SISTEMA DE CONTROL DE PEDIDOS PARA APP DE DELIVERY

LIAM TAYLOR

ANDRES SILVA

MANUEL BUITRAGO

SANTIAGO BARRANTES

MARIA PAULA VELANDIA SEDANO

MARTHA DIAZ

DOCENTE

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA

MISIÓN TIC 2022

2021

|  |
| --- |
| **Sprint Planning** |

**Fecha:** 10/09/21

**Horario:** 6:00 pm a 6:30 pm

**Recursos:** Se utilizó el enlace de meet <https://meet.google.com/ufk-taua-ydw> para una reunión virtual. Se deja el enlace de la reunión como evidencia.

**Participantes:**



|  |  |
| --- | --- |
| **Rol** | **Nombre** |
| Scrum Master | Andres Silva |
| Product Owner | Maria Paula Velandia Sedano |
| Ingeniero De Soporte | Liam Taylor |
| Ingeniero De Software | Manuel Buitrago |
| Auxiliar | Santiago Barrantes |

**Descripción del proyecto:**

la empresa XYZ quieren realizar un software que apoye a los restaurantes de la ciudad con sus procesos de venta de productos gastronómicos, este con la finalidad de promocionar a los restaurantes mostrando sus menús, en la cual cada restaurante tiene muchos productos y a su vez un producto puede estar en varios restaurantes, además hay usuarios que pueden comprar varios productos y un producto puede ser comprado por varios usuarios. El encargado del restaurante necesita tener una aplicación informática para gestionar la información de cada producto de su restaurante. ¿

De cada producto se requiere la siguiente información; Identificación del producto, nombre, categoría (que puede ser: desayuno, comida rápida, típica, postres, hamburguesa, pollo, saludable, parrilla, oriental, mexicana, italiana, sándwich y bebidas), precio este puede variar según el restaurante y descripción del producto.

De cada restaurante se requiere la siguiente información: Identificación de la empresa, nombre, teléfono, correo electrónico, dirección, tipo de restaurante (que puede ser: desayunos, comida rápidas, típicas, postres, hamburguesas, pollo, saludable, parrillas, oriental, mexicana, italiana, sándwich y bebidas) además puede pertenecer a varios.

De cada usuario se requiere la siguiente información: identificación, tipo de identificación, Nombre, teléfono, email y dirección.

Se necesita gestionar los productos donde los restaurantes pueden: agregar productos desde un formulario, modificar la información del producto, ver la información del producto y eliminar el producto.

Se necesita que los restaurantes se gestionan de tal manera que puedan: registrarse en la plataforma con un formulario, puedan modificar sus datos si es necesario, pueda consultar su información y por último pueda eliminar su cuenta.

Los usuarios pueden, registrarse y eliminar su cuenta, además de poder comprar productos y cancelar el pedido.

Para entrar al sistema por parte del restaurante se necesita una validación con un usuario y contraseña de tal forma que les permita las funcionalidades de agregar, modificar, consultar y eliminar sus productos. Y por parte del usuario se requiere logear con email y contraseña para comprar los productos.

|  |
| --- |
| **Sprint 1** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **G6\_PROYECTO** | | | | | | | |
|  | CODIGO | [QUIEN, SUJETO] | [EVENTO, OBJETO] | [DEBE, DEBERA, NO DEBE, NO DEBERA] | [RESULTADO, CONSECUENCIA, PARA] | REQUERIMIENTO | CRITERIOS DE ACEPTACION |
|  |
| **Sprint 1** | H1 | Santiago Barrantes | Desarrollará las relaciones entre los componentes del sistema. | debe | tener representada la lógica interna del sistema. | Santiago Barrantes Desarrollara las relaciones entre los componentes del sistema. debe tener representada la lógica interna del sistema. | \*Documentación de los componentes \*Se debe especificar las relaciones entre componentes \*Un componente puede tener más de una relación. |  |
|  |
|  |
| H2 | Liam Tylor y Manuel Buitrago | modelaran la estructura de la base de datos del sistema. | deben | tener organizada la información, haciendo optima la asignación y la petición de datos. | Liam Tylor y Manuel Buitrago modelaran la estructura de la base de datos del sistema. deben tener organizada la información, haciendo optima la asignación y la petición de datos. | \*modelo Entidad relación o \* modelo Relacional  y documentación \*Cada tabla debe tener su respectivo identificador, el cual debe ser único. |  |
|  |
|  |
| H3 | Andres Silva | realizar la sincronización del modelo de la base de datos con el sistema. | debe | complementar el correcto funcionamiento del sistema. | Andres Silva realizar la sincronización del modelo de la base de datos con el sistema. debe complementar el correcto funcionamiento del sistema. | \*Documentación de la BD \*La base de datos tiene que estar montada en algún gestor de bases de datos. |  |
|  |
|  |
| H4 | María Velandia | diseñara la estructura de navegación entre interfaces del sistema. | debe | tener representada la lógica navegacional del sistema. | María Velandia diseñara la estructura de navegación entre interfaces del sistema. debe tener representada la lógica navegacional del sistema. | \*Se debe especificar qué acciones se pueden ejecutar en cada interfaz \*Se mostrarán atributos de las interfaces (cual será el home, y cual necesita autentificación) \*Se debe presentar de manera jerárquica. |  |
|  |
|  |

