CSE 345/545 Foundations of Computer Security Course Project

Real Estate Aggregator

1. **Introduction**:

The Real Estate Aggregator System is a platform designed to enable secure property transactions and document verification in the real estate industry. The focus of this project is to create a portal that facilitates the secure exchange and verification of property-related documents and to enable secure transactions during property buying/selling/renting.

2. **Requirements**:

The project requirements will be adapted to cater to the real estate domain while maintaining the core security and document verification elements.

The two **key requirements** are:

   a. Property purchases - The portal must act as an <u>escrow</u> service. The property name transfer and the fund transfer should be done through the portal.
   b. Property leasing/renting - The portal must act as an aggregator. There must be **two separate contracts**, one between the portal and the leaser and one between the lessee and the portal.

Required Functionalities:

   ● Properties will be categorized by buy/sell or rent.
   ● An escrow based payment system (can be in installments for the case of buying)
   ● Maintenance of profile information of the user/organization.
   ● Creation of digital contract (e-signed by the involved parties) when both parties agree to the terms and conditions specified.
   ● eKYC: You must implement a mechanism to invoke an API for eKYC, which will be utilized for high-level tasks such as registration and transactions. The API endpoint and credentials will be provided to you in advance.

- Rental Contracts Creation and Signing:
  - The portal creates rental contracts (through e-sign) when both parties agree to terms.
  - Contracts include property details, buyer/seller info, date, and terms.
  - Provide a mechanism to verify that the contract is indeed sanctioned by the portal.

3. **Users**:

- Each user (except the admin) will upload documents (proof of identity) which must be automatically verified by the portal. Once verified the users can use the platform.

3.1.1 Sellers and Lessors:

- Ability to list/delete/modify their uploaded property listings.
- Ability to view past sell/rent out history.
- Ability to report a malicious buyer which would then be sent to the admin.
- Sellers and Lessors upload property-related documents (ownership proofs, identification documents) which must be automatically verified for authenticity.

3.1.2 Buyers and Lessees:

- Users can search and select properties based on type, budget, location, amenities and dates of availability.
- Ability to report a malicious listing which would then be sent to the admin.
- Ability to view past purchase history.

3.1.3: Aggregator (Platform Moderators)

- Admin can view all the users and their details.
- An admin can moderate any suspicious activity/listings on the platform.
- Ability to automatically verify the contracts(through e-sign without using any external libraries).

2.5. Other Requirements

- **Public Key Certificates:** The secure Real Estate website must use public key infrastructure (PKI) and SSL/TLS (HTTPS) to enforce the application's security. You can establish your own certificate issuing authority for this

project. A minimum of two functions must employ PKI, and you may decide the extent of the PKI applicability to the functions.

- **OTP:** The secure real estate website must employ OTP (One Time Password) technique with a virtual keyboard feature to validate highly sensitive transactions for at least two of the functions in requirements. You may decide the extent of the OTP applicability to the functions.
- The secure real estate website should allow multiple users to use the system simultaneously.
- Secure transaction logging is required to enable external audits.
- Must employ security features to defend against attacks on the secure real estate website (the TAs and students will test the project).
- Make sure you read about payment gateway compliance and store only the data allowed to be stored. For example, keeping card details and private keys on your local database is not permitted.

3. Programming Languages and Framework

Each group can choose the programming language of their preference through consensus. Each group can use the following: OS: Windows (XP or any newer version) or Linux. DBMS: MySQL, Mongo DB, SQLite, or Postgres. Web server: Nginx, IIS, or Apache. However, if you choose to use other types of OS, database systems, or web servers for the project, you need to discuss it with the TA since the TA may be unable to help you with the project.

4. Milestones

The TAs will continuously evaluate your progress in bi-weekly meetings. Groups will be required to demo their project to their TAs at the end of each month and will be graded according to the following milestones:

- August Milestone   [2.5%]
  - Decide the full tech stack, including OS, web server, and database (a rough idea).
  - Install and configure the web server along with SSL/TLS certificates.
  - Host any sample **HTTPS** website on the VM provided.

- September Milestone   [2.5%]
  - Host an HTTPS website on the web server that enables the following functionalities
    - Users of all categories can log in/signup on to the website to view and edit their profile information.
    - Creation of digital contract (e-signed by the involved parties) when both parties agree to the terms and conditions specified.

- Each user (except the admin) will upload documents (proof of identity) which must be automatically verified by the portal.
- Sellers and Lessors should be able to list/delete/modify their uploaded property listings.
- Buyers and Lessees can search and select properties based on type, budget, location, amenities and dates of availability.
- Admin can view all the users and their details.
- Portal should be able to automatically verify the user documents (through e-sign without using any external libraries).
- Ability to automatically verify the contracts(through e-sign without using any external libraries).

- October Milestone
  - Lessors/Sellers can delete/modify their own documents.
  - eKYC: You must implement a mechanism to invoke an API for eKYC, which will be utilized for high-level tasks such as registration and transactions. The API endpoint and credentials will be provided to you in advance.
  - A payment gateway system.
  - Admin can remove any suspicious patient or organization from the platform. Ability to automatically verify the contracts(through e-sign without using any external libraries).
  - In addition, the website must also use OTP for at least two of the functions in the requirements.
  - Complete any other requirements mentioned
- Finalize your tech stack and VM requirements. *No changes to these will be accepted after September.*

IMPORTANT: **10% BONUS marks** will be awarded to the groups who have used blockchain for the automatic verification system of property listings