DESARROLLO DE LA APLICACIÓN AGAUPRO (AGILIZACIÓN Y AUTOMATIZACIÓN DE PROCESOS) PARA MEJORAR LA ATENCIÓN AL CLIENTE EN LOS RESTAURANTES

**INTRODUCCION**

Hoy en día los restaurantes se enfrentan a diversas situaciones competitivas en el mundo del comercio, por el cual cada vez estos se ven obligados a mejorar en cada una de las necesidades que satisfagan tanto para el establecimiento comercial como las del cliente, siendo la última como un elemento primordial, ya que los clientes determina la demanda de los productos y servicios que se ofrecen en estos establecimientos.

Sin embargo no hay que olvidar que la calidad e inocuidad de los productos prevalece a la productividad del restaurante, al igual que un espacio agradable y de una buena ejecución de los diferentes métodos de trabajo y atención dentro del establecimiento. Por lo tanto cabe resaltar que algunos de los procesos que se llevan a cabo, bien sea desde la atención al cliente hasta la liquidación del pago no son tan eficientes para efectuar mejoras, siendo necesario el uso e implementación de la tecnología.

La automatización de los procesos mediante la implementación de dispositivos es una óptima respuesta para el mejoramiento de las necesidades de los clientes. En cuanto la finalidad de estos dispositivos, consta de poder tomar el pedido sin necesidad de un mesero, al igual que liquidar pagos por medio de tarjeta de crédito sin necesidad de ir al cajero. Por lo dicho anteriormente, habría una reducción de tiempo y de costos, es decir que los minutos que se gastan los clientes en esperar un mesero para tomar su pedido, se reducen a solo segundos, así mismo necesitando menos personal, ahorrando gastos.

# **JUSTIFICACIÓN**

El presente proyecto está basado en la implementación de un software para cada uno de los procesos que se llevan a cabo en los restaurantes de Florencia-Caquetá, con el fin de llegar a una posible solución, mediante el problema o necesidad identificada.

Esto se llevó a cabo por que el sistema del establecimiento comercial se estaba viendo afectado por la alta cantidad de tiempo que se ocupaba para realizar el proceso de toma de pedidos, por lo que fue necesario implementar una posible mejora a través de una página web donde los clientes podrán reservar sus mesas. Así siendo más eficiente y eficaz para el desarrollo de las actividades, al igual para que el cliente que vaya hacer uso del servicio quede más satisfecho.

**OBJETIVOS**

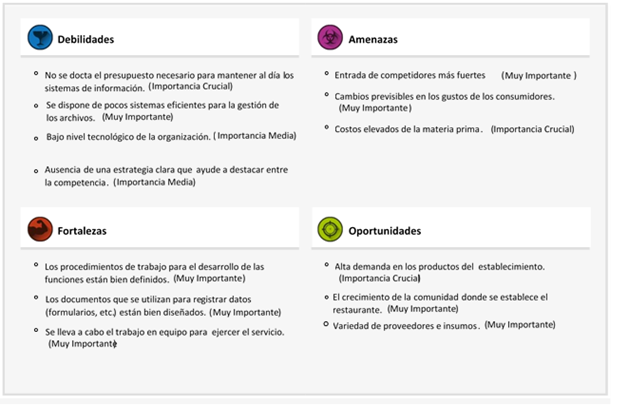
General:

Desarrollar una página web, en donde el cliente pueda hacer reservas y visualizar los platos del dia.

Específicos:

1. Analizar el sistema de información actual de los restaurantes de Florencia - Caquetá y cada uno de sus diferentes aspectos.
2. Definir los requisitos del sistema, a través de técnicas de elicitación de requisitos.
3. Realizar los modelos y diseños correspondientes del sistema, teniendo en cuenta los requisitos.
4. ANALISIS DE SITUACION

A través del análisis DOFA, obtenemos los diferentes aspectos que se están viviendo en los restaurantes de Florencia-Caquetá.



2. DEFINICIÓN DE REQUISITOS

**Requisitos funcionales:**

1.La página web permitirá la visualización de los platos en un menú a los usuarios.

2.La página web permitirá el registro de un menú con los platos del día, donde se permita ingresar los platos con sus precios.

3.La página web permitirá la modificación de un plato, bien sea en su nombre, su descripción o precio.

4.La página web permitirá el registro del cliente.

5.El sistema permitirá la visualización de las mesas disponibles en determinadas fechas

6.La página web guardará la información de la mesa reservada.

**Requisitos no funcionales:**

1. La página web sólo será de uso exclusivo para los funcionarios y clientes del restaurante.

2. La página web indicará los errores de procedimientos inválidos.

3. Su interfaz gráfica será de manera ilustrativa, con colores representativos a los del restaurante.

4. Su interfaz gráfica será enfocada en una página web, que vendrá predeterminada en las tablets.

5.Las imágenes deben estar en formato TIFF para mayor calidad.

