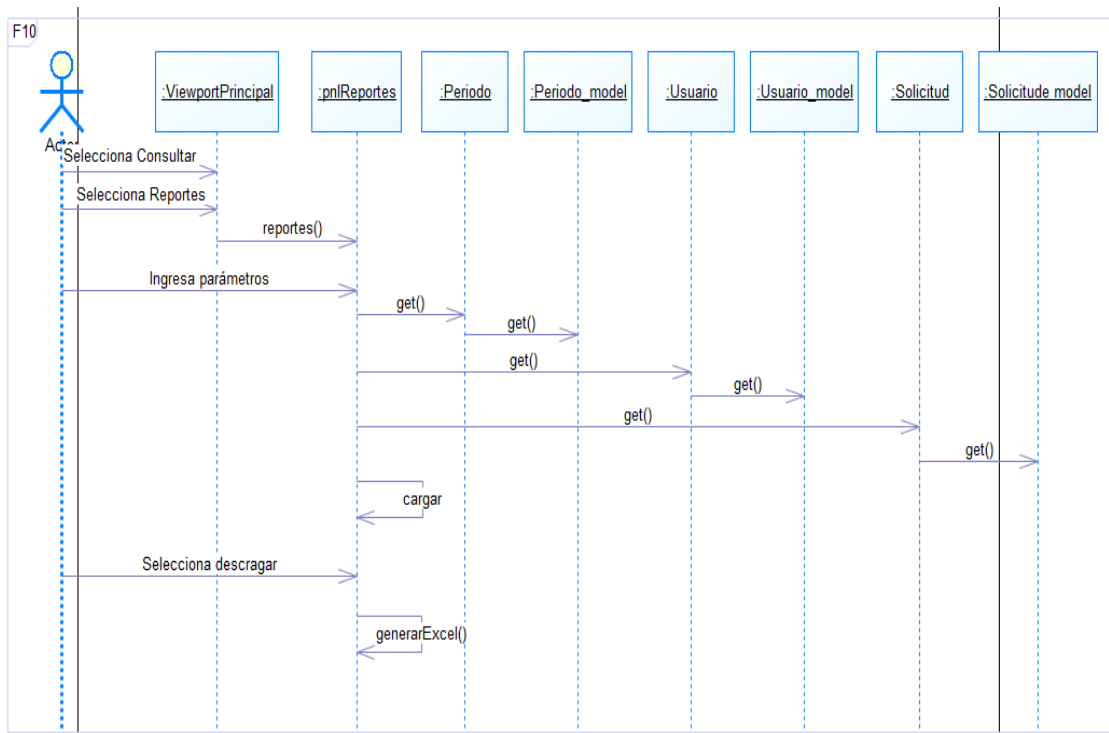


Figura 66: F10 Reportes

### 3.4 Diagrama Entidad Relación

Mediante el diagrama de entidad relación se determinan las dependencias que existen entre los distintos elementos del sistema, dando partida para la implementación de este proyecto (Ver Figura 67).

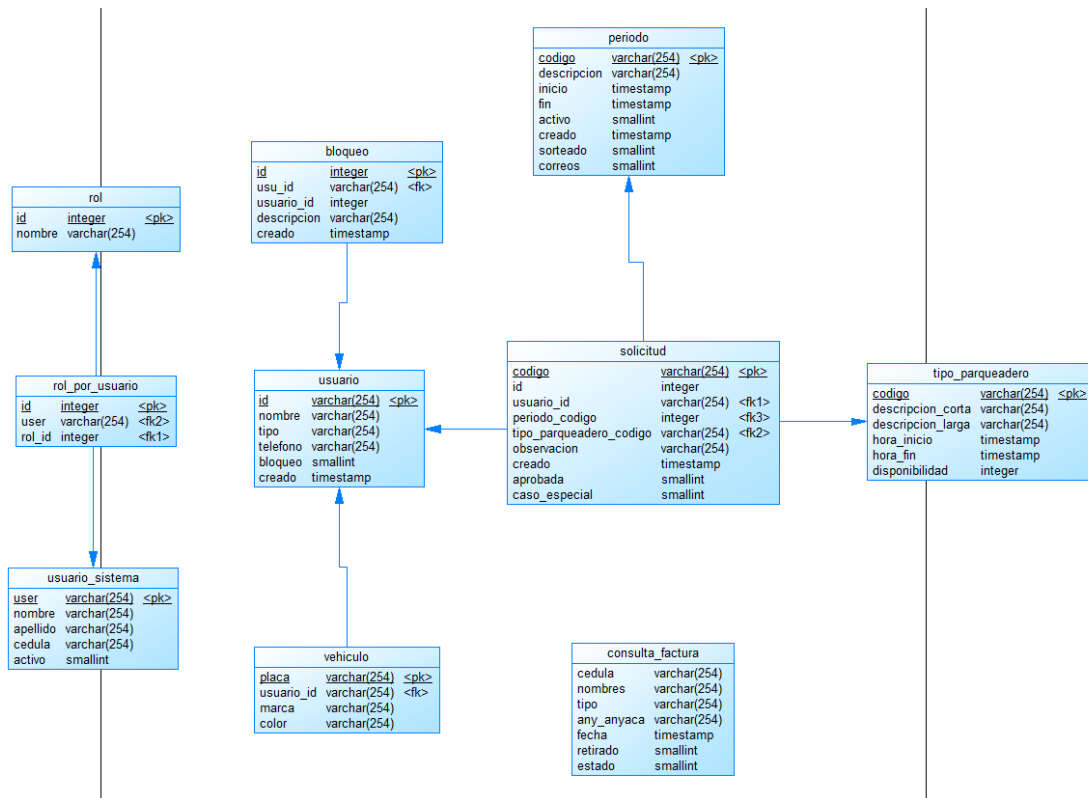


Figura 67: Diagrama entidad relación

### 3.5 Plan de pruebas de integración

Las pruebas de integración analizan las interfaces entre los distintos componentes o subsistemas, de modo que se asegure que son llamados cuando sea necesario y que los datos que transmiten entre los mismos son los requeridos por el otro módulo.

En este proyecto se realizará en el orden establecido de las funcionalidades ya que varias contienen dependencias por lo que al momento de probar debe ser realizado de la misma forma dado que es el modo como el sistema debe funcionar con relación a otros módulos. Tomando en cuenta que dichas funcionalidades ya fueron probadas de manera individual.

## 4 Capítulo IV: Desarrollo

El desarrollo del sistema comprende el levantamiento del ambiente donde se va a implementar, además de la codificación de frontend y backend.

Para la preparación del ambiente se comienza con la instalación Wamp, el cual utiliza por defecto una base de datos MySQL, donde una vez instalado se lo ejecuta y se debe verificar que se inicialicen todos sus servicios.

Posteriormente para el backend se instaló PHP Storm en su versión más actualizada, para poder utilizar este software es importante crear una cuenta en JetBrains y solicitar una licencia de estudiantes, la cual habilitará el sistema por un tiempo limitado. Sin embargo, una vez transcurrido este tiempo se puede volver a renovar la licencia de estudiante.

Para la implementación del frontend se utiliza la herramienta Sencha Architect en su versión 4.2.4, donde se activa el software mediante la licencia de este producto.

En la Figura 68 se puede visualizar que la primera etapa del desarrollo comienza con el Login del sistema, el cual se conecta con una base de datos donde constan todos los usuarios activos de la comunidad universitaria, una vez validado esto se realiza otra comprobación hacia la base de datos del sistema de modo que tenga acceso al mismo.



Usuario:

Contraseña:

Aceptar

Desarrollado por la Dirección Informática de la PUCE

Figura 68: Login

Posteriormente en la Figura 69 se representa el desarrollo de los usuarios del sistema, mismos que tienen acceso a realizar acciones en el sistema.

