**Software Specification and Design**

**Proposal –**

HAI (Homeowner’s Assistant for Insurance) is a personal filing system for all your insured belongings! Our system should make filing for insurance easier by storing insurance documents electronically and centrally. Users can easily pull up all their insurance documents either through our app or the web. A User also can request an email summary of all their items in our system.

**Functional Requirements –**

1. Manage user content:

a. User can add insured items to be stored.

i. User will select item category.

ii. User will add item photograph/image of insurance papers and item.

iii. User will add item value (and have the option to have the item appraised/suggested price given based on previous items in our database).

iv. User will add item make, model, serial, and comments, as well as insurance company information.

b. User can select item to view and update item.

c. User can remove posted item (self-content).

d. User can access their information from the web.

e. User will have to login to manage content.

2. Interactive GUI:

a. User can interact with GUI to perform all tasks mentioned in above section.

b. The GUI should be extremely easy to navigate as Users do not need any added difficulty after their items have been lost, stolen or damaged.

3. Data Storage

a. Database as well as local storage.

b. User will retain all personally identifiable information on device.

c. Users will also have access to their information via the web.

**Non-Functional Requirements –**

1. Operational Requirements:

a. The system will operate in either (or both) iOS and Android environment.

2. Performance Requirements:

a. The system will post user content live within 2 seconds.

b. The system will email user’s their content.

3. Security Requirements:

a. The system will require password authentication.

b. The system will require email authentication for user account.

4. Cultural and Political Requirements:

a. The system should provide support for multiple languages.

b. The system will remind user(s) to obey all local and regional laws regarding insurance fraud.

**Use-Case Diagram –A close up of a map

Description automatically generated**

**Use Case Descriptions –**

1. LOG IN/OUT

ID: 1

Importance Level: High

Primary Actor: User/Admin

Use Case Type: Detailed

Description: How a User would log in to our system

Trigger: User enters email (username) and password

Type: Internal

Relationships –

Association: REGISTER USER PROFILE

Include:

Extend:

Generalization:

Normal Flow of Events:

1. Enter email (username) and password

2. Click Log In

SubFlows:Alternate/Exceptional Flow:

1. Incorrect email (username) or password

2. VALUATE AN ITEM

ID: 2

Importance Level: High

Primary Actor: User

Use Case Type: Detailed

Description: How a User would have their item valuated through our system

Trigger: User clicks on price suggestor button and enter item description

Type: Internal

Relationships –

Association:

Include:

Extend: Add an item

Generalization:

Normal Flow of Events:

1. User clicks on price suggestor button

2. User enters item description

3. User is given a estimation of their product’s value based on most recent inputs for similar items

SubFlows:Alternate/Exceptional Flow:

1. No similar items found

**Class Diagram –A screenshot of a social media post

Description automatically generated**

**Activity Diagrams –A close up of a map

Description automatically generated**

A close up of a map

Description automatically generated

**Sequence Diagram – A close up of a map

Description automatically generated**

**Communication Diagram –A close up of text on a white background

Description automatically generated**

**Concept behind smart algorithm –**

Item appraisal is done by searching for all items that are similar to the item being appraised (through user description of an item: make, model, year). It is based on the median of all the prices. A deviate decimal is calculated by how much deviation there is from our suggestion to the 10 most recent values inputted by users in order to keep our estimates current.

**Future considerations for smart algorithm –**

Further improve accuracy by having users also describe the ware of the item and compare values only to those items that have been describe as having the same amount of ware. For instance, a user will have the condition options of new, like new, and old.