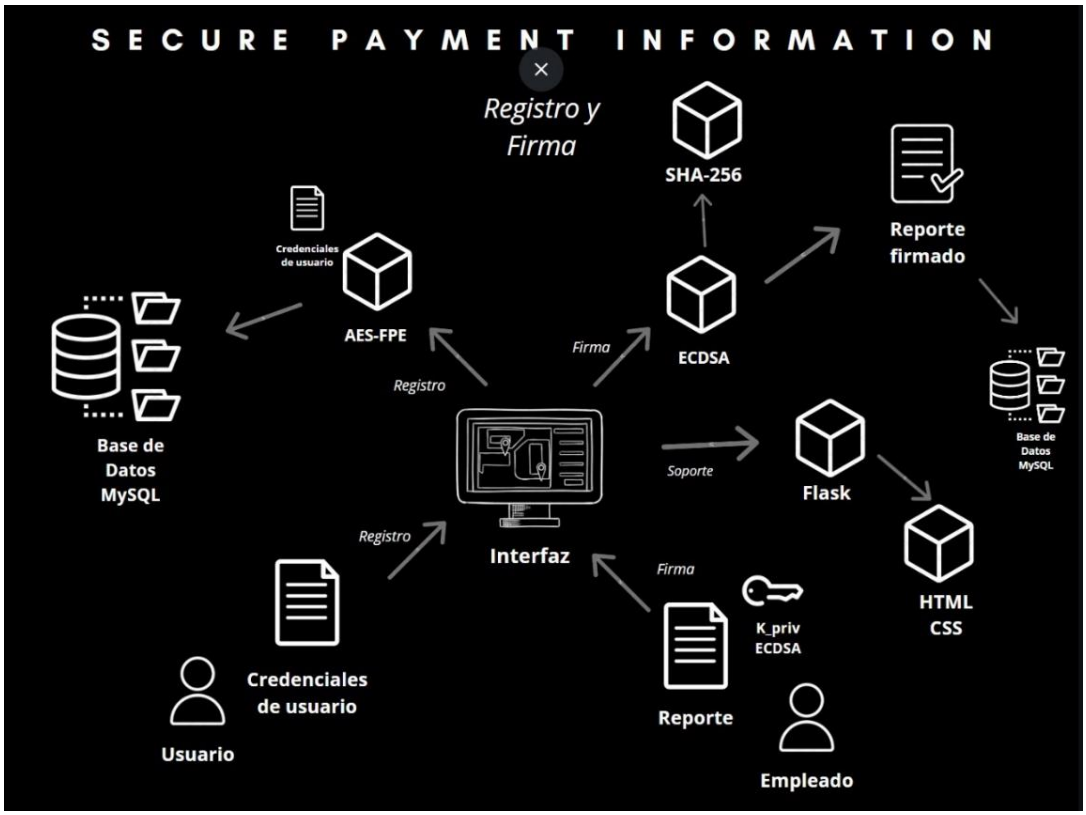
Secure Payment Information

Avances de proyecto del 30 de octubre.

La arquitectura de nuestra propuesta de solución esta dividida en tres esquemas, cada uno representa un camino o funcionalidad que el sistema tendrá. Estos esquemas están compuestos por nuestros actores (Propietario, Empleado y Usuario), los algoritmos criptográficos por implementar y las tecnologías que darán soporte a todo el sistema. A continuación, se muestra cada uno de los esquemas propuestos, así como una explicación de la secuencia y sus componentes.