Pontificia Universidad Católica del Ecuador

PUCE ARQ

Administración > Usuarios del sistema

Usuarios >

Solicitudes >

Consultar >

Sistema >

Salir >

Nuevo +

Usuarios					Roles	
User	Nombre	Apellido	Cedula	Activo	Rol	Asignado
pparedes9...	Paul	Paredes	05035567...	Sí	Administración	<input checked="" type="checkbox"/>
palmeida3...	Pablo	Almeida	17227725...	Sí	Usuarios	<input checked="" type="checkbox"/>
jjgonzalez	Juan Javier	Gonzalez	1	Sí	Solicitudes	<input checked="" type="checkbox"/>
cpavon020	Carlos	Pavon	0	Sí	Consultar	<input checked="" type="checkbox"/>
cmlopez	Marilyn	Lopez	0	Sí	Sistema	<input checked="" type="checkbox"/>
banguilo851	Bryan	Angulo	17250537...	Sí		

ANGULO CAMUENDO BRYAN LENIN

Desarrollado por la Dirección Informática de la PUCE.

Figura 69: Usuarios del sistema

De igual manera la disponibilidad y periodos son tablas las cuales no dependen de otras, sino que sirven de catálogo para las demás (Ver Figura 70 y 71).

Pontificia Universidad Católica del Ecuador

PUCE ARQ

Administración > Periodos de solicitud

Usuarios >

Solicitudes >

Consultar >

Sistema >

Salir >

Nuevo +

Código	Descripción	Inicio	Fin	Activo
2020-02	Periodo académico	2020-11-28	2021-01-31	<input checked="" type="checkbox"/>

ANGULO CAMUENDO BRYAN LENIN

Desarrollado por la Dirección Informática de la PUCE.

Figura 70: Periodos

Pontificia Universidad Católica del Ecuador

PUCE ARQ

Administración

Disponibilidad

Nuevo +

Codigo	Descripcion Corta	Descripcion Larga	Disponibilidad
M	SEMPRESENCIAL	Parqueaderos Fin de Semana - SEM...	1
N	NOCHE	Parqueadero Nocturno (18:00 en a...	1
G	GIRÓN	Parqueadero Abierto Girón - 12 de ...	1
A	ABIERTO	Parqueadero Abierto PUCE, uso co...	2

ANGULO CAMUENDO BRYAN LENIN

Desarrollado por la Dirección Informática de la PUCE

Figura 71: Disponibilidad

Una vez concluido con la administración se procede a realizar lo que comprende a los usuarios, lo que es el ingreso y la modificación de los mismos, como se puede visualizar en la figura 72 y 73.

Pontificia Universidad Católica del Ecuador

PUCE ARQ

Administración

Usuarios

Ingresar usuario

Limpiar

Cédula:  Buscar

Nombre:

Teléfono:

Tipo:

Placa:

Marca:

Color:

Guardar

ANGULO CAMUENDO BRYAN LENIN

Desarrollado por la Dirección Informática de la PUCE

Figura 72: Ingreso usuarios

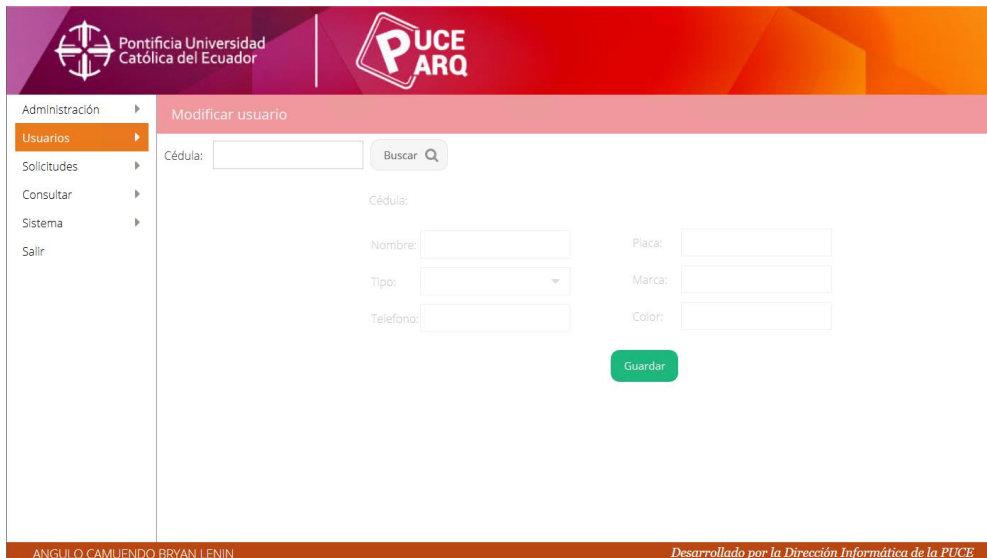


Figura 73: Modificación usuarios

De igual manera se procede a la implementación de la funcionalidad de las solicitudes (Ver Figura 74 y 75).

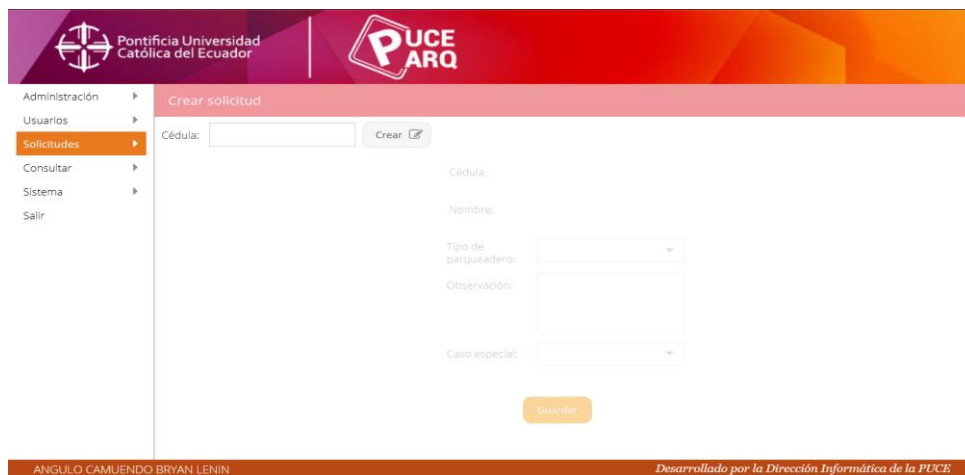


Figura 74: Ingreso solicitudes

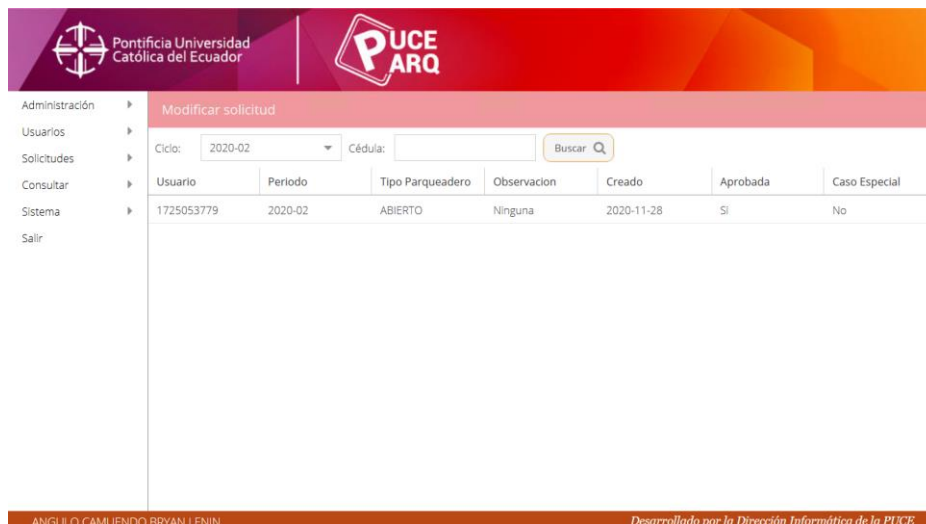


Figura 75: modificación y consulta

Una vez implementado las funcionalidades anteriores, como se indica en la Figura 76 se procede a realizar el bloqueo de los usuarios que realicen acciones no adecuadas en el ámbito de los parqueaderos.

