

Global Distributed Software Development

HSF Marketplace

WiSe2019 Group: G3

Milestone #1

November 14, 2019

Table of Contents

Executive summary:	2
Persona and main user cases	4
Data Management	9
Functional Requirements	17
Competitive analysis:	19
System Architecture and Technologies	21
Team Members and their roles:	23

1. Executive summary

In a market already saturated with a number of the e-commerce sites, it is a challenging task for any new competitor to stand a chance, but our team wanted to face this challenge and hence we put together a vision of how we could make our place amongst the giants and stand out as a unique product.

Our product “HSF-Marketplace” is an ecommerce site targeting the students of Hochschule Fulda. Being a focused product, it is extremely easy for us to align ourselves with the expectations of our targeted audience by closely observing their shopping patterns.

In our product, every user can be both the buyer and the seller hence giving everybody the freedom to not just use our product as a tool for purchasing something, but also the ability to host their own products. This enables us to empower the students to not just sell something they have used and want to sell out for a sum, but also enables them to start their own small-scale businesses which can help them achieve much-required financial stability in their student life.

We are developing our e-commerce platform with ease of use in mind. Our platform will be able to cater to both the layman and the advanced users. With our approach to building the platform with user-centered development, we will be able to deliver the key options in the most intuitive manner.

As the user can be either a buyer or seller, our platform will provide a well-detailed dashboard for tracking their past activities. The sellers will be able to keep track of what they sold, to whom they sold, how well their product was reviewed. They can also use the dashboard to give permanent or time-limited discounts. The buyers will also have a well-detailed dashboard giving them information about their deliveries, their past purchases and also provide recommendations on the basis of their past activities, the buyer will also have a unique option of generating requests, through which they will be able to request for items that are not available on our platform. The request will be open to all the sellers and can be made available by them if they can.

Our platform has two unique features. We are implementing a reward system which gives the users virtual credits the more they shop, after earning a certain amount of credits, the seller will be able to get a discount on the products. Our platform will also have some social aspects, which leads us to the second unique feature, that allows the user to shop together with their friends, the user will have a friend list and whatever they plan to buy can be shared to their friends and the friends can write share their opinions about the product and hence helping to decide whether or not to buy an item.

This platform can become a reality only with the help of the blood, sweat, and tears of a well-coordinated team of software developers and platform experts, fortunately for us, we have the developers with more than 2 years of experience in the industry in all the different fields. Jobayer Mojumdar our frontend lead developer has developed e-commerce sites in his past projects, our database manager Kumkum Rajput has worked in the industry for the same for over 5 years, Dan, our team lead has worked as frontend and back end developer in total for about 1 years and has some

experience in team leading, Usman has experience with creating web sites and especially with HTML, CSS and JavaScript, on the backend Asmita Upreti has worked for over 2 years in iOS development and is currently working as a mobile software developer and Nikhil Yadav who is also working on the backend has experience of PHP for over 2 years and working currently on the same as a software developer.

2. Persona and main use cases

Our application will have three key personas involved.

Buyer:

First persona in our project is the **buyer**. A buyer can be any person, in our case is limited only to SFSU students, who can access the application, have a registered account and who is looking to buy something from the list of items offered on the website. In order to access our platform and buy items, a buyer must hold a registered account on the platform.

Buyer's privileges are:

- accessing our platform using his registered account
- navigate through the web site
- contact sellers
- add items to shop list
- has access only to his own shop list.

Seller:

Our second persona is **Seller**. A seller can be any person, who can access the application, have a registered account and who is looking to post an item on our platform to be sold. A seller, before posting the item on our platform, must log in with his registered account, send an approval request, and after the item is approved by administration, it is displayed on the web, and wait to get contacted by somebody who wants to buy that specific item.

Seller's privileges are:

- Accessing our platform
- Navigate through web site
- Add items on the web to be sold
- Have access to his list of items from the web (sell items list)

Admin:

Our third persona is **Admin** (Administrator). An administrator is a specially trained person, with certain skills in the IT industry, who operates and administers the whole system. His account is special and different from others giving him the ability to decide what items are allowed on the web, have most privileges, access to the whole system and all actions can be monitored by an administrator. Every single item must be approved by Admin before it goes on the web.

