**TRUECALLER WEB APPLICATION SYNOPSIS**

**CORE FEATURES OF THE APPLICATION**

**1.USER AUTHENTICATION**

**2.CALLER IDENTIFICATION**

**3.SPAM DETECTION**

**4.SEARCH FUNCTIONALITY**

**EACH FEATURES IMPLEMENTATION LOGIC**

**1.USER AUTHENTICATION**

* Setting up routes for register and login
* An option to login with email/phone
* Password Reset feature
* Otp response through services like twilio or firebase
* Using jwt for authentication and Oauth for authorisation is essential.
* Link to learn jwt: https://www.geeksforgeeks.org/json-web-token-jwt/

**2.CALLER IDENTIFICATION**

* Caching systems like Redis or Memcached to reduce repetitive database queries and API calls for frequently searched numbers must be implemented.
* The search function should retrieve a mobile number either from a database or an api call, Free tier plan can be used from Numverify or TrueCanm
* A database entry should be created for a frequently searched number so that search data is optimised.
* Use celery to search for frequently searched numbers

**3.SPAM DETECTION**

* Use APIs like **Numverify**, **TrueCNAM**, or **Twilio Lookup** to check spam scores or reputation
* Allow users to flag numbers as spam.
* Collect datasets from public sources like Kaggle or spam-detection datasets

**4.SEARCH FUNCTIONALITY**

* The search function should retrieve a mobile number either from a database or an api call, Free tier plan can be used from Numverify or TrueCanm
* A database entry should be created for a frequently searched number so that search data is optimised.
* Caching systems like Redis or Memcached to reduce repetitive database queries and API calls for frequently searched numbers must be implemented.

**Tech Stack to be used**

REACT

HTML&CSS

JAVASCRIPT

FLASK

MySql/MongoDB

VERCEL FOR DEPLOYMENT