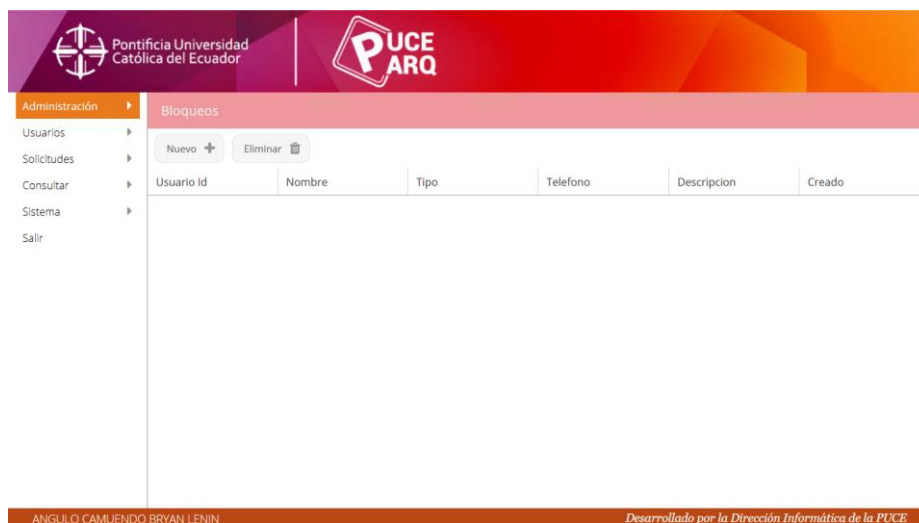


Figura 76: Bloqueos

Posteriormente se implementa las funcionalidades principales del sistema lo que comprende el sorteo, las notificaciones y el envío de los comprobantes (Ver Figuras 77, 78 y 79).

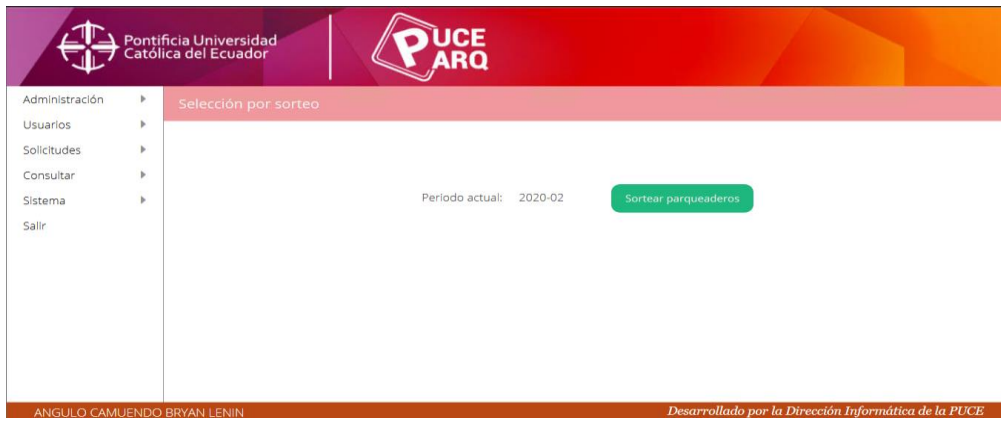


Figura 77: Sorteo

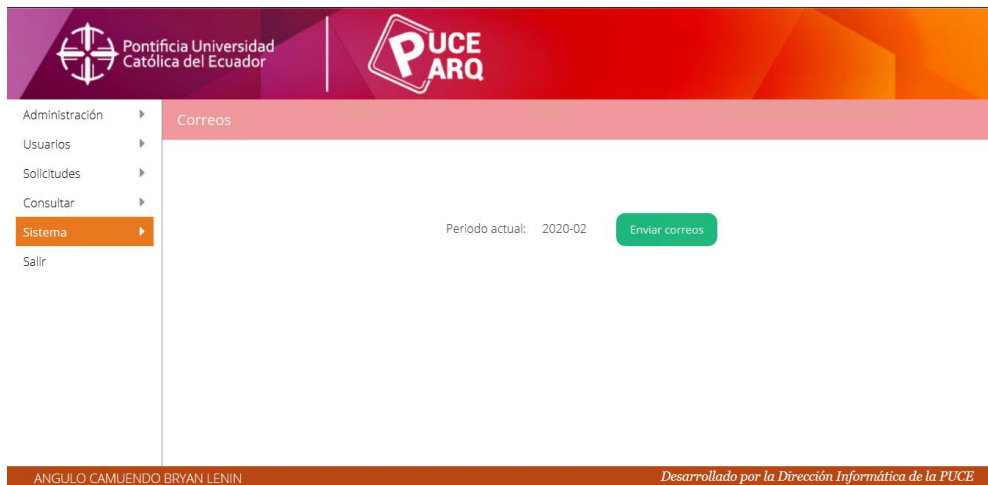


Figura 78: Notificaciones



Figura 79: Envío de comprobantes

Para finalizar se realiza la reportaría del sistema en la cual existen búsquedas por diferentes parámetros obteniendo así información entendible para el usuario, misma que puede ser descargada en formato xlsx, como se puede ver en la Figura 80.

Pontificia Universidad Católica del Ecuador

**PUCE**  
**PARQ**

Reportes

Solicitudes Usuario

Parqueadero:  Ciclo:  Aprobados:

Todos los parqueaderos:  Todos los ciclos:  No aprobados:

Usuario	Nombre	Teléfono	Placa	Periodo	Tipo Parqu...	Observacion	Creado	Aprobada
---------	--------	----------	-------	---------	---------------	-------------	--------	----------

ANGULO CAMUENDO BRYAN LENIN *Desarrollado por la Dirección Informática de la PUCE*

Figura 80: Reportes

## 5 Capítulo V: Pruebas

Las pruebas son realizadas en base a las ya descritas en requerimientos como lo son las pruebas por componentes mismas que se cumplió correctamente. Además, las que se encuentran descritas en diseño que comprenden la interacción entre distintos módulos del sistema, siendo las mismas favorables.

Primero se procedió a verificar el Login ya que es donde se inicia el sistema ingresando valores para validar su funcionamiento. Se puede visualizar en la Figura 81 a continuación.



Usuario:

Contraseña:

Desarrollado por la Dirección Informática de la PUCE

*Figura 81 Login*

Luego para verificar la pantalla de usuarios se realiza la inserción de usuarios con acceso al sistema, en el cual al llenar el formulario permite verificar esta funcionalidad como se indica en la Figura 82.



Nuevo usuario del sistema

Usuario:

Nombre:

Apellido:

Cédula:

Activo:

*Figura 82 Crear Usuario del Sistema*

Como se indica en las Figuras 83 y 84, se crean los parqueaderos y periodos los cuales sirven de catálogos para las siguientes verificaciones.

Nuevo período

Código: 2021-01

Descripción: Nuevo periodo

Inicio: 2021-04-25

Fin: 2021-04-25

Cancelar Guardar

Figura 83 Creación Período

Nuevo parqueadero

Código: ABIERTO

Descripción larga: PARQUEADERO DENTRO DEL CAMPUS

Descripción corta: ABIERTO

Disponibilidad: 150

Cancelar Guardar

Figura 84 Creación Parqueadero

Para la etapa la siguiente etapa en las Figuras 85 y 86 se crea un usuario y una solicitud para el mismo, tomando en cuenta que para ingresar una solicitud es necesario asignar un usuario ya creado.

Ingresar usuario

Limpiar

Cédula: 1725053779 Buscar

Nombre: .O CAMUENDO BRYAN LENIN

Telefono: 0987027450

Tipo: Est. Presencial

Placa: PDF3518

Marca: CHEVROLET

Color: GRIS

Guardar

Figura 85 Creación Usuario

Crear solicitud

Cédula:  Crear

Cédula: 1725053779

Nombre: ANGULO CAMUENDO BRYAN LENIN

Tipo de parqueadero: Presencial

Observación: no

Caso especial: No

Guardar

Figura 86 Creación Solicitud

Una vez se tiene todos los catálogos listos se procede a realizar el sorteo de las solicitudes para comprobar que todo se encuentre funcionando de la manera esperada, dando como resultado un mensaje que se ha realizado el sorteo con éxito como lo indica la Figura 87.