Admin's privileges are:

- Approving or denying items from sellers
- Access database of registered users

- Access database of items
- Remove user
- Ban user
- Give fidelity point to active user base on different criteria (new feature)

Unregistered user:

Our last persona is an unregistered user. Unregistered user can be any person who accessed our website, and who is not required to have a registered account. It has no privileges except adding items to his wish list, which can be accessed after an account is registered

Use Cases:

Login: As a user I want to be able to log into the system. When a user accesses our website, he is directed to a login page where registered log in name and password for that account must be provided in order to move on to the next step.

1. **Home page:** As a user I want to be able to see all available items. After the user has logged in into account, our system redirects user to website's main page, where all available to buy items, are displayed in a list.
2. **Search Product:** All users of the application should be able to browse through all the available products and search for a specific product by name or category. There are multiple categories for a product and searching for a product is much of categorizing as per the requirement of the buyer.
3. **Buy Product:** As a user I want to be able to buy any available items. If the user decides to buy a desired item from our platform, he can select the item, and proceed with buy procedure.
4. **Contact:** As a user I want to be able to contact the seller of the item. Every single item, displayed on our website, contains details about the owner of the item, so the user has option to contact user via mobile phone, or contact directly on website using our messenger feature.
5. **Selling Product:** As a user I want to be able to register and item to be sold. If a user has some items for selling, they can be registered on our service provided by our platform.
6. **Profile:** As a user I want to be able to see my registered items. Whenever a user registers an account on our platform, all his personal data are saved in his profile section, which can be

accessed later, including personal list of posted items, which can be accessed and displayed using profile section.

7. **Admin Dashboard:** As an admin I want to be able to see all users. One admin has successfully logged in, access is provided to all data in our system, including to all registered users which can be displayed at wish.
8. **Post approve/reject:** As an admin I want to be able to approve or reject item to be posted on the site. One of the main admin's responsibilities is to approve or deny items requested items by sellers to be posted on the website. This be done via special section, where only requested items for approval will be displayed.
9. **Rewards:** As an admin I want to give fidelity/rank point to users based on their activity and some more criteria. As a reward for good behavior, or for long use of our platform and services and more criteria, Admin can reward users with fidelity/rank points, which users can benefit from.

