Requirements Document

Members:
Denis Bajic
Lohengrin Mindiola Diaz
Serapheim Dimitropoulos
Meagan Lenoir

HawksMarket
Team K

Problem Statement

College life requires students to either buy or to get rid of many things. Imagine a freshman -- she has to buy textbooks and electronics for her class. She may need to find an apartment near campus to live or register for dorming. On the other hand, you have students graduating that could benefit a lot if they were able to sell a lot of their stuff, like furniture or books. The problem is that the existing methods to buy and sell do not perfectly meet the students' needs. They are either not reliable or too complicated due to their generality.

The existing methods are the following:
- <u>Amazon</u>: Buyers cannot see what they bought until it arrives to their mailbox after they paid. In addition, shipping can take a long time depending on the shipping method and location.
- <u>IIT Bookstore</u>: Specific for books and too expensive. You can sell your books there for less that you will sell them somewhere else. Buying used books is as expensive as buying new ones from somewhere else.
- <u>IIT Facebook Page</u>: Basically, it is not meant for that.
- <u>Apartment websites</u>: Specific to apartment finding. Locations widely vary instead of being close to campus. Entries there are posted late and the feedback is not personal.
- <u>CraigsList</u>: Not reliable!

Product Description

Our product aims to provide the means to make this procedure a lot easier and simpler for IIT students. We will be building an web application that will streamline the procedure of buying from and selling to other IIT students. Students will just post the items that they want to sell with a full description, pictures and a price. Other students can look/search the list of posted items and express their interest in buying them. After they do that, the seller of the item gets notified of the interested buyer and their info. After this, they can arrange to meet up to finalize the buying process themselves.

The benefits over the existing methods are the following:
- Buyers can ask seller to inspect item before purchase.
- Reliable - the service is exclusive to IIT students.
- No shipping time.
- Multiple Categories -Not limited to book purchase/selling or apartment finding.
- Room for negotiation.

● Well-defined target market.
● No communication overhead.
● A chance to meet more people from IIT.

User Categories

Seller:

A seller creates posts of items that he wants to sell. He has to provide a Title, a description, picture(s) and tags for each entry. The tags should be descriptive so buyers can later search for the item easier. Sellers are notified whenever a buyer is interested in their item. Then it is up to them to choose a student from his list of buyer and contact her to arrange the transaction details in person. Sellers are also responsible for deleting their sold items from the website.
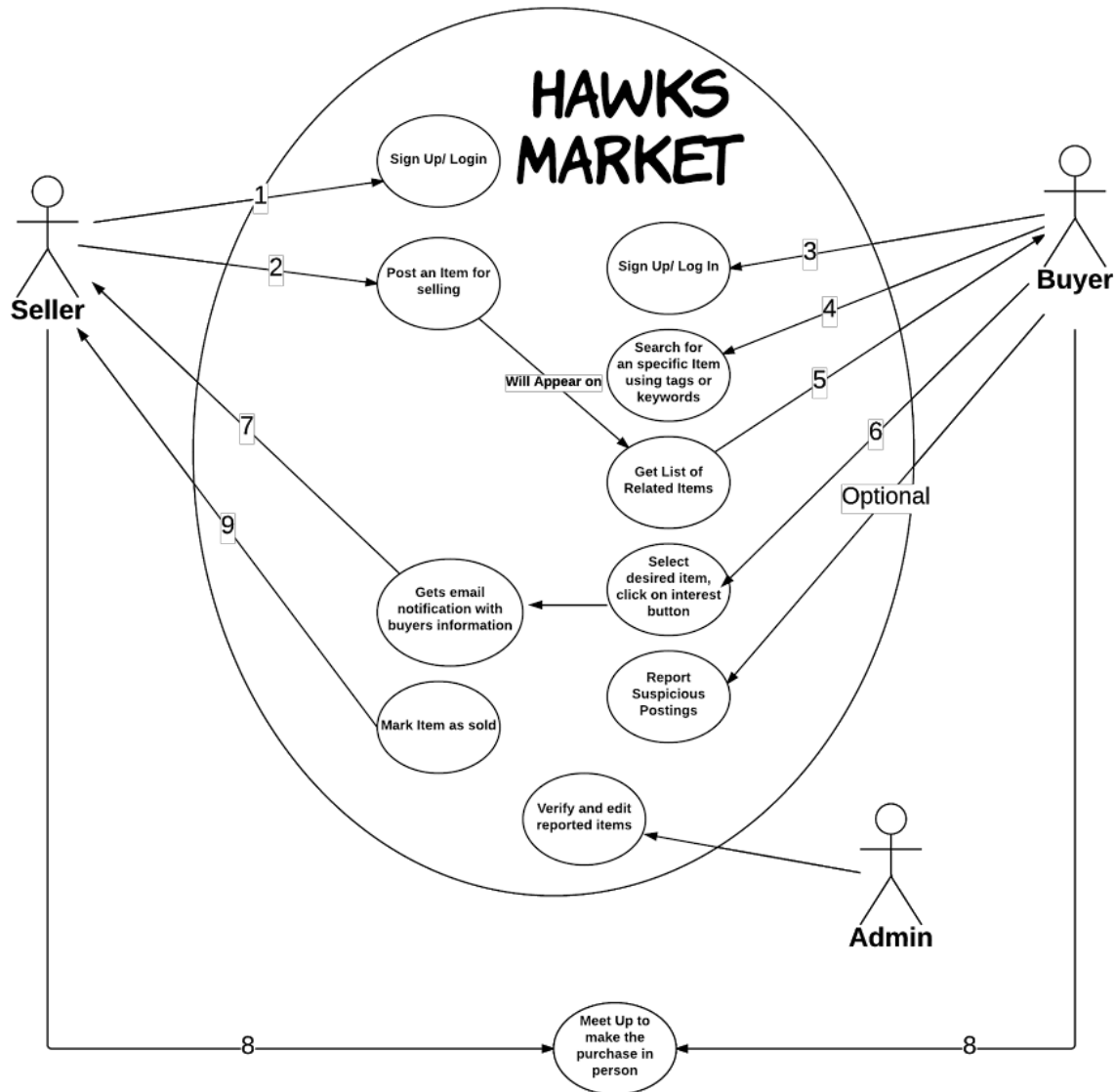
Buyer:

A buyer can search for an item that he/she needs by item description or tag. When he searches through the list and finds an item that he wants, he clicks the according interest button. Then he waits until the buyer contacts him/her. In the case that one of the search results are not relevant with what he/she searched for or it is inappropriate, he/she can report the item.

Note: A student account can post items and buy items. So every student can be a buyer and/or a seller.

Admin:

An admin has the role of going through the reported items and decide if the reports are actually valid and need to be removed or not.

Use Cases

Data Flow

We can see that the critical pieces of data are the item posts, the information within them and the buyer information. We can see from the use case diagram how this information flows from seller to the buyer. The seller posts the item information, then the buyer is able to search for items and get the post previously posted by the seller as result of his/her search. After, the buyer expresses interest, his/her information in sent via email to the seller, and he/she can use this information to contact the seller to arrange the meeting.

Requirements

<u>User Requirements:</u>
- The application shall allow easy communication between two students that will lead to the purchase of an item.
- The application shall allow the buyers to show the items they want to sell to other students.
- The application shall provide a way to allow the buyers notify the seller about their interest regarding an item.

<u>Functional Requirements:</u>

1. **Authentication and Basic Functionalities:**
   a. Users shall be able to sign up/login only with a @hawk.iit.edu account
   b. A User shall have to verify his email address before being able to access the system. The application will send a verification email to the user to confirm his/her address
   c. A User shall be able to set up and change the password
   d. A User shall have the option to add in a phone number as an optional form of communication if they want to be contacted through it
   e. A User shall be able to select to perform tasks as a seller or buyer in the same account. Similar to a social network, where the user can upload posts and like posts at the same time, but in this case the user can both post and be interested in items.
2. **UI Visualization:**
   a. The application shall display by default a list of the categories the user can quickly search for Books, Apartments, Electronics, and Furniture. Of course the application shall have a search box so the user can search for item categories outside the ones of the predefined list.
   b. The application shall also display a list of the top ten tags used in item entries, so users can see what is trending at the current moment.
   c. Once the application gets selected, a new layout will appear and will include a New Post Option if the user wants to post and Item, and below there will be the most recents post from that category. The application shall also have a search bar on the top, where the user can filter his search by description keywords and tags

d. On the top of the screen there will be one tab. If the user clicks on the first tab, it will display a list that includes all the items that the user has clicked the interest button from.

e. The application shall allow the user to click on an item post and get a new layout with the picture(s) of an item, the title, a full description (including price), and an interest button.