Selección por sorteo

Datos Guardados

El sorteo se ha realizado exitosamente

Sortear parqueaderos

Aceptar

Figura 87 Sorteo Exitoso

Al realizar el sorteo es posible enviar las notificaciones a los usuarios para lo cual se evidencia dicha funcionalidad en la Figura 88 con un correo que le llega al estudiante favorecido.

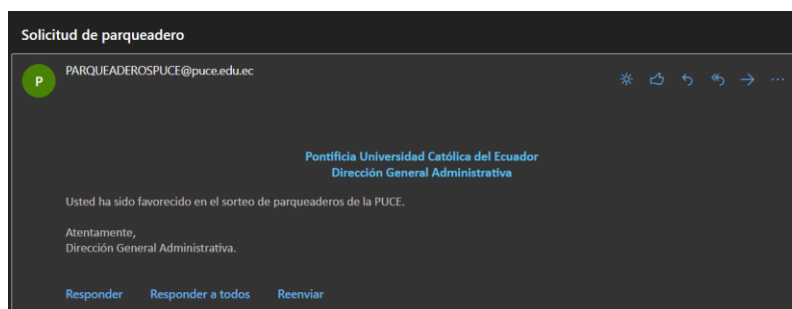
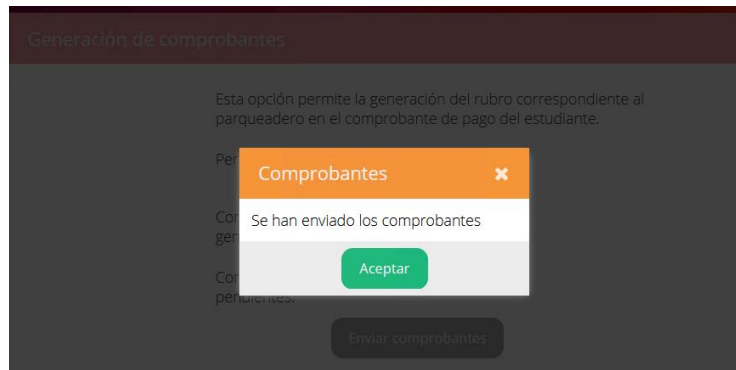


Figura 88 Correo

Y como paso final se verifica él envió de comprobantes por medio del botón que da como resultado un mensaje de envío exitoso como se puede visualizar en la Figura 89.



*Figura 89 Envío de comprobantes*

También se realizaron pruebas por un experto, en este caso se solicitó a la Dirección de Informática para que el sistema sea revisado, cumpla con las especificaciones realizadas y se desenvuelva de la manera correcta ante los distintos manejos por el usuario. Dando un resultado favorable, de tal modo que el sistema fue aprobado y puesto en producción (Ver Figura 90).

Quito, 27 de noviembre del 2020

Doctor  
Charles Escobar  
**DECANO DE LA FACULTAD DE INGENIERÍA**  
Presente,

De mi consideración,  
Por medio de la presente, se informa que el estudiante Bryan Lenin Angulo Camuendo de la Facultad de Ingeniería, Escuela de Sistemas ha desarrollado satisfactoriamente el sistema web acerca de la gestión de parqueaderos de la PUCE, mismo que ha sido probado y puesto en producción, dando como resultado que se encuentra funcionando correctamente y sobre el cual el estudiante realiza su trabajo de titulación.



Atentamente

Ing. Pablo Almeida  
**Desarrollador de Software - Departamento de  
Desarrollo de Software PUCE**

**C.I:** 1722772553  
**Teléfono:** 0999861331

*Figura 90: Pruebas*

## 6 Capítulo VI: Manual de usuario

### 6.1 Manual de usuario

El manual de usuario esta realizado bajo los estándares de manuales que se manejan en la Dirección de Informática.

#### 6.1.1 Introducción

El sistema del Puceparq cuenta con diferentes módulos que permiten al usuario crear, consultar, actualizar y eliminar la información manejada, además realizar el sorteo, enviar los comprobantes y correos.

#### 6.1.2 Módulos/Componentes

##### 6.1.2.1 Login

El módulo de login permite al usuario ingresar al sistema (Ver Figura 91).



Figura 91: Login

Para ingresar se debe colocar el nombre de usuario y contraseña que se utiliza para acceder a los diferentes sistemas de la universidad (Correo institucional). Si existe un error en el nombre de usuario o contraseña o el usuario no tiene permisos para acceder al sistema se muestra el error de la Figura 92.

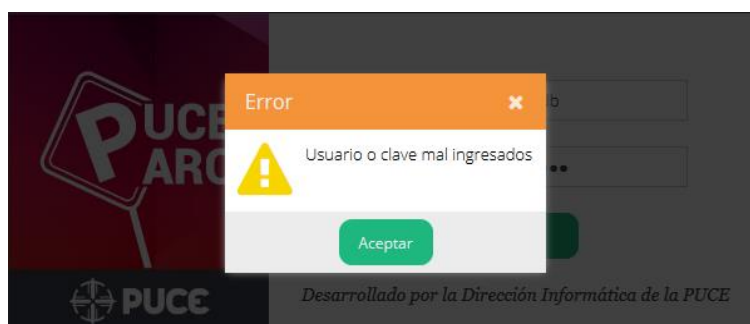


Figura 92: Error login

### 6.1.2.2 Página principal

Una vez ingresada la información se muestra la página principal evidenciada en la Figura 93.



Figura 93: Página principal

La página principal cuenta con cinco botones para manejar la información del sistema y un botón para cerrar la sesión debajo de los mismos. En la barra inferior se indica el nombre del usuario activo.

Al seleccionar los botones muestra un menú con las diferentes opciones como se puede ver en la Figura 94.

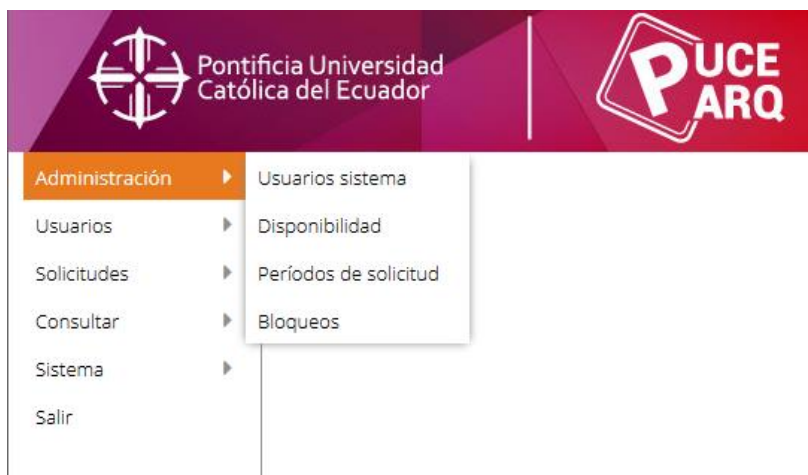


Figura 94: Submenú

Al acceder a una de las opciones se abre una pantalla con la información deseada.

En general cada ventana tiene su título, los botones para administrar y una lista con la información (Ver Figura 95).

Codigo	Descripcion
2019-01	Test
2020	

Figura 95: Ventana

### 6.1.2.3 Manipulación de la información

El sistema permite crear, modificar, consultar o eliminar usuarios, solicitudes y vehículos. También procesos como sortear, enviar comprobantes y correos.

Si existe un error en alguno de los campos al momento de ingresar la información dicho campo va a marcarse con un contorno rojo si se coloca el mouse sobre el cuadro se indica cual es el error relacionado.

### 6.1.2.4 Administración

#### 6.1.2.4.1 Crear un usuario del sistema

Como lo indica la Figura 96 para crear un usuario del sistema se deben realizar los siguientes pasos:

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Usuarios sistema**.
3. Presione el botón **Nuevo**.
4. Se presentará una nueva ventana con los campos a llenar (si se selecciona en el campo activo No, el usuario no tendrá acceso al sistema, esto se podrá modificar posteriormente).
5. Presione el botón **Guardar**.

