

DISEÑO DE APLICACIÓN PARA EMPRESA DE RENTA DE PRENDAS.

OBJETIVOS:

- Mejorar la experiencia del usuario al momento de querer rentar una prenda
- Automatizar el proceso de visita a la tienda al permitirle al cliente acceder a los horarios disponibles para agendar cita
- Diseñar una aplicación funcional para todos sus actores principales
- Optimizar el proceso de renta y ofrecerle la opción al cliente de rentar directamente en línea
- Ofrecer al usuario la opción de envío y/o recolección de prendas a domicilio

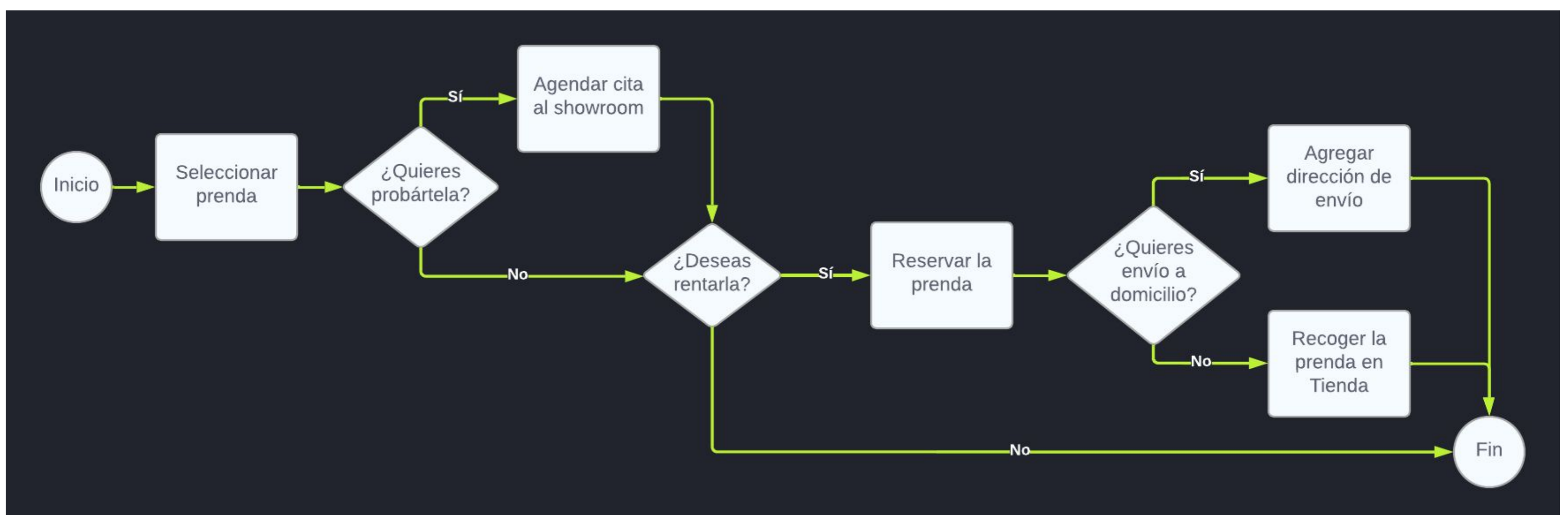
ACTORES :

En nuestra solución del sistema podemos identificar 4 actores:

Actor	Tipo	Casos de Uso
Cliente	Primario	Actor principal y representa a todos los clientes que desean rentar un vestido y/o accesorios para un evento dentro de la aplicación.
Tienda	Primario	Representa a la(s) persona(s) que se encarga de subir y mantener actualizada toda la información disponible en la aplicación y de citas disponibles, al igual que el manejo de usuarios.
Repartidor	Primario	Representa a las y los repartidores de la tienda que se encargan de llevar y recolectar todas las prendas de la tienda al domicilio y viceversa, al igual que el registro de los envíos y el estado de las prendas.
Base de Datos	Secundario	Representa a la base de datos, donde se tiene guardada toda la información de los clientes, citas realizadas a través de la aplicación, artículos disponibles y datos en general.

Para poder cumplir con este escenario es necesario que cada actor asuma un rol un poco diferente al actual.

DIAGRAMA DE PROCESO PROPUESTO:



CONCLUSIONES:

- Para tener un proceso más eficiente es necesario tener en cuenta un nuevo trabajador de la tienda con la función de repartidor.
- Al ser una aplicación es muy importante que el uso de todos los datos de los usuarios de la aplicación sea ético y responsable. Para esto se debe tener las clases con sus atributos y roles muy bien definidos.

REFERENCIAS:

- Weitzenfeld, A. (s. f.). Ingeniería de Software Orientada a Objetos con UML, Java e Internet. Thomson.
- Berzal, F. (s. f.). Relaciones entre clases: Diagramas de clases UML.