Universidad Autónoma de Yucatán

Facultad de matemáticas

Profesor: **Edgar Cambranes Martínez**

Carlos Eduardo Avila Criollo

Jose de la Rosa Baeza Pérez

Sebastian Echeverria Lopez

Rigel Bustamante Lara

Proyecto: LA TRANKERA

Programación estructurada

**Definición del proyecto**

Nuestro proyecto software es un programa destinado a restaurantes que realiza comandas generadas por un mesero hacia los cocineros a partir de un menú en una base de datos ya preestablecida.

En una primera versión el programa funcionara en computador y el objetivo es que en una futura versión pueda funcionar en teléfonos móviles.

**Definición de requisitos del sistema**

**Actores del sistema**

Lo primero que hicimos fue definir a los actores que van a interactuar con el sistema y definimos 3

* **cliente:** Es la persona que proporcionara la información de que va a consumir a partir de un menú ya prestablecido en una base de datos.
* **Mesero:** Es la persona que generara la comanda con los comentarios necesarios a partir de la información recibida por el cliente y enviarla a la cocina.
* **Cocinero:** Es la persona que se encargara de recibir la comanda y mandar un aviso de que esta lista la comanda

**Requisitos funcionales de usuario**

* **RS-1:** El cocinero podrá bloquear los elementos del menú que ya no estén disponibles
* **RS-2:** El mesero podrá agregar el numero de la mesa de la comanda enviada
* **RS-3:** El mesero podrá mandar la comanda al cocinero
* **RS-4:** El cocinero podrá enviar una notificación cuando la comanda este lista
* **RS-5:** El cocinero podrá eliminar las comandas que ya se realizaron de su base de datos
* **RS-6:** El mesero podrá hacer comentarios sobre los productos como agregar o quitar un ingrediente

**Requisitos funcionales del sistema**

* **RS-7:** El programa tendrá 2 usuarios directos el mesero y el cocinero y un usuario indirecto que será el cliente
* **RS-8:** El programa se comunicará atreves de una red local
* **RS-9:** El menú solo contendrá productos hechos en la cocina

**Requisitos no funcionales del sistema**

* **RSNF-1:** La comanda no demorara más de 10 segundos en llegar a la cocina
* **RSNF-2:** El programa no calculara el total de las ventas
* **RSNF-3:** El programa funcionara correctamente en el 90% de los casos
* **RSNF-4:** El programa no registrara que productos se vendieron
* **RSNF-5:** La notificación de la cocina no tardara más de 10 segundos en llegar

**Casos de uso**

Logramos generar 3 casos de uso que serian los casos de uso más frecuentes que podrían ocurrir:

* **CU-1:** El mesero podrá hacer comentarios sobre la comanda para realizar cambios en los productos como agregar o quitar ingredientes si es necesario pero una vez que esta sea enviada ya no se podrán hacer modificaciones a través del programa.
* **CU-2:** El cocinero podrá eliminar las comandas que ya se hayan realizado y se haya enviado la notificación de que están listas de su base de datos para evitar acumulamiento y confusión con nuevas comanda que sean recibidas.
* **CU-3:** En caso de que un cliente quiera agregar algo a la comanda el mesero deberá generar una nueva y volver a enviarla a la cocina con los nuevos pedidos**.**

**Estándar de codificación**

Para realizar el proyecto software elegimos el lenguaje de programación Python, ya que después de investigar encontramos librerías que nos podrían servir para trabajar sobre trasmisión de datos en redes locales y desarrollar para móviles.

El programa estará modularizado en bloques para entrada, proceso y salida de datos, se evitará usar variables globales en todos los casos que sea posible y también estará formado con funciones en los casos que sea necesario para reducir el tamaño y complejidad del software, así como también de usaran comentarios para describir la funcionalidad de la función.

Los tamaños máximos de las líneas de código serán de 79 caracteres para hacer el código más legible

Para el nombrado de variables decidimos agregar un guion antes del nombre de las variables para evitar conflictos con palabras reservadas

Ejemplo: \_variable

**Proceso de desarrollo**

Se investigo acerca de una problemática en la que podríamos basar nuestro proyecto y nos topamos con un problema en un restaurant en el que para enviar comandas hasta la cocina era complicado.

Nos reunimos cada miércoles para investigar del proyecto y sobre que lenguaje de programación podríamos usar que tuviera las bibliotecas que necesitábamos para poder realizar el proyecto.

Luego de obtener la aprobación del proyecto nos reunimos con el dueño para platicar los requisitos que iba a poder realizar el programa y los que no serían posible agregar.

Para la realización del proyecto nos reuniremos cada fin de semana para avanzar en el desarrollo e irnos familiarizando con el lenguaje para aplicar las bibliotecas que necesitamos.

Para la realización del proyecto acordamos usar un ciclo de vida de cascada ya que es el que mas se adapta y es mas sencillo de usar en un proyecto pequeño como el que se va a realizar.

Esta métrica se basa en las asistencias de cada uno de los integrantes del equipo a las reuniones de trabajo para avanzar en el proyecto y las horas que haga, cuando se comience el desarrollo del software se contabilizara las líneas de código que genere en el tiempo en el que este trabajando.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombres** | **Contribución** | **Asistencia/hrs** | **Porcentaje** |
| Carlos Eduardo Avila Criollo | Requerimientos, proceso de desarrollo | 5/5, 6hrs | 25% |
| José de la Rosa Baeza Pérez | Idea principal, acceso con el cliente, casos de uso | 5/5, 6hrs | 25% |
| Sebastián Echeverria López | Requerimientos, casos de uso | 5/5, 6hrs | 25% |
| Rigen Bustamante Lara | Requerimientos, proceso de desarrollo, casos de uso | 5/5, 6hrs | 25% |