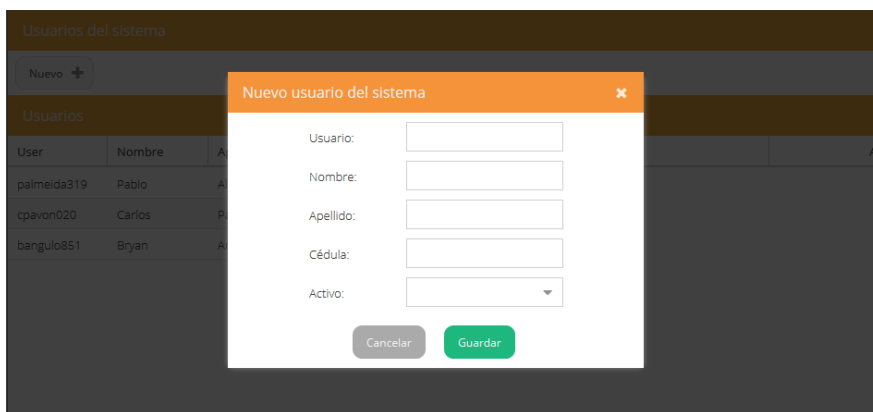


Figura 96: Crear usuario del sistema

Se debe tener en cuenta que si se intenta ingresar un usuario sin haber llenado todos los campos saldrá un mensaje de error como el de la Figura 97.

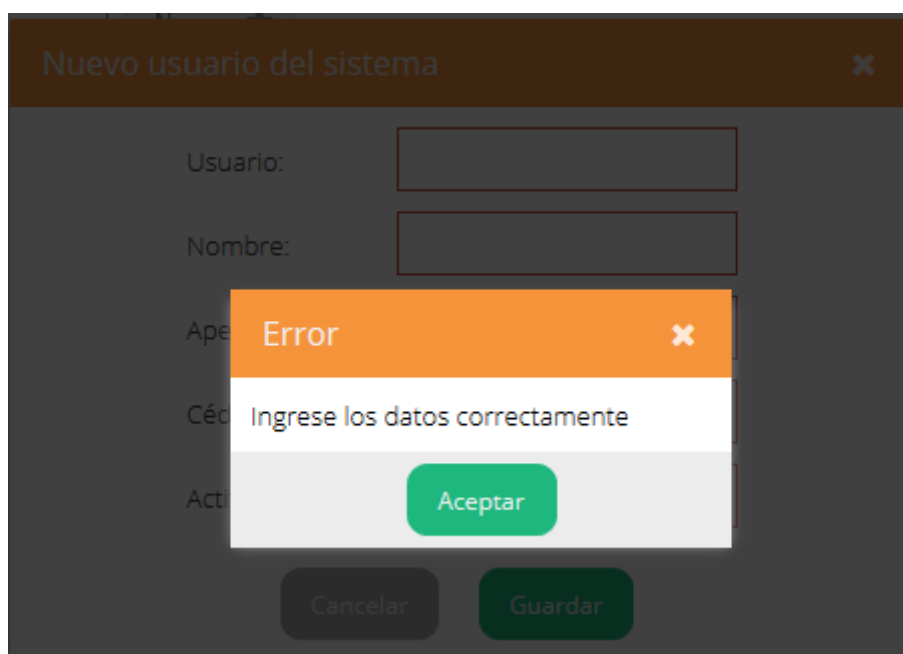


Figura 97: Error ingreso usuario sistema

#### 6.1.2.4.2 Modificar roles a un usuario del sistema

Para modificar los roles de un usuario en el sistema se deben realizar los siguientes pasos (Ver Figura 98):

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Usuarios sistema**.
3. En la tabla Usuarios debe seleccionar a quien desea asignar roles.
4. Se presentará en la tabla de Roles los accesos que el usuario tiene, representados por un checkbox aprobado.
5. Doble clic sobre el rol.

## 6. Presione el botón **Actualizar**.

Usuarios					Roles	
User	Nombre	Apellido	Cedula	Activo	Rol	Asignado
palmeida319	Pablo	Almeida	1722772553	Sí	Administración	<input checked="" type="checkbox"/>
cpavon020	Carlos	Pavon	0	Sí	Usuarios	<input checked="" type="checkbox"/>
bangulo851	Bryan	Angulo	1725053779	Sí	Solicitudes	<input checked="" type="checkbox"/>
					Consultar	<input type="button" value="Actualizar"/> <input type="button" value="Cancelar"/>
					Sistema	<input type="button" value="Actualizar"/> <input type="button" value="Cancelar"/>

Figura 98: Modificar roles

### 6.1.2.4.3 Modificar a un usuario del sistema

Para modificar a un usuario del sistema se deben realizar los siguientes pasos representados en la Figura 99.

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Usuarios sistema**.
3. Doble clic sobre el usuario a modificar.
4. Presione **Actualizar**.

Usuarios					
User	Nombre	Apellido	Cedula	Activo	
palmeida319	Pablo	Almeida	1722772553	Sí	
cpavon020	<input type="text" value="Carlos"/>	<input type="text" value="Pavon"/>	<input type="text" value="0"/>	<input type="text" value="Sí"/>	
bangulo851	Bryan	Angulo			
					<input type="button" value="Actualizar"/> <input type="button" value="Cancelar"/>

Figura 99: Modificar usuario del sistema

### 6.1.2.4.4 Modificar la disponibilidad de los parqueaderos

Como lo indica la Figura 100 para modificar la disponibilidad de los parqueaderos se deben realizar los siguientes pasos:

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Disponibilidad**.
3. Doble clic sobre el parqueadero a modificar.
4. Presione el botón **Actualizar**.

Disponibilidad			
Codigo	Descripcion Corta	Descripcion Larga	Disponibilidad
M	SEMIPRESENCIAL	Parqueaderos Fin de Serr	1
N	NOCHE	Nocturno...	1
G	GIRÓN	Parqueadero Abierto G...	1
A	ABIERTO	Parqueadero Abierto P...	2

Figura 100: Disponibilidad de parqueaderos

#### 6.1.2.4.5 Crear un nuevo período

Para crear un nuevo período se deben realizar los siguientes pasos (Ver Figura 101):

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Períodos de solicitud**.
3. Presione el botón **Nuevo**.
4. Se presentará una nueva ventana con los campos a llenar.
5. Presione el botón **Guardar**.

Figura 101: Crear un período

Tomar en cuenta que si se intenta ingresar un periodo nuevo con el mismo código no se ingresara ya que solo debe existir uno por código.

#### 6.1.2.4.6 Modificar un período de solicitud

Para modificar un periodo de solicitud se deben realizar los siguientes pasos (Ver Figura 102):

1. Se debe ir a la sección de **Administración**.

2. Se desplegará opciones y deberá seleccionar **Períodos de solicitud**.
3. Doble clic sobre el período a modificar.
4. Presione el botón **Actualizar**.



Figura 102: Modificar un período

#### 6.1.2.4.7 Asignar un bloqueo

Para asignar un bloqueo se deben realizar los siguientes pasos como se encuentra representado en la Figura 103.

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Bloqueos**.
3. Presione el botón **Nuevo**.
4. Se presentará una nueva ventana.
5. Llenar el campo cédula y presionar **Buscar**.
6. Llenar los campos faltantes.
7. Presione el botón **Guardar**.