1. Registro de usuarios y firma de reportes de ventas

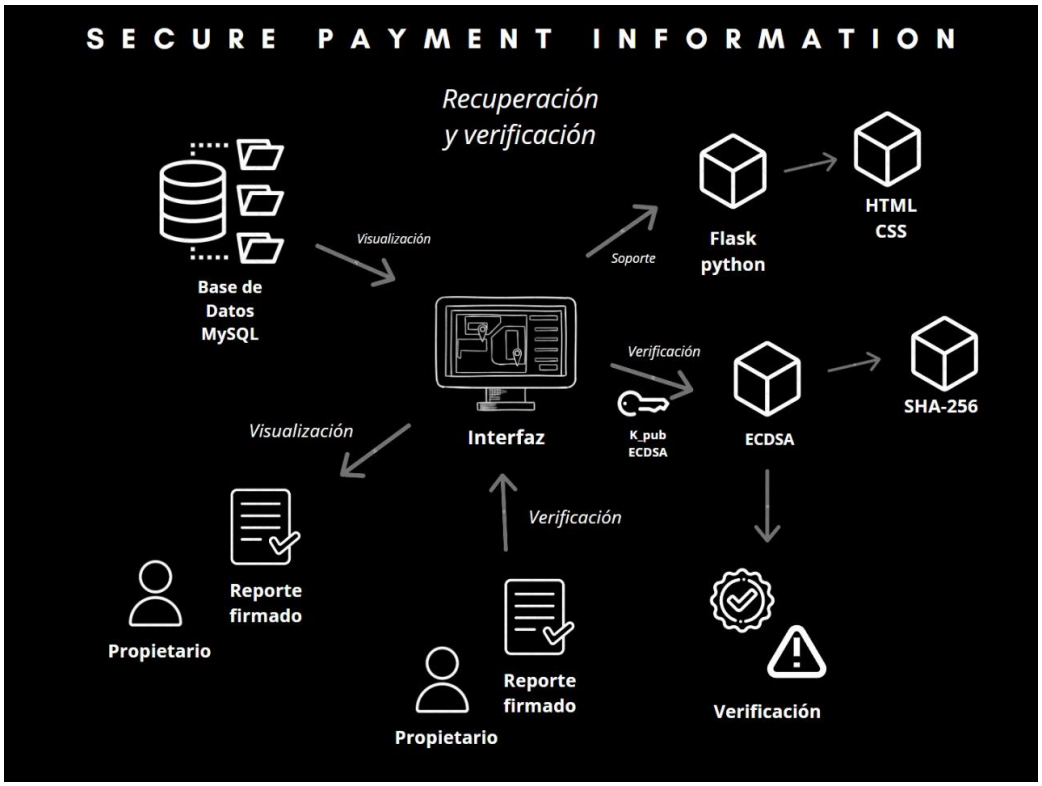


En este esquema se tienen dos solicitudes al sistema. La primera es el registro, en el cual se considera que, además de los datos del usuario, se ingresa su número de tarjeta personal. Estos datos son proporcionados mediante la interfaz, y el número de tarjeta se cifra usando AES en modo FPE (Format Preserving Encryption) antes de ser almacenado en la base de datos junto con el resto de la información del usuario.

La segunda solicitud al sistema es la firma de los reportes de ventas. El empleado debe ingresar el reporte a firmar y su llave privada, como lo indica nuestro algoritmo de firma ECDSA. Es importante mencionar que este algoritmo también requiere una función hash criptográfica, motivo por el cual existe esa conexión en el diagrama. Como resultado, se devuelve el documento firmado, el cual se almacena en la base de datos.

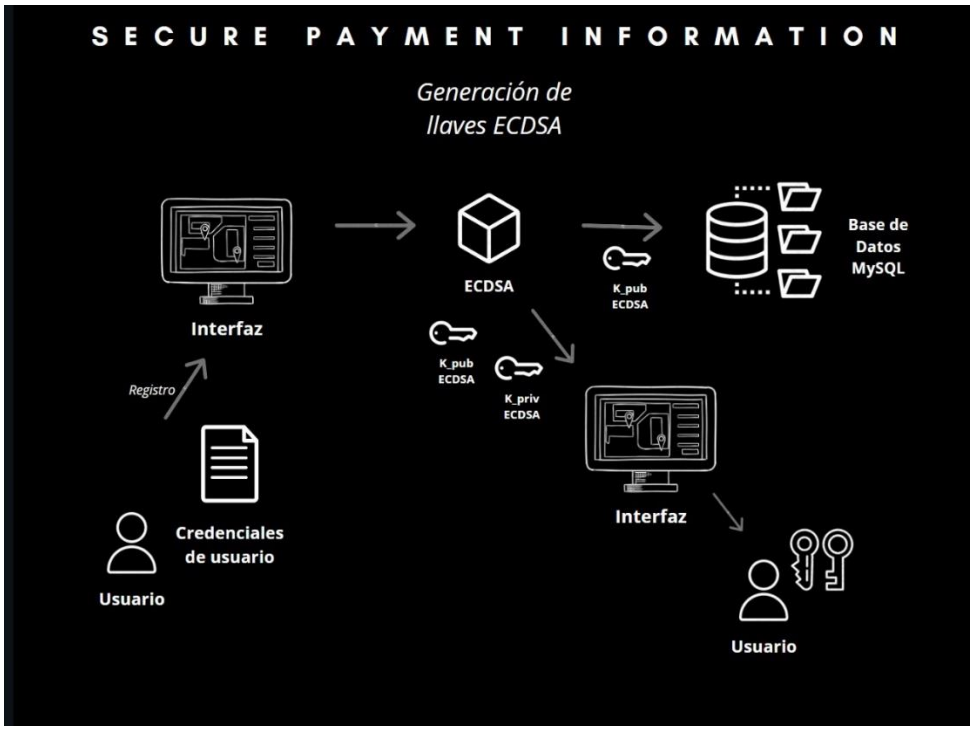
La conexión de soporte se refiere al framework Flask, que se utilizará para desplegar el sistema como un servicio web, así como a las tecnologías de desarrollo web que se emplearán para la interfaz gráfica.

2. Recuperación de documentos firmados y su verificación



En este esquema, encontramos los módulos de recuperación de documentos firmados y de verificación de estos. La solicitud se realiza mediante la interfaz gráfica a la base de datos para recuperar los reportes previamente firmados. Si el propietario así lo desea, se procede a la verificación aplicando nuevamente ECDSA; este algoritmo devolverá un valor de verdadero o falso, el cual se desplegará en la interfaz gráfica para el propietario.

3. Generación de llaves ECDSA



En nuestro último esquema, tenemos el módulo de generación de llaves. Estas se generan mediante ECDSA en el momento de registrar a un nuevo usuario. Una vez generadas, se devuelven al usuario (empleado) y la llave pública se almacena en la base de datos, con el fin de poder acceder a ella en el momento de la verificación de firma por parte del propietario.