|  |
| --- |
| **Daily Meeting** |

**Fecha:** 15/09/21

**Horario:** 5:00 pm a 5:15 pm

**Recursos:** Se utilizo el enlace <https://meet.google.com/mkq-hbcd-ona> para una reunión virtual.

**Participantes:**

|  |  |
| --- | --- |
| **Rol** | **Nombre** |
| Scrum Master | Andres Silva |
| Product Owner | Maria Paula Velandia Sedano |
| Ingeniero De Soporte | Liam Taylor |
| Ingeniero De Software | Manuel Buitrago |
| Auxiliar | Santiago Barrantes |

**Descripción:**

Se asigno cada historia de usuario de lo que iba hacer cada integrante del esquipo en el primer Sprint, de tal forma que quedaron así:

H1 – Diagramar los casos de usos – se le asigno a Santiago Barrantes

H2 – Modelo de datos (entidad relación) – se le asigno a Manuel Buitrago

H2 – Modelo de datos (Modelo relacional) – se le asigno a Liam Tylor

H3 – Implementación de la base de datos – se le asigno a Andres Silva

H4 – Diagrama navegacional – se le asigno a María Velandia

Además se resolvieron dudas de Git y Git Hub.

**Fecha:** 16/09/21

**Horario:** 1:00 pm a 1:15 pm

**Recursos:** Se utilizó el chat de nuestro grupo de whatssap

**Participantes:**

|  |  |
| --- | --- |
| **Rol** | **Nombre** |
| Scrum Master | Andres Silva |
| Product Owner | Maria Paula Velandia Sedano |
| Ingeniero De Soporte | Liam Taylor |
| Ingeniero De Software | Manuel Buitrago |
| Auxiliar | Santiago Barrantes |

**Descripción del proyecto:**

Se llego un Acuerdo para cambiar el proyecto y agregar un modulo mas que es el de usuario el cual tiene la posibilidad de comprar los productos que el restaurante haya puesto en el mercado.

Además, se visualizó avances de su parte de trabajo de María Velandia y Andres Silva.

Los demás se comprometieron a entregar el 18/09/21 la parte que le scorrespondio

**Fecha:** 19/09/21

**Horario:** 6:00 pm a 6:15 pm

**Recursos:** Se utilizó el enlace de meet <https://meet.google.com/jyc-vwkp-oxe> para una reunión virtual. Se deja el enlace de la reunión como evidencia.

**Participantes:**

Imagen que contiene Icono

Descripción generada automáticamente

|  |  |
| --- | --- |
| **Rol** | **Nombre** |
| Scrum Master | Andres Silva |
| Product Owner | Maria Paula Velandia Sedano |
| Ingeniero De Soporte | Liam Taylor |
| Ingeniero De Software | Manuel Buitrago |
| Auxiliar | Santiago Barrantes |

**Descripción del proyecto:**

Se revisaron los avances de cada participante del proyecto en la cual Santiago Barrantes, María Velandia, Liam Tylor y Andres Silva Ya habían terminado su parte aunque Liam Tylor presentaba problemas en subirlo al Git Hub Y Manuel Buitrago se comprometió a terminar su parte.

|  |
| --- |
| **Product Increment** |

**H1: Diagrama de Uso**

Se realizo el diagrama de casos de uso con sus respectivas entidades y los procesos que pueden realizar

**Imagen que contiene Icono

Descripción generada automáticamenteDiagrama

Descripción generada automáticamente**

**Anexo H2: Modelos Entidad Relacional**

Se realizo el modelo entidad relacion, con sus respectivas tablas, Pk, tipos de datos y sus relaciones.