6. El sistema debe ser desarrollado en HTML con el CSS y el JavaScript.

7. Cualquier fallo de la página web, se ha de guardar todos los procedimientos recientes y generar un tiempo de espera.

|  |  |  |  |
| --- | --- | --- | --- |
| **CÓDIGO** | **TIPO REQUISITO** | **NOMBRE DEL REQUISITO** | **DESCRIPCIÓN DEL REQUISITO** |
| RF-001 | Funcional | Visualización de platos | La página web permitirá la visualización de los platos en un menú a los usuarios. |
| RF-002 | Funcional | Gestionar el menú diario. | La página web permitirá el registro de un menú con los platos del día, donde se permita ingresar los platos con sus precios. |
| RF-003 | Funcional | Modificar los platos del menú. | La página web permitirá la modificación de un plato, bien sea en su nombre, su descripción o precio. |
| RF-004 | Funcional | Registrar cliente. | La página web permitirá el registro del cliente. |
| RF-005 | Funcional | Visualización de mesas | El sistema permitirá la visualización de las mesas disponibles en determinadas fechas |
| RF-006 | Funcional | Registrar mesa | La página web guardará el la información de la mesa reservada |

|  |  |  |  |
| --- | --- | --- | --- |
| **CÓDIGO** | **TIPO REQUISITO** | **NOMBRE DEL REQUISITO** | **DESCRIPCIÓN DEL REQUISITO** |
| RNF-001 | **No Funcional** | Exclusividad | La página web sólo será de uso exclusivo para los funcionarios y clientes del restaurante. |
| RNF -002 | **No Funcional** | Solución de problemas | La página web indicará los errores de procedimientos inválidos. |
| RNF -003 | **No Funcional** | Implementación gráfica | Su interfaz gráfica será de manera ilustrativa, con colores representativos a los del restaurante. |
| RNF -004 | **No Funcional** | Implementación gráfica | Su interfaz gráfica será enfocada en una página web, que vendrá predeterminada en las tablets. |
| RNF -005 | **No Funcional** | Implementación de contenidos multimedia. | Las imágenes deben estar en formato TIFF para mayor calidad. |
| RNF -006 | **No Funcional** | Programación en lenguaje determinado. | La página web debe ser desarrollado en HTML con el CSS y el JavaScript. |
| RNF -007 | **No Funcional** | Alternativa de errores | Cualquier fallo de la página web, se ha de guardar todos los procedimientos recientes y generar un tiempo de espera. |

**Priorización de requisitos**

Por usuario

Cliente:

● El sistema permitirá el registro del cliente.

● El sistema permitirá la visualización del menú o mostrar un menú claro a los usuarios.

● El sistema permitirá que el cliente valide su pedido.

● El sistema permitirá la selección de forma de pago.

Gerente o Administrador:

● El sistema registrara un menú con los platos del día, donde se permita ingresar los platos con sus precios.

● El sistema permitirá la modificación de un plato, bien sea en su nombre, su descripción o precio.

# **FORMULACIÓN DEL PROBLEMA O NECESIDAD**

Desde años atrás los restaurantes han venido presentando algunos inconvenientes a la hora de prestar sus servicios, tales como la atención al cliente el cual ha influido a que sus ventas no tengan un buen rendimiento. De esta manera, al momento de la llegada de un cliente, se debe disponer de personal que lo atienda, generando más gastos para el restaurante e incrementando más tiempo (Minutos) en cada proceso (Exhibición del menú y toma de pedido), haciendo que el cliente no se sienta satisfecho con su atención.

La falta de automatización en los procesos de un restaurante ha hecho que estos busquen una mejor estrategia de atención a través de las tecnologías, tratando de implementar medios para que esta sea una ventaja óptima en la reducción de gastos y tiempo, es decir que los procesos se realicen en cuestión de segundos.

**POSIBLE SOLUCIÓN**

Desarrollar una página web, en donde el cliente pueda realizar sus respectivas reservas y visualizar los platos activos del día, sin necesidad de estar en el restaurante. Prácticamente se trata de reservar las mesas que se desee para una fecha específica, teniendo en cuenta que el cliente se puede registrar para llevar a cabo este proceso.

**METODOLOGÍA PARA EL DESARROLLO DE SOFTWARE**

Modelo basado en prototipo: También conocido como desarrollo con prototipación o modelo de desarrollo evolutivo, se inicia con la definición de los objetivos globales para el software, luego se identifican los requisitos conocidos y las áreas del esquema en donde es necesaria más definición. Este modelo se utiliza para dar al usuario una vista preliminar de parte del software. Este modelo es básicamente prueba y error ya que si al usuario no le gusta una parte del prototipo significa que la prueba fallo por lo cual se debe corregir el error que se tenga hasta que el usuario quede satisfecho.

