

# **Universidad Autónoma Metropolitana – Azcapotzalco**

**División de Ciencias Básicas e Ingeniería**

**Licenciatura en Ingeniería en Computación**

## **Sistema de información para reservaciones en el servicio de cafetería UAM-A con generación de reporte**

Modalidad: Proyecto Tecnológico

Trimestre 2018-Invierno

Sandra Zavala Rojas  
208203963  
sandra\_zavala\_r@hotmail.com

### **Asesor**

Dr. José Alejandro Reyes Ortíz  
Profesor Asociado  
Departamento de Sistemas

### **Coasesor**

Dr. Leonardo Daniel Sánchez Martínez  
Profesor Asociado  
Departamento de Sistemas

Fecha de entrega: 23 de marzo de 2018


Yo, José Alejandro Reyes Ortiz, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Yo, Leonardo Daniel Sánchez Martínez, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Yo, Sandra Zavala Rojas, doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



*Con este trabajo concluyo una linda y larga etapa de mi vida, agradezco a mis padres que son mis pilares, sin ellos nada de esto sería posible, mis amigos por estar conmigo en buenos y malos momentos, mis asesores por trasmitirme su conocimiento, a ti Eduardo que siempre me alentaste a dar lo mejor de mí y a Dios que a pesar de todos mis errores siempre me diste la oportunidad de empezar de nuevo.*

## RESUMEN

En el presente reporte describe el desarrollo del Sistema de Reservación en Línea para el servicio de cafetería, el cual puede ser visualizado desde cualquier navegador, dispositivo móvil o tableta.

El Sistema de Reservación en Línea para el servicio de cafetería fue pensado principalmente para todos aquellos alumnos que se ven en la necesidad de desayunar y/o comer en la cafetería de la Unidad, ya que para recibir este servicio tienes que esperar demasiado tiempo. No obstante, este sistema también puede ser usado por los trabajadores de la Unidad.

El Sistema de Reservación en Línea para el servicio de cafetería consta de dos módulos principales: **USUARIO** el cual se encarga de realizar las reservaciones, cancelaciones y penalizaciones de los dos principales servicios que ofrece la cafetería de la unidad los cuales son desayuno y comida; y **ADMINISTRADOR** este se encarga de validar las reservaciones hechas por los usuarios, así como de obtener datos estadísticos y generación de reportes de los servicios ofrecidos (desayuno y comida) por las cafetería de la Unidad.

Las principales ventajas de desarrollar este sistema son:

- Disminuir el tiempo de espera de los alumnos o trabajadores para recibir el servicio de la cafetería.
- Garantizar que todos los usuarios que hicieron reservación reciban los alimentos que este servicio ofrece.
- Mejorar el servicio de cafetería, así como un mejor manejo de los recursos.

# ÍNDICE

1.	INTRODUCCIÓN.....	1
2.	ANTECEDENTES .....	1
2.1	Sistema de gestión para la oficina parroquial de San Juan Bautista Coyoacán. ....	1
2.2	Prototipo para agendar citas médicas.....	1
2.3	Sistema Gestor de Nutrición.....	1
2.4	LODGIFY.....	2
2.5	INFOCAJA 2.0. ....	2
2.6	Reservación de computadoras del edificio T. ....	2
3.	JUSTIFICACIÓN .....	2
4.	OBJETIVOS.....	3
4.1	Objetivo general .....	3
4.2	Objetivos específicos .....	3
5	MARCO TEÓRICO.....	3
5.1	PHP .....	3
5.1.1	Historia.....	3
5.1.2	¿Qué es PHP? .....	4
5.2	MySQL .....	4
5.2.1	Historia.....	4
5.2.2	¿Qué es MySQL?.....	5
5.3	HTML.....	5
5.3.1	Historia.....	5
5.3.2	¿Qué es HTML? .....	5
5.3.3	HTML 5.....	6
5.4	JavaScript .....	6
5.5	CSS .....	6
5.6	Semáforos PHP .....	7
5.7	AdminLTE .....	7
5.8	Barcode .....	7
5.9	FPDF .....	7
5.10	PHPExcel .....	8
5.11	SweetAlert2.....	8
6	DESARROLLO DEL PROYECTO .....	8
6.1	BASE DE DATOS.....	8
6.1.1	Tabla usuario .....	9

6.1.2	Tabla reservacion .....	10
6.1.3	Tabla cancelacion.....	11
6.1.4	Tabla tipo_reservacion .....	11
6.1.5	Tabla penalización.....	12
6.1.6	Tabla trabajador.....	12
6.1.7	Tabla privilegio.....	13
6.1.8	Diagrama del Modelo Entidad Relación de la Base de Datos .....	14
6.2	SISTEMA USUARIO .....	14
6.2.1	Registro.....	14
6.2.2	Login .....	17
6.2.3	Reservación.....	19
6.2.4	Cancelación .....	24
6.3	SISTEMA ADMINISTRADOR.....	30
6.3.1	Login .....	30
6.3.2	Crear, actualizar y eliminar Administradores .....	32
6.3.3	Crear, editar y eliminar Usuarios .....	39
6.3.4	Reservar, cancelar o quitar penalización .....	43
6.3.4.1	Reservaciones.....	43
6.3.4.2	Cancelaciones.....	46
6.3.4.3	Penalizaciones .....	47
6.3.5	Validación .....	48
6.3.5.1	Validar Asistencia por día.....	49
6.3.5.2	Validar Asistencia en General.....	50
6.3.6	Datos Estadísticos.....	51
6.3.6.1	Estadísticas y Reportes.....	51
6.3.6.2	Gráficas .....	57
6.3.6.2.1	Reservación .....	57
6.3.6.2.2	Cancelaciones.....	59
6.3.6.2.3	Penalización.....	60
6.3.6.3	Historiales .....	61
6.3.7	Hardware y Software .....	65
6.3.7.1	Hardware .....	65
6.3.7.2	Software.....	65
7	RESULTADOS .....	65
8	ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS .....	66

8.1	Registro y Login .....	66
8.2	Reservación .....	66
8.3	Penalización.....	67
8.4	Cancelación .....	67
8.5	Administrador .....	67
8.6	Verificar Asistencia .....	67
8.7	Datos Estadísticos y Reportes, Gráficas e Historiales .....	68
9	CONCLUSIONES .....	68
10	REFERENCIAS .....	69

## INDICE DE FIGURAS

Figura 1. Proceso que se realiza al visitar una página PHP. ....	4
Figura 2. Logo de MySQL. ....	4
Figura 3. Tabla Usuario. ....	9
Figura 4. Tabla reservación. ....	10
Figura 5. Tabla cancelación. ....	11
Figura 6. Tabla tipo_reservacion. ....	11
Figura 7. Tabla Penalización. ....	12
Figura 8. Tabla trabajador. ....	12
Figura 9. Tabla privilegio ....	13
Figura 10. Modelo de la Base de Datos. ....	14
Figura 11. Formulario para el registro de los Usuarios. ....	15
Figura 12. Contraseña correcta vs contraseña incorrecta. ....	15
Figura 13. Resultados de la verificación de la función is_numeric(). ....	16
Figura 14. Mensaje de error al intentar registrar un Usuario ya registrado. ....	16
Figura 15. Mensajes que se pueden mostrar al registrar un Usuario. ....	17
Figura 16. Tabla con información de Usuarios. ....	17
Figura 17. Formulario Login Usuario. ....	18
Figura 18. Resultado de la Base de Datos, Usuario Registrado vs Usuario no Registrado. ....	18
Figura 19. Resultados de la verificación de la función password_verify(). ....	19
Figura 20. Mensajes que se pueden mostrar al logear un Usuario. ....	19
Figura 21. Formulario Reservación. ....	19
Figura 22. Mensaje de error. Fuera de la hora para hacer la Reservación. ....	20
Figura 23. Mensaje de error. Ya no hay reservaciones. ....	21
Figura 24. Respuesta de la consulta a la Base de Datos para validar el cupo. ....	21
Figura 25. Mensaje de error. El usuario ya hizo una reservación. ....	21
Figura 26. Respuesta de la consulta a la Base de Datos para validar si el Usuario ya ha hecho una reservación previa. ....	21
Figura 27. Respuesta de la consulta a la Base de Datos para validar si alguna vez ha reservado el Usuario. ....	22
Figura 28. Valor de confirmación igual a true vs confirmación igual a false. ....	22
Figura 29. Mensaje de error. El usuario está Penalizado. ....	23
Figura 30. Descarga de la reservación. ....	23
Figura 31. Ejemplo del PDF de una reservación. ....	23



Figura 32. Formulario Cancelación. ....	24
Figura 33. Mensaje error. El usuario no ha reservado aun. ....	24
Figura 34. Resultado de la consulta a la Base de Datos para saber si el usuario ya reservo. ....	24
Figura 35. Mensaje error. El usuario no está en fecha ni hora. ....	25
Figura 36. Mensaje error. La reservación ya fue cancelada. ....	25
Figura 37. Resultado de la Base de Datos, confirmación igual a uno vs confirmación igual a cero. ....	25
Figura 38. Descarga de la cancelación en PDF ....	26
Figura 39. Ejemplo del PFD de una cancelación. ....	26
Figura 40. Tabla con cancelaciones registradas. ....	26
Figura 41. Formulario de Editar Usuario. ....	27
Figura 42. Opciones de Historial. ....	27
Figura 43. Historial de Reservaciones. ....	28
Figura 44. Historial de Cancelaciones. ....	28
Figura 45. Historial de Penalizaciones. ....	29
Figura 46. Interfaz completa del Sistema de Reservación en Línea para el Servicio de Cafetería (Usuario). ....	29
Figura 47. Formulario de Login Administrador. ....	30
Figura 48. Resultado de la Base de Datos, Usuario Registrado vs Usuario no Registrado. ....	31
Figura 49. Resultados de la verificación de la función password_verify(). ....	31
Figura 50. Mensajes que se pueden mostrar al logear un Administrador. ....	32
Figura 51. Interfaz Administrador con privilegio "administrador". ....	33
Figura 52. Interfaz Administrador con privilegio tipo "validar". ....	34
Figura 53. Comparativa de los resultados de la Base de Datos del campo privilegio. ....	34
Figura 54. Formulario Crear Administrador. ....	35
Figura 55. Contraseña incorrecta vs contraseña incorrecta. ....	36
Figura 56. Resultados de la verificación de la ..... ..	36
Figura 57. Mensaje de error al intentar registrar un Administrador ya registrado. ....	37
Figura 58. Mensajes que se pueden mostrar al crear un Administrador. ....	37
Figura 59. Tabla con información de los Administradores. ....	37
Figura 60. Lista de Administradores. ....	38
Figura 61. Formulario Editar Administrador. ....	39
Figura 62. Formulario Crear Usuario. ....	40
Figura 63. Contraseña correcta vs contraseña incorrecta. ....	40

Figura 64. Resultados de la verificación de la .....	41
Figura 65. Mensaje de error al intentar registrar un Usuario ya registrado.....	41
Figura 66. Lista de Usuarios Registrados.....	42
Figura 67. Formulario Editar Usuario. ....	42
Figura 68. Formulario Reservar, Sistema Administrador. ....	43
Figura 69. Mensaje al Administrador de reservaciones agotadas. ....	44
Figura 70. Existe usuario vs No Existe usuario. ....	44
Figura 71. Mensaje al Administrador, no se encontró al Usuario.....	45
Figura 72. Mensaje al Administrador, Reservación exitosa. ....	45
Figura 73. Reservaciones que puede cancelar el Administrador.....	46
Figura 74. Mensaje al Administrador, Cancelación exitosa.....	46
Figura 75. Usuarios a los que se les puede quitar la Penalización.....	47
Figura 76. Mensaje al Administrador, el usuario ya no está penalizado. ....	48
Figura 77. Menú de validar Asistencia. ....	48
Figura 78. Lista de usuarios que hicieron algún tipo de reservación. ....	49
Figura 79. Mensaje de Asistencia registrada. ....	49
Figura 80. Formulario Validar Reservación. ....	50
Figura 81. Resultados de la verificación de la .....	50
Figura 82. Resultado de la consulta a la Base de Datos,.....	51
Figura 83. Mensajes que se pueden mostrar al registrar una asistencia al servicio de cafetería. ....	51
Figura 84. Datos Estadísticos Generales y Generación de Reportes. ....	52
Figura 85. Descarga de los reportes.....	52
Figura 86. Reporte_Usuario.xlsx.....	53
Figura 87. ReporteReservaciones.xlsx. ....	54
Figura 88. ReporteCancelacion.xlsx. ....	55
Figura 89. ReportePenalizacion.xlsx.....	56
Figura 90. Datos Estadísticos por Servicio. ....	56
Figura 91. Datos Estadísticos por Día. ....	57
Figura 92. Gráfica de Reservaciones en General por día. ....	58
Figura 93. Gráfica de Reservaciones de Desayuno por día.....	58
Figura 94. Gráfica de Reservaciones de Comida por día. ....	58
Figura 95. Gráfica de Cancelaciones en General por día. ....	59
Figura 96. Gráfica de Cancelaciones de Desayuno por día.....	59
Figura 97. Gráfica de Cancelaciones de Comida por día.....	60

Figura 98. Gráfica de Penalizaciones en General por día.....	60
Figura 99. Gráfica de Penalizaciones de Desayuno por día. ....	61
Figura 100. Grafica de Penalizaciones de Comida por día. ....	61
Figura 101. Historial de Reservaciones. ....	62
Figura 102. Historial de Cancelaciones. ....	63
Figura 103. Historial de Penalizaciones.....	64
Figura 104. Formulario Editar Administrador. ....	64

## 1. INTRODUCCIÓN

La Coordinación de Servicios Universitarios tiene la encomienda de favorecer el desarrollo de la comunidad universitaria proporcionando servicios de alimentación, atención médica, recreación deportiva, reproducción de documentos, venta de libros y papelería; así como el apoyo en medios audiovisuales, en tiempo y espacios adecuados [1].

En esta ocasión solo nos enfocaremos a la **sección de cafetería**, el cual es un espacio donde se ofrecen desayunos y comidas a un promedio de cuatro mil miembros de la comunidad universitaria al día. La cafetería ofrece alimentos frescos, elaborados higiénicamente con productos de alta calidad y balanceados nutricionalmente [2].

Las actividades de los estudiantes y trabajadores de la Universidad Autónoma Metropolitana Unidad Azcapotzalco (UAM-A) son vastas y diversas, por lo que muchos de ellos pasan demasiado tiempo en la institución, viéndose en la necesidad de consumir alimentos fuera de sus casas, algunos de ellos recurren al servicio de cafetería lo que ocasiona largas filas y mucho tiempo de espera para hacer uso de este servicio

La demanda de este servicio ocasiona que los alimentos ofrecidos sean insuficientes, por ende, tanto alumnos como trabajadores tienen incertidumbre respecto a la posibilidad de obtener dicho servicio.

Al realizar un sistema de reservaciones para el servicio de cafetería, se pretende minimizar el tiempo de espera de alumnos y trabajadores, garantizando el consumo de los alimentos que ofrece este servicio.

## 2. ANTECEDENTES

### 2.1 Sistema de gestión para la oficina parroquial de San Juan Bautista Coyoacán [3]

Este trabajo es un sistema de gestión para las solicitudes de las ceremonias eucarísticas de una oficina parroquial. La similitud con la propuesta es que es una aplicación web, registro de ceremonia (Tipo de ceremonia, día de la ceremonia, hora de la ceremonia). La diferencia es que está programada en el lenguaje JAVA, está enfocado en administrar un catálogo, utiliza Oracle SQL, se basa en mejorar una opción de software ya existente.

### 2.2 Prototipo para agendar citas médicas [4]

Este trabajo es una agenda compartida para gestionar citas médicas a través de un dispositivo móvil con un sistema operativo Android. La similitud con la propuesta es que está enfocado a la gestión de citas médicas (día y hora de la cita) así como la cancelación de dichas citas. La diferencia es que esta es una aplicación móvil y utiliza SQLite.

### 2.3 Sistema Gestor de Nutrición [5]

Es un sistema de información web que permite hacer el registro de paciente, guardar el historial del paciente, dieta propuesta, propone una posible cita de acuerdo a su historial de citas y además realiza estadísticas. La similitud con la propuesta es que es una aplicación web, propone futuras citas (día y hora de la cita) y generación de reportes para fines estadísticos. La diferencia es que se realizó en plataforma Eclipse y se hizo uso de frameworks Spring.

## **2.4 LODGIFY [6]**

Este software es una plataforma ERP<sup>1</sup> *all-in-one* que permite a sus clientes que son propietarios de casas rurales crear páginas web personalizadas, hacer reservaciones *online*, gestionar calendarios entre otras cosas. Es semejante a este proyecto ya que permite al usuario final reservar una propiedad si es que está disponible, a través de una aplicación web. La diferencia es el tipo de servicio que se le ofrece al cliente final, cabe mencionar que este servicio implica un costo.

## **2.5 INFOCAJA 2.0 [7]**

Es un software comercial que permite la gestión total de un restaurante, bar, cafetería, etc., permite hacer reservaciones, registrar la asistencia de los empleados, el inventario, pedidos a domicilio, permite tener una base de datos de los clientes con lo que han pedido a domicilio, para que el dueño sepa las preferencias de los clientes, tiene la capacidad de geolocalización de sus clientes, entre otras características. Es semejante a este proyecto ya que una de sus funcionalidades es permitir al cliente hacer reservaciones y cancelaciones, estas reservaciones o cancelaciones son a través de un dispositivo móvil.

## **2.6 Reservación de computadoras del edificio T [8]**

La Universidad Autónoma Metropolitana también ha hecho algunos trabajos relacionados con reservaciones. “La reservación de computadoras del edificio T”, específicamente de la sala general, este trabajo estuvo a cargo del ingeniero Rafael Alvarado Ramírez. Para la parte web se utiliza un servidor Linux y PHP 5.3, soporta 300 usuarios concurrentes, es una aplicación realizada en el lenguaje JAVA. Una de las similitudes es que este sistema permite al usuario reservar por día.

## **3. JUSTIFICACIÓN**

El desarrollar un sistema de información de reservaciones de cafetería, ayudará a que los alumnos y trabajadores no pierdan tanto tiempo haciendo largas filas. Garantizando que los alumnos y trabajadores reciban los alimentos que este servicio ofrece. Por otra parte, ayudará a que el jefe de cafetería en turno tome mejores decisiones en cuanto a la cantidad de alimento que se tendrá que realizar para que no sea insuficiente y el desperdicio de este sea mínimo y con esto ofrecer un mejor servicio a la comunidad universitaria.

Los alumnos y trabajadores podrán hacer sus reservaciones desde cualquier dispositivo. En un principio del total de platillos servidos, solo se podrá reservar un 30% de los platillos. Los alumnos y trabajadores que generen su reservación tendrán una fila especial, para poder formarse en esta fila tendrán que mostrar de manera impresa o digital la reservación, ya que esta será validada por el personal de cafetería, con el fin de actualizar la base de datos y llevar un control de la asistencia a la cafetería por reservación.

Es importante mencionar que ya existen algunas aplicaciones con la finalidad de hacer reservaciones, por lo que el proyecto no es en sí mismo original, pero la finalidad es apoyar a la comunidad universitaria.

---

<sup>1</sup> ERP es un conjunto de sistemas de información que permite la integración de ciertas operaciones de una empresa, especialmente las que tienen que ver con la producción, la logística, el inventario, los envíos y la contabilidad [9].

En el desarrollo de este proyecto no se tomará en cuenta el servicio de cena que ofrece la cafetería, ya que no es muy demandado y no es constante. Se asume un funcionamiento “ideal” del servicio en cuanto a las funciones y horarios de operación del mismo.

## 4. OBJETIVOS

### 4.1 Objetivo general

Desarrollar un sistema de información con el fin de disminuir el tiempo de espera para el consumo de alimentos y garantizar el buen uso de los recursos del servicio de cafetería de la UAM-A.

