

REQUISITOS TIComo

- **Requisitos funcionales**

- De Almacenamiento de Información:
 - Los datos que hay que almacenar en la base de datos serán: admin, restaurante, cliente, rider.
 - De los administradores nos interesa almacenar:
 - Nombre (obligatorio)
 - Apellido (obligatorio)
 - Zona (obligatorio)
 - Email (obligatorio)
 - Password (obligatorio)
 - De los restaurantes almacenaremos:
 - Nombre (obligatorio)
 - Razón Social (obligatorio)
 - CIF (obligatorio)
 - Dirección Completa (obligatorio)
 - Teléfono (obligatorio)
 - Email (obligatorio)
 - Categoría (obligatorio)
 - De los clientes almacenaremos:
 - Nombre (obligatorio)
 - Apellidos (obligatorio)
 - NIF (obligatorio)
 - Dirección completa (obligatorio)
 - Teléfono (obligatorio)
 - Email (obligatorio)
 - Password (obligatorio)
 - De los riders almacenaremos:
 - Nombre (obligatorio)
 - Apellidos (obligatorio)
 - NIF (obligatorio)
 - Tipo de vehículo (obligatorio)
 - Matricula (en el caso de que sea vehículo a motor)
 - Carnet (en el caso de que sea vehículo a motor)
 - Email (obligatorio)
 - Password (obligatorio)
 - De los platos almacenaremos:
 - Nombre (obligatorio)
 - Foto (obligatorio)
 - Descripción (obligatorio)
 - Precio (obligatorio)

- Apto para veganos (obligatorio)

- De Reglas de Negocio:
 - Los administradores tendrán la opción de consultar cuentas, añadir, modificarlas, eliminarlas y activarlas/desactivarlas (no se borrará la cuenta, pero se deshabilita el acceso del usuario a esta)
- De Interfaz:
 - La interfaz del sistema tendrá una sección de consulta y edición de perfil de usuario.
 - Tendremos una sección en la interfaz donde pueda darse de alta el restaurante y posteriormente su carta correspondiente
- De Casos de uso:
 - Para acceder a la herramienta, los diferentes tipos de usuario deberán identificarse ante el sistema con un usuario y contraseña. Para eso debe utilizarse el patrón RBAC.
 - Debe permitir a los usuarios registrarse en el sistema, solicitar, gestionar y valorar sus pedidos de comida. Una vez entregado el pedido se podrá valorar tanto al restaurante como al rider
 - Debe permitir a los riders consultar la lista de pedidos pendientes de servir y marcar los pedidos como entregados. Cada rider sólo podrá tener en proceso de entrega un número máximo de pedidos definido en el sistema.
 - Debe permitir a los administradores gestionar la lista de restaurantes y la carta de cada uno de ellos, así como consultar los pedidos realizados y la facturación por restaurante, día y mes. También se podrá acceder a las valoraciones de restaurantes y riders.
 - La aplicación debe tener un módulo de gestión de usuarios, que sea solamente accesible por el administrador. Dicho módulo debe tener las siguientes funciones: Crear (solo administradores y riders), modificar, consultar y eliminar usuarios.
 - El usuario puede entrar en el perfil y modificar todos sus datos excepto el NIF e email.
 - El administrador podrá acceder a una sección para modificar todos los datos de todos los usuarios.

- **Requisitos de calidad**

- Disponibilidad
- Eficiencia
- Flexibilidad
- Integridad:
 - Conexión segura entre el servidor y el cliente

- La información sensible (datos del usuario) debe ser protegida adecuadamente
- El CIF de la información fiscal de los restaurantes debe estar cifrado.
- El NIF de los clientes y riders tiene que estar cifrado.
- El password de los usuarios tiene que estar cifrado.
- Al registrarse el usuario tendrá que introducir y repetir la contraseña como método de confirmación de contraseña.
- Interoperabilidad
- Confiabilidad
- Robustez:
 - Debe establecerse una política de contraseñas segura.
 - Debe tener el mínimo de vulnerabilidades e identificarlas adecuadamente.
- Usabilidad
- Mantenibilidad
 - Se debe cumplir con las mejores prácticas de programación por ello se debe reducir el número de defectos en el código, no existiendo defectos de las severidades más graves.
- Portabilidad:
 - Debe ser accesible desde cualquier navegador web.
- Comprobabilidad

- **Requisitos no funcionales (restricciones)**

- Se debe realizar un AGR del sistema en cumplimiento con la norma ISO27001.
- Conexión mediante https.
- Con el objetivo de asegurar el correcto funcionamiento de la aplicación las pruebas que se realicen deben cubrir al menos el 60% del código (mayor al 60%). Por ello, no se permitirá la existencia de código en la aplicación que no sea utilizado.
- De cara a facilitar los cambios en el código se debe evitar la repetición de fragmentos de código, así se exigirá que el código duplicado sea inferior al 10% (menor al 10%). También para mejorar la analizabilidad del código se debe evitar la creación de métodos con una complejidad ciclomática excesiva y en caso de crearlo se debe justificar su existencia.
- Para poder verificar que nuestro código cumple con estos criterios de calidad nos ayudaremos de Sonar y Sonar Lint.
- El nombre de los clientes tendrá un número de caracteres limitados.
- Como aplicación para la base de datos utilizaremos MONGODB
- La matrícula de los riders (en caso de tener) tiene que seguir un formato de matrícula correcto previamente establecido.
- Una cuenta de correo electrónico solo pertenece a un solo perfil de usuario.
- A la hora de realizar un pedido, no se podrá editar el plato que elija el usuario
- Los riders tienen un número máximo de pedidos simultáneamente.

- Únicamente las cuentas de los riders y de los usuarios se podrán deshabilitar
- Cuando se deshabilita la cuenta, se negará el acceso al usuario, y únicamente podrá volver a habilitarlo contactando con los administradores
- Para eliminar, tanto un usuario, como un rider, todos los pedidos que estén unidos a su cuenta tendrán que estar en estado “entregado”
- El restaurante únicamente se podrá eliminar, y solo se podrá realizar a través de los administradores
- Aunque un pedido este en estado “entregado”, no se eliminará de la base de datos
- Los pedidos únicamente se le mostrarán al rider para cogerlo cuando estén en estado de “preparación”
- En el sistema habrá tantos administradores como sean necesarios. Cualquier administrador puede gestionar otros administradores
- El email de los usuarios servirá como user login.
- El usuario puede modificar todos sus datos excepto el NIF e email.

HISTORIAS DE USUARIO

- HDU1: Administración de información de riders
- HDU2: Administración de información de clientes
- HDU3: Administración de información de administradores
- HDU4: Creación de sistema de Login
- HDU5: Creación de sistema de registro
- HDU6: Creación de sistema de gestión de restaurantes
- HDU7: Creación de sistema de gestión de la carta
- HDU8: Creación de la interfaz de usuario de sistema
- HDU9: Creación sistema pedidos
- HDU10: Creación de sistema de consulta y configuración perfil
- HDU11: Creación de sistemas de facturación de restaurantes