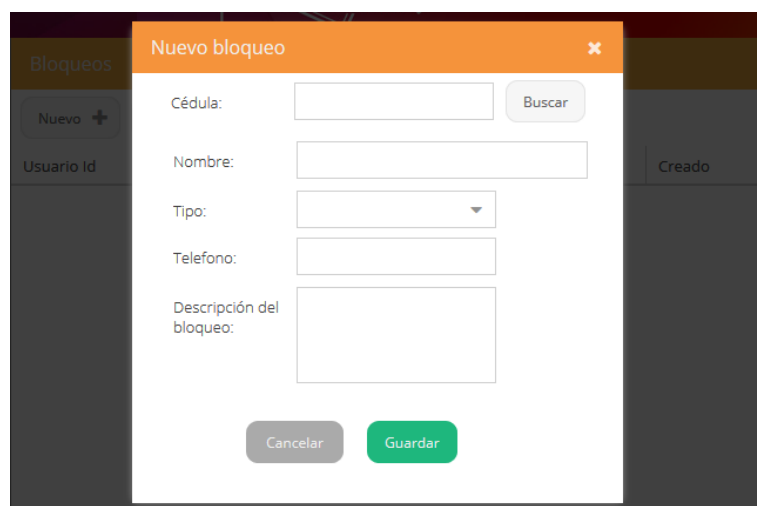
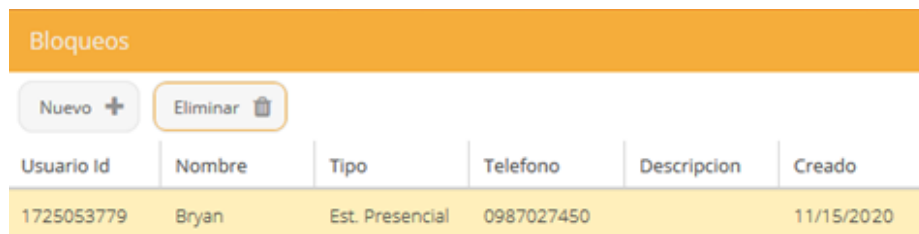


Figura 103: Bloqueo

#### 6.1.2.4.8 Quitar un bloqueo

En la Figura 104 representa como quitar un bloqueo para lo cual se debe realizar los siguientes pasos:

1. Se debe ir a la sección de **Administración**.
2. Se desplegará opciones y deberá seleccionar **Bloqueos**.
3. Seleccione al usuario al que se le va a quitar el bloqueo.
4. Presione el botón **Eliminar**.



Usuario Id	Nombre	Tipo	Telefono	Descripcion	Creado
1725053779	Bryan	Est. Presencial	0987027450		11/15/2020

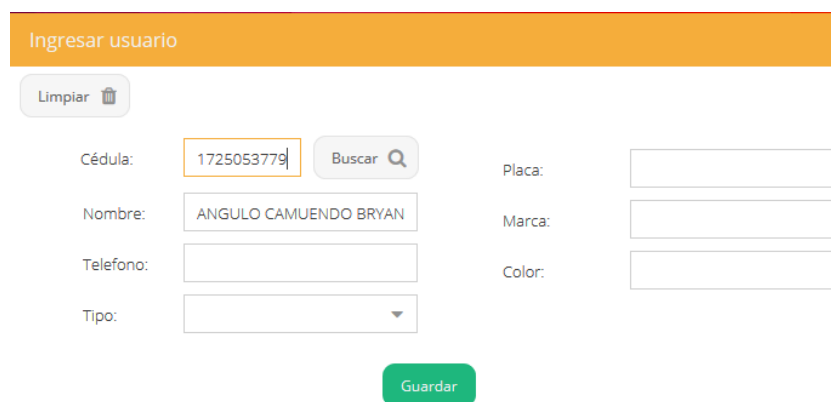
Figura 104: Quitar bloqueo

#### 6.1.2.5 Usuarios

##### 6.1.2.5.1 Ingresar un usuario

Para ingresar un usuario se debe realizar los siguientes pasos evidenciados en la Figura 105:

1. Se debe ir a la sección de **Usuarios**.
2. Se desplegará opciones y deberá seleccionar **Ingresar usuario**.
3. Llenar el campo cédula y presionar **Buscar**.
4. Se habilitarán los campos a llenar.
5. Presione el botón **Guardar**.



Limpiar

Cédula:  Buscar

Nombre:

Telefono:

Tipo:

Placa:

Marca:

Color:

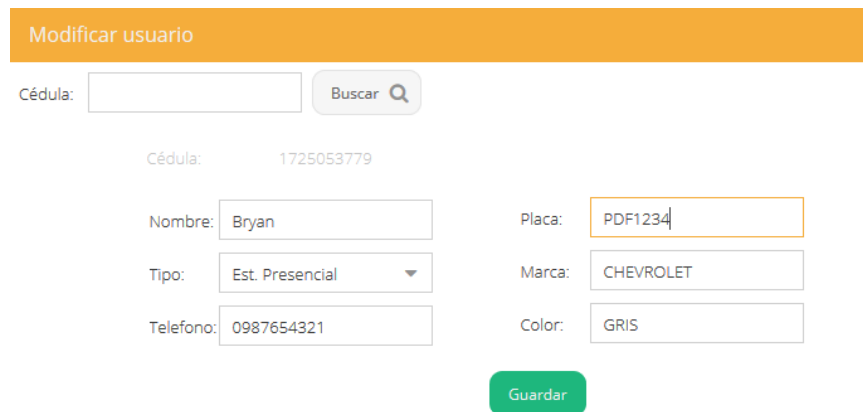
Guardar

Figura 105: Ingresar usuario


#### 6.1.2.5.2 Modificar un usuario

Para modificar un usuario se deben realizar los siguientes pasos (Ver Figura 106):

1. Se debe ir a la sección de **Usuarios**.
2. Se desplegará opciones y deberá seleccionar **Modificar usuario**.
3. Llenar el campo cédula y presionar **Buscar**.
4. Se habilitarán los campos con la información que el usuario.
5. Presione el botón **Guardar**.



Modificar usuario

Cédula:  Buscar 

Cédula: 1725053779

Nombre:	<input type="text" value="Bryan"/>	Placa:	<input type="text" value="PDF1234"/>
Tipo:	<input type="text" value="Est. Presencial"/>	Marca:	<input type="text" value="CHEVROLET"/>
Teléfono:	<input type="text" value="0987654321"/>	Color:	<input type="text" value="GRIS"/>

Figura 106: Modificar usuario

#### 6.1.2.6 Solicitudes

##### 6.1.2.6.1 Crear una solicitud

Según la Figura 107 para crear una solicitud se deben realizar los siguientes pasos:

1. Se debe ir a la sección de **Solicitudes**.
2. Se desplegará opciones y deberá seleccionar **Crear solicitud**.
3. Llenar el campo cédula y presionar **Buscar**.
4. Se habilitarán los campos a llenar.
5. Presione el botón **Guardar**.

Crear solicitud

Cédula:

Cédula: 1725053779

Nombre: Bryan

Tipo de parqueadero:

Observación:

Caso especial:

Figura 107: Crear solicitud

Tomar en cuenta que al no llenar todos los campos obligatorios saldrá un mensaje de error como se puede ver en la Figura 108.

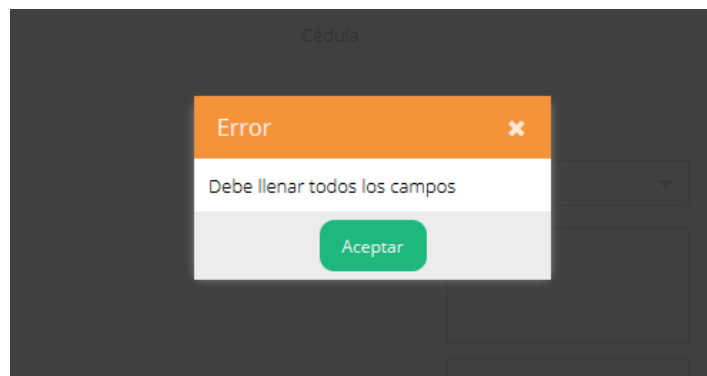


Figura 108: Error crear solicitud

#### 6.1.2.6.2 Modificar una solicitud

Para modificar la aprobación de una solicitud se deben realizar los siguientes pasos (Ver Figura 109):

1. Se debe ir a la sección de **Solicitudes**.
2. Se desplegará opciones y deberá seleccionar **Modificar solicitud**.
3. Debe llenar el ciclo o la cédula, no es necesario la cédula completa.
4. Presione **Buscar**.
5. Doble clic sobre la solicitud a modificar.
6. Presione el botón **Actualizar**.