### 4.2 Objetivos específicos

- Crear una Base de datos para almacenar la información del Sistema.
- Desarrollar e implementar un módulo de registro tanto para usuarios como administrador.
- Desarrollar e implementar un módulo de acceso tanto para usuarios como administrador.
- Desarrollar e implementar un módulo de reservación.
- Desarrollar e implementar un módulo de cancelar reservaciones.
- Desarrollar e implementar un módulo de generación de reportes.
- Desarrollar e implementar un módulo de validación.

## 5 MARCO TEÓRICO

### 5.1 PHP

#### 5.1.1 Historia [10]

En 1994 un programador llamado Rasmus Lerdorf, fue quien desarrollo un código que le ayudaría a crear su página Web personal de manera sencilla y lo llamo *Personal Home Page Tools (PHP Tools)*. De las primeras tres letras en ingles surge el nombre del lenguaje, que finalmente se liberó al público de manera gratuita en 1995.

Dos años después, junto a otras personas rescriben parte del código del lenguaje y se lanza una nueva versión de PHP (Un lenguaje simple, rápido y dinámico que permite crear páginas Web interactivas con muy poco código).

En ese momento decide que el nombre de *Personal Home Page* ya le quedaba corto y decide cambiar el significado de las siglas a *PHP Hypertext Preprocessor*.

Uno de los sitios más importantes y con mayor número de visitas al día en el mundo es *Facebook* y está desarrollado con PHP, esto es solo para ejemplificar la potencia de este lenguaje.

### 5.1.2 ¿Qué es PHP? [11]

PHP es un lenguaje de alto nivel que se ejecuta desde un servidor, un lenguaje de servidor es aquél que se ejecuta en el servidor donde están alojadas las páginas Web. La ventaja de ejecutar el código en el servidor es que todas nuestras páginas van a poder ser vistas en cualquier ordenador, independientemente del navegador que se tenga.

La Figura 1 ilustra el funcionamiento de PHP a través de un servidor Web.



Figura 1. Proceso que se realiza al visitar una página PHP.

## 5.2 MySQL

### 5.2.1 Historia [12]



Figura 2. Logo de MySQL.

Comenzó con la intención de utilizar el sistema de base de datos mSQL para conectar sus tablas, utilizando sus propias rutinas. Sin embargo, después de algunas pruebas, llegaron a la conclusión de que mSQL no era lo suficientemente rápido o lo suficientemente flexible para sus necesidades. Esto dio como resultado una nueva interfaz SQL, esta API fue diseñada para permitir que el código que fue escrito para uso con mSQL sea portable para el uso con MySQL.

Fue concebido originalmente por la compañía sueca MySQL AB y en 2008 fue adquirida por *Oracle*.

MySQL lleva el nombre de la hija de la cofundadora Monty Widenius, My. En la Figura 2 se muestra el nombre y logo de MySQL.

El nombre del *MySQL Dolphin* (logo de MySQL) es "Sakila" que fue elegido en el concurso "Nombre del delfín". El ganador fue Ambrose Twebaze, un desarrollador de software de código abierto de Suazilandia, África. Según Ambrose, el nombre de Sakila tiene sus raíces en SiSwati, el idioma local de Suazilandia. Sakila es también el nombre de una ciudad en Arusha, Tanzania.

### 5.2.2 ¿Qué es MySQL? [13]

MySQL es un sistema de gestión de base de datos relacional de código abierto, basado en lenguaje de consulta estructurado (SQL), es desarrollado, distribuido y respaldado por *Oracle Corporation*. Una base de datos relacional almacena datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto impacta en la velocidad de acceso y proporciona flexibilidad para la modificación. Las tablas mantienen relaciones que hacen posible combinar datos de las diferentes tablas.

## 5.3 HTML

### 5.3.1 Historia [14]

El origen de HTML se remonta a 1980, cuando el físico Tim Berners-Lee propuso un nuevo sistema de "hipertexto" para compartir documentos. El "hipertexto" permitía que los usuarios accedieran a la información relacionada con los documentos electrónicos que estaban visualizando.

Tras finalizar el desarrollo de su sistema de "hipertexto", Tim Berners-Lee lo presentó en una convocatoria, organizada para desarrollar un sistema de "hipertexto" para Internet, al unir fuerzas con el ingeniero de sistemas Robert Cailliau, presentaron la propuesta ganadora llamada *WorldWideWeb* (W3).

El primer documento formal con la descripción de HTML se publicó en 1991 bajo el nombre *HTML Tags* (Etiquetas HTML).

La primera propuesta oficial para convertir HTML en un estándar se realizó en 1993 por parte del organismo IETF (*Internet Engineering Task Force*), aunque se consiguieron avances significativos, la propuestas no consiguió convertirse en estándar oficial, hasta que en 1995, el organismo IETF organiza un grupo de trabajo de HTML y consigue publicar el 22 de septiembre de ese mismo año el estándar HTML 2.0. A partir de 1996, los estándares de HTML los publica otro organismo de estandarización llamado W3C (*World Wide Web Consortium*).

### 5.3.2 ¿Qué es HTML? [15]

Definiéndolo de forma sencilla, HTML (*Hyper Text Markup Language*) es el lenguaje con el que se escriben las páginas web. Es un lenguaje muy sencillo que permite confeccionar archivos de texto, mostrándolo de forma estructurada y atractiva, con enlaces que hacen referencia a otros documentos o fuentes de información relacionada, y que permite incorporar archivos multimedia como gráficos, sonidos, imágenes, videos, etc.

El propio W3C<sup>2</sup> define el lenguaje HTML como "un lenguaje reconocido universalmente y que permite publicar información de forma global". Desde su creación, el lenguaje HTML ha pasado de ser un lenguaje utilizado exclusivamente para crear documentos electrónicos a ser un lenguaje que se utiliza en muchas aplicaciones electrónicas como buscadores, tiendas online y banca electrónica.

---

<sup>2</sup> *World Wide Web Consortium*.



### 5.3.3 HTML 5 [16]

HTML5 es la última versión de HTML, con nuevos elementos, atributos y comportamientos. Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance.

### 5.4 JavaScript [17]

JavaScript es un lenguaje de Programación creado por Netscape con el objetivo de integrarse en HTML y facilitar la creación de páginas interactivas.

En 1997 se propuso que JavaScript fuera adoptado como estándar de la ECMA (*European Computer Manufacturers Association*) y en junio del mismo año fue adoptado como estándar ECMA, con el nombre de *ECMAScript*.

JScript es la implementación de *ECMAScript* de Microsoft, muy similar a JavaScript, pero con ciertas diferencias en el modelo de objetos del navegador que hacen que ambas versiones sean incompatibles con frecuencia. Para evitar estas incompatibilidades el W3C<sup>1</sup> diseñó el estándar Modelo de Objeto del Documento (DOM) que incorporan los navegadores.

No hay que confundir Java con JavaScript. Java es un lenguaje de programación que permite crear aplicaciones independientes y necesita ser compilado, mientras que JavaScript es un lenguaje que funciona como extensión de HTML.

El código de programación de JavaScript, se introduce directamente en el documento HTML y no necesita ser compilado, es el propio navegador el que se encarga de traducir el código.

Gracias a JavaScript podemos desarrollar programas que se ejecuten directamente en el navegador, de manera que éste pueda tomar decisiones sin la necesidad de acceder al servidor.

### 5.5 CSS [18]

CSS (Hojas de Estilo en Cascada), es un lenguaje que describe el estilo de un documento HTML, describe cómo deben mostrarse los elementos HTML, ahorra mucho trabajo ya que puede controlar el diseño de múltiples páginas web a la vez.

Cuando un navegador lee una hoja de estilo, formateará el documento HTML de acuerdo con la información en la hoja de estilos.

Hay dos formas de insertar una hoja de estilo:

- Hoja de estilo externo
- Hoja de estilo interna

## 5.6 Semáforos PHP [19]

En ciertas ocasiones necesitamos que una parte del código se ejecute por un solo usuario a la vez. Estos bloqueos de código se utilizan para evitar problemas de concurrencia. En PHP es posible implementar estos bloqueos mediante semáforos.

Según el manual de PHP los semáforos se pueden usar para proveer acceso exclusivo a recursos en la máquina actual, o limitar el número de procesos que puede usar un recurso simultáneamente.

## 5.7 AdminLTE [20]

AdminLTE es un panel de administración para Bootstrap creado por el estudio *Almsaeed*. Es una solución de código abierto basada en un diseño modular que permite una construcción y personalización sencillas. La idea es que cada uno de estos elementos sea un *plugin*<sup>3</sup> o un *widget*<sup>4</sup> a través del cual uno va creando la interfaz de usuario tanto por la parte de *front-end*<sup>5</sup> como de *back-end*<sup>6</sup>.

## 5.8 Barcode [21]

Este es un generador de código de barras en PHP. Crea imágenes SVG, PNG, JPG y HTML a partir de los estándares de códigos de barras más utilizados.

La base de código proviene en gran parte del generador de códigos de barras TCPDF<sup>7</sup> de Nicola Asuni.

## 5.9 FPDF [22]

FPDF es una clase escrita en PHP que permite generar documentos PDF directamente desde PHP, es decir, sin usar la biblioteca PDFlib<sup>8</sup>. La F de FPDF significa *Free* (gratis y libre). Sus principales características:

- Elección de la unidad de medida, formato de página y márgenes
- Gestión de cabeceras y pies de página
- Salto de página automático
- Salto de línea y justificación del texto automáticos
- Admisión de imágenes (JPG, PNG y GIF)

---

<sup>3</sup> *Plugin* es una aplicación que se relaciona con otra para agregarle una función nueva y generalmente muy específica.

<sup>4</sup> *Widget* es un programa pequeño que te da acceso a las funciones que más usas.

<sup>5</sup> *Front-end* es el encargado de las vistas en una página web (es lo que el usuario puede ver).

<sup>6</sup> *Back-end* es todo lo que el usuario no puede ver, es aquel que se encuentra del lado del servidor, en otras palabras es el encargado de la manipulación de los datos.

<sup>7</sup> TCPDF es una *Open Source* Clase/Biblioteca para el Popular Lenguaje de Programación Web PHP.

<sup>8</sup> PDFlib es una librería de PHP para creación de archivos PDF.

- Colores
- Enlaces
- Compresión de página

### **5.10 PHPExcel [23]**

PHPExcel es una librería creada en PHP que nos permitirá crear, modificar y leer diferentes formatos de Excel, como .xls (Excel 2007), .xlsx (OfficeOpenXML). Para utilizar la librería es necesario tener instalado PHP 5.2 en adelante.

### **5.11 SweetAlert2 [24]**

Reemplaza las alertas por default de JavaScript por unas cajas de diálogo más coloridas y animadas, no importa si está utilizando una computadora de escritorio, un dispositivo móvil o una tableta, en todos ellos se visualizarán las alertas. Tiene 5 principales tipos de mensaje: de éxito, de error, de advertencia, de información y de pregunta.

## **6 DESARROLLO DEL PROYECTO**

El Sistema de Reservación en línea para el servicio de cafetería de la UAM-AZC se divide en tres partes.

- Base de Datos.
- Sistema Usuario.
- Sistema Administrador.

### **6.1 BASE DE DATOS**

El propósito de usar una Base de Datos es guardar toda la información del Sistema y esta consta de siete tablas.

- usuario
- reservacion
- cancelacion
- tipo\_reservacion
- penalizacion
- trabajador
- privilegio

A continuación se detallará cada una de ellas.

### 6.1.1 Tabla usuario

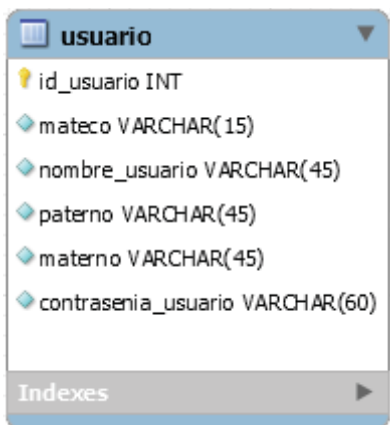


Figura 3. Tabla Usuario.

El propósito de la tabla usuario es almacenar toda la información requerida por el sistema relacionada con nuestros usuarios. Tal y como se muestra en la Figura 3, esta tabla cuenta con los siguientes atributos.

*id\_usuario.*

Este atributo identifica de manera única a los usuarios dentro del Sistema, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

*mateco.*

En este atributo se guardará la matricula del alumno o el número económico del trabajador, es de tipo VARCHAR, forzosamente tiene que tener un valor que no puede ser repetido.

*nombre\_usuario.*

En este atributo, como su nombre lo indica, se guardará el nombre o los nombres del usuario, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

*paterno.*

En este atributo se guardará el apellido paterno de los usuarios, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

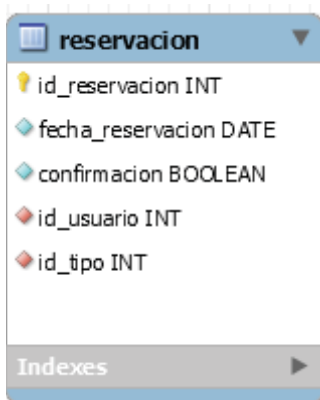
*materno.*

En este atributo se guardará el apellido materno de los usuarios, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

*contrasenia\_usuario.*

En este atributo se guardará la contraseña de los usuarios, es de tipo VARCHAR, forzosamente tiene que tener un valor y puede que se repita.

## 6.1.2 Tabla reservacion



El propósito de la tabla reservación es almacenar toda la información necesaria con respecto a las reservaciones del Sistema. Tal y como se muestra en la Figura 4, esta tabla cuenta con los siguientes atributos.

*id\_reservacion.*

Este atributo identifica de manera única las reservaciones hechas en el Sistema, se podría decir que es el folio de la reservación, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

Figura 4. Tabla reservación.

*fecha\_reservacion.*

En este atributo se guardará la fecha en que fue hecha la reservación por parte del usuario, es de tipo DATE, forzosamente tiene que tener un valor y este se puede repetir.

*confirmacion.*

En este atributo se registrará la asistencia o inasistencia por parte del usuario a la cafetería, es de tipo BOOLEAN, forzosamente tiene que tener un valor, que inicialmente es "false", una vez que se confirma la asistencia el valor pasa a ser "true".

*id\_usuario.*

En este atributo se guardará el ID de los usuarios que hicieron una o más reservaciones, es de tipo entero, forzosamente tiene que tener un valor y este se puede repetir. Este atributo es una llave foránea.

*id\_tipo.*

En este atributo se guardará el ID del tipo de reservación que hizo el usuario, es de tipo entero, forzosamente tiene que tener un valor y este se puede repetir. Este atributo es una llave foránea.

### 6.1.3 Tabla cancelacion

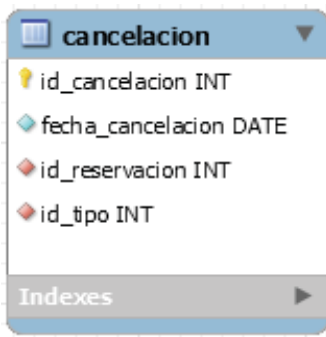


Figura 5. Tabla cancelación.

El propósito de la tabla cancelación es almacenar toda la información necesaria con respecto a las cancelaciones del Sistema. Tal y como se muestra en la Figura 5, esta tabla cuenta con los siguientes atributos.

*id\_cancelacion.*

Este atributo identifica de manera única las cancelaciones hechas en el Sistema, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

*fecha\_cancelacion.*

En este atributo se guardará la fecha en que fue hecha la cancelación por parte del usuario, es de tipo DATE, forzosamente tiene que tener un valor y este se puede repetir.

*id\_reservacion.*

En este atributo se guardará el ID de las reservaciones hechas por los usuarios, es de tipo entero, forzosamente tiene que tener un valor y no puede ser repetido. Este atributo es una llave foránea.

*id\_tipo.*

En este atributo se guardará el ID del tipo de reservación que hizo el usuario, es de tipo entero, forzosamente tiene que tener un valor y este se puede repetir. Este atributo es una llave foránea.

### 6.1.4 Tabla tipo\_reservacion



Figura 6. Tabla tipo\_reservacion.

El propósito de la tabla tipo\_reservacion es almacenar los tipos de servicio que ofrece la sección de cafetería. Tal y como se muestra en la Figura 6, esta tabla cuenta con los siguientes atributos.

*id\_tipo.*

Este atributo identifica de manera única el tipo de servicio que eligió el usuario en el Sistema, ya sea para reservar o cancelar, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

*nombre\_tipo.*

En este atributo, como su nombre lo indica, se guardará los nombres de los servicios que ofrece la sección de cafetería, es de tipo VARCHAR, forzosamente tiene que tener un valor y este no puede ser repetido.

### 6.1.5 Tabla penalización



Figura 7. Tabla Penalización.

El propósito de la tabla penalización es almacenar toda la información necesaria con respecto a las penalizaciones del Sistema. Tal y como se muestra en la Figura 7, esta tabla cuenta con los siguientes atributos.

*id\_penalizacion.*

Este atributo identifica de manera única las penalizaciones hechas en el Sistema, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

*fecha\_penalizacion.*

En este atributo se guardará la fecha en que fue realizada la penalización al usuario, es de tipo DATE, forzosamente tiene que tener un valor y este se puede repetir.

*id\_reservacion.*

En este atributo se guardará el ID de las reservaciones hechas por los usuarios, es de tipo entero, forzosamente tiene que tener un valor y no puede ser repetido. Este atributo es una llave foránea.

### 6.1.6 Tabla trabajador



Figura 8. Tabla trabajador.

El propósito de la tabla trabajador es almacenar toda la información de los trabajadores requerida por el Sistema administrador. Tal y como se muestra en la Figura 8, esta tabla cuenta con los siguientes atributos.

*id\_trabajador.*

Este atributo identifica de manera única a los trabajadores dentro del Sistema, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es la llave primaria de esta tabla.

*num\_economico.*

En este atributo se guardará el número económico del trabajador, es de tipo VARCHAR, forzosamente tiene que tener un valor que no puede ser repetido.

*nombre\_trabajador.*

En este atributo, como su nombre lo indica, se guardará el nombre o los nombres del trabajador, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

*paterno\_trabajador.*

En este atributo se guardará el apellido paterno de los trabajadores, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

*materno\_trabajador.*

En este atributo se guardará el apellido materno de los trabajadores, es de tipo VARCHAR, forzosamente tiene que tener un valor y este puede ser repetido.

*contrasenia.*

En este atributo se guardará la contraseña de los trabajadores, es de tipo VARCHAR, forzosamente tiene que tener un valor y puede que se repita.

*id\_privilegios.*

En este atributo se guardará el ID del privilegio que tienen el trabajador, es de tipo entero, forzosamente tiene que tener un valor y puede ser repetido. Este atributo es una llave foránea.

### 6.1.7 Tabla privilegio



Figura 9. Tabla privilegio.

El propósito de la tabla privilegio es almacenar los tipos de privilegios que se tienen del lado del administrador. Tal y como se muestra en la Figura 9, esta tabla cuenta con los siguientes atributos.

*id\_privilegio.*

Este atributo identifica de manera única el tipo de privilegio en el Sistema, es de tipo entero, forzosamente tiene que tener un valor que no puede ser repetido y es

la llave primaria de esta tabla.

*nombre\_privilegio.*

En este atributo, como su nombre lo indica, se guardarán los nombres de los privilegios que puede tener un trabajador, es de tipo VARCHAR, forzosamente tiene que tener un valor y este no puede ser repetido.



### 6.1.8 Diagrama del Modelo Entidad Relación de la Base de Datos

En la Figura 10 se muestra el diagrama completo del modelo Entidad Relación de la Base de Datos que se usó para el desarrollo del Sistema de Reservación en Línea de la Cafetería.