Diagrama

Descripción generada automáticamente

**Anexo H2: Base De Datos Relacional**

**DESCRIPCIÓN**

**Anexo H3: Sql**

Se realizo la base de datos en Heidi con ayuda de Xamp. Y se crearon las tablas correspondientes con SQL.

**CREATE DATABASE if NOT EXISTS g6\_proyecto;**

**USE g6\_proyecto;**

**DROP TABLE if EXISTS restaurante\_producto;**

**DROP TABLE if EXISTS producto\_usuario;**

**DROP TABLE if EXISTS restaurante\_categoria;**

**DROP TABLE if EXISTS restaurante;**

**DROP TABLE if EXISTS producto;**

**DROP TABLE if EXISTS usuartio;**

**DROP TABLE if EXISTS categoria;**

**CREATE TABLE if NOT EXISTS restaurante(**

**idRestaurante INT PRIMARY KEY,**

**nombre VARCHAR(25) NOT NULL,**

**telefono INT NOT NULL,**

**email VARCHAR(45) NOT NULL,**

**direccion VARCHAR(30) NOT NULL**

**);**

**CREATE TABLE if NOT EXISTS categoria(**

**idCategoria INT PRIMARY KEY,**

**nombreCategoria VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE if NOT EXISTS restaurante\_categoria(**

**idRC INT PRIMARY KEY,**

**idRestaurante INT,**

**idCategoria INT,**

**CONSTRAINT fk\_rest\_cate FOREIGN KEY(idRestaurante) REFERENCES restaurante(idRestaurante),**

**CONSTRAINT fk\_cate\_rest FOREIGN KEY(idCategoria) REFERENCES categoria(idCategoria)**

**);**

**CREATE TABLE if NOT EXISTS producto(**

**idProducto INT PRIMARY KEY,**

**nombre VARCHAR(25) NOT NULL,**

**categoria VARCHAR(25) NOT NULL,**

**idCategoria INT,**

**CONSTRAINT fk\_cate\_prod FOREIGN KEY(idCategoria) REFERENCES categoria(idCategoria)**

**);**

**CREATE TABLE if NOT EXISTS usuario(**

**idUsuario INT PRIMARY KEY,**

**tipoIdentificacion VARCHAR(25) NOT NULL,**

**nombre VARCHAR(30) NOT NULL,**

**telefono INT NOT NULL,**

**email VARCHAR(50) NOT NULL,**

**direccion VARCHAR(30) NOT null**

**);**

**CREATE TABLE if NOT EXISTS restaurante\_producto(**

**id INT PRIMARY KEY,**

**precioProducto INT NOT NULL,**

**descripcionProdcuto VARCHAR(100) NOT NULL,**

**idRestaurante INT,**

**idProducto INT,**

**CONSTRAINT fk\_rest\_prod FOREIGN KEY(idRestaurante) REFERENCES restaurante(idRestaurante),**

**CONSTRAINT fk\_prod\_rest FOREIGN KEY(idProducto) REFERENCES producto(idProducto)**

**);**

**CREATE TABLE if NOT EXISTS producto\_usuario(**

**id INT PRIMARY KEY,**

**idProducto INT,**

**idUsuario INT,**

**CONSTRAINT fk\_prod\_usa FOREIGN KEY(idProducto) REFERENCES producto(idProducto),**

**CONSTRAINT fk\_usua\_prod FOREIGN KEY(idUsuario) REFERENCES usuario(idUsuario)**

**);**

**Anexo H4: Diagrama De Navegación Del Sistema De Promoción De Restaurantes** De Delivery

Para la realización del diagrama navegacional se utilizó el software libre Draw.io, en su última versión.

**Imagen que contiene Icono

Descripción generada automáticamenteDiagrama

Descripción generada automáticamente**