Modificar solicitud

Ciclo: 2020-02 Cédula:  Buscar

Usuario	Periodo	Tipo Parqueadero	Observacion	Creado	Aprobada	Caso Especial
1725053779	2020-02	ABIERTO	Ninguna	2020-11-28	Si	No

Actualizar Cancelar

Figura 109: Modificar solicitud

### 6.1.2.6.3 Eliminar solicitud

Para eliminar una solicitud debe realizar los siguientes pasos, que se pueden visualizar en la Figura 110:

1. Se debe ir a la sección de **Solicitudes**.
2. Se desplegará opciones y deberá seleccionar **Eliminar solicitud**.
3. Seleccionar el bloqueo que se desea eliminar.
4. Presione el botón **Eliminar**.
5. Acepte el mensaje de confirmación.

Eliminar solicitud

Cédula:  Buscar Eliminar

Id	Periodo	Tipo Parquee...	Observacion
1725053779	2020	GIRÓN	
1720579968	2020	ABIERTO	

Figura 110: Eliminar solicitud

Tomar en cuenta que si se intenta eliminar una solicitud sin previamente haber selecciona una el sistema presentara un mensaje de error que se encuentra en la Figura 111.

ula: fntrhrthrh Buscar Eliminar

Periodo	Tipo Parqueadero	Observacion	Creado
---------	------------------	-------------	--------

**Error** ✕

Debe escoger un registro

Aceptar

Figura 111: Error eliminar solicitud

### 6.1.2.7 Consulta

#### 6.1.2.7.1 Consultar vehículo

Para consultar un vehículo se deben realizar los siguientes pasos (Ver Figura 112):

1. Se debe ir a la sección de **Consultar**.
2. Se desplegará opciones y deberá seleccionar **Buscar**.
3. Debe llenar a lo menos un campo y presionar **Buscar**, si presiona **Buscar** sin haber llenado ningún campo se presentarán todos los vehículos.

Id	Nombre	Tipo	Telefono	Bloqueo	Placa	Marca	Color
1725053...	Bryan	Est. Pres...	0987027...	SI	PDF3518	CHEVROL...	GRIS

Figura 112: Consultar vehículo

#### 6.1.2.7.2 Reportes

Para consultar los reportes esta ventana consta de dos partes.

##### Solicitud

Consultar una solicitud se deben realizar los siguientes pasos de la Figura 113:

1. Se debe ir a la sección de **Consultar**.
2. Se desplegará opciones y deberá seleccionar **Reportes**.
3. Seleccionar la pestaña **Solicitudes**.
4. Se presentarán diferentes parámetros por lo que se puede realizar la búsqueda.
5. Presiona el botón **Buscar**.

Reportes						
Solicitudes		Usuario				
Parqueadero:	<input type="text"/>	Ciclo:	<input type="text"/>	Aprobados:	<input checked="" type="checkbox"/>	Buscar
Todos los parqueaderos:	<input checked="" type="checkbox"/>	Todos los ciclos:	<input type="checkbox"/>	No aprobados:	<input type="checkbox"/>	
Usuario	Nombre	Teléfono	Placa	Periodo	Tipo Parqu...	Observac
1725053779	Bryan Lenin...	0987027450	PDF3518	2020	NOCHE	
1725053779	Bryan Lenin...	0987027450	PDF3518	2019-01	NOCHE	
1725053779	Bryan Lenin...	0987027450	PDF3518	2021	SEMIPRESE...	
1725053779	Bryan Lenin...	0987027450	PDF3518	2313	NOCHE	
1725053779	Bryan Lenin...	0987027450	PDF3518	1478	NOCHE	
1725053779	Bryan Lenin...	0987027450	PDE3518	2022	NOCHE	

Figura 113: Reportes solicitud

## Usuario

Para consultar un usuario que ha sido favorecido en el sorteo al menos una vez como en la Figura 114, se deben realizar los siguientes:

1. Se debe ir a la sección de **Consultar**.
2. Se desplegará opciones y deberá seleccionar **Reportes**.
3. Seleccionar la pestaña **Usuario**.
4. Se presentarán diferentes parámetros por lo que se puede realizar la búsqueda.
5. Presiona el botón **Buscar**.

Reportes							
Solicitudes		Usuario					
Tipo:	<input type="text" value="Est. Presencial"/>	Ciclo:	<input type="text"/>	Todos los ciclos:	<input checked="" type="checkbox"/>	Buscar	
Id	Nombre	Tipo	Telefono	Bloqueo	Placa	Marca	Color
1725053...	Bryan	Est. Prese...	09870274...	SI	PDF3518	CHEVROL...	GRIS

Figura 114: Reportes usuario

## 6.1.2.8 Sistema

### 6.1.2.8.1 Sorteo

Para realizar el sorteo se deben realizar los siguientes pasos (Ver Figura 115):

1. Se debe ir a la sección de **Sistema**.
2. Se desplegará opciones y debe seleccionar **Sorteo**.
3. Presione el botón **Sortear**.
4. Acepte el mensaje de confirmación.



Figura 115: Sorteo

#### 6.1.2.8.2 Enviar correos

Para enviar los correos se deben realizar los siguientes pasos como lo indica la Figura 116.

1. Se debe ir a la sección de **Sistema**.
2. Se desplegará opciones y debe seleccionar **Enviar correos**.
3. Presione el botón **Enviar correos**.
4. Acepte el mensaje de confirmación.

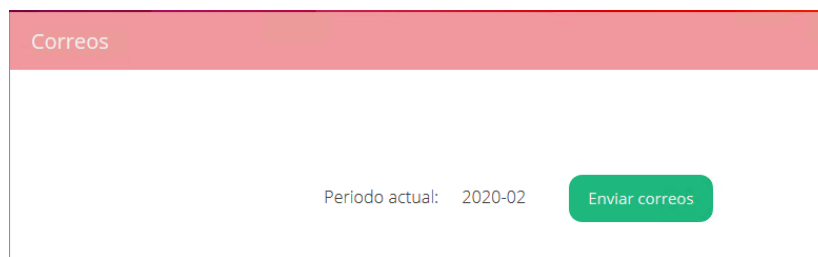


Figura 116: Correos

#### 6.1.2.8.3 Comprobantes

Para enviar los comprobantes se deben realizar los siguientes pasos (Ver Figura 117):

1. Se debe ir a la sección de **Sistema**.
2. Se desplegará opciones y debe seleccionar **Comprobantes**.
3. Presione el botón **Enviar comprobantes**.

## Generación de comprobantes

Esta opción permite la generación del rubro correspondiente al parqueadero en el comprobante de pago del estudiante.

Periodo activo: 2020-02

Comprobantes generados: 0

Comprobantes pendientes: 1

Enviar comprobantes

Figura 117: Comprobantes

## 6.2 Manual técnico

### 6.2.1 Administración

- Al crear un usuario del sistema, si se lo pone como activo este tendrá acceso al sistema caso contrario no, también el mismo debe ser asignado roles para el despliegue de los distintitos módulos que comprende el sistema.
- Cuando se modifica los valores de la disponibilidad de los parqueaderos, también cambian en la base de datos y afectan a la disponibilidad del sorteo.
- Los periodos nuevos se crean como inactivos de modo que es posible crearlos mientras hay otro periodo activo.
- Los bloqueos consideran si esta creado el usuario en caso de que estos buscan en la base de datos SARS en la tabla OPR\_USUARIO y traen el nombre y si esta creado en el sistema le adjunta un bloqueo, en caso de ya tener un bloqueo lo borra y le pone el actual, cabe recalcar que con un bloqueo basta para que el sistema no le asigne parqueadero.

### 6.2.2 Usuarios

- Al crear un nuevo usuario busca el nombre en la base de datos SARS en la tabla OPR\_USUARIO y trae el nombre, luego al momento de guardarlo consulta que este usuario sea activo o suspendido y por último que no esté en el sistema ingresado tanto el vehículo como el usuario.

- Cuando se modifica un usuario busca si existe el usuario en el sistema, en caso de que exista llena con la información los campos a ser modificados y al guardar hace un update de los mismos.