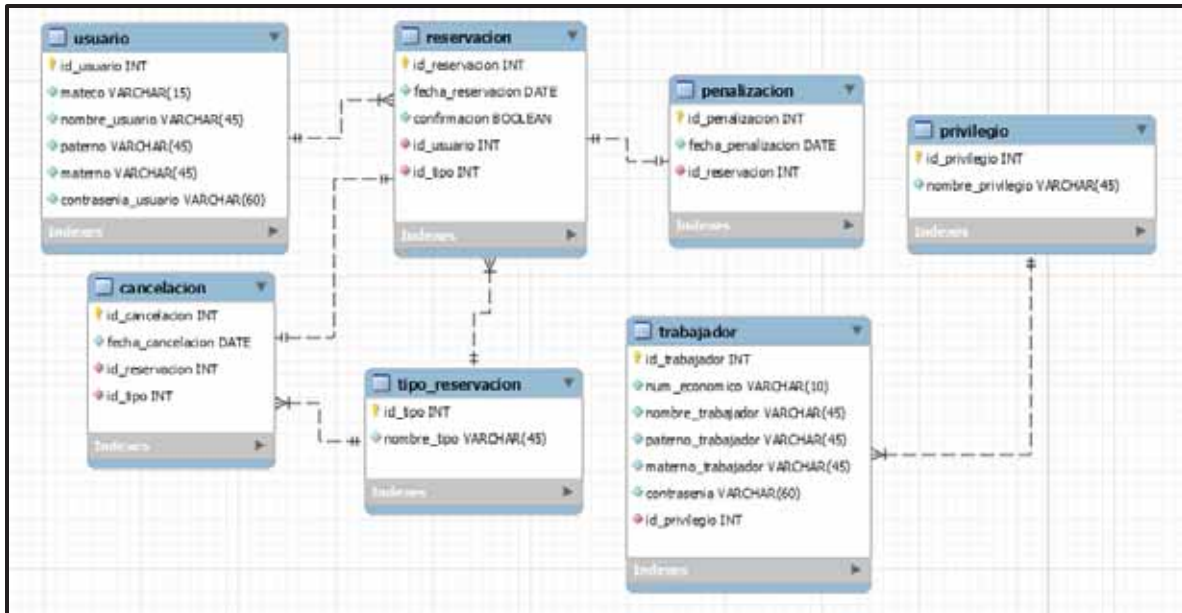


Figura 10. Modelo de la Base de Datos.

## 6.2 SISTEMA USUARIO

El Sistema Usuarios es donde los usuarios podrán hacer las reservaciones o cancelaciones para el servicio de cafetería y este se divide en:

- Registro.
- Login.
- Reservación.
- Cancelación.

### 6.2.1 Registro

Para que el usuario pueda tener acceso al Sistema de Reservación en Línea para el servicio de cafetería, antes deberá registrarse en él, llenado los campos del formulario que se muestra en la Figura 11, una vez que el usuario llena los campos pasa por diferentes validaciones.

**Servicio** de Cafeteria

Registro de Usuario

Matricula / No. Económico

Nombre

Apellido Paterno

Apellido Materno

Contraseña

Repetir Contraseña

Registrar

[Iniciar Sesión](#)

**Figura 11. Formulario para el registro de los Usuarios.**

La primera de ellas es que hasta que no esté bien la contraseña no se habilitará el botón de “Registrar” tal y como se muestra en la Figura 12, esto se hace con JavaScript usando las validaciones de *Bootstrap* que son **has-success** y **has-error**.

<div style="border: 1px solid green; padding: 5px; margin-bottom: 5px;">****</div> <div style="border: 1px solid green; padding: 5px; margin-bottom: 5px;">****</div> <p style="color: green; font-weight: bold;">CORRECTO!!!</p> <div style="background-color: #3498db; padding: 5px; text-align: center; color: white; font-weight: bold;">Registrar</div>	<div style="border: 1px solid red; padding: 5px; margin-bottom: 5px;">*****</div> <div style="border: 1px solid red; padding: 5px; margin-bottom: 5px;">*****</div> <p style="color: red; font-weight: bold;">NO SON IGUALES!!!</p> <div style="background-color: #3498db; padding: 5px; text-align: center; color: white; font-weight: bold;">Registrar</div>
--	--

**Figura 12. Contraseña correcta vs contraseña incorrecta.**

Una vez que el usuario llena todos los campos del formulario “*Registro de Usuario*” y el campo de “*Contraseña*” y “*Repetir Contraseña*” son correctos, se verifica lo siguiente:

- Lo que puso en el campo “*Matricula / No. Económico*” sea un número.
- El usuario no debe existir en la Base de Datos.

Para verificar que el usuario puso solo números en el campo “*Matricula / No. Económico*”, se hace con una función de PHP llamada *is\_numeric('cadena')*, esta verifica que la cadena que se le mando sea un número, si es así el valor que devuelve es “*true*”, pero si la cadena que recibió no es un número en valor que devuelve es “*false*” (ver Figura 13).

Es número bool(true)	No es número bool(false)
-------------------------	-----------------------------

Figura 13. Resultados de la verificación de la función *is\_numeric()*.

Para verificar que el usuario no existe en la Base de Datos, se hace tratando de insertar al usuario a la Base de Datos como la matricula o número económico son únicos, si este ya existe mandará un mensaje de error, tal y como se muestra en la Figura 14 y no se permitirá insertar al usuario en la Base de Datos. En caso contrario se insertará al usuario en la Base de Datos y su registro quedará hecho.

```
["errno"]=>
int(1062)
["error"]=>
string(41) "Duplicate entry '789654' for key 'matEco'"
["error_list"]=>
array(1) {
  [0]=>
  array(3) {
    ["errno"]=>
    int(1062)
    ["sqlstate"]=>
    string(5) "23000"
    ["error"]=>
    string(41) "Duplicate entry '789654' for key 'matEco'"
  }
}
```

Figura 14. Mensaje de error al intentar registrar un Usuario ya registrado.

Al usuario se le hace saber si se pudo registrar con éxito, si sus datos están incorrectos o si ya está registrado en el Sistema (ver Figura 15).

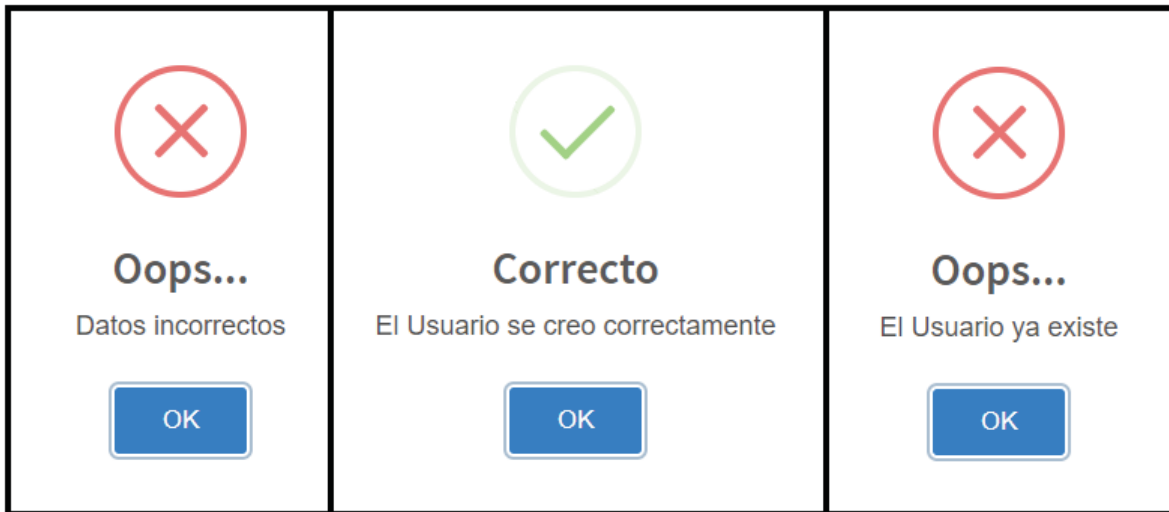


Figura 15. Mensajes que se pueden mostrar al registrar un Usuario.

Tal y como se muestra el Figura 16 las contraseñas se insertan en la Base de Datos encriptadas y para esto se utiliza la función de PHP `password_hash('contraseña', 'algoritmo que se utilizara para hacer el hash')`.

id_usuario	matEco	nombre_usuario	paterno	materno	contrasenia_usuario
1	2173002538	Alejandro	Cruz	Acasio	\$2y\$12\$GqCmNdkkzkqUViBy.IMForJ6TqtsvFX8KdcP23vS3e9Mfy453vtm
2	2156038586	Rebeca	Castro	Ortal	\$2y\$12\$ASKntNBX8cz7XUe/mHAb9.TzU40pPHxEckRWvvd1FMSA/FneF9Ege
3	2154095531	David	Colin	Diaz	\$2y\$12\$SdEa/CcXzrYogBQlaLwR.0.6lc84Wg7zXYrbMMttzEdCEgns74K
4	2148914415	Karen	Rivera	Vidal	\$2y\$12\$CncbK7WPOTmEz1M6ScX5OKBBIzIzdDdGzWjQ/q5xQl85kSn8D2Ua
5	2147214415	Monica	Jaramillo	Ruiz	\$2y\$12\$uLuEM3xpWN.US6wRDq2Pi.xPoabzM5StyBukFWOcEDoI6J0gb4F0u
7	12	Carla	Magaña	López	\$2y\$12\$VyQTbUh7vq3UYT9tnIyvFuzgFBbuigfwJIKdxkltvk..gZduoncu

Figura 16. Tabla con información de Usuarios.

## 6.2.2 Login

Esta es la parte donde el usuario podrá tener acceso al Sistema de Reservación en Línea para el servicio de cafetería. Para eso tendrá que llenar los campos del formulario que se muestra en la Figura 17.

Figura 17. Formulario Login Usuario.

Una vez que el usuario llenó los campos, estos se tendrán que pasar por diferentes validaciones:

- Que el usuario exista en la Base de Datos
- Que la contraseña que escribió el usuario sea la misma que la de la Base de Datos.

Para verificar que el usuario existe en la Base de Datos, se hace una consulta de acuerdo con el número que proporcionó en el campo “*usuario*”, si esta nos devuelve un número mayor a cero, quiere decir que existe un registro con los datos del usuario (ver Figura 18) y se hace la verificación de la contraseña.

<p>Usuario Registrado</p> <p><code>int(1)</code></p>	<p>Usuario NO esta Registrado</p> <p><code>int(0)</code></p>
--	--

Figura 18. Resultado de la Base de Datos, Usuario Registrado vs Usuario no Registrado.

Recordemos que la contraseña que está en la Base de Datos esta encriptada y que la contraseña que proporcionó el usuario no está encriptada. Cuando se encripta la contraseña para insertarla en la Base de Datos no hay manera de desencriptarla, es como una carretera de un solo sentido tenemos forma de llegar, pero no tenemos forma de regresar. Hay una forma de comparar la contraseña que proporcionó el usuario con la contraseña que está registrada en la Base de Datos, usando la función de PHP `password_verify` (*'contraseña que proporcionó el usuario', 'contraseña almacenada en la Base de Datos'*), compara ambas contraseñas si son iguales manda un valor de “*true*”, pero si son diferentes manda un valor de “*false*” (ver Figura 19).

Datos correctos bool(true)	Datos incorrectos bool(false)
-------------------------------	----------------------------------

Figura 19. Resultados de la verificación de la función *password\_verify()*.

Si el usuario está registrado en la Base de Datos y llenó correctamente los campos del login se le mandará un mensaje de éxito, en caso contrario se le mandará un mensaje de error (ver Figura 20).

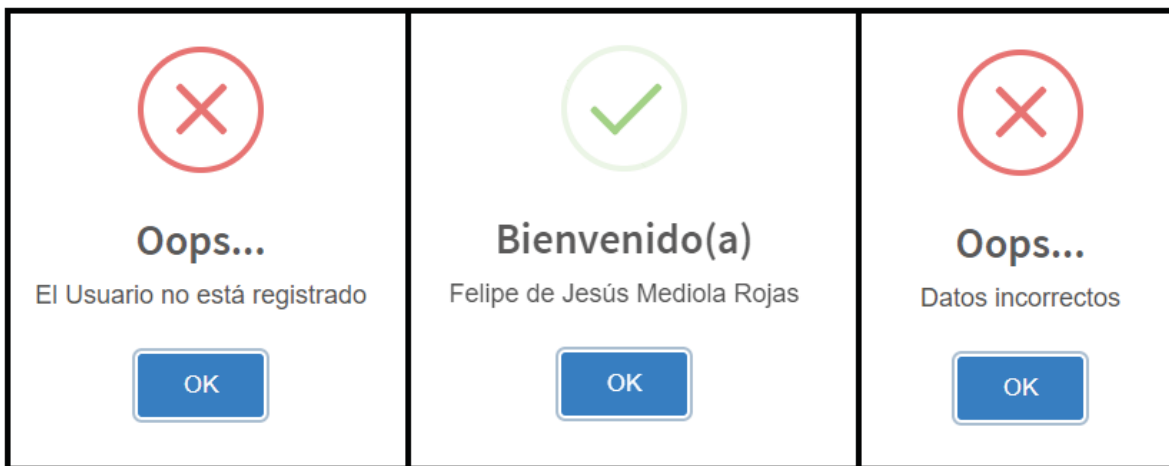


Figura 20. Mensajes que se pueden mostrar al logear un Usuario.

### 6.2.3 Reservación

Para que el usuario pueda hacer una reservación tiene que elegir una de las dos opciones, ya sea para desayuno o para comida (ver Figura 21), por default siempre estará seleccionada la opción de desayuno, pero el usuario es libre de cambiar la opción a comida.

The image shows a form titled 'Reservación'. Under the heading 'Tipo de Reservación', there are two radio button options: 'Desayuno' (which is selected) and 'Comida'. At the bottom of the form is a blue button labeled 'Reservar'.

Figura 21. Formulario Reservación.

Una vez que el usuario hace clic en el botón Reservar (ver Figura 21) se ejecuta un semáforo en PHP que funciona de la siguiente manera

- Se obtiene un ID para un semáforo determinado (***sem\_get()***).
- Se pide acceso al semáforo (***sem\_acquire()***)
- Se realizan los procesos que se deseen, en este caso la reservación.
- Se libera el semáforo (***sem\_release()***), de modo que el semáforo le de paso a otros usuarios que puedan estar esperando.

Una vez que se logra acceder al semáforo se hacen varias validaciones. La primera es que este dentro de la hora permitida para hacer la reservación.

- Para el desayuno la reservación estará permitida de las 7:00 en adelante.
- Para la comida la reservación estará permitida de las 13:00 en adelante.

Si no está dentro de la hora permitida para hacer la reservación se le informará al usuario, tal y como se muestra en la Figura 22.



The image shows a web form titled "Registro de Reservaciones" with a subtitle "elige una opcion". The form has a section labeled "Reservación" containing a red-bordered box with the text "NOTA: Todavía NO es la hora de reservar." Below this, there is a section titled "Tipo de Reservación" with two radio button options: "Desayuno" (which is selected) and "Comida". At the bottom of the form is a blue button labeled "Reservar".

Figura 22. Mensaje de error. Fuera de la hora para hacer la Reservación.

La segunda validación que se hace es saber si todavía hay cupo para hacer reservaciones, ya que solo se permiten hacer 50 reservaciones para desayuno y 50 reservaciones para comida al día, en caso de ya no haber cupo se le notifica al usuario, tal y como se muestra en la Figura 23. Para esta validación se hace una consulta a la Base de Datos la cual regresa el número de registros de reservaciones que hay de acuerdo al tipo que eligió el usuario y la fecha en que se hizo la reservación (ver Figura 24).

Registro de Reservaciones elige una opción

Reservación

**NOTA: Reservaciones agotadas por el día de hoy.**

Tipo de Reservación

Desayuno

Comida

Reservar

Figura 23. Mensaje de error. Ya no hay reservaciones.

Todavía hay Reservaciones	Reservaciones agotadas
int(10)	int(50)

Figura 24. Respuesta de la consulta a la Base de Datos para validar el cupo.

La tercera validación que se hace es saber si el usuario ya hizo una reservación del servicio que solicitó, desayuno o comida. Esto se hace con una consulta a la Base de Datos la cual regresara el número de reservaciones hechas por el usuario dependiendo del tipo de servicio y la fecha (ver Figura 26), si el número que se obtiene es mayor a cero se le notifica al usuario que ya ha hecho una reservación, tal y como se muestra en la Figura 25.

Registro de Reservaciones elige una opción

Reservación

**NOTA: Ya has hecho una reservación el día de hoy.**

Tipo de Reservación

Desayuno

Comida

Reservar

Figura 25. Mensaje de error. El usuario ya hizo una reservación.

No has hecho una reservacion	Ya has hecho una reservacion
int(0)	int(1)

Figura 26. Respuesta de la consulta a la Base de Datos para validar si el Usuario ya ha hecho una reservación previa.

La cuarta validación que se hace es saber si el usuario nunca ha hecho una reservación de acuerdo con el tipo de servicio que pidió (desayuno o comida). Esto se hace con una consulta la Base de Datos de todas las reservaciones que ha hecho el usuario (ver Figura 27), dependiendo del tipo de servicio si esta devuelve un número igual a cero, quiere decir que nunca ha hecho una reservación y por lo tanto se le permite hacer la reservación (ver Figura 30) de lo contrario se tiene que verificar que el usuario no este penalizado.



Puedes Reservar int(0)	Verificar Penalizacion int(10)
---------------------------	-----------------------------------

Figura 27. Respuesta de la consulta a la Base de Datos para validar si alguna vez ha reservado el Usuario.

La quinta y última validación consiste en saber si el usuario está penalizado o no. Esta se divide en dos partes. En la primera parte se busca saber si el usuario asistió o canceló su última reservación y se hace con una consulta a la Base de Datos de la última reservación hecha por el usuario dependiendo del tipo de servicio que solicitó. Si el atributo confirmación tiene el valor de 1 o "true" (ver Figura 28), se le permite hacer la reservación, de lo contrario se penaliza al usuario. En la segunda parte se busca saber si el usuario está penalizado y se hace con una consulta a la Base de Datos, se compara la fecha de la penalización con la fecha en que solicitó la reservación y si la fecha de la penalización es mayor que la fecha en que se solicitó la reservación, quiere decir que el usuario está penalizado y se le hace saber al usuario, tal y como se muestra en la Figura 29, en caso contrario se le deja hacer la reservación (ver Figura 30).

Confirmado	No Confirmado
<pre>array(5) {   ["id_reservacion"]=&gt;   string(2) "38"   ["fecha_reservacion"]=&gt;   string(10) "2017-12-20"   ["confirmacion"]=&gt;   string(1) "1"   ["id_usuario"]=&gt;   string(2) "12"   ["id_tipo"]=&gt;   string(1) "1" }</pre>	<pre>array(5) {   ["id_reservacion"]=&gt;   string(2) "31"   ["fecha_reservacion"]=&gt;   string(10) "2017-12-18"   ["confirmacion"]=&gt;   string(1) "0"   ["id_usuario"]=&gt;   string(1) "9"   ["id_tipo"]=&gt;   string(1) "1" }</pre>

Figura 28. Valor de confirmación igual a true vs confirmación igual a false.

Registro de Reservas elige una opción

Reservación

**NOTA: Estás penalizado NO puedes reservar.**

Tipo de Reservación

Desayuno

Comida

Reservar

Figura 29. Mensaje de error. El usuario está Penalizado.

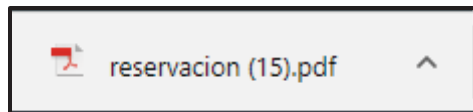


Figura 30. Descarga de la reservación.

En la Figura 31 se muestra un ejemplo de una reservación.