-Análisis y especificación

-Diseño y construcción

-Evaluación

-Modificación

-Diseño técnico

-Programación y prueba

-Operación y mantenimiento

Modelo de Kanban: En contra de lo que se pudiera pensar, el sistema Kanban ideado por Ohno es relativamente sencillo. Solo su nombre, en japonés, ya explica mucho. “Kan” significa visible o visual, y “ban”, tarjeta o tablón. Si tomamos como referencia el conjunto de una fábrica, o también cualquier otro tipo de organización, como por ejemplo un equipo de desarrollo, el sistema Kanban se organiza con un gran tablón dividido en columnas, normalmente siete:

Objetivos: se marcan a largo plazo, con la idea de que todos los miembros del equipo los tengan en mente. Es una columna opcional, no siempre está presente.

Pendiente: esta columna engloba las tareas pendientes que se pueden afrontar de forma inmediata. En el lugar más alto de esa columna colocaremos la tarea pendiente que tiene la máxima prioridad, y en cuanto empecemos, la pasaremos a las siguientes columnas.

Preparación: también es opcional. Aquí incluimos aquellas tareas que necesitan cierta discusión interna antes de ser afrontadas. Cuando lo tengamos claro, pasamos a la siguiente columna.

Desarrollo: en este espacio situamos la tarea hasta que la terminemos. Si algo falla, regresa a la columna anterior.

Prueba: comprobamos que todo funciona bien. En función de ese examen, la tarea avanza en el tablón o retrocede.

Aplicación: la existencia de esta columna depende de las características de cada tarea. Hablamos, por ejemplo, de tareas como colocar una nueva versión de una aplicación en un servidor.

Hecho: cuando ya no tenemos que preocuparnos más de algo porque hemos terminado la tarea.

Utilidad: Al proyecto se le implementara cada una de las fases de la metodología prototipo, siendo esta la principal. Como secundaria se implantará la metodología Kanban el cual será usado en las actividades de cada una de las fases de la metodología prototipo.

**DIAGRAMAS DE CASOS DE USO**

Actores del sistema:

Cliente

- Ver los platos disponibles

- Crear cuenta

- Iniciar sesión

- Ver mesas disponibles

- Seleccionar mesa

- Cerrar sesión

Administrador

-Iniciar sesión administrador.