3. **Seller Functionalities:**

a. Sellers shall be able to post many products. The required field for the product are title, at least one picture, and a full description that includes price, and tags, so other users can find the item by searching. Without any of these fields, the item cannot be posted. Also, they seller should select a category for the item.

b. Sellers shall receive an email when each student shows interest. The email shall contain the name of the student, a form of contact(email is required, phone number is optional), and the date the student clicked on the interest button.

c. Sellers shall be able to choose to receive a daily summary of interested people instead of an email per interest.

d. Sellers shall have the freedom to choose which buyer they are going to arrange a face-to-face meeting with to sell the item.

e. Sellers shall be able to mark an item as sold, and remove it from his/her posts

4. **Buyer Functionalities**

a. Buyers shall be able to select a item category, and filter by tag or by keyword in the description using the search bar.

b. Buyers shall be able to click on a product a visualize its details

c. Buyers shall be able to press a button to show they are interested in buying the item in an active posting, they then will have to wait for the seller to contact them.

d. Buyers shall be able to click on the interest list tab to see all the items they have expressed interest in, and verify that they are still available or have been sold.

e. Buyers shall not be able to contact the seller since they will not have access to their information

f. Buyers can remove items from their interest list.

g. Buyers shall be able to press the "Report Post Button" in case the post is not coherent or it is inappropriate. Then, he can submit a comment regarding the post.

5. **Admin Functionalities**

a. The admin shall have access to the list of reported posts

b. The admin shall go over the reports and delete the ones that violate our regulations

Non-Functional Requirements:

1. **Product Requirements**
   a. Usability: The system shall have an interactive user interface.
   b. Efficiency: The system shall perform decently in terms of performance when handling requests. (IIT users (students + faculty) can be 100,000 max online simultaneously!)
   c. Dependability: Even though the system shall be available/reliable at all times, the safety and the integrity of user data should be our number one goal.
   d. Security: Each users data shall not be visible to other users (only in the case a user expressed interest, the seller will get the user's information, e. g email) neither their items of interest or the items that they have reported as inappropriate.
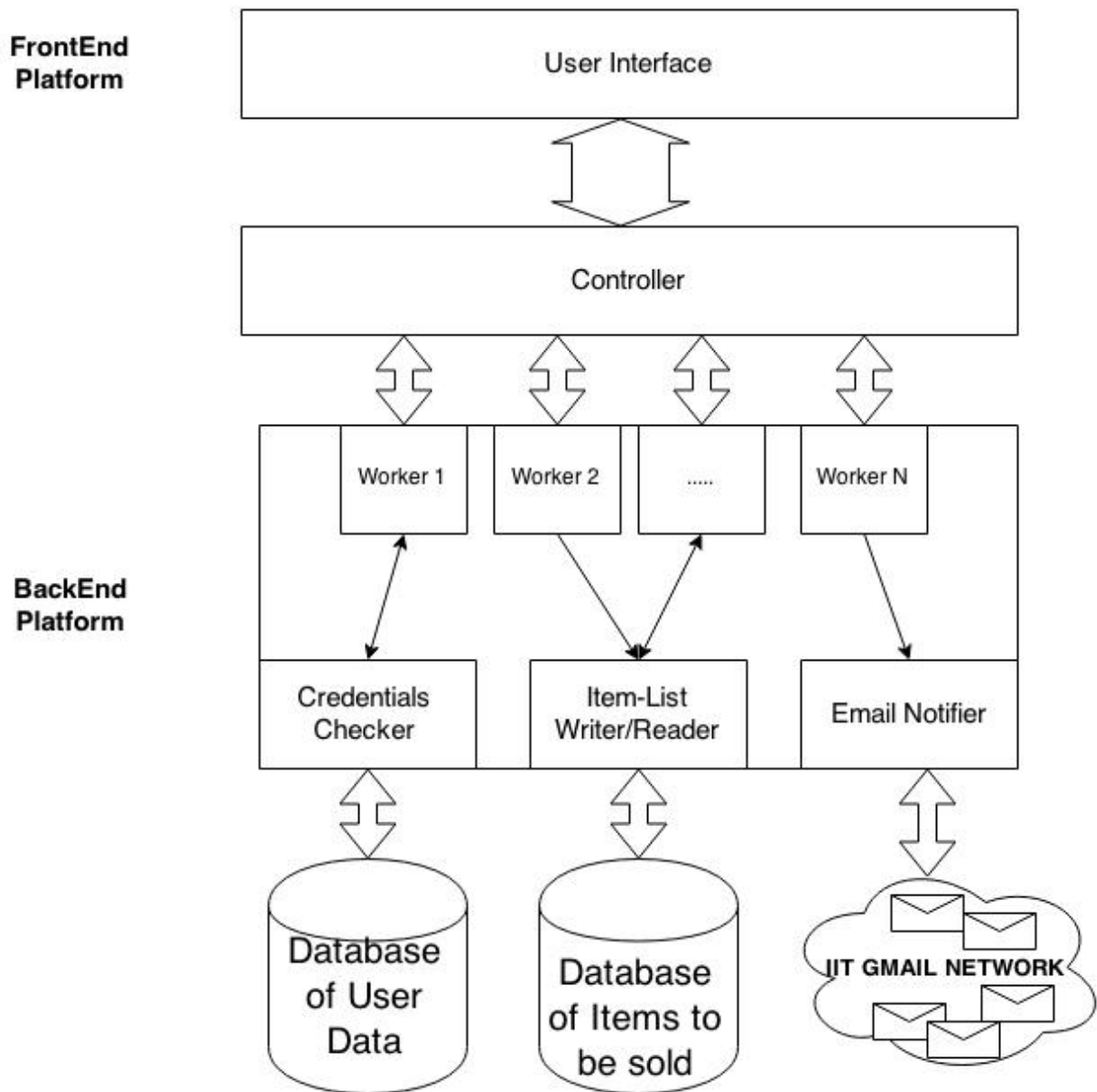
2. **Organizational Requirements**
   a. Environmental Requirements: Not Applicable.
   b. Operational Requirements: The user categories defined above should have and be limited to their according operations mentioned above. Nothing less nothing more.
   c. Development Requirements: A working and presentable prototype shall be ready by the end of the semester for a quick demonstration.