Coordinación de Servicios Universitarios

Sistema de reservación en línea

41

Nombre: Felipe de Jesús Mediola Rojas

Matricula: 15

Reservación hecha para desayuno

Esta reservación es válida solo para el día Vienes 22-12-2017

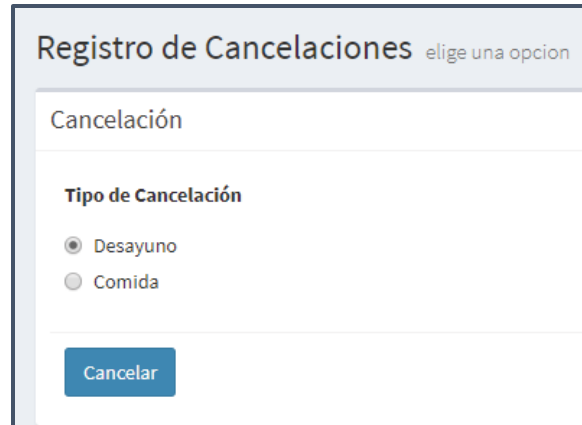
NOTA: Esta reservación no es un vale tendrás que pagar por el servicio

Desayuno \$1.50
Comida \$2.50

Figura 31. Ejemplo del PDF de una reservación.

## 6.2.4 Cancelación

Para que el usuario pueda hacer una cancelación tiene que elegir una de las dos opciones ya sea para desayuno o para comida (ver Figura 32), por default siempre estará seleccionada la opción de desayuno, pero el usuario es libre de cambiar la opción a comida.



Registro de Cancelaciones elige una opción

Cancelación

**Tipo de Cancelación**

Desayuno

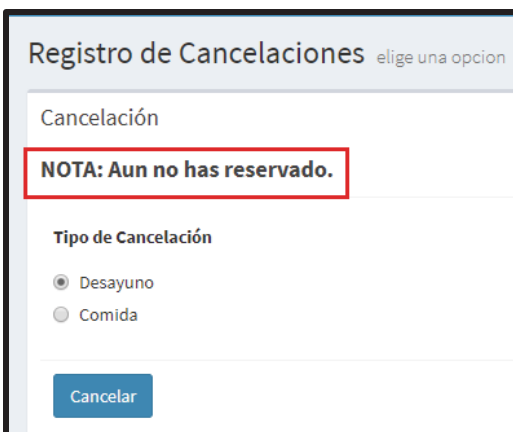
Comida

Cancelar

Figura 32. Formulario Cancelación.

Una vez que el usuario hace clic en el botón Cancelar (ver Figura 32) se hacen varias validaciones.

La primera validación que se hace es saber si el usuario ya hizo una reservación para poder cancelarla. Esto se hace con una consulta a la Base de Datos, en la cual se obtiene el número de registros de las reservaciones hechas por el usuario de acuerdo al tipo que solicitó para hacer la cancelación y la fecha en que solicito la cancelación (ver Figura 34). Si esta consulta nos devuelve un número mayor a cero quiere decir que si hay una reservación para cancelar, de lo contrario el usuario todavía no ha hecho una reservación y se le hace saber al usuario, tal y como se muestra en la Figura 33.



Registro de Cancelaciones elige una opción

Cancelación

**NOTA: Aun no has reservado.**

**Tipo de Cancelación**

Desayuno

Comida

Cancelar

Figura 33. Mensaje error. El usuario no ha reservado aún.

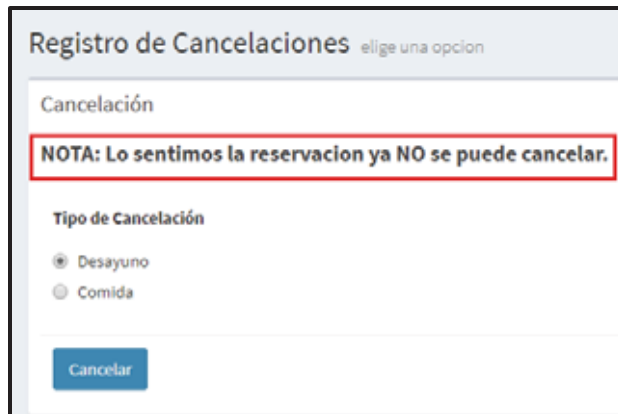
Tiene reservaciones hechas	No tiene reservaciones hechas
int(1)	int(0)

Figura 34. Resultado de la consulta a la Base de Datos para saber si el usuario ya reservo.

La segunda validación que se hace es saber que el usuario este dentro de la hora y día permitido para hacer la cancelación.

- Para el desayuno la cancelación está permitida antes de las 12:59.
- Para la comida la cancelación está permitida antes de las 15:59.
- El día tiene que ser el mismo en que se hizo la reservación.

Si no está dentro de la hora y día permitido para hacer la cancelación se le informará al usuario, tal y como se muestra en la Figura 35.



Registro de Cancelaciones elige una opcion

Cancelación

**NOTA: Lo sentimos la reservación ya NO se puede cancelar.**

Tipo de Cancelación

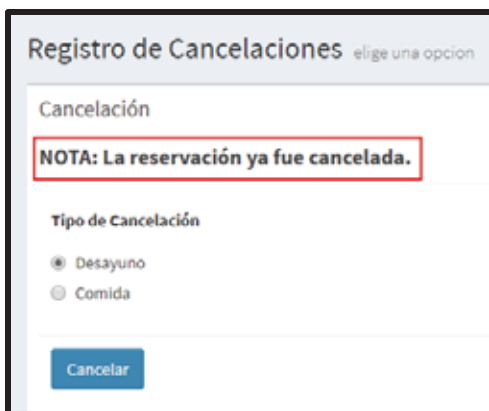
Desayuno

Comida

Cancelar

Figura 35. Mensaje error. El usuario no está en fecha ni hora.

La tercera y última validación que se hace es checar que la reservación no se haya validado o este ya cancelada. Para esto se hace una consulta a la Base de Datos de la reservación del usuario, si el valor del atributo “confirmacion” es igual a uno la reservación ya no se puede cancelar (ver Figura 37). Se le informa al usuario, tal y como se muestra en la Figura 36 en caso contrario se hace la cancelación.



Registro de Cancelaciones elige una opcion

Cancelación

**NOTA: La reservación ya fue cancelada.**

Tipo de Cancelación

Desayuno

Comida

Cancelar

Figura 36. Mensaje error. La reservación ya fue cancelada.

El usuario ya Canelo	El usuario no Canelo
<pre>array(5) {   ["id_reservacion"]=&gt;   string(2) "34"   ["fecha_reservacion"]=&gt;   string(10) "2017-12-18"   ["confirmacion"]=&gt;   string(1) "1"   ["id_usuario"]=&gt;   string(1) "9"   ["id_tipo"]=&gt;   string(1) "2" }</pre> <pre>array(4) {   ["id_cancelacion"]=&gt;   string(1) "5"   ["fecha_cancelacion"]=&gt;   string(10) "2017-12-18"   ["id_reservacion"]=&gt;   string(2) "34"   ["id_tipo"]=&gt;   string(1) "2" }</pre>	<pre>array(5) {   ["id_reservacion"]=&gt;   string(2) "31"   ["fecha_reservacion"]=&gt;   string(10) "2017-12-18"   ["confirmacion"]=&gt;   string(1) "0"   ["id_usuario"]=&gt;   string(1) "9"   ["id_tipo"]=&gt;   string(1) "1" }</pre> <p>NULL</p>

Figura 37. Resultado de la Base de Datos, confirmación igual a uno vs confirmación igual a cero.

Una vez que se registra la cancelación en la Base de Datos, se descarga un archivo PDF en el cual se le informa al usuario que la reservación ha sido cancelada (ver Figura 38).

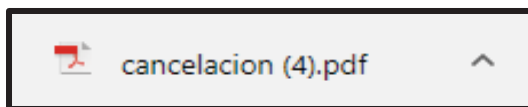


Figura 38. Descarga de la cancelación en PDF.

En la Figura 39 se puede apreciar un ejemplo de una cancelación.

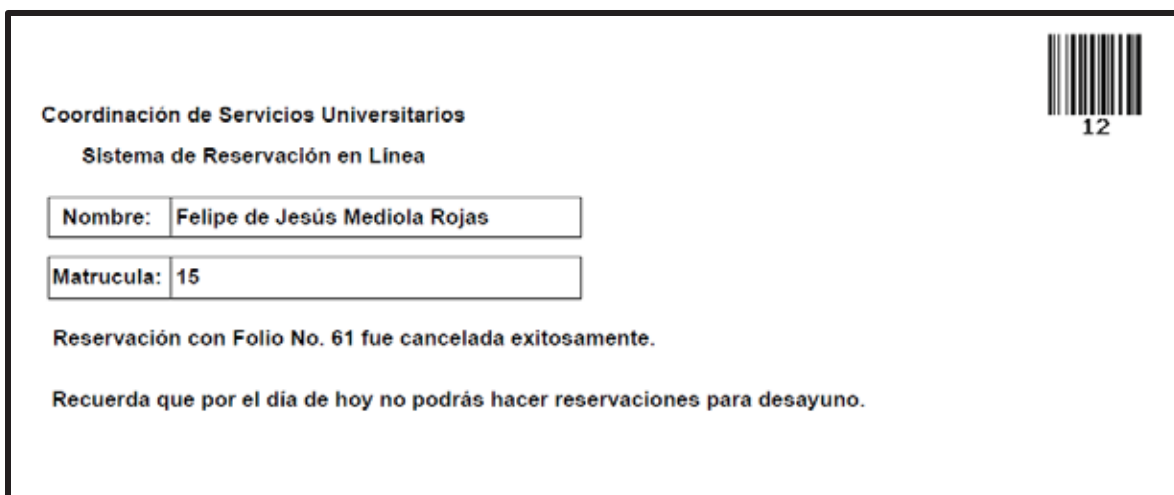


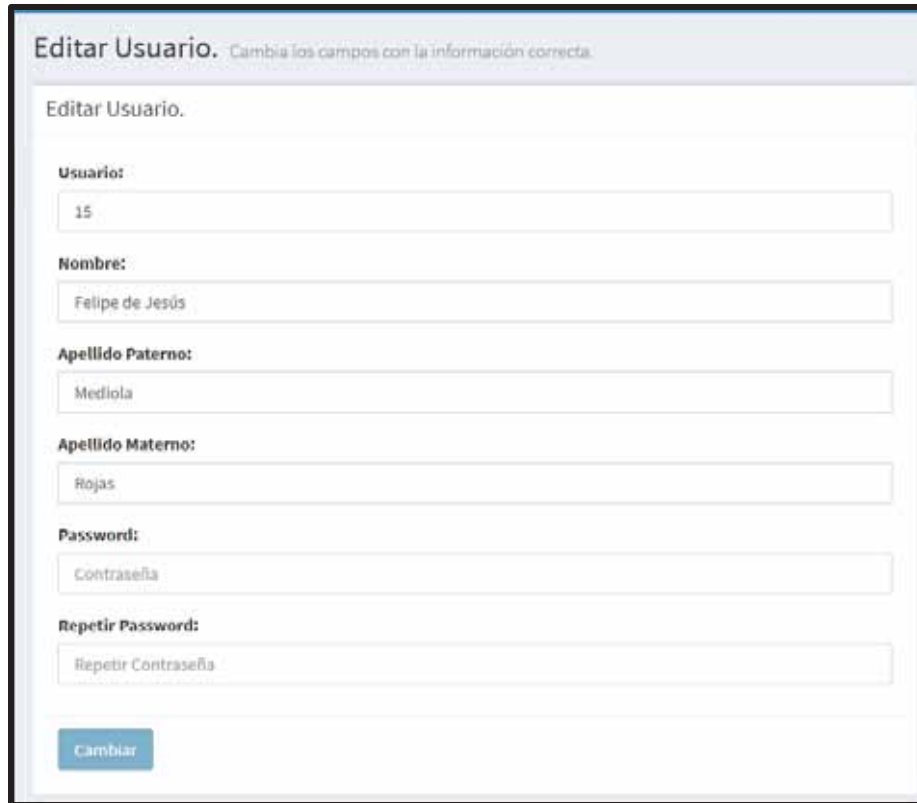
Figura 39. Ejemplo del PFD de una cancelación.

En la Figura 40 se muestran algunos de los registros de la tabla cancelación.

id_cancelacion	fecha_cancelacion	id_reservacion	id_tipo
1	2017-11-28	11	1
2	2017-12-18	13	1
3	2017-12-18	18	1
4	2017-12-18	29	2
5	2017-12-18	34	2
9	2017-12-20	38	1
10	2017-12-22	41	1

Figura 40. Tabla con cancelaciones registradas.

Adicionalmente el Sistema de Usuario, permite que el usuario pueda hacer ajustes a sus datos, tal y como se muestra en la Figura 41.



Editar Usuario. Cambia los campos con la información correcta.

Editar Usuario.

**Usuarios:**  
15

**Nombre:**  
Felipe de Jesús

**Apellido Paterno:**  
Mediola

**Apellido Materno:**  
Rojas

**Password:**  
Contraseña

**Repetir Password:**  
Repetir Contraseña

**Cambiar**

Figura 41. Formulario de Editar Usuario.

Este funciona similar al Registro de Usuario, la diferencia es que en vez de insertar un nuevo registro en la Base de Datos, se modifica un registro ya existente. Como se puede observar en la Figura 41, el formulario ya viene precargado con los datos del Usuario. Los datos pueden ser modificados en caso de tener algún error, de igual manera el botón de “Cambiar” no se activara hasta que el usuario tenga iguales los campos “Contraseña” y “Repetir Contraseña”.

Otra opción que tiene el Usuario es un Historial de Reservaciones, Cancelaciones y Penalizaciones (ver Figura 42).

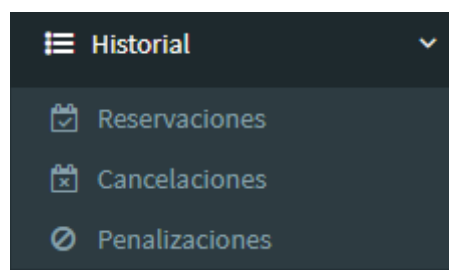


Figura 42. Opciones de Historial.

Los Historiales muestran la siguiente información

- Folio
- Fecha
- Tipo de servicio

En las Figura 43 se observa un ejemplo del Historial de Reservaciones y en la Figura 44 se observa un ejemplo del Historial de Cancelaciones. En el Historial de Penalización hay una pequeña variante, ya que se indica si el usuario está o no está penalizado, tal y como se muestra en la Figura 45.

Folio	Fecha de Reservacion	Tipo
31	2017-12-18	desayuno
34	2017-12-18	comida
41	2017-12-22	desayuno

Mostrando 1 a 3 de 3 resultados

Figura 43. Historial de Reservaciones.

Folio	Fecha de Reservacion	Tipo
34	2017-12-18	comida
41	2017-12-22	desayuno

Mostrando 1 a 2 de 2 resultados

Figura 44. Historial de Cancelaciones.

Historial de Penalizaciones - Servicio de Cafetería

Información sobre las Penalizaciones

Buscar

Folio	Fecha de Penalización	Tipo
31 <b>No penalizado</b>	2017-12-21	desayuno
Folio	Fecha de Penalización	Tipo

Mostrando 1 a 1 de 1 resultados

Anterior **1** Siguiente

Figura 45. Historial de Penalizaciones.

En la Figura 46 se muestra toda la interfaz del Sistema de Reservación en Línea para el servicio de cafería (Usuario).

**Servicio de Cafetería** Felipe de Jesús Mediola Rojas

Felipe de Jesús Mediola Rojas  
Online

Buscar...

**Menú**

- Inicio
- Reservacion
- Cancelacion
- Historial**
  - Reservaciones
  - Cancelaciones
  - Penalizaciones

Servicio de Cafetería Re [Ajustes](#) [Cerrar Sesión](#)

Avisos

Por el momento no hay avisos !!!

Copyright © 2016-2018 UAM-AZC. Todos los derechos reservados. Version 2.4.0

Figura 46. Interfaz completa del Sistema de Reservación en Línea para el Servicio de Cafetería (Usuario).



## 6.3 SISTEMA ADMINISTRADOR

El Sistema Administrador es donde los Administradores podrán hacer validaciones a las reservaciones, crear, editar o eliminar administradores o usuarios, hacer reservaciones, cancelaciones o quitar penalizaciones, obtener datos estadísticos y reportes con respecto a los servicios ofrecidos, este se divide en:

- Login.
- Crear, editar o eliminar Administradores.
- Crear, editar o eliminar Usuarios.
- Reservar, cancelar o quitar penalización.
- Validar.
- Datos Estadísticos y Reportes.

### 6.3.1 Login

Esta es la parte donde el Administrador podrá tener acceso al Sistema Administrador, para eso tendrá que llenar el formulario que se muestra en la Figura 47.



The image shows a login form for the 'Servicio de Cafeteria Administrador'. The form has a light gray header with the title 'Servicio de Cafeteria Administrador'. Below the header, the text 'Iniciar Sesión' is centered. There are two input fields: 'Usuario' with a user icon and 'Password' with a lock icon. At the bottom, there is a blue button labeled 'Iniciar Sesión'.

Figura 47. Formulario de Login Administrador.

Una vez que el usuario llena los campos, estos tendrán que pasar por diferentes validaciones:

- Que el Administrador exista en la Base de Datos
- Que la contraseña que escribió el Administrador sea la misma que la de la Base de Datos.

Para verificar que el Administrador existe en la Base de Datos se hace una consulta con el número que proporciono en el campo “*Usuario*”, si esta nos devuelve un número mayor a cero quiere decir que existe un registro con los datos del Administrador (ver Figura 48) y se pasa a verificar la contraseña.

Administrador Registrado <code>int(1)</code>	Administrador NO Registrado <code>int(0)</code>
---	--

Figura 48. Resultado de la Base de Datos, Usuario Registrado vs Usuario no Registrado.

Recordemos que la contraseña que está en la Base de Datos esta encriptada y que la contraseña que proporciono el Administrador no está encriptada. Cuando se encripta la contraseña para insertarla en la Base de Datos no hay manera de desencriptarla, es como una carretera de un solo sentido tenemos forma de llegar, pero no tenemos forma de regresar. Hay una forma de comparar la contraseña que proporciono el Administrador con la contraseña que está registrada en la Base de Datos, usando la función de PHP ***password\_verify***(*contraseña que proporcionó el Administrador*, *contraseña almacenada en la Base de Datos*), compara ambas contraseñas si son iguales manda un valor de “*true*”, pero si son diferentes manda un valor de “*false*” (ver Figura 49).

Datos correctos <code>bool(true)</code>	Datos incorrectos <code>bool(false)</code>
--	---

Figura 49. Resultados de la verificación de la función *password\_verify()*.

Si el Administrador está registrado en la Base de Datos y llenó correctamente los campos del login se le mandará un mensaje de éxito, en caso contrario se le mandará un mensaje de error (ver Figura 50).

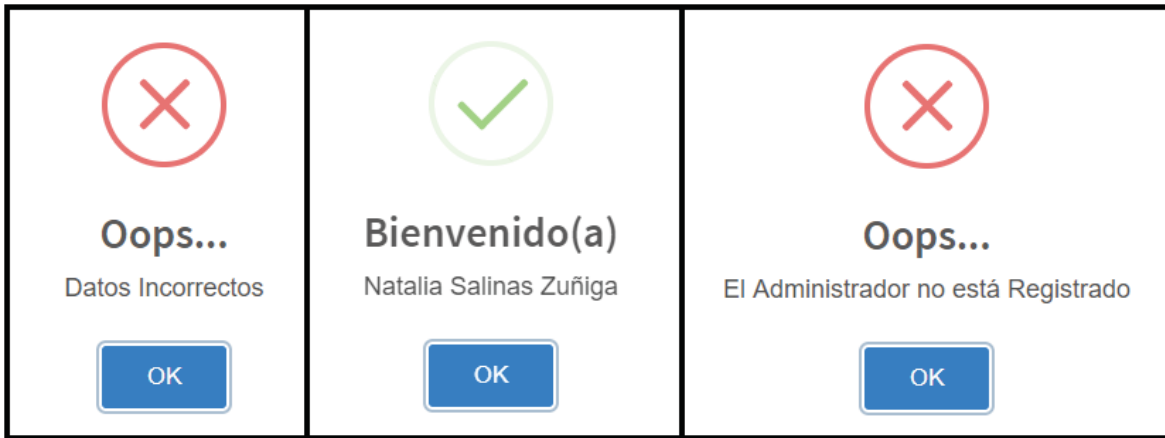


Figura 50. Mensajes que se pueden mostrar al logear un Administrador.