-Ver reservas de mesas

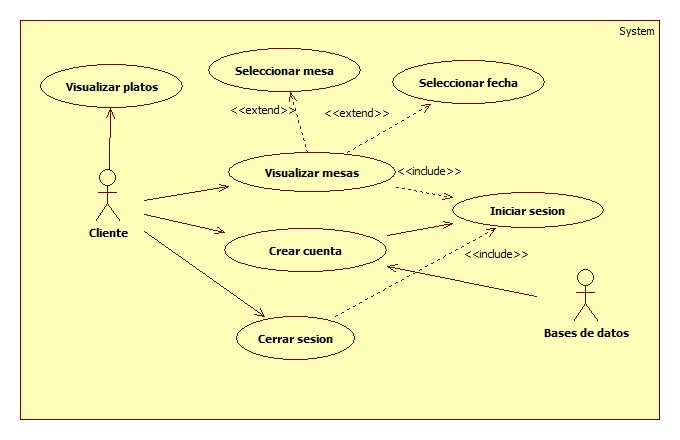
-Registrar platos

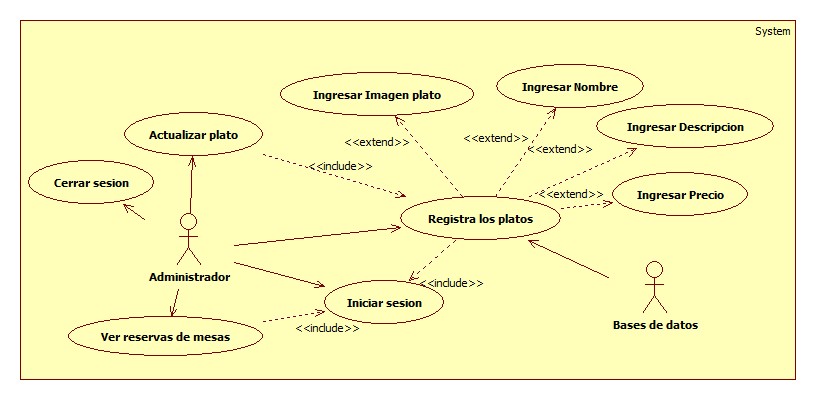
-Ingresar imagen para plato

-Ingresar nombre para plato

-Ingresar descripción para plato

-Ingresar precio para plato





|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 001-UC | | |
| Nombre de UC: | Ver los platos disponibles | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Cliente, base de datos, administrador | | |
| Descripción: | Dada la selección de la sección de menú, se desplegará la vista de cada plato disponible. | | |
| Disparador: |  | | |
| Pre-condiciones: | Debe existir al menos un plato guardado y disponible en la base de datos. | | |
| Post-condiciones: |  | | |
| flujo Normal: | 1. El usuario selecciona la sección menú. 2. El sistema muestra los platos disponibles. | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | Siempre. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 002-UC | | |
| Nombre de UC: | Crear cuenta. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Base de datos. | | |
| Descripción: | La persona que quiera realizar un reserva en el sistema debe crear su cuenta. | | |
| Disparador: |  | | |
| Pre-condiciones: |  | | |
| Post-condiciones: |  | | |
| flujo Normal: | 1. Crear cuenta. | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | Siempre. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 003-UC | | |
| Nombre de UC: | Iniciar sesión. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Cliente, base de datos. | | |
| Descripción: | El usuario cliente inicia sesión con credenciales. | | |
| Disparador: |  | | |
| Pre-condiciones: | Debe existir la cuenta del cliente. | | |
| Post-condiciones: |  | | |
| flujo Normal: | 1. Seleccionar en el menú del sistema iniciar sesión. | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | Siempre. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 004-UC | | |
| Nombre de UC: | Ver los mesas disponibles | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Cliente, base de datos, administrador | | |
| Descripción: | El cliente puede realizar la visualización de la disponibilidad de las mesas con sus respectivos horarios. | | |
| Disparador: |  | | |
| Pre-condiciones: | El cliente debe iniciar sesión en el sistema. | | |
| Post-condiciones: |  | | |
| flujo Normal: | 1. Inicia sesión. 2. Visualiza las mesas del restaurante. | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | A veces. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 005-UC | | |
| Nombre de UC: | Seleccionar mesas | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Cliente, base de datos. | | |
| Descripción: | El cliente después de ver la disponibilidad del equipo puede hacer reserva de la mesa que tenga disponibilidad. | | |
| Disparador: |  | | |
| Pre-condiciones: |  | | |
| Post-condiciones: | El cliente debe Iniciar sesión, la mesa debe estar disponible. | | |
| flujo Normal: |  | | |
| Flujo Alternativo: | 1. selecciona la mesa. 2. reserva la mesa. | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | a veces. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 006-UC | | |
| Nombre de UC: | Cerrar sesión. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Cliente. | | |
| Descripción: | El cliente puede cerrar la sesión. | | |
| Disparador: |  | | |
| Pre-condiciones: | El cliente debe haber iniciado sesión. | | |
| Post-condiciones: |  | | |
| flujo Normal: |  | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | Siempre. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 007-UC | | |
| Nombre de UC: | Iniciar sesión administrador | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Base de datos, administrador | | |
| Descripción: | El usuario administrador inicia sesión al sistema. | | |
| Disparador: |  | | |
| Pre-condiciones: |  | | |
| Post-condiciones: |  | | |
| flujo Normal: |  | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | A veces. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 008-UC | | |
| Nombre de UC: | Ver reservas de la mesa. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Base de datos, administrador. | | |
| Descripción: | El sistema le deje ver al administrador detalles sobre las reservas. | | |
| Disparador: |  | | |
| Pre-condiciones: |  | | |
| Post-condiciones: |  | | |
| flujo Normal: |  | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | a veces. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 009-UC | | |
| Nombre de UC: | Registrar plato. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Base de datos, administrador | | |
| Descripción: | El administrador puede registrar un plato, en esto se incluye la imagen del plato, el nombre, su descripción y el precio del mismo. | | |
| Disparador: |  | | |
| Pre-condiciones: | Debe iniciar sesión el administrador. | | |
| Post-condiciones: |  | | |
| flujo Normal: | 1. Iniciar sesión el administrador. 2. Registrar plato. | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | A veces. | | |
| Notas: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID-UC | 009-UC | | |
| Nombre de UC: | Visualizar platos. | | |
| Creado por: | Nelson Cortes | Actualizado por: |  |
| Fecha de la Creación | 02/09/2019 | Fecha de la Ultima actualización |  |
| Actores: | Base de datos, administrador | | |
| Descripción: | El administrador puede visualizar los platos de la vista del cliente. | | |
| Disparador: |  | | |
| Pre-condiciones: |  | | |
| Post-condiciones: |  | | |
| flujo Normal: |  | | |
| Flujo Alternativo: |  | | |
| Excepciones: |  | | |
| Prioridad: | Deseada. | | |
| Frecuencia de Uso: | a veces. | | |
| Notas: |  | | |