Requirements document for HWS

Date: 4/23//2021

By: Saboor Ahmed

1. **Background/Intro:**

The owners want to make an application that will help potential clients sign up online for Home Watching Services.

1. **Goal:**

The goal of this project is to expand Home Watching Services and gain more clients in nearby towns.

1. **Objectives:**

* Build a System capable of registering multiple users online.
* Build a system that can read sensors.
* This system should be user-friendly (With bluish color and gold on top).
* The system should have a quick and easy monthly billing method,

1. **Scope:**

* This system must allow users to register online.
* This system must allow users to add tickets from anywhere and at anytime.
* This system must allow users to get notifications when task is done.
* This system must allow users to sell products.
* This system must allow users to judge products and sellers by leaving reviews.
* The system must allow users to search for any products they wish for.

1. **Risks:**

Since this is the first major online retail site, there are a limited number of resources in creating this project. Due to the limited resources, this would affect the budget we are using for this project, as well as the timeline. In addition, this would also affect the scope of this project, since the technology to build this system is still relatively new, it may not be able to fulfill the requirements.

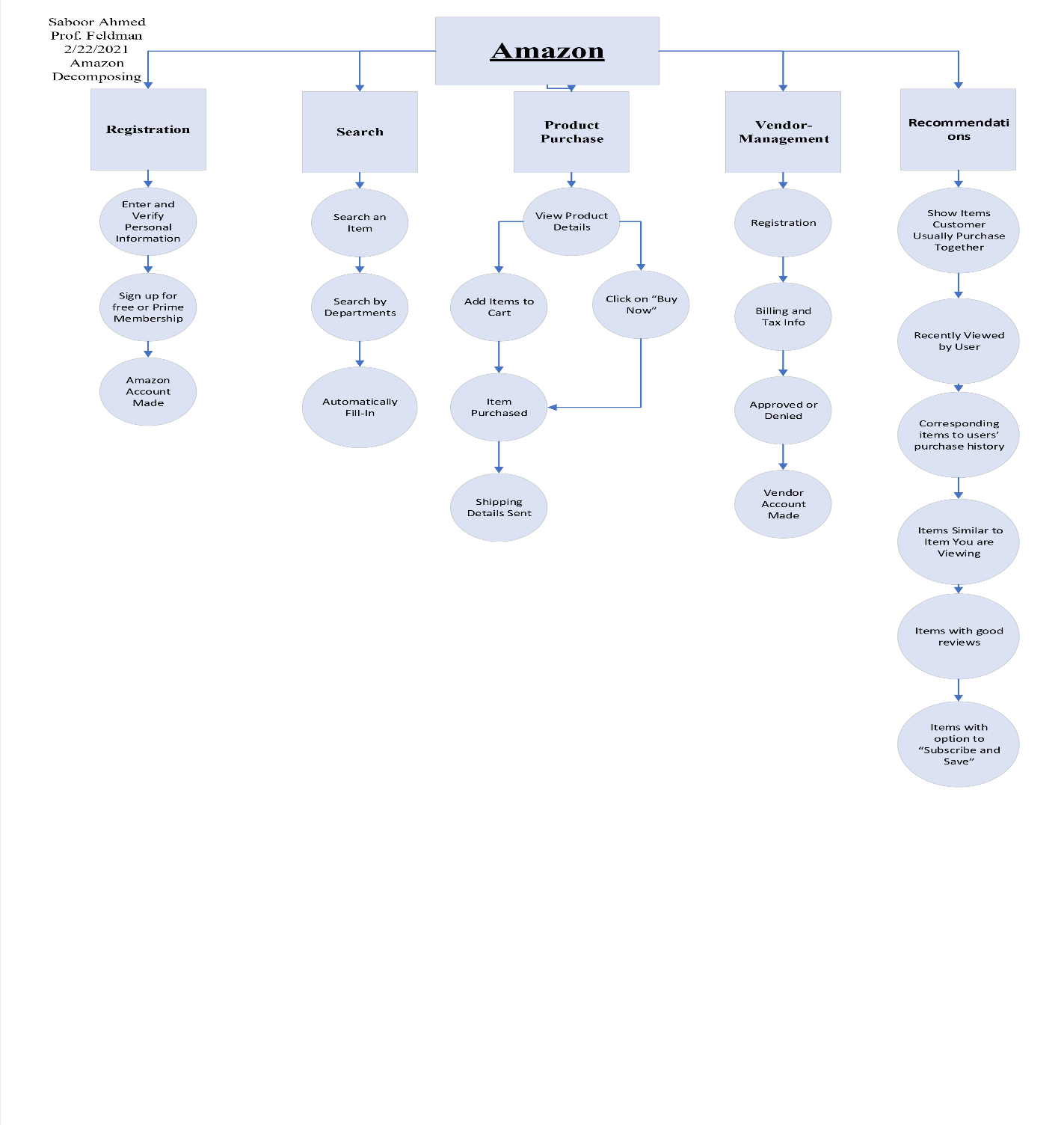
1. **Constraint:**

The budget for this project is fixed at $100,000. This budget is financed as paid-in capital by the owner.