### 6.3.2 Crear, actualizar y eliminar Administradores

Para crear un Administrador, el Administrador tiene que tener el privilegio de “*administrador*”, si no es así no podrá hacerlo, para esto se hace una consulta a la Base de Datos para saber qué tipo de privilegio tiene el Administrador (ver Figura 53).

- Si es un Administrador con privilegio de “*administrador*”.
- Si es un Administrador con privilegio de “*validar*”.

En la Figura 51 se muestra la Interfaz del Administrador con privilegio “*administrador*”.

**Servicio de Cafeteria** Natalia Salinas Zuñiga

**Natalia Salinas Zuñiga**  
Online

Buscar...

**Menú de Administración**

- Inicio
- Estadísticas <
- Graficas <
- Reservaciones <
- Cancelaciones <
- Penalizaciones <
- Asistencia <
- Usuarios <
- Administradores <

**COORDINACIÓN DE SERVICIOS UNIVERSITARIOS.**  
**Servicio de Cafeteria.** Administrador

**Avisos**

Por el momento no hay avisos.!!

Copyright © 2016-2018 **UAM-AZC**. Todos los derechos reservados. Version 2.4.0

Figura 51. Interfaz Administrador con privilegio "administrador".

En la Figura 52 se muestra la Interfaz del Administrador con privilegio “validar”.



Figura 52. Interfaz Administrador con privilegio tipo "validar".

Trabajador con privilegio Validar	Trabajador con privilegio Administrador
<pre>array(7) {   ["id_trabajador"]=&gt;   string(2) "17"   ["num_economico"]=&gt;   string(2) "54"   ["nombre_trabajador"]=&gt;   string(6) "Monica"   ["paterno_trabajador"]=&gt;   string(9) "Jaramillo"   ["materno_trabajador"]=&gt;   string(5) "Mejia"   ["contrasenia"]=&gt;   string(60) "\$2y\$12\$U8Qze:   ["id_privilegios"]=&gt;   string(1) "2" }</pre>	<pre>array(7) {   ["id_trabajador"]=&gt;   string(2) "18"   ["num_economico"]=&gt;   string(2) "22"   ["nombre_trabajador"]=&gt;   string(7) "Natalia"   ["paterno_trabajador"]=&gt;   string(7) "Salinas"   ["materno_trabajador"]=&gt;   string(7) "Zuñiga"   ["contrasenia"]=&gt;   string(60) "\$2y\$12\$MhvKj:   ["id_privilegios"]=&gt;   string(1) "1" }</pre>

Figura 53. Comparativa de los resultados de la Base de Datos del campo privilegio.

Una vez que se comprobó que el Administrador tiene privilegio de “administrador”, para poder crear un Administrador se tiene que llenar el formulario que se muestra en la Figura 54, una vez que el Administrador llena los campos pasa por diferentes validaciones.

Crear Administrador llena los siguientes campos para crear un nuevo administrador

Crear Administrador

**Usuario:**

**Nombre:**

**Apellido Paterno:**

**Apellido Materno:**

**Tipo de Administrador**

ADMINISTRADOR

SOLO VALIDAR

**Password:**

**Repetir Password:**

Figura 54. Formulario Crear Administrador.

La primera validación que se hace es que hasta que no esté bien la contraseña no se habilitara el botón de “Crear” tal y como se muestra en la Figura 55, esto se hace con JavaScript usando las validaciones de Bootstrap que son *has-success* y *has-error*.

<p><b>Password:</b></p> <input type="password" value="...."/>	<p><b>Password:</b></p> <input type="password" value="....."/>
<p><b>Repetir Password:</b></p> <input type="password" value="...."/>	<p><b>Repetir Password:</b></p> <input type="password" value="....."/>
<p>CORRECTO!!!</p>	<p>NO SON IGUALES!!!</p>
<p>Crear</p>	<p>Crear</p>

Figura 55. Contraseña incorrecta vs contraseña incorrecta.

Una vez que el Administrador llenó todos los campos de “*Crear Administrador*” y el campo de “*Password*” y “*Repetir Password*” son correctos, se verifica lo siguiente:

- Lo que se puso en el campo “*Usuario*” sea un número.
- El Administrador no debe existir en la base de datos

Para verificar que el Administrador puso solo números en el campo “*Usuario*”, se hace con una función de PHP llamada *is\_numeric('cadena')*, esta verifica que la cadena que se le mando sea un número, si es así el valor que devuelve es “*true*”, pero si la cadena que recibió no es un número en valor que devuelve es “*false*” (ver Figura 56).

Es número	No es número
bool(true)	bool(false)

Figura 56. Resultados de la verificación de la función *is\_numeric()*.

Para verificar que el Administrador no existe en la Base de Datos, se hace tratando de insertar al Administrador a la Base de Datos, como el “*Usuario*” es único ya que es el número de empleado, si este ya existe mandará un mensaje de error, tal y como se muestra en la Figura 57 y no se permitirá insertar al Administrador en la Base de Datos, en caso contrario se insertará al Administrador en la Base de Datos y su registro quedará hecho.

```

["errno"]=>
int(1062)
["error"]=>
string(47) "Duplicate entry '41251' for key 'num_economico'"
["error_list"]=>
array(1) {
  [0]=>
  array(3) {
    ["errno"]=>
    int(1062)
    ["sqlstate"]=>
    string(5) "23000"
    ["error"]=>
    string(47) "Duplicate entry '41251' for key 'num_economico'"
  }
}

```

Figura 57. Mensaje de error al intentar registrar un Administrador ya registrado.

Al Administrador se le hace saber si se pudo crear con éxito el Administrador, si sus datos están incorrectos o si ya está registrado (ver Figura 58).

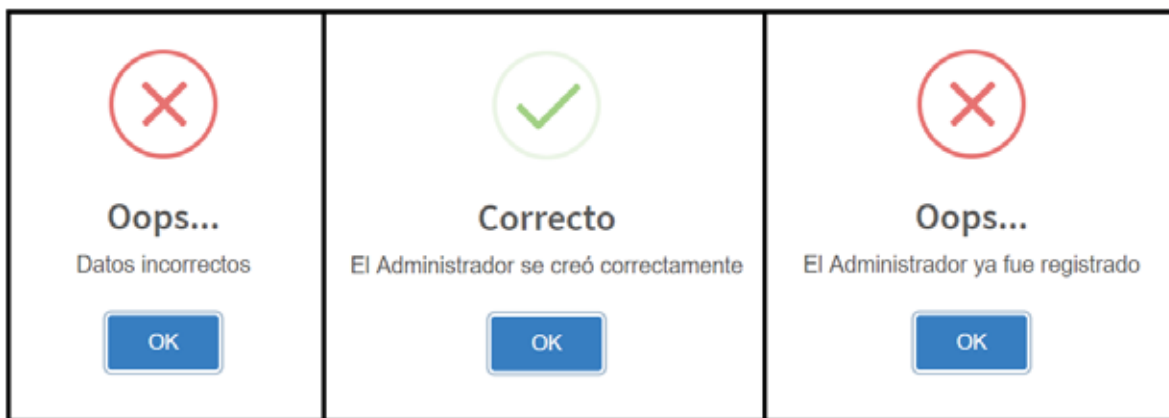


Figura 58. Mensajes que se pueden mostrar al crear un Administrador.

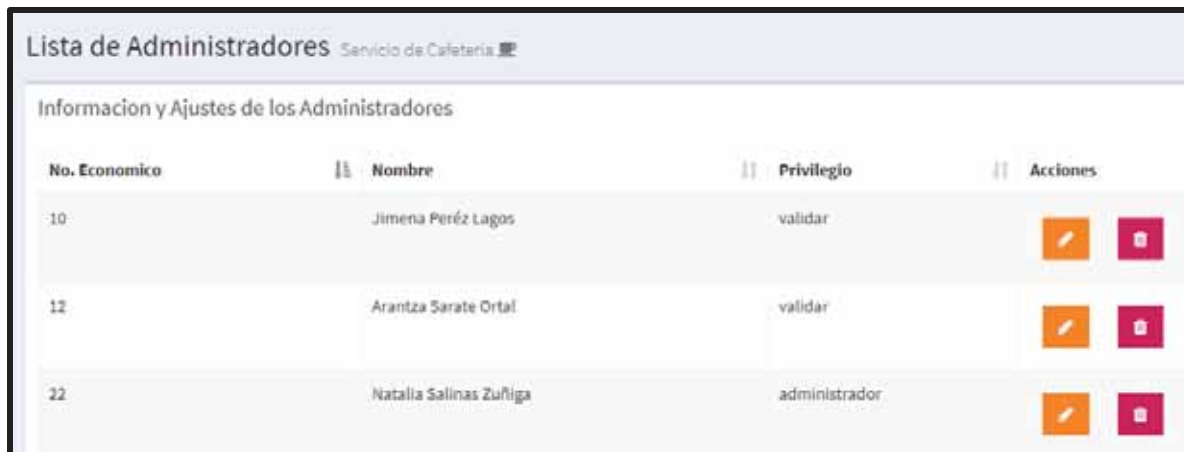
Tal y como se muestra en la Figura 59 la contraseña se inserta en la Base de Datos encriptada y para eso se utiliza la función de PHP **password\_hash**(*contraseña*, *algoritmo que se utilizara para hacer el hash*).

id_trabajador	num_economico	nombre_trabajador	paterno_trabajador	materno_trabajador	contrasenia	id_privilegios
2	12345	Samuel	Sarate	Solis	\$2y\$12\$soLLk66y6QZoW46zjyGi1OHVpt2tuQOya2ve...	2
12	30	Carlos	Castro	Mejia	\$2y\$12\$P6aXFPZUVDa7.UDHirmhxeQnbg/lV3EDS63i2...	2
13	33	Daniela	Mendiola	Solorio	\$2y\$12\$1y3F/q0GCV3FZq6udqdDe9Ip24L/BAEzX8A6...	1
14	36	Oscar	Reyes	Bravo	\$2y\$12\$sCpoY/c/U08G7uhqJKXLr.yC.NVoISnWM6h3...	2
15	10	Jimena	Peréz	Lagos	\$2y\$12\$3yei6F3tOa8ldozOVFH.5eHzF.VJF0zYhYZxj...	2
17	54	Monica	Jaramillo	Mejia	\$2y\$12\$U8QzeyvqyrQYtmA7qb8CzeRINDu/TQzpVaL...	2
18	22	Natalia	Salinas	Zuñiga	\$2y\$12\$MhvkjSEFfj/16rddjB.2M.imTpWR3E4Gn4JnHL...	1

Figura 59. Tabla con información de los Administradores.



Para editar a un Administrador, el Administrador tiene que tener el privilegio de “administrador”. Tal y como se muestra en la Figura 60 aparecerá una lista con todos los Administradores, se seleccionará el Administrador que se quiere editar. El editar un Administrador funciona similar al “Crear Administrador”, la diferencia es que en vez de insertar un nuevo registro en la Base de Datos se modifica un registro ya existente. Como se puede observar en la Figura 61 el formulario ya viene precargado con los datos del Administrador, de igual manera el botón de “Cambiar” no se activará hasta que el Administrador tenga iguales los campos “Contraseña” y “Repetir Contraseña”









No. Economico	Nombre	Privilegio	Acciones
10	Jimena Pérez Lagos	validar	 
12	Arantza Sarate Ortal	validar	 
22	Natalia Salinas Zuñiga	administrador	 

Figura 60. Lista de Administradores.

**Editar Administrador.** Cambia los campos con la información correcta.

Editar Administrador.

**Usuario:**

**Nombre:**

**Apellido Paterno:**

**Apellido Materno:**

**Tipo de Administrador**

ADMINISTRADOR

SOLO VALIDAR

**Password:**

**Repetir Password:**

Figura 61. Formulario Editar Administrador.

Para borrar a un Administrador, el Administrador tiene que tener el privilegio de “administrador”. Tal y como se muestra en la Figura 60, aparecerá una lista con todos los Administradores y se seleccionará el Administrador que se quiere borrar.

### 6.3.3 Crear, editar y eliminar Usuarios

Para que un Administrador pueda crear un usuario no se necesita saber qué tipo de privilegio tiene el Administrador (ver Figura 51 y 52).

Para crear a un usuario se tendrá que llenar el formulario que se muestra en la Figura 62, una vez que el Administrador llena los campos pasa por diferentes validaciones.

Crear Usuario llena los siguientes campos para crear un nuevo Usuario

Crear Usuario

**Usuario:**

**Nombre:**

**Apellido Paterno:**

**Apellido Materno:**

**Password:**

**Repetir Password:**

Figura 62. Formulario Crear Usuario.

La primera validación que se hace es que hasta que no esté bien la contraseña no se habilitará el botón de “*Crear*” tal y como se muestra en la Figura 63, esto se hace con JavaScript usando las validaciones de *Bootstrap* que son ***has-success*** y ***has-error***.

<p><b>Password:</b></p> <input type="text" value="...."/>	<p><b>Password:</b></p> <input type="text" value="....."/>
<p><b>Repetir Password:</b></p> <input type="text" value="...."/>	<p><b>Repetir Password:</b></p> <input type="text" value="....."/>
<p>CORRECTO!!!</p>	<p>NO SON IGUALES!!!</p>
<p><input type="button" value="Crear"/></p>	<p><input type="button" value="Crear"/></p>

Figura 63. Contraseña correcta vs contraseña incorrecta.

Una vez que el Administrador llena todos los campos del formulario “*Crear Usuario*” y el campo de “*Contraseña*” y “*Repetir Contraseña*” son correctos, se verifica lo siguiente:

- Lo que se puso en el campo “*Usuario*” sea un número.
- El usuario no debe existir en la Base de Datos

Para verificar que el Administrador puso solo números en el campo “*Usuario*”, se hace con una función de PHP llamada *is\_numeric('cadena')*, esta verifica que la cadena que se le mando sea un número, si es así el valor que devuelve es “*true*”, pero si la cadena que recibió no es un número en valor que devuelve es “*false*” (ver Figura 64).

Es número	No es número
bool(true)	bool(false)

Figura 64. Resultados de la verificación de la función *is\_numeric()*.

Para verificar que el usuario no existe en la Base de Datos, se hace tratando de insertar al usuario en la Base de Datos, como la matricula o número económico son únicos, si este ya existe mandará un mensaje de error, tal y como se muestra en la Figura 65 y no se permitirá insertar al usuario en la Base de Datos. En caso contrario se insertará al usuario en la Base de Datos y su registro quedara hecho.

```
["errno"]=>
int(1062)
["error"]=>
string(41) "Duplicate entry '789654' for key 'matEco'"
["error_list"]=>
array(1) {
  [0]=>
  array(3) {
    ["errno"]=>
    int(1062)
    ["sqlstate"]=>
    string(5) "23000"
    ["error"]=>
    string(41) "Duplicate entry '789654' for key 'matEco'"
  }
}
```

Figura 65. Mensaje de error al intentar registrar un Usuario ya registrado.

Para editar a un usuario no se tiene que verificar el tipo de privilegio que tiene el Administrador (ver Figura 51 y 52).

Como se muestra en la Figura 66 aparecerá una lista con todos los usuarios registrados y se seleccionará al usuario que se quiere editar. El editar un usuario funciona similar al “*Crear Usuarios*”, la diferencia es que en vez de insertar un nuevo registro en la Base de Datos se modifica un registro ya existente. Como se puede observar en la Figura 67 el formulario ya viene precargado con los datos del usuario, de igual manera el botón de “Cambiar” no se activará hasta que el Administrador tenga iguales los campos “Contraseña” y “Repetir Contraseña”







Lista de Usuarios <small>Servicio de Cafetería</small>			
Información y Ajustes de los Usuarios			
Usuario	Nombre		Acciones
12	Carla Magaña López		 
15	Felipe de Jesús Mediola Rojas		 
16	Daniela Jimenez Arrebola		 

Figura 66. Lista de Usuarios Registrados.

**Editar Usuario.** Cambia los campos con la información correcta.

Editar Usuario.

Usuario:

Nombre:

Apellido Paterno:

Apellido Materno:

Password:

Repetir Password:

Figura 67. Formulario Editar Usuario.

Para borrar a un usuario no se tiene que verificar el tipo de privilegio que tiene el Administrador (ver Figura 51 y 52). Como se muestra en la Figura 66 aparecerá una lista con todos los usuarios registrados y se seleccionará al usuario que se quiere borrar.

### 6.3.4 Reservar, cancelar o quitar penalización

#### 6.3.4.1 Reservas

Si el usuario tuviera algún problema al realizar la reservación, el Administrador tiene la opción realizar la reservación al usuario.

El Administrador tendrá que llenar el formulario que se muestra en la Figura 68, una vez que el Administrador llena el formulario este verifica si todavía haya reservaciones disponibles en caso de no haber se le informa al Administrador tal y como se muestra en la Figura 69.



The image shows a web form titled "Reservación de Usuario". The form has a sub-header "Reservar". Below this, there is a label "Usuario:" followed by a text input field containing the placeholder text "Usuario". Underneath, there is a label "Tipo de Reservación" followed by two radio button options: "Desayuno." (which is selected) and "Comida.". At the bottom of the form is a blue button labeled "Reservar".

Figura 68. Formulario Reservar, Sistema Administrador.

Reservación de Usuario

Reservar

**NOTA: Ya no hay reservaciones por el día de hoy.**

Usuario:

Tipo de Reservación

Desayuno.

Comida.

Reservar

Figura 69. Mensaje al Administrador de reservaciones agotadas.

Una vez que se comprueba que todavía hay reservaciones, se tiene que verificar que el usuario existe (ver Figura 70), si el usuario existe se hace la reservación, pero si el usuario no existe se le hace saber al Administrador tal y como se muestra en la Figura 71.

El usuario existe bool(true)	El usuario no existe NULL
---------------------------------	------------------------------

Figura 70. Existe usuario vs No Existe usuario.

Reservación de Usuario

Reservar

**NOTA: No se encontró al usuario, verificar si está registrado.**

**Usuario:**

**Tipo de Reservación**

Desayuno.

Comida.

Reservar

Figura 71. Mensaje al Administrador, no se encontró al Usuario.

En la Figura 72 se muestra el mensaje de éxito que ve el Administrador cuando una reservación se registró correctamente.

Reservación de Usuario

Reservar

**NOTA: Reservación exitosa.**

**Usuario:**

**Tipo de Reservación**

Desayuno.

Comida.