#### 6.2.3 Solicitudes

- Cuando se crea una solicitud primero busca si el usuario ya tiene una solicitud para el periodo activo, en caso de tener una solicitud se puede ingresar una nueva, pero la anterior será borrada. Todas las solicitudes creadas son ingresadas como aprobadas.
- Al modificar una solicitud estas pueden buscarse por los ciclos o por cédula y solo se puede modificar el activo.
- Para eliminar una solicitud se la busca por medio de la cédula y presenta las solicitudes solo de periodo activo.

#### 6.2.4 Consulta

- Al buscar se presenta los usuarios y vehículos que cumplan con los parámetros ingresados para la búsqueda.
- En reportes en aceptación trae la información dependiendo de los parámetros con los que se haga la búsqueda.
- En reportes por usuario cuando se selecciona administrativo hace la consulta de todos los administrativos y los ciclos, mientras que al elegir otra opción se puede consultar con el parámetro ciclo de manera opcional.

#### 6.2.5 Sistema

- En el sorteo primero se verifica el ciclo activo y se lo presenta en la pantalla, también se revisa si ese ciclo ya fue sorteado, de ser así se bloquea el botón y se presenta un mensaje de que ya fue sorteado para ese ciclo, caso contrario primero se les asigna parqueadero a todos los usuarios de posgrado y semipresenciales, luego se consulta la disponibilidad de cada parqueadero restante es decir ABIERTO, GIRON y NOCHE, y se realiza un sorteo para cada uno de ellos. Posteriormente se bloquea el botón y se presenta un mensaje.
- Al enviar los correos primero se verifica el ciclo y se lo presenta en pantalla, también se revisa si ya se enviaron los correos para ese ciclo, de ser así se bloquea el botón y se presenta un mensaje, caso contrario

se consulta todos los favorecidos en el sorteo y se busca en la base de datos SARS en la tabla de OPR\_USUARIO los usuarios de cada uno de ellos, y se le adjunta a cada uno de ellos "@puce.edu.ec;" luego se los envía los correos como oculto para que no puedan saber a quiénes fueron los favorecidos. Posteriormente se bloquea el botón y se presenta un mensaje.

- Para mandar los comprobantes primero se busca el ciclo y se lo presenta en pantalla, luego se busca cuantos comprobantes ya fueron enviados a la tabla de consulta\_factura para el ciclo actual y se presenta por pantalla, también se busca cuantos deberían haber, verificando cuantas solicitudes aprobadas están en la tabla de solicitud y se hace una resta entre el total con los generados y así se sabe si hay pendientes, en caso de no existir comprobantes pendientes se bloquea el botón y se presenta un mensaje. Al enviar los comprobantes se borra todos los que había en la tabla de consulta\_factura y se vuelve a cargar todos. Posteriormente se bloquea el botón ya que no existirán comprobantes pendientes y se presenta un mensaje.

### 6.3 Conclusiones y Recomendaciones

#### 6.3.1 Conclusiones

Las funcionalidades establecidas en los requerimientos del presente trabajo de disertación han sido realizadas correctamente lo que cumple con las necesidades de la comunidad universitaria.

El uso de buenas prácticas al momento de la construcción del software permite que sea un sistema mantenible y actualizable ya que al estar construido por capaz es de fácil adaptación a cambios futuros o integraciones en este producto.

Mediante el sistema se logra minimizar los gastos en stickers ya que ahora para la generación de reportes se valida con los usuarios activos de la comunidad universitaria, porque en el sistema anterior no se lo hacía y se realizaba la impresión de una cantidad aproximada pero no la real.

Al utilizar las mismas herramientas que la dirección de informática el producto final se integra mucho más fácil con los demás sistemas que se maneja en este departamento, pudiendo tomar datos en tiempo real desde otros programas.

Se recomienda que en un futuro se implemente una versión móvil ya que esto facilitaría la accesibilidad a los usuarios al momento de realizar cualquier acción sobre el sistema, para lo cual el frontend móvil debería conectarse al mismo backend ya que es donde están implementadas todos los requerimientos.

### 6.3.2 Recomendaciones

Se recomienda que para cualquier actualización del sistema se realicen por módulos y una vez estos módulos se encuentren funcionando de la manera esperada se los integre al sistema y se vuelvan a realizar las pruebas de los partes que tengan relación con los nuevos procesos.

Es recomendable mantener al sistema conectado a las fuentes del personal de la universidad ya que realiza verificaciones al momento de la generación de reportes, y en caso de que en un futuro se cambie dichas tablas se tenga en cuenta para realizar la modificación correspondiente en el sistema.

En caso de requerir información de otros sistemas de la comunidad universitaria se lo puede realizar sin ningún cambio significativo porque la mayoría de los backend de los sistemas se encuentran bajo un mismo proyecto de Codeigniter.

## Bibliografía

- Almeida, P. (2020). Sistema Parquaderos. Quito, Pichincha, Ecuador.
- Alvarez, M. (23 de Noviembre de 2009). *DesarrolloWeb*. Obtenido de <https://desarrolloweb.com/articulos/codeigniter.html>
- Alvarez, M. (28 de Julio de 2020). *DesarrolloWeb*. Obtenido de <https://desarrolloweb.com/articulos/que-es-mvc.html>
- Alvear, J., & Mazón, J. (2007). *Elaboración y Análisis de Métricas para el Proceso de Desarrollo de Software para Empresas Desarrolladoras de Software del Ecuador*. Guayaquil.
- ANT. (2020). *Agencia Nacional de Tránsito*. Obtenido de <http://www.amt.gob.ec/>
- Castejón, J. (2004). *Arquitectura y diseño de sistemas web modernos*.
- De La Cruz, F. (2020). *Ingeniería de Software I*. Quito.
- Dirección de Informática. (2020). Base de datos de parqueaderos. Quito, Ecuador.
- Dominguez, P. (2 de Junio de 2020). *OPENCLASSROOMS*. Obtenido de <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>
- EllisLab. (2020). *CodeIgniter*. Obtenido de <https://codeigniter.com/>
- Enríquez, J., Farías, E., Flores, E., Honores, C., Llanos, R., & Lopez, W. (2017). *Metodologías de desarrollo de Software*. Chimbote.
- Espitia, N., Armao, O., & Carbajo, J. (2016). *Modelo Vista Controlador (MVC)*. Caracas: Universidad Alejandro de Humboldt.
- Frederick, S., & Ramsay, C. (2008). *Learning Ext JS: Build Dynamic, Desktop-style User Interfaces for Your Data-driven Web Applications*. Packt Publishing Ltd.
- González, A., & Objeto, O. (2007). *Ingeniería de software: Metodologías*.
- González, Y., & Romero, F. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática*, 47-57.

- IEEE. (2000). Recommended Practice for Architectural Description of Software-Intensive Systems-IEEE Std 1471-2000. New York.
- INEC. (2020). Obtenido de <https://www.ecuadorencifras.gob.ec/vehiculos-matriculados-serie-historica-2008-2014/>
- ITCB. (19 de Junio de 2017). *Codeigniter*. Obtenido de [https://www.codeigniter.es/user\\_guide/overview/at\\_a\\_glance.html](https://www.codeigniter.es/user_guide/overview/at_a_glance.html)
- Juanta Andalucía. (2020). *Junta Andalucía*. Obtenido de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/264>
- Marin, J. (13 de Agosto de 2019). *EcuRed*. Obtenido de [https://www.ecured.cu/index.php?title=Sencha\\_Ext\\_JS&action=history](https://www.ecured.cu/index.php?title=Sencha_Ext_JS&action=history)
- Mateu, C. (2004). *Desarrollo de aplicaciones web*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.
- Pérez, D., & Cobo, Á. (2005). *PHP y MySQL: tecnologías para el desarrollo de aplicaciones web*. España: Ediciones Díaz de Santos.
- Quirola, J. (2020). *Requerimientos Funcionales y No Funcionales*.
- Romero, E. (2018). *Evaluación del entrenamiento físico y técnico en fútbol de las categorías formativas del Club Liga Deportiva Universitaria de Quito*. Quito.
- Sencha INC. (2020). *Sencha*. Obtenido de <https://www.sencha.com/products/extjs/>
- Sotomayor, J. A. (2017). *Sistemas de Gestión de Información y Conocimientos*. Lima.