3. **External Requirements**
   a. Regulatory Requirements: Sellers shall make no more than one entry per item. Buyers shall express their interest at most once. An admin's sole role is to regulate the entries submitted by the sellers so no inappropriate entries stay on the system. The user is required by a disclaimer to remove the items sold, in order to prevent more people from expressing interest in the item. If after a month, an item has at least 5 users interested, and the seller has not removed the item, the admin will be required to remove the item.
   b. Ethical Requirements: Sellers shall post items that they are really willing to sell/trade. An Admin's decision of whether a post is or is not appropriate shall be objective and unbiased.

Models

System model:

FrontEnd Platform

User Interface

Controller

BackEnd Platform

Worker 1 Worker 2 ..... Worker N

Credentials Checker

Item-List Writer/Reader

Email Notifier

Database of User Data

Database of Items to be sold

IIT GMAIL NETWORK

Major components and Interaction:

As you can see from the figure above every request on the system starts from the users interacting with the Front-End of the system. Basically the user interface. Whenever a specific event is triggered by the user through the user interface the controllers route a type of request to be handled by the Back-End.

The Back-End consists of workers that handle the requests sent by the controller. Each worker interacts with a specific module depending of the type of the request that it is handling. For example, interacting with user credentials for authentication, has the worker talking to the Credentials Checker which checks the user database for authentication purposes. The Item-List Reader/Writer module handles the creation, modification and deletion of entries by manipulating data from the specific database. The Email-Notifier module's sole purpose is to just notify Sellers of Buyers interested in their items and it is placed on the back-end simply because it needs to have direct and authenticated access to specific user data for contact information.
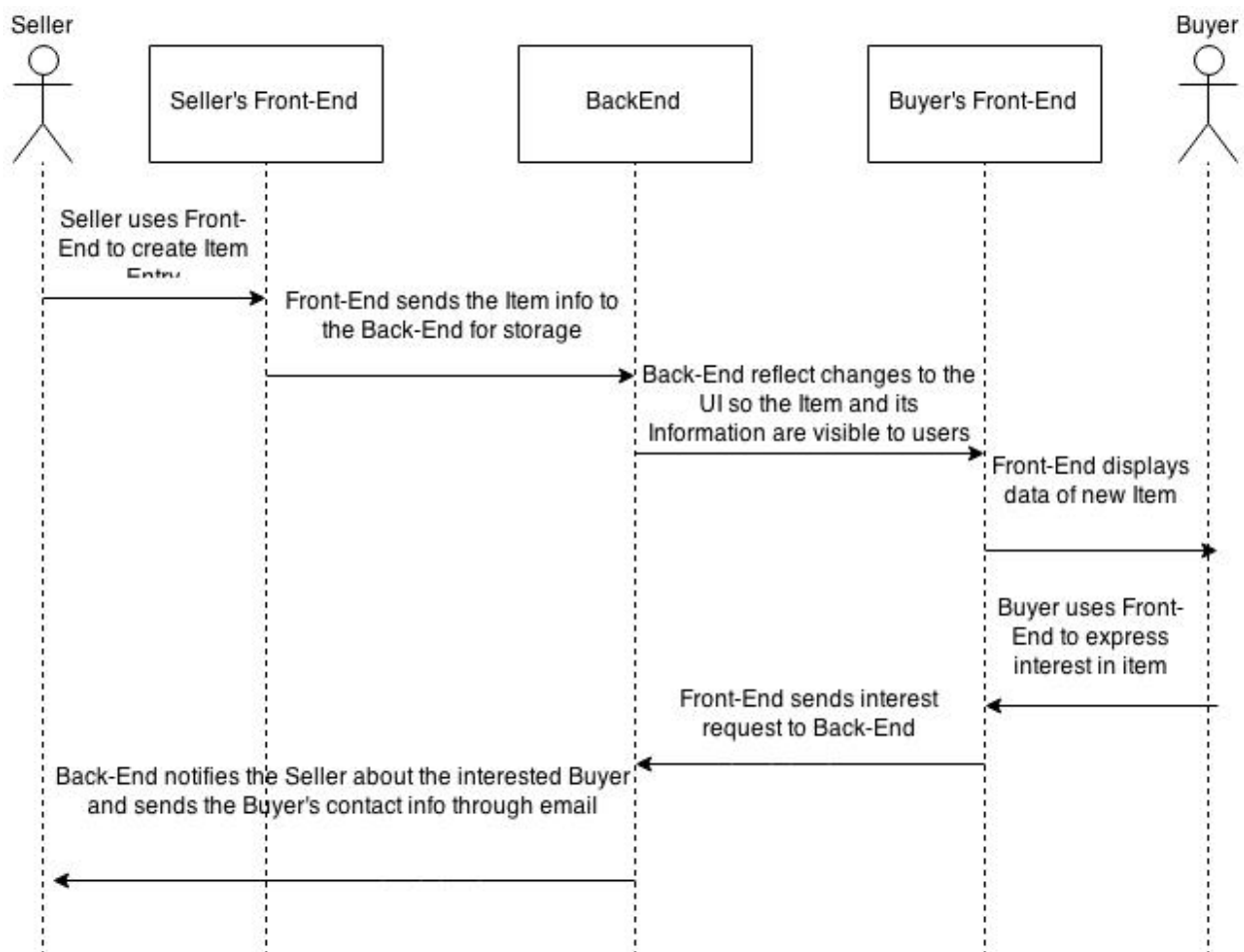