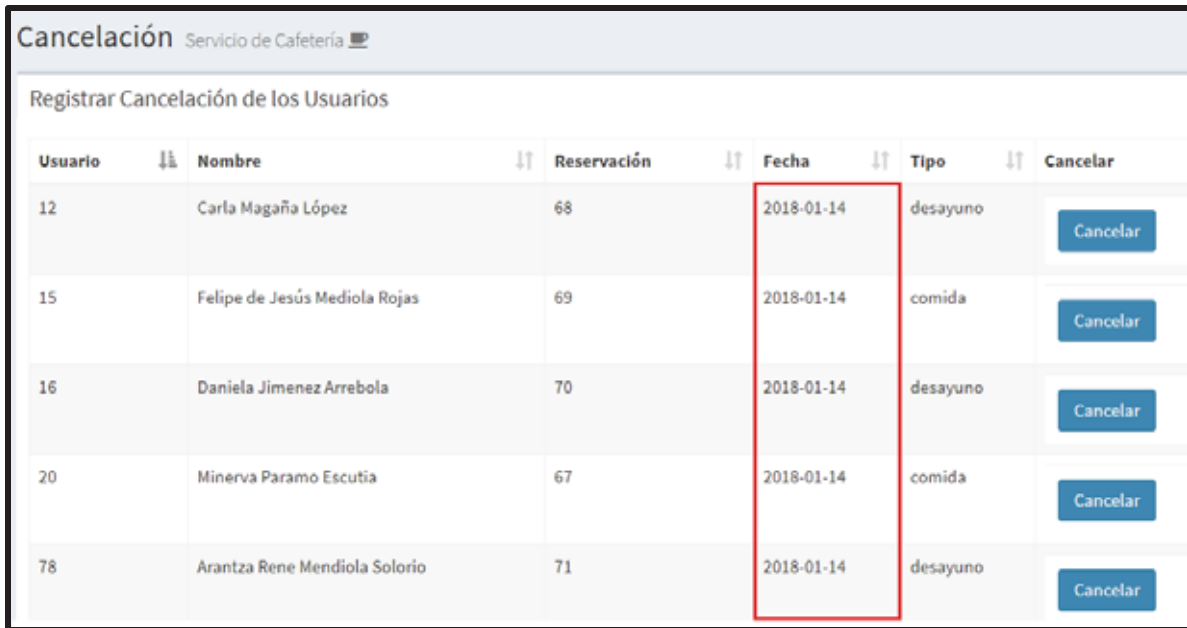
Reservar

Figura 72. Mensaje al Administrador, Reservación exitosa.



### 6.3.4.2 Cancelaciones

Si el usuario tuviera algún problema al realizar la cancelación de su reservación, el Administrador tiene la opción realizar la cancelación de las reservaciones hechas en un día tal y como se muestra en la Figura 73.



The screenshot shows a web interface titled 'Cancelación Servicio de Cafetería'. Below the title is a sub-header 'Registrar Cancelación de los Usuarios'. A table lists reservations with columns: 'Usuario', 'Nombre', 'Reservación', 'Fecha', 'Tipo', and 'Cancelar'. The 'Fecha' column contains the date '2018-01-14' for all entries, which is highlighted by a red rectangular box. Each row has a blue 'Cancelar' button.

Usuario	Nombre	Reservación	Fecha	Tipo	Cancelar
12	Carla Magaña López	68	2018-01-14	desayuno	Cancelar
15	Felipe de Jesús Mediola Rojas	69	2018-01-14	comida	Cancelar
16	Daniela Jimenez Arrebola	70	2018-01-14	desayuno	Cancelar
20	Minerva Paramo Escutia	67	2018-01-14	comida	Cancelar
78	Arantza Rene Mendiola Solorio	71	2018-01-14	desayuno	Cancelar

Figura 73. Reservaciones que puede cancelar el Administrador.

El Administrador solo seleccionará la reservación que quiere cancelar, en la Figura 74 se muestra el mensaje de éxito que ve el Administrador cuando una reservación es cancelada correctamente.



The screenshot shows the same 'Cancelación' interface as Figure 73. A red box highlights a message 'Cancelación exitosa.' centered above the table. The table below is identical to the one in Figure 73, showing a list of reservations with 'Cancelar' buttons.

Usuario	Nombre	Reservación	Fecha	Tipo	Cancelar
12	Carla Magaña López	68	2018-01-14	desayuno	Cancelar
15	Felipe de Jesús Mediola Rojas	69	2018-01-14	comida	Cancelar
16	Daniela Jimenez Arrebola	70	2018-01-14	desayuno	Cancelar
20	Minerva Paramo Escutia	67	2018-01-14	comida	Cancelar

Figura 74. Mensaje al Administrador, Cancelación exitosa.

### 6.3.4.3 Penalizaciones

Si el usuario tuviera algún problema con respecto a la penalización, el Administrador tiene la opción de quitar la penalización.

En la Figura 75 se muestra todos los usuarios que están o estuvieron penalizados.

Usuario	Nombre	Folio	Inicio de Penalización	Fin de Penalización	Tipo	Quitar
12 Penalizado	Carla Magaña López	63	2018-01-13	2018-01-16	desayuno	Quitar
15 No penalizado	Felipe de Jesús Mediola Rojas	54	2017-12-31	2018-01-14	desayuno	Quitar
16 Penalizado	Daniela Jimenez Arrebola	70	2018-01-14	2018-01-17	desayuno	Quitar
20 Penalizado	Minerva Paramo Escutia	65	2018-01-13	2018-01-16	desayuno	Quitar
2148914415 No penalizado	Karen Rivera Vidal	10	2017-11-21	2017-11-24	comida	Quitar
2148914415 Penalizado	Karen Rivera Vidal	9	2017-11-21	2018-01-24	desayuno	Quitar
2154095531 No penalizado	David Colin Diaz	7	2017-10-15	2017-10-18	desayuno	Quitar

Figura 75. Usuarios a los que se les puede quitar la Penalización.

Si el Administrador considera que es conveniente quitar la penalización al usuario, seleccionará a este y la penalización se le quitará (ver Figura 76).

Lista de Penalizaciones Servicio de Cafetería

Quitar Penalización a Usuarios

El Usuario ya no está Penalizado.

Usuario	Nombre	Folio	Inicio de Penalización	Fin de Penalización	Tipo	Quitar
15 No penalizado	Felipe de Jesús Mediola Rojas	54	2017-12-31	2018-01-14	desayuno	Quitar
16 Penalizado	Daniela Jiménez Arrebola	70	2018-01-14	2018-01-17	desayuno	Quitar
20 Penalizado	Minerva Paramo Escutia	65	2018-01-13	2018-01-16	desayuno	Quitar
2148914415 No penalizado	Karen Rivera Vidal	10	2017-11-21	2017-11-24	comida	Quitar
2148914415 Penalizado	Karen Rivera Vidal	9	2017-11-21	2018-01-24	desayuno	Quitar
2154095531 No penalizado	David Colin Díaz	7	2017-10-15	2017-10-18	desayuno	Quitar

Figura 76. Mensaje al Administrador, el usuario ya no está penalizado.

### 6.3.5 Validación

Para que un Administrador pueda validar un usuario no se necesita saber qué tipo de privilegio tiene el Administrador (ver Figura 51 y 52).

Para poder registrar la asistencia a la cafetería de un usuario el Administrador puede hacerlo de dos formas (ver Figura 77).

- Validar Asistencia por día.
- Validar Asistencia en General.
- 

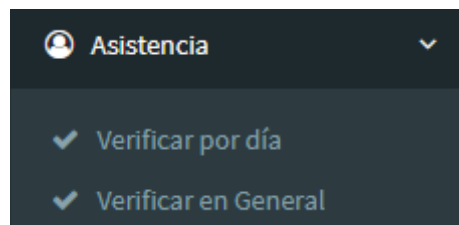
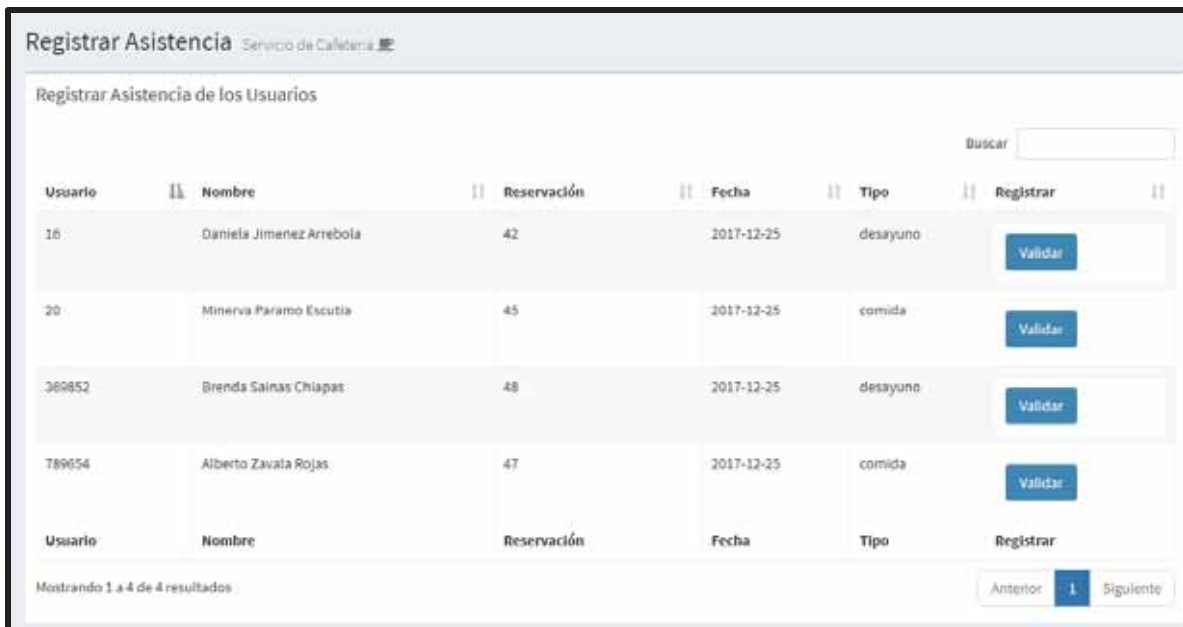


Figura 77. Menú de validar Asistencia.

### 6.3.5.1 Validar Asistencia por día

En este tipo de validación el Administrador visualizaría una tabla con todos los usuarios que hicieron una reservación ya sea para desayuno o comida, tal y como se muestra en la Figura 78, esta lista se hace con una consulta a la Base de Datos, con todos los registros requeridos.

En la ultima columna “Registrar” de la tabla (ver Figura 78) aparece un boton el cual el Administrador dara clic y la asistencia del usuario quedara registrada (ver Figura 79).



Registrar Asistencia Servicio de Cafetería

Registrar Asistencia de los Usuarios

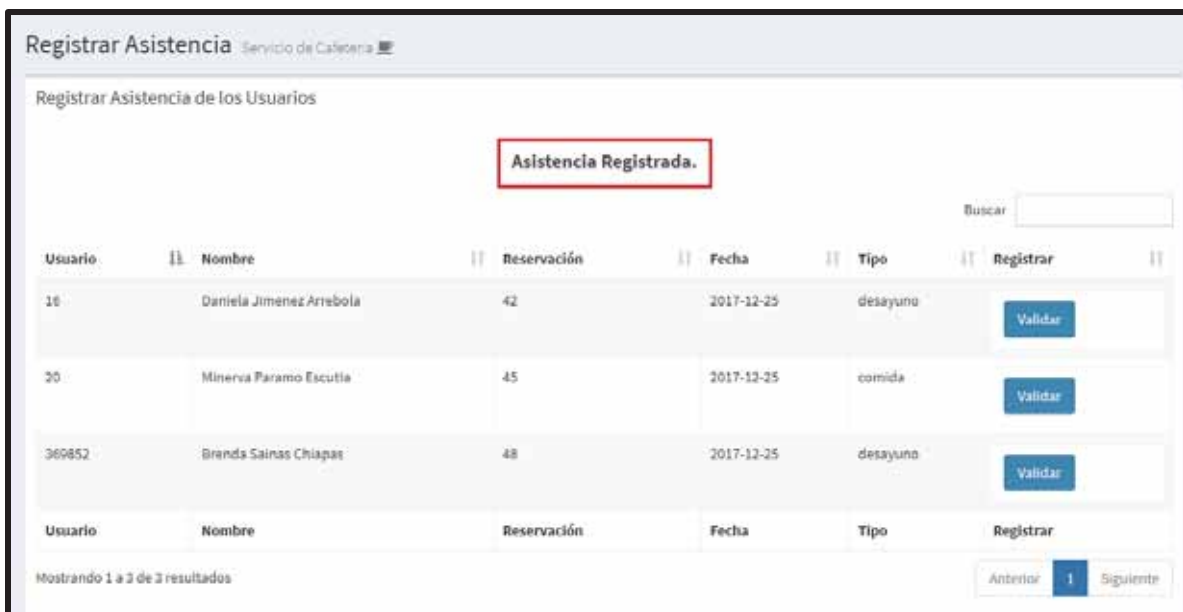
Buscar

Usuario	Nombre	Reservación	Fecha	Tipo	Registrar
16	Daniela Jimenez Arrebola	42	2017-12-25	desayuno	Validar
20	Minerva Paramo Escutia	45	2017-12-25	comida	Validar
360852	Brenda Sainas Chiapas	48	2017-12-25	desayuno	Validar
789654	Alberto Zavala Rojas	47	2017-12-25	comida	Validar

Mostrando 1 a 4 de 4 resultados

Anterior 1 Siguiente

Figura 78. Lista de usuarios que hicieron algún tipo de reservación.



Registrar Asistencia Servicio de Cafetería

Registrar Asistencia de los Usuarios

Asistencia Registrada.

Buscar

Usuario	Nombre	Reservación	Fecha	Tipo	Registrar
16	Daniela Jimenez Arrebola	42	2017-12-25	desayuno	Validar
20	Minerva Paramo Escutia	45	2017-12-25	comida	Validar
360852	Brenda Sainas Chiapas	48	2017-12-25	desayuno	Validar

Mostrando 1 a 3 de 3 resultados

Anterior 1 Siguiente

Figura 79. Mensaje de Asistencia registrada.

### 6.3.5.2 Validar Asistencia en General

En este tipo de validación el Administrador tendrá que llenar el formulario que se muestra en la Figura 80.

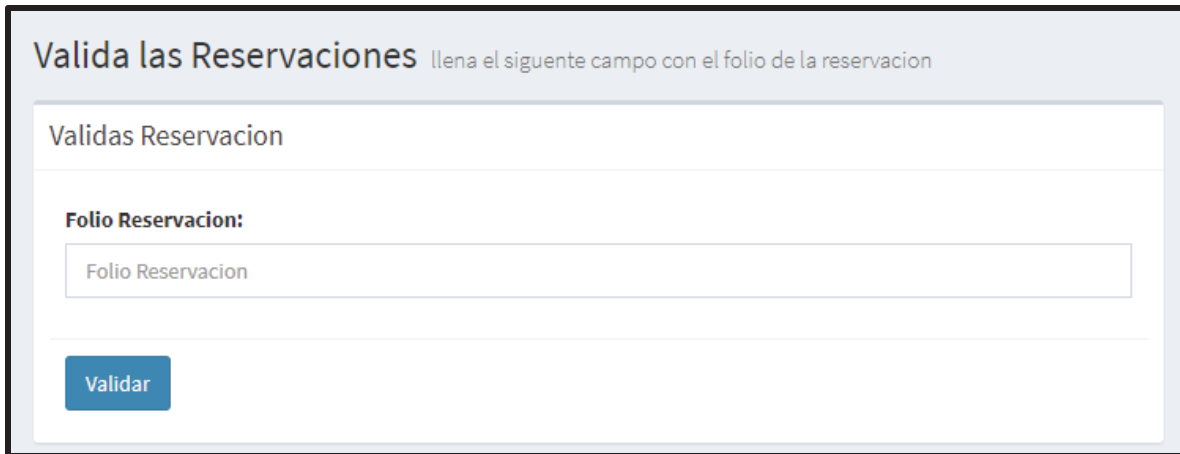


Figura 80. Formulario Validar Reservación.

En el tendrá que poner el folio de la reservación, una vez que el Administrador da clic en el botón “Validar”, se hacen varias validaciones.

La primera validación que se hace es verificar que lo que el Administrador puso en el campo “Folio Reservación” sea un número y esto se hace usando la función de PHP llamada *is\_numeric('cadena')*, esta verifica que la cadena que se le mando sea un número, si es así el valor que devuelve es “true”, pero si la cadena que recibió no es un número en valor que devuelve es “false” (ver Figura 81).

Es número	No es número
<code>bool(true)</code>	<code>bool(false)</code>

Figura 81. Resultados de la verificación de la función *is\_numeric()*.

La segunda validación que se hace es verificar que el folio proporcionado por el Administrador exista, para esto se hace una consulta a la Base de Datos la cual nos devolverá un número (ver Figura 82), si este es igual a cero quiere decir que la reservación no existe y se le hace saber al Administrador, en caso de que sea diferente a cero, existe la reservación y se registra la asistencia al servicio de cafetería.

La reservación existe  int(1)	La reservación no existe  int(0)
--	---

Figura 82. Resultado de la consulta a la Base de Datos, existe reservación vs no existe reservación.

En la Figura 83 se muestra el mensaje de éxito y los mensajes de error que arroja el sistema al Administrador.

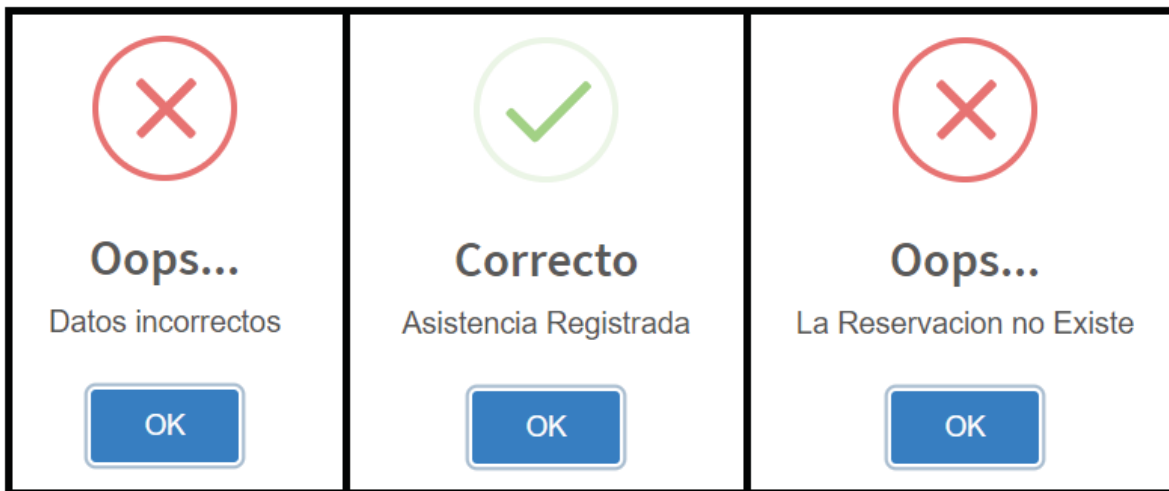


Figura 83. Mensajes que se pueden mostrar al registrar una asistencia al servicio de cafetería.

### 6.3.6 Datos Estadísticos

Para que un Administrador tenga acceso a los datos estadísticos no importara que privilegio tenga, todos los Administradores tendrán acceso a ellos, y se dividen en:

- Estadísticas y Reportes.
- Graficas
- Historiales.

#### 6.3.6.1 Estadísticas y Reportes

En esta parte el Administrador podrá ver cuantos usuarios hay registrados en el sistema, cuantas reservaciones, cancelaciones y penalizaciones se han hecho en total, por servicio (desayuno o comida) y por día.



Figura 84. Datos Estadísticos Generales y Generación de Reportes.

En la Figura 84 se muestran las estadísticas en general, también se puede observar que en cada uno de los *widgets* en la parte de abajo dice “Reporte”, cuando el Administrador da clic en “Reporte” se descargara un reporte en Excel (ver Figura 85).

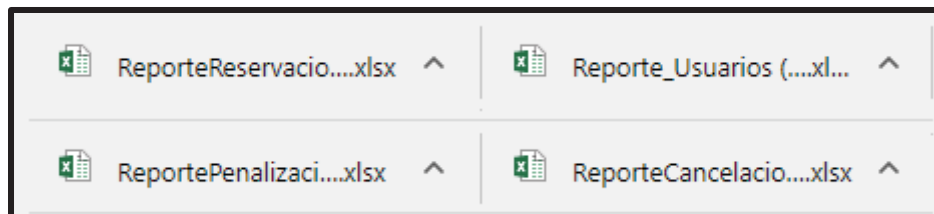


Figura 85. Descarga de los reportes.