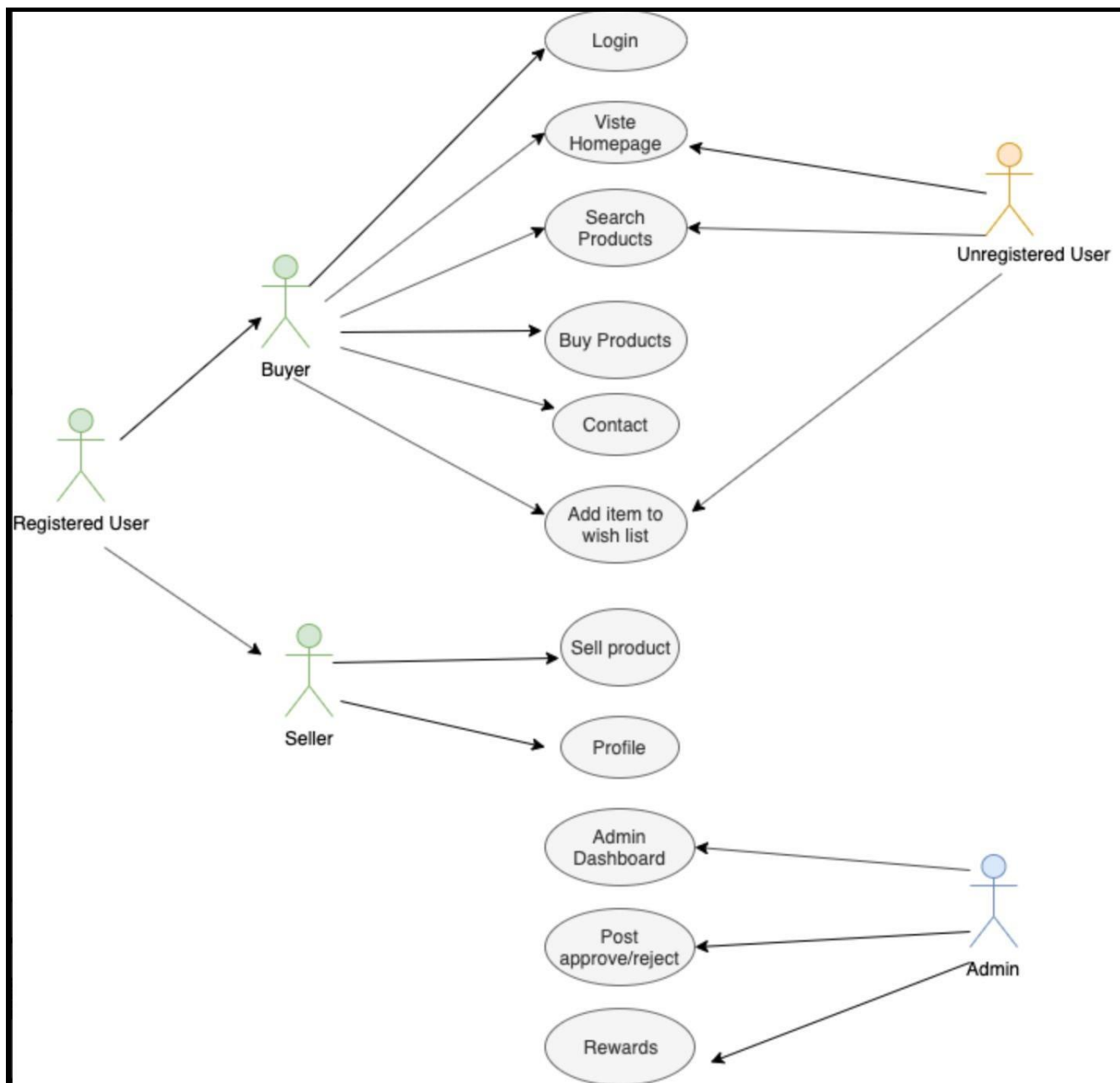


Fig: Use Case Diagram

3. Data Management

Data Description

User: User is a generic term used for all types of users accessing the application. It is categorized with multiple access roles and interfaces.

Sellers: A registered user who sells his product.

Buyers: A registered user who buys the product.

Guest Users: An unregistered user who navigate the website and can make wishlist of his favorites.

Product: An item to be sold. A product can be categorised in multiple categories. User is able to search for products on the basis of multiple categories. These categories are added by an admin. All the categorisation is managed by admin. Admin has access roles to delete or add a category. Seller is supposed to add products for selling on the basis of categories only. Admin has access rights to map products to particular categories on the basis of categories while the product is under screening.

Admin: Administrator is someone who manages all the roles for different users. Admin is something who gets all the requests to approve for products and all types of users. Admin manages the whole content of the application.

The database consists of following tables-

- **Table_SuperUser:** Admin information is added to the database with a unique ID. Along with other necessary columns, this table has Role column which identifies a particular admin user.

SUserID
SFirstName,
SLastName,
Password,
Email,
Role,
Privilege

- **Table_Users:** User information is added to the database with unique ID. This has columns as-

UserID,
FirstName,
LastName,
Password,
Email,

City,
State,
Zip,
VerifiedEmail,
VerificationCode,
UserIP,
UserPhone,
Country, Address

- Table_Product: Complete product information is stored in this table. ProductID,

ProductSKU, Product Name,
ProductCost,
ProductCategory,
ProductCategoryID,
Image, Thumbnail,
Description,
ProductStock

- Table_Orders: Customer ordered products, status and delivery information is stored in this table.

OrderID,
Amount,
OrderShipAddress,
OrderCity,
OrderState,
OrderZip,
OrderCountry,
OrderEmail,
OrderShippeddate,
OrderTrackingNo

Apart from this, according to our “unique” (whatever we may decide) idea, the group can decide what else we can have. E.g.-

Table_BuyingHistory- which can have the following information

Products/category shopped,
Price range,
Average monthly shopping,

Order intervals etc.

Table_AbandonedProducts - Shoppers don't add products to the cart by accident. There are many reasons why some products are left for later and sometimes never purchased. Some prominent ones are:

Product exceeds budget, and shoppers are waiting for prices to fall. Shopper wants to look at other similar options before purchasing. The need of the product is not urgent.

A quick glance at an abandoned cart with demographics and buying behavior in mind will be enough to figure out what ought to be done to trigger a sale. Some options could be:

Offering a discount on a particular product
Notifying shopper about awaiting product

Table_Wishlist

Can be a goldmine of insights for us. Studying wishlist will tell us about shopper aspirations and potential purchases. Shoppers can be notified whenever something in their wishlist does following-:

Decline in price
Special offers
Product upgrades

While hosting special sales, wishlist data can be highly valuable. So, we can make the effort to analyze wishlist data and plan surprises for registered users.