"Partner" systems:

Since this is an independent app it does not have any direct partner systems. The only system that the application interacts with remotely is the Gmail-Network set up specifically for IIT students and faculty.

Data model

There only two categories of data handled by the architecture of the app. There user-centric private data like login credentials and email, and item-centric data like description, tags, pictures and seller. A user's user-centric data is only visible and modified by her. The only exception is when she expresses an interest for an item. Then her contact info is visible to the seller of the item. Item-centric data is visible to everyone since this is the core functionality of the application, to display information of items that people want to buy and sell.

Flow of information:

Seller | Seller's Front-End | BackEnd | Buyer's Front-End | Buyer

Seller uses Front-End to create Item Entry

Front-End sends the Item info to the Back-End for storage

Back-End reflect changes to the UI so the Item and its Information are visible to users

Front-End displays data of new Item

Buyer uses Front-End to express interest in item

Front-End sends interest request to Back-End

Back-End notifies the Seller about the interested Buyer and sends the Buyer's contact info through email

<u>Storage and retrieval</u>:

Persistent storage of user-centric data and credentials will be stored and retrieved from an encrypted database for security. The storage of Item-List will be stored and retrieved from a database too, although not as heavily encrypted. One reason behind that is that the risk of exposing private data from the Item-List data is small since there are not that many (if any at all). The other reason is that since this database will be accessed the most, unneeded security checks would make it slow. But taking them away, we would expect performance to increase in terms of number of transactions per unit time. As far availability and reliability are concerned both databases will have generated snapshots as backups once in a while, but that is a feature to be added in later releases and not in a prototype.