En la Figura 86 se muestra el “Reporte\_Usuario.xlsx” el cual consiste de:

- Usuario.
- Nombre del usuario
- Apellido paterno del usuario.
- Apellido materno de usuario.

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					

Figura 86. Reporte\_Usuario.xlsx.

En la Figura 87 se muestra el “ReporteReservaciones.xlsx” el cual consiste de una gráfica y dos tablas estructuradas de la siguiente manera:

- Tabla Reporte
  - Usuario.
  - Nombre completo de usuario.
  - Fecha de reservación.
  - Folio de reservación.
  - Tipo de servicio.
- Tabla Grafica
  - Numero.
  - Fecha.
  - Numero de reservaciones echas en una fecha específica.



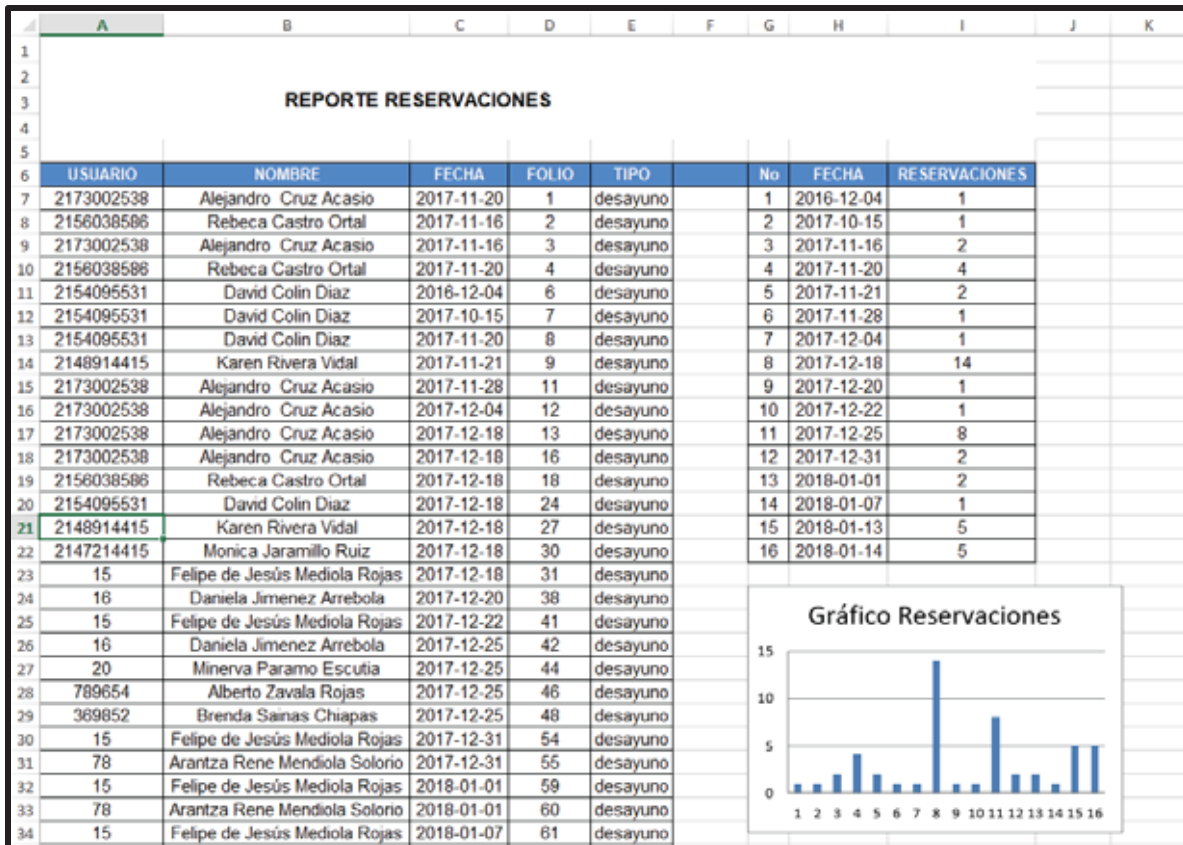


Figura 87. ReporteReservaciones.xlsx.

En la Figura 88 se muestra el “ReporteCancelacion.xlsx” el cual consiste de una gráfica y dos tablas estructuradas de la siguiente manera:

- Tabla Reporte
  - Usuario.
  - Nombre completo de usuario.
  - Fecha de la cancelación.
  - Folio cancelación.
  - Folio de reservación.
  - Tipo de servicio.
- Tabla Grafica
  - Numero.
  - Fecha.
  - Numero de cancelaciones echas en una fecha específica.



Figura 88. ReporteCancelacion.xlsx.

En la Figura 89 se muestra el “ReportePenalizacion.xlsx” el cual consiste de una gráfica y dos tablas estructuradas de la siguiente manera:

- Tabla Reporte
  - Usuario.
  - Nombre completo de usuario.
  - Fecha de inicio de la penalización.
  - Fecha de fin de la penalización.
  - Folio de penalización.
  - Folio de reservación.
  - Tipo de servicio.
- Tabla Grafica
  - Numero.
  - Fecha.
  - Numero de penalizaciones echas en una fecha específica.



Figura 89. ReportePenalizacion.xlsx.

En la Figura 90 se muestran los datos de reservaciones, cancelaciones y penalizaciones por servicio (desayuno o comida).



Figura 90. Datos Estadísticos por Servicio.

En la Figura 91 se muestran los datos de reservaciones, cancelaciones y penalizaciones por día y por servicio (desayuno o comida).



Figura 91. Datos Estadísticos por Día.

### 6.3.6.2 Gráficas

En estas se podrá ver el número de usuarios que han solicitado algún servicio con respecto a la fecha que se solicitó el servicio y se dividen en tres:

- Reservación
- Cancelación
- Penalización

#### 6.3.6.2.1 Reservación

En la gráfica de la Figura 92 se podrá observar en número de usuarios que han reservado en total con respecto a la fecha que solicitaron la reservación.

En la gráfica de la Figura 93 se podrá observar en número de usuarios que han hecho reservaciones de tipo desayuno con respecto a la fecha que solicitaron la reservación.

En la gráfica de la Figura 94 se podrá observar en número de usuarios que han hecho reservaciones de tipo comida con respecto a la fecha que solicitaron la reservación.



Figura 92. Gráfica de Reservaciones en General por día.

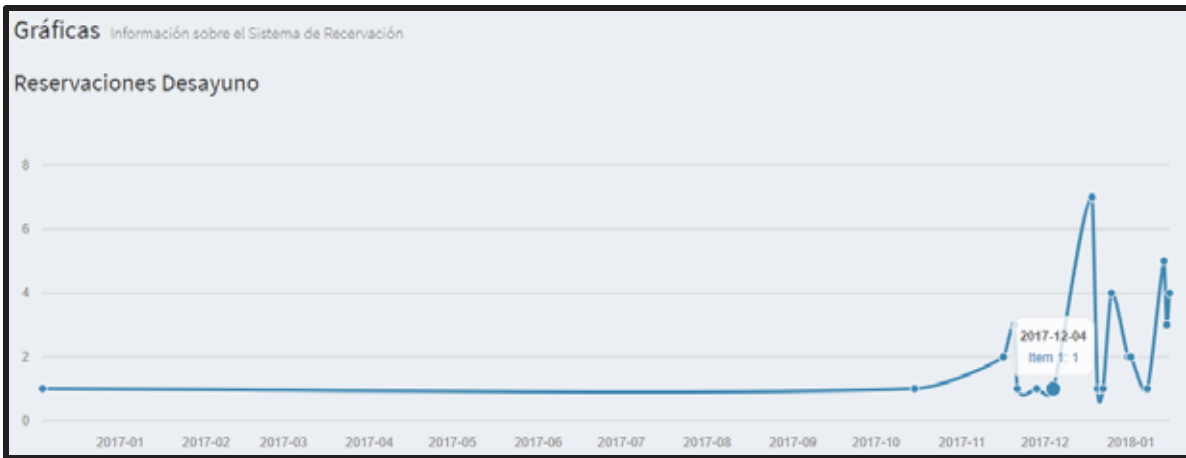


Figura 93. Gráfica de Reservaciones de Desayuno por día.

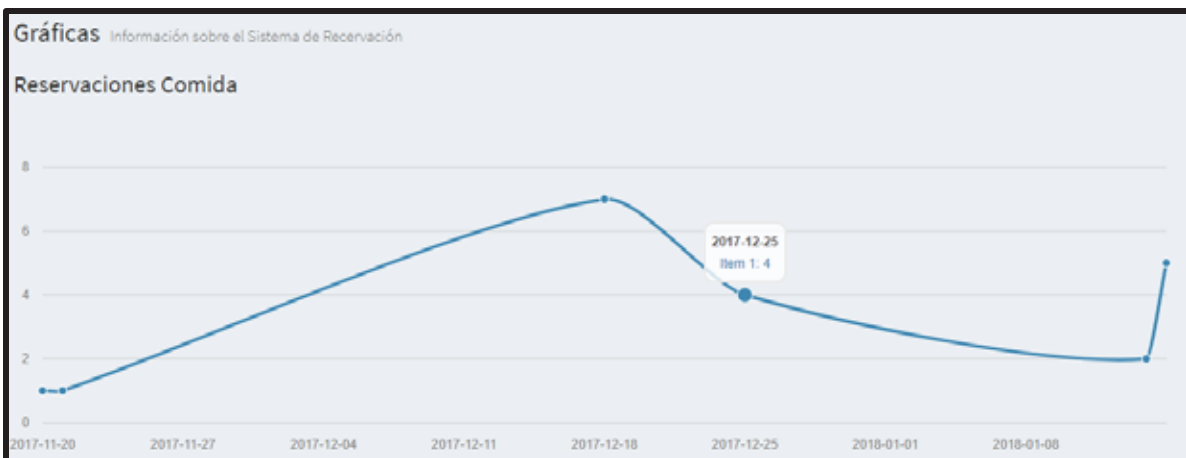


Figura 94. Gráfica de Reservaciones de Comida por día.

### 6.3.6.2.2 Cancelaciones

En la gráfica de la Figura 95 se podrá observar en número de usuarios que han cancelado reservaciones en total con respecto a la fecha que hicieron la cancelación.

En la gráfica de la Figura 96 se podrá observar en número de usuarios que han cancelado reservaciones de tipo desayuno con respecto a la fecha que hicieron la cancelación.

En la gráfica de la Figura 97 se podrá observar en número de usuarios que han cancelado reservaciones de tipo comida con respecto a la fecha que hicieron la cancelación.

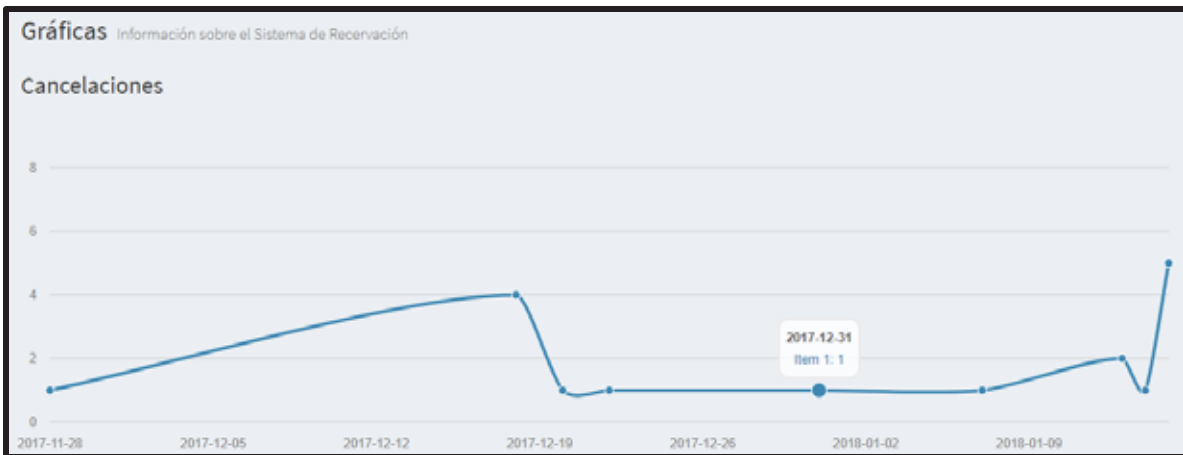


Figura 95. Gráfica de Cancelaciones en General por día.

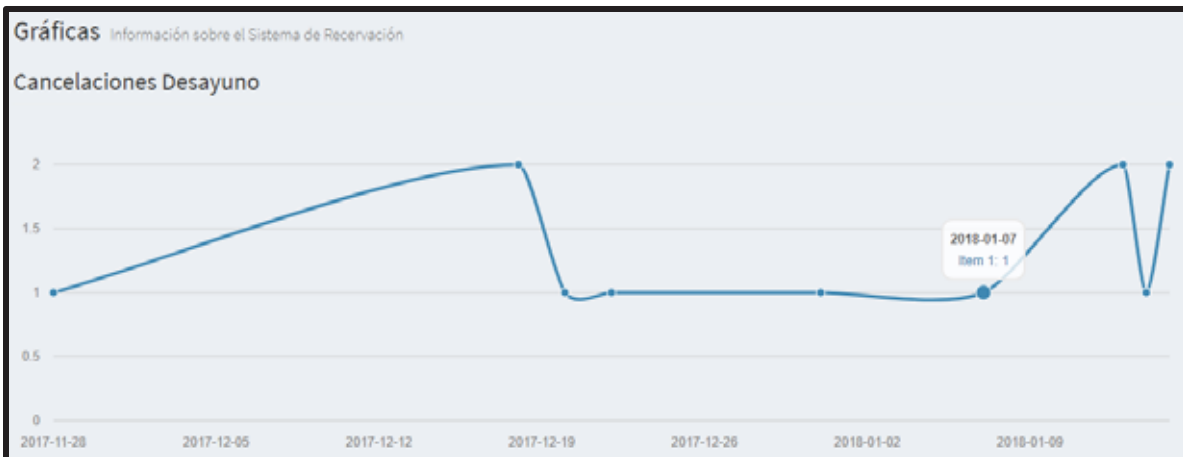


Figura 96. Gráfica de Cancelaciones de Desayuno por día.

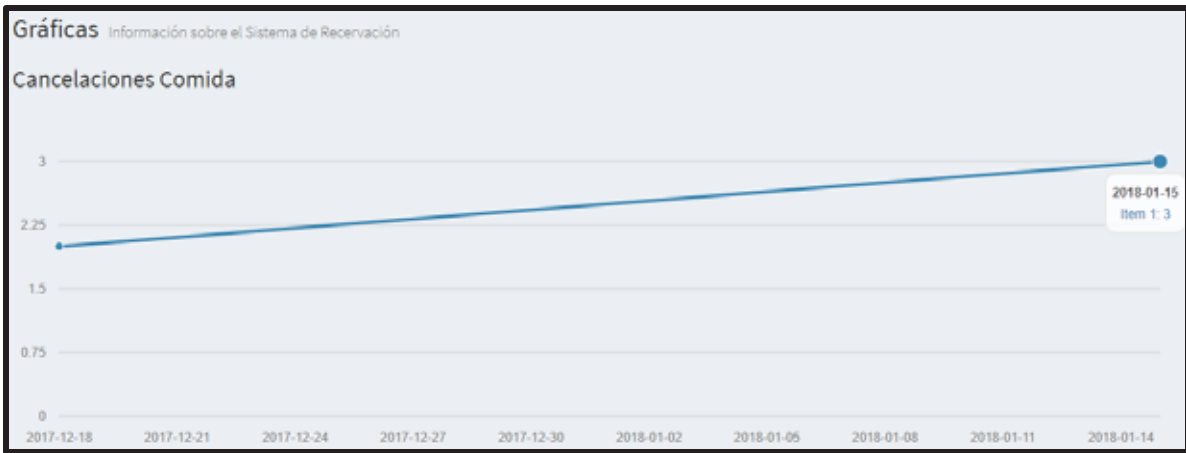


Figura 97. Gráfica de Cancelaciones de Comida por día.

### 6.3.6.2.3 Penalización

En la gráfica de la Figura 98 se podrá observar en número de usuarios que están o estuvieron penalizados con respecto a la fecha en que termina la penalización.

En la gráfica de la Figura 99 se podrá observar en número de usuarios que están o estuvieron penalizados de tipo desayuno con respecto a la fecha en que termina la penalización.

En la gráfica de la Figura 100 se podrá observar en número de usuarios que están o estuvieron penalizados de tipo comida con respecto a la fecha en que termina la penalización.

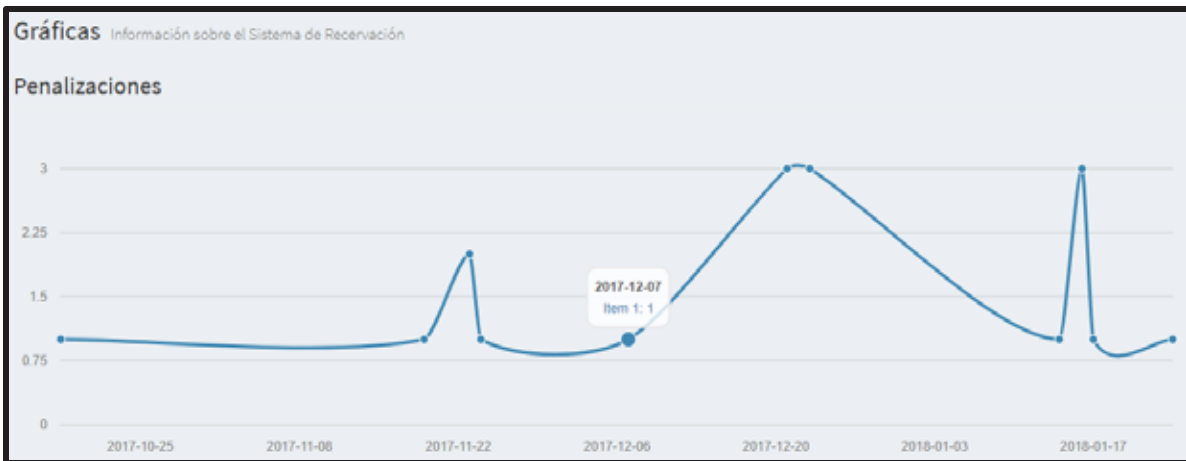


Figura 98. Gráfica de Penalizaciones en General por día.

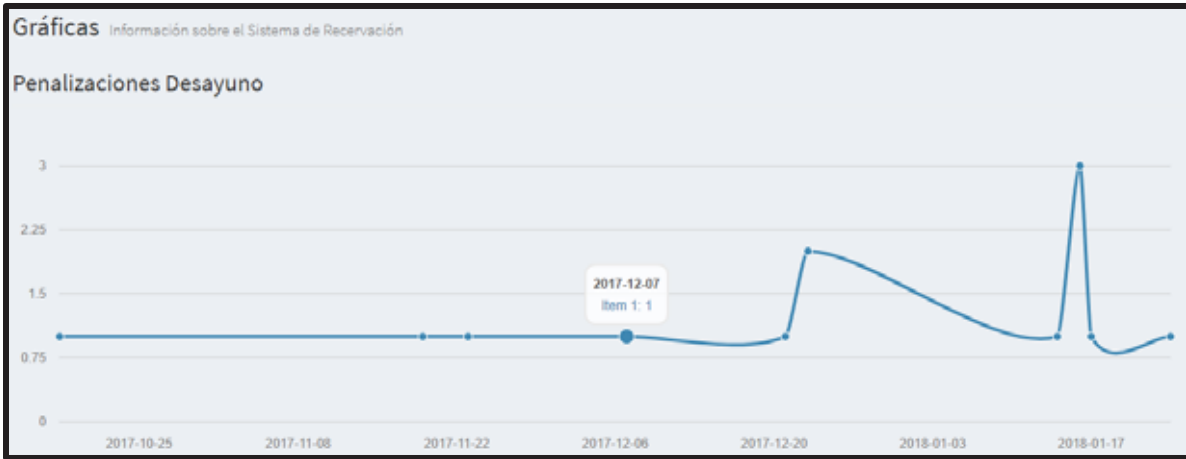


Figura 99. Gráfica de Penalizaciones de Desayuno por día.