The timeline for this project is 12 months.  This is to prevent potential competitors from entering the market.

1. **Out of Scope:**

* This site will not support showing similar products to what the customer is buying, and products that are usually purchased together.
* This site will not allow users to subscribe to items.
* This system will not be available on multiple platforms
* This site will not support the categorization of the products they are selling.
* This site will not support Prime account or two-day delivery.

**Decomposition Diagram:**

**8.** **Functional requirements**

8.1. **Registration**

8.1.1. Enter and Verify Personal Information

8.1.1.1. System will require the user to enter the full name

8.1.1.2. System will require the user to provide an email

8.1.1.3 System will require the user to create a password

containing at least 6 Characters

8.1.1.4. System will require the user to verify email

8.1.1.5 The system shall create an Amazon Account

8.1.2. Sign up for a Prime Account

8.1.2.1. System shall ask user to input a payment method

8.1.2.2. System shall charge the user an annual in return for prime services

8.2. **Search**

8.2.1. Search an item

8.2.2.1. System should allow the user to view the results of their search

8.2.2.2. System should allow the user to view details of each item in the

search list

8.2.2. Search by department

8.2.2.1. System should allow the user to search by departments

8.2.2.2. System should allow the user to search for an item in the department

8.2.3. Automatically Fill-In

8.2.3.1. System should predict potential responses underneath the search bar while the user is searching.

8.3. **Product Purchase**

8.3.1. View Details of an Item

8.3.1.1 System should allow the user to select size or quantity when viewing details

8.3.1.1.1. System should allow users to click on the “Buy now” option which

would purchase the product immediately.

8.3.1.2. System should allow the user to add the item to the digital cart

8.3.1.3. System should allow the user to continue shopping or proceed to checkout

8.3.2. Proceed to checkout

8.3.2.1. System should allow the user to select a name to be delivered to

8.3.2.2. System should allow the user to input the address to be delivered to.

8.3.2.3. System should allow the user to input a payment a method

8.3.2.4. System should email the product’s purchase confirmation and shipping details to the user.

8.4. **Vendor Management**

8.4.1. Registration

8.4.1.1. System should ask for the business name and product name

8.4.1.2. System should ask the seller about the product details

8.4.1.3. System should allow the seller to agree or disagree with the “AmazonSeller Agreement”

8.4.2. Tax Information

8.4.2.1. System will require the user to upload W2 along with other required documents

8.4.3. Approved or Denied

8.4.3.1. System shall approve the document if all the paperwork meets

specifications

8.4.3.1.2. System shall charge a registration fee to the seller.

8.4.3.1.3. System shall charge an annual fee based on the seller’s annual Revenue.

8.4.3.2. System shall deny the application if the paperwork does not meet

specifications.

8.5. **Recommendation**

8.5.1. Show items customers usually purchase together

8.5.1.1. System shall show the customer items that other users usually purchase with the item that is being viewed.

8.5.2. Items recently viewed

8.5.2.1. System shall show the user items that he/she recently viewed on the website

8.5.3. Show items that correspond to users purchase history

8.5.4. Show user items that have good reviews

8.5.4.1. System shall show user items that have good reviews and are related to the an item that the user is viewing

8.5.5. Show items with “Subscribe and Save” option.

8.5.5.1 System shall convey items to the user which have the “Subscribe and Save” option, and this saves the user from repeatedly buying the same item. Instead, the item will be automatically shipped out when the old item finishes.

**9. Non-Functional Requirements:**

* System should be large enough to store data for millions of users
* System should be large enough to service millions of customers daily
* System should be capable of listing millions of products.
* System should be able to keep data secure for all its users.

**10. Stakeholders**

Stakeholders, Sponsors and Owner

**11. Approvals** <leave this blank for now. We will revisit later>

* **Interpretability**
* “Black Boxes”
* Flawed Assumptions
* **Bias**
* Someone Designs the System
* **Data**
* Data Quality
* Not Enough Data