Table_PageInsights

Some website pages are more critical than others from sales perspective, and it's important to know how they are performing individually. Page performance

Traffic sources

Improvement areas Bounce rate

By gathering diverse set of page data, we might be able to make amazing changes to improve our visitor experience.

Table_UserProfiling

Below data figures influences product expansion as well as marketing approach: Gender Age

Marital status

Income group

This helps in generating reports based on user data.

Table_SiteSearch

Search is a popular channel of product discovery on ecommerce platforms. Search queries made by registered users can tell: What they are looking for?

What type of keywords they use?

How many times search results in purchases?

Valuable data can also be collected from searches ran by unregistered users. Popular online stores use the site data to pitch similar products to the shoppers using a dedicated section.

4. Functional Requirements

- 1.** Users will register for the website
- 2.** Users will be able to login
- 3.** Registered and non-registered user will be able to see the home page with product list
- 4.** Buyer will be able to search products
- 5.** Buyer will be able to put an item in a wish list for buying
- 6.** Buyer will be able to view the seller's details for contact
- 7.** Buyer will be able to contact the seller
- 8.** Sellers will be able to post an Ad for the product sale
- 9.** User will be able to see a profile page with his information.
- 10.** Admin will be able to see the all registered user list and maintain users.
- 11.** Admin will be able to modify the Ad before it has been approved by the admin
- 12.** Admin will be able to rewards user.

5. List of non-functional requirements:

- 1.** Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO).
- 2.** Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers.
- 3.** Selected application functions must render well on mobile devices.
- 4.** Data shall be stored in the team's chosen database technology on the team's deployment server.
- 5.** No more than 50 concurrent users shall be accessing the application at any time
- 6.** Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
- 7.** The language used shall be English.
- 8.** Application shall be very easy to use and intuitive.
- 9.** No email clients shall be allowed.
- 10.** Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
- 11.** Site security: basic best practices shall be applied (as covered in the class)
- 12.** Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
- 13.** The website shall prominently display the following exact text on all pages "SFSU / Fulda Software Engineering Project, Fall 2019. For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).

6. Competitive analysis:

Along with the increasing trend of online shopping, the web application which provides customers with this facility is increasing too. E-commerce business is booming with the increasing awareness of customer regarding online shopping. Thus, thousands of such e commerce application can be found on the internet. Our project “HSF-Marketplace” is also a buying and selling application and to make it withstand the competition on the web, we are analyzing our competitors .

To do so we found three successful applications that are already in the web which is very successful there are amazon, Alibaba and MediaMarkt.

Features	Amazon	Media Markt	Alibaba	HSF
Product Suggestion	✓	✓	✓	✓
Search History	✓	✓	✓	✓
Product Review	✓	✓	✓	✓
Product wishlist without registration	x	x	x	✓
Shop Together	x	x	x	✓
Shopping Reward	x	x	x	✓
Abandoned product discount	x	x	x	✓

The above table shows the list of planned features which makes our application standout among the competitors.

- Product Suggestion: On the basis of pervious purchases, the buyers will get product suggestion.
- Search History: Buyers would be able to see their past purchases on the application.
- Product Review: Buyers will be able to write short reviews about the product they purchased.
- Product wish list without registration: Buyers would be able to create a wish list of items without registering in the system.
- Shop Together: Buyers can share the product which he/she wants to buy with their friends on the same platform.

- Shopping Reward: Buyer gets rewarded on each purchase which can be used to get a discount after achieving a certain amount of reward.
- Abandoned product discount: Buyer gets discount notification for such product which has been forgotten in the cart for a long time when it gets discounted.

7. System Architecture and Technologies

Server	AWS
Memory (Ram)	8GB Ram
Operating system	Linux-Amazon Version 2.9
Database	MySQL Version 2.9
Webserver	AWS Version No public info by the vendor
Server Language	PHP Version 7 - X64 bit

8. Team Members and their roles:

Team Member	Role Assigned
Dan Jalba	Team Lead and Documentation manager
Asmita Upreti	Back End
Kumkum Rajput	Database Engineer
Nikhil Yadav	Back End Lead
M. Usman Aslam	Front End
Jobayer Mojumdar	Git Manager and Front End lead

9. Checklist:

Team found a time slot to meet outside of the class.	DONE
GitHub master chosen.	DONE
Team decided and agreed together on using the listed SW tools and deployment server.	DONE
Team ready and able to use the chosen back and front-end frameworks and those who need to learn are working on learning and practicing	DONE
Team lead ensured that all team members read the final M1 and agree/understand it before submission	DONE
GitHub organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	DONE