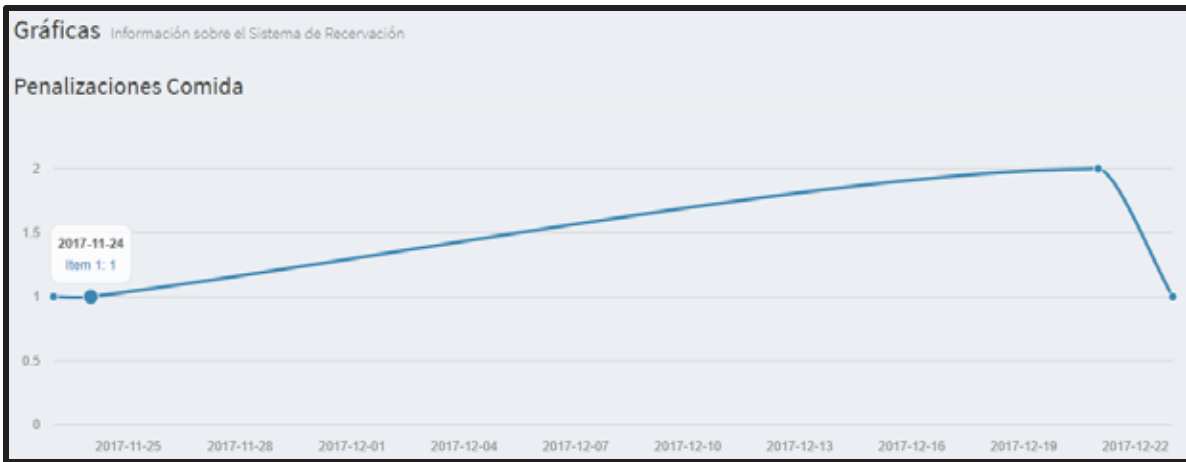


Figura 100. Grafica de Penalizaciones de Comida por día.

### 6.3.6.3 Historiales

En los historiales el Administrador podrá ver toda la información de los usuarios que han hecho reservaciones, cancelaciones y los usuarios que están o han estado penalizados, podrá hacer búsquedas por fecha, nombre, tipo de servicio, etc. También podrá ordenar los historiales de forma alfabética (ascendente o descendente), de mayor a menor o viceversa, según sea el caso.

Los historiales de reservaciones se dividen en tres

- General
- Desayuno
- Comida

En la Figura 101 se muestra un historial de reservaciones en general.



**Historial de Reservas** Servicio de Cafetería

Información sobre las Reservas

Buscar

Usuario	Nombre	Folio	Fecha de Reservacion	Tipo
2173002538	Alejandro Cruz Acasio	1	2017-11-20	desayuno
2156038586	Rebeca Castro Ortal	2	2017-11-16	desayuno
2173002538	Alejandro Cruz Acasio	3	2017-11-16	desayuno
2156038586	Rebeca Castro Ortal	4	2017-11-20	desayuno
2173002538	Alejandro Cruz Acasio	5	2017-11-20	comida
2154095531	David Colin Diaz	6	2016-12-04	desayuno
2154095531	David Colin Diaz	7	2017-10-15	desayuno
2154095531	David Colin Diaz	8	2017-11-20	desayuno
2148914415	Karen Rivera Vidal	9	2017-11-21	desayuno
2148914415	Karen Rivera Vidal	10	2017-11-21	comida

Mostrando 1 a 10 de 60 resultados

Anterior **1** 2 3 4 5 6 Siguiente

**Figura 101. Historial de Reservas.**

Los historiales de cancelaciones de dividen el tres

- General
- Desayuno
- Comida

En la Figura 102 se muestra un historial de cancelaciones en general.

**Historial de Cancelaciones** Servicio de Cafetería

Información sobre las Cancelaciones

Buscar

Usuario	Nombre	Folio	Fecha de Cancelación	Tipo
12	Carla Magaña López	80	2018-01-15	desayuno
15	Felipe de Jesús Mediola Rojas	34	2017-12-18	comida
15	Felipe de Jesús Mediola Rojas	41	2017-12-22	desayuno
15	Felipe de Jesús Mediola Rojas	61	2018-01-07	desayuno
15	Felipe de Jesús Mediola Rojas	62	2018-01-13	desayuno
15	Felipe de Jesús Mediola Rojas	73	2018-01-15	comida
16	Daniela Jimenez Arrebola	38	2017-12-20	desayuno
16	Daniela Jimenez Arrebola	74	2018-01-15	desayuno
20	Minerva Paramo Escutia	75	2018-01-15	comida
78	Arantza Rene Mendiola Solorio	55	2017-12-31	desayuno

Mostrando 1 a 10 de 17 resultados

Anterior **1** 2 Siguiente

**Figura 102. Historial de Cancelaciones.**

Los historiales de penalizaciones se dividen en tres

- General
- Desayuno
- Comida

Estos historiales le indican al Administrador si un usuario está o no está penalizado, en la Figura 103 se muestra un historial de penalizaciones en general.

**Historial de Penalizaciones** Servicio de Cafetería

Información sobre las Penalizaciones

Buscar

Usuario	Nombre	Folio	Fecha de Penalización	Tipo
15 <b>No penalizado</b>	Felipe de Jesús Mediola Rojas	31	2017-12-21	desayuno
2148914415 <b>No penalizado</b>	Karen Rivera Vidal	9	2017-11-24	desayuno
2148914415 <b>No penalizado</b>	Karen Rivera Vidal	10	2017-11-24	comida
2148914415 <b>Penalizado</b>	Karen Rivera Vidal	27	2017-12-23	desayuno
2154095531 <b>No penalizado</b>	David Colin Diaz	7	2017-10-18	desayuno
2154095531 <b>No penalizado</b>	David Colin Diaz	8	2017-11-23	desayuno
2154095531 <b>Penalizado</b>	David Colin Diaz	24	2017-12-23	desayuno
2154095531 <b>Penalizado</b>	David Colin Diaz	23	2017-12-23	comida
2150038586 <b>No penalizado</b>	Rebeca Castro Ortal	2	2017-11-19	desayuno
2150038586 <b>No penalizado</b>	Rebeca Castro Ortal	17	2017-12-21	comida
Usuario	Nombre	Folio	Fecha de Penalización	Tipo

Mostrando 1 a 10 de 14 resultados

Anterior **1** 2 Siguiete

Figura 103. Historial de Penalizaciones.

Adicionalmente el Sistema de Administrador, permite que el Administrador pueda hacer ajustes a sus datos.

**Editar Administrador.** Cambia los campos con la información correcta.

Editar Administrador.

**Usuario:**

**Nombre:**

**Apellido Paterno:**

**Apellido Materno:**

**Password:**

**Repetir Password:**

**Cambiar**

Figura 104. Formulario Editar Administrador.

Este funciona similar al Crear Administrador, con dos pequeñas diferencias

- La primera es que en vez de insertar un nuevo registro en la Base de Datos se modifica un registro ya existente
- La segunda es que no puede modificar el atributo privilegio. Eso solo lo puede hacer un Administrador con privilegio “*administrador*”.

Tal y como se puede observar en la Figura 104, el formulario ya viene precargado con los datos del Administrador, estos pueden ser modificados en caso de tener algún error, de igual manera el botón de “*Cambiar*” no se activará hasta que el usuario tenga iguales los campos “*Contraseña*” y “*Repetir Contraseña*”.

### **6.3.7 Hardware y Software**

Para el desarrollo del Sistema de Reservación en Línea para el Servicio de Cafetería se utilizó una computadora DELL con las siguientes características.

#### **6.3.7.1 Hardware**

- Procesador CORE i5, 2.4 GHz.
- Memoria RAM de 8 GB.
- Disco Duro de 1TB.
- Sistema Operativo Windows 10 Home.

#### **6.3.7.2 Software**

- PHP 7
- MySQL 6.3
- Servidor Apache

## **7 RESULTADOS**

A continuación, se detallaron los resultados obtenidos del desarrollo del proyecto del Sistema de Reservación en Línea para el Servicio de Cafetería con generación de reportes.

- En el Sistema Usuario.
  - Se logró el registro de los usuarios.
  - Se logró el login de los usuarios.
  - Se logró la reservación a los dos principales tipos de servicios que ofrece el servicio de cafetería los cuales son desayuno y comida, el cual culmina con la generación de un documento PDF.
  - Se logró limitar las reservaciones.
  - Se logró la cancelación de las reservaciones, la cual culmina con la generación de un documento PDF.
  - Se logró la penalización a usuarios automáticamente.
  - Adicionalmente se logró que el usuario pueda editar los datos que proporciona en el registro y

- Se logró crear un historial por usuario de reservaciones, cancelaciones y penalizaciones.
- En el Sistema Administrador.
  - Se logró el login del administrador.
  - Se logró identificar a los Administradores con privilegio de tipo “*administrador*”, esto se hizo con el propósito de que solo ellos pudieran crear más Administradores.
  - Se logró registrar la asistencia al servicio de cafetería solicitado por el usuario.
  - Se lograron datos estadísticos de los servicios ofrecidos por el servicio de cafetería (desayuno y comida), estos fueron en general y por día.
  - Se logró la generación de reportes de usuarios, reservaciones, cancelaciones y penalizaciones utilizando la información de los datos estadísticos, estos reportes se generan en archivos .xlsx
  - Se lograron hacer graficas de las reservaciones, cancelaciones hechas por los usuarios y de los usuarios penalizados por día.
  - Se lograron historiales de reservaciones, cancelaciones y penalizaciones de todos los usuarios y por servicio (desayuno y comida).
  - Adicionalmente se logró que el Administrador pueda editar sus datos.
- Base de Datos.
  - Se logró desarrollar un diseño de la Base de Datos, el cual se utilizó para elaborar la estructura final de la Base de Datos.

## 8 ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

Las pruebas realizadas al Sistema de Reservación en Línea para el Servicio de Cafería se hicieron tanto en la parte de Usuario como en la parte del Administrador. Las pruebas que se realizaron fueron las siguientes.

### 8.1 Registro y Login

- Una vez que el Usuario o Administrador es registrado en el Sistema, se revisó que todos los datos se registraran correctamente en la Base de Datos, prestando mayor atención en la contraseña ya que esta tenía que insertarse encriptada y en símbolos propios del español como son acentos y “eñes”.
- Se garantizó el uso de sesiones, con el fin de que nadie pueda ver información exclusiva para Usuarios o Administradores del sistema.

### 8.2 Reservación

- Se hicieron pruebas con el horario para garantizar que no se permita el acceso a las reservaciones antes de tiempo.
- Con el propósito de hacer pruebas se limitó a 30 reservaciones por servicio (desayuno o comida), mismas que fueron insertadas desde la Base de Datos para comprobar que realmente se cumple la validación de cupo, ya que no se permite hacer más de 50 reservaciones al día por servicio (desayuno o comida).
- Se intentó que un usuario hiciera dos reservaciones al día para un mismo servicio (desayuno o comida), el resultado obtenido fue la negación de la

reservación, con esto se corroboro que los usuarios solo pueden hacer una reservación por servicio (desayuno o comida) al día.

- En el documento PDF se revisó que todos los datos del Usuario fueran correctos, que se respetaran acentos, “eñes” y caracteres propios del español, también se revisó que el número de la reservación del documento PDF fuera el mismo que el de la Base de Datos.

### **8.3 Penalización**

- Se penalizaron a diferentes usuarios desde la Base de Datos, con el propósito de hacer pruebas con usuarios penalizados, dando como resultado la negación de la reservación.
- Se comprobó que las penalizaciones se registraran automáticamente al momento de revisar el estado de la confirmación de la reservación, si esta estaba en cero se procedía a registrar la penalización.
- Se comprobó que los usuarios penalizados pudiera realizar reservaciones una vez transcurrido el tiempo de la penalización, dando como resultado la descarga de la reservación.

### **8.4 Cancelación**

- Se verificó que solo se pudiera cancelar una reservación, siempre y cuando fuera del mismo día en que se hizo la reservación y estuviera dentro del horario permitido.
- Se intentó hacer la cancelación de una reservación que aún no existía, dando como resultado la negación de la cancelación.
- Se intentaron cancelar varias reservaciones de diferentes tipos de servicio (desayuno o comida) fuera del horario permitido, dando como resultado la negación de la cancelación.
- En el documento PDF se revisó que todos los datos del Usuario fueran correctos, que se respetaran acentos, “eñes” y caracteres propios del español, también se revisó que el número de la cancelación del documento PDF fuera el mismo que el de la Base de Datos.

### **8.5 Administrador**

- Se verificó que solo aquellos Administradores con privilegio “*administrador*”, pudieran actualizar, crear o eliminar Administradores.
- Se verificó que cualquier tipo de Administrador pudiera actualizar, crear o eliminar a un Usuario.

### **8.6 Verificar Asistencia**

- Se hicieron pruebas con folios incorrectos (mesclando números, letras y símbolos) a la hora de validar la asistencia de una reservación, dando como resultado la negación de la asistencia al servicio de cafetería.
- Se intentó validar reservaciones que no estaban registradas en el sistema, dando como resultado la negación de la asistencia al servicio de cafetería.

## 8.7 Datos Estadísticos y Reportes, Gráficas e Historiales

- En estos, se estuvo observando su comportamiento, ya que cada vez que se actualizaba la Base de Datos se tiene que actualizar estos datos y mostrara la información correcta de los Datos Estadísticos y Reportes, Graficas e Historiales, dando como resultado la visualización de la información correcta y actualizada en los Datos Estadísticos y Reportes, Graficas e Historiales.

## 9 CONCLUSIONES

El Sistema de Reservación en Línea para el Servicio de Cafetería consta de dos módulos principales, *Usuario* el cual se encarga de registrar a los usuarios, hacer las reservaciones, cancelaciones y penalizaciones, y *Administrador* este es el encargado de verificar la asistencia al Servicio de Cafetería, así como de proporcionar los datos estadísticos y reportes, graficas e historiales.

El cumplimiento de los objetivos presentados en la propuesta de Integración se describe a continuación:

- Crear una Base de Datos para almacenar la información del Sistema. Este objetivo se cumplió completamente, ya que se pudo diseñar y elaborar una Base de Datos la cual es capaz de guardar toda la información del Sistema.
- Desarrollar e implementar un módulo de registro tanto para usuarios como administrador. Este objetivo se cumplió completamente, ya que se logró una interfaz la cual permite el registro de usuarios y administradores, con una pequeña diferencia, que los usuarios se registran y los administradores los registran.
- Desarrollar e implementar un módulo de acceso tanto para usuarios como administrador. Este objetivo se cumplió completamente, ya que se logró una interfaz la cual permite el acceso a todos los usuarios y administradores registrados en el Sistema.
- Desarrollar e implementar un módulo de reservación. Este objetivo se cumplió completamente, ya que todo usuario registrado en el sistema puede generar reservaciones de los dos tipos que se ofrecen (desayuna y comida).
- Desarrollar e implementar un módulo que cancela reservaciones. Este objetivo se cumplió completamente, ya que todo usuario registrado en el sistema puede generar cancelaciones de los dos tipos que se ofrecen (desayuna y comida), siempre y cuando esté en fecha y hora.
- Desarrollar e implementar un módulo de generación de reportes. Este objetivo se cumplió completamente, ya que se logró obtener reportes de usuarios, reservaciones, cancelaciones y penalizaciones haciendo uso de los datos estadísticos.
- Desarrollar e implementar un módulo de validación. Este objetivo se cumplió completamente, ya que el administrador es capaz de registrar la asistencia al servicio solicitado por el usuario.

## 10 REFERENCIAS

- [1] [2] Catalogo Institucional, [En línea]. Available: [https://www.azc.uam.mx/informacion\\_general/CatInstitucional](https://www.azc.uam.mx/informacion_general/CatInstitucional). Último acceso: 23 de diciembre 2017].
- [3] P. R. Ramírez and E. O. García., “Sistema de gestión para la oficina parroquial de San Juan Bautista Coyoacán”, Proyecto Terminal, Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, D.F., México, 2013.
- [4] M. A. Serrano Lices, “Prototipo para agendar citas médicas”, Proyecto Terminal, Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, D.F., México, 2015.
- [5] A. Hernández Hernández, "Sistema gestor de nutrición", Proyecto Terminal, Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, D.F., México, 2014.
- [6] Lodgify, [En línea]. Available: <http://www.lodgify.com/>. [Último acceso: 23 de diciembre 2017].
- [7] Infocaja, [En línea]. Available: <http://infocaja.mx/>. [Último acceso: 23 de diciembre 2017].
- [8] Ingeniero Rafael Alvarado Ramírez, Jefe de Proyecto de Sección de Desarrollo de Sistemas, Universidad Autónoma Metropolitana, Unidad Azcapotzalco, Coordinación de Servicios de Computo, Edificio T. planta baja.
- [9] Aner, [En línea]. Available: <http://www.aner.com>. [Último acceso: 23 de diciembre 2017].
- [10] Beati, Hernán, PHP Creación de páginas Web dinámicas. 1ª edición, Buenos Aires, Alfaomega Grupo Editorial Argentino.
- [11] Pavón, Jacobo, Creación de un portal con PHP y MySQL. 4ª edición, Alfaomega Grupo Editorial, S.A. de C.V., México.
- [12] History of MySQL, [En línea]. Available: <https://dev.mysql.com/doc/refman/5.5/en/history.html>. [Último acceso: 23 de diciembre 2017].
- [13] QueH. Darwen, “Introducción a los sistemas de base de datos”, Massachusetts, Prentice Hall, 2001.
- [14] Historia HTML, [En línea]. Available: [http://librosweb.es/libro/xhtml/capitulo\\_1/breve\\_historia\\_de\\_html.html](http://librosweb.es/libro/xhtml/capitulo_1/breve_historia_de_html.html). [Último acceso: 23 de diciembre 2017].
- [15] What is HTML?, [En línea]. Available: [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp). [Último acceso: 23 de diciembre 2017].
- [16] HTML 5, [En línea]. Available: <https://developer.mozilla.org/es/docs/HTML/HTML5>. [Último acceso: 23 de diciembre 2017].



- [17] Orós, Juan Carlos, Guía práctica de XHTML, JavaScript y CSS 1ª edición, Alfaomega Grupo Editorial, S.A. de C.V., México.
- [18] CSS, [En línea]. Available: <https://www.w3schools.com/css/>. [Último acceso: 23 de diciembre 2017].
- [19] Semáforos PHP [En línea]. Available: <http://php.net/manual/es/function.sem-get.php> [Último acceso: 23 de diciembre 2017].
- [20] AdminLTE, [En línea]. Available: <https://adminlte.io/>. [Último acceso: 23 de diciembre 2017].
- [21] Barcode, [En línea]. Available: <https://github.com/picqer/php-barcode-generator>. [Último acceso: 23 de diciembre 2017].
- [22] FPDF, [En línea]. Available: <http://www.fpdf.org/>. [Último acceso: 23 de diciembre 2017].
- [23] PHPExcel, [En línea]. Available: <https://phpexcel.codeplex.com/>. [Último acceso: 23 de diciembre 2017].
- [24] SweetAlert2, [En línea]. Available: <https://limonte.github.io/sweetalert2/>. [Último acceso: 23 de diciembre 2017].