

Milestone 1

2020 Spring CmpE352 Group 9

April 2020

Contents

1 Executive Summary	3
1.1 Project Description	3
1.2 Project Status	3
1.3 Future Goals	3
2 List and Status of Deliverables	4
3 Evaluation of the Status of Deliverables	5
4 Tools and Processes	6
5 Member Reports	8
6 Project	16
6.1 Communication Plan	16
6.2 Requirements	16
6.3 Scenarios and Mockups	23
6.3.1 Scenario 1: Vendor Selling an Item	23
6.3.2 Scenario 2: Customer Buying an Item	34
6.3.3 Scenario 3: Customer Cancelling an Item	43
6.4 UML Software Designs	51
6.4.1 Use Case Diagram	51
6.4.2 Class Diagram	52
6.4.3 Sequence Diagram	52
6.5 Project Plan & RAM	59

1 Executive Summary

1.1 Project Description

The aim of our project “Bazaar” is to create an e-commerce platform where customers and vendors meet and conduct an online shopping experience. The platform allows three types of users: Guest Users, Customers, and Vendors. While Guest Users can explore products and vendors; Customers can also rate the products/vendors and comment on them, interact with vendors, and create lists. After the buying and selling is carried out between customers and vendors, both parties can keep track of their orders. In addition to those three types of users, there are so called admins who can administrate the platform and communicate with the vendors to handle problems when one is raised. The functionalities are not limited to those mentioned and all are elaborated as in **Requirements**.

1.2 Project Status

From the start of this class, we have kept improving ourselves about software engineering and completing tasks one by one. In the very first weeks we tried to find a meeting day that is appropriate for everyone. After we set the meeting day, we have created our communication plan. Then we started to talk over assignments and share the workload equally. All of us made his/her own research about Git and GitHub. We quickly created our wiki page and personal pages, shared repositories we like. We decided on our project name and logo.

Requirements Specification has been the most time-consuming process so far. After we came up with the draft of ‘Requirements’, it had been upgraded according to the customer meeting and feedbacks. A further Requirements Specification task was creating the scenarios and their mockups. We worked as groups for the following scenarios: vendor selling an item, customer buying an item, and customer cancelling an item. The next and our first Design assignment was preparing the UML Diagrams and we splitted up into three groups preparing one diagram each: Use Case Diagram, Class Diagram, and Sequence Diagram. However we had to give a 2 weeks break due to the COVID epidemic. We changed our meeting system and schedule. After that we created the diagrams.

Knowing that they will be used as a guide for future tasks, we put great effort to deliver these documents complete and correct, and we have done that without missing any deadline. Nevertheless, we made mistakes but tried our best to revise the documents in consideration of the feedbacks; in fact, we start every meeting with feedbacks and assessment of the previous week to accomplish this.

1.3 Future Goals

We overcame every difficulty so far. We have learned a lot about software engineering. Although we are under circumstances that we never lived on, we will

keep talking about our project. From now on we will talk more about project implementation and design. The next thing we will do is deciding implementation languages and implement an API for the sense of practice. We aim to implement frontend, backend and android parts of our project. On the road we can revise our project plan. We will be proceeding more carefully since we have learned from our mistakes. At the end, we believe we will complete "Bazaar" successfully and we will be proud of this outcome.

2 List and Status of Deliverables

Deliverable	Status	Description
Github Wiki Page	Done	Literally all the information about the project is accessible there.
Github Readme	Done	There is a list of all group members with links to their personal pages, and a photo showing 8 of them.
Github Issues	In progress	Here we have issues with labels and deadlines.
Meeting Notes	Done	There are information about our regular meetings like time, place, duration, agenda, attendees etc.
Communication Plan	Done	There is a table showing all different communication options of the team as well as when to use which.
Requirements	Done	Here we have a glossary, functional and non-functional requirements. These are statements about the expected capabilities of our system.
User Scenarios & Mockups	Done	We have 3 different scenarios each having a persona, preconditions, a story and the flow of scenario depicted by representative photos.
UML Diagrams	Done	We have class, use case and sequence diagrams that provide us a clear picture and understanding of the main framework. They will guide us implementing the project.
Project Plan and RAM	Done	Our project plan was drawn using ProjectLibre software. Current and historical RAMs are available too where one can clearly see which task was given whom.

3 Evaluation of the Status of Deliverables

1. Github Wiki:

From the first day of the creation of the wiki page, we put everything in order. The home page of the wiki is well designed. You can find anything you want easily if it exists thanks to the design. The page is enriched by proper icons for the titles. We constantly update the wiki and add new pages. All group members contribute to the wiki from time to time. Wiki include personal wiki pages of group members, meeting notes, some useful documents, communication plan, and project documents. To conclude, our wiki page is a showcase of our status at the project.

2. Github Readme:

Up to now, the Readme consists of 3 parts. In the first part, there is a definition of our project. The second part is our group photo. There are 8 of us in the photo however we are 10. The third part is the member list. Names of the list route to their corresponding wiki page.

3. UML Diagrams

Working with diagrams was hard but it's at the main core of our project. This was the first step of understanding what our project would look like. We worked on hard to think what would be our class diagrams while ensuring our requirements. We tried to create classes and functions and tried to design a system in which our possible scenarios could implemented with. Our members know that we will need to add and drop some functionalities in our design diagrams but we are trying to make sure our design diagrams provide a smooth baseline for further progress. We used our feedbacks to improve our diagrams and did necessary changes. Our vision about the implementation part of project's functionalities greatly improved with these diagrams. So we attach importance to our diagrams for further progress.

4. Meeting Notes:

In the meeting notes we aimed successful information flow between each member of the group. Even if some member was not in the weekly meeting for some reason, we needed to be sure that member could catch up our discussions via meeting notes. In addition to that, we often take a look on meeting notes for finding out who was responsible for any work or what had been done after the last meeting. Meeting notes are essential for communication between team members and our goal is improving these notes further such that any member could catch up with our discussions any time.

5. Communication Plan:

We are using Slack and Discord in the parts where we are divided into sub-groups. For instance, when we were working with the class diagrams,

we had a slack and discord channel for that. Also, we use doodle to open a poll and piazza to communicate with customer and GitHub integrated into Slack to get a commit, issue updates, etc. in real-time. Currently, sharing docs over Drive, using WhatsApp as a communication channel and GitHub issues remain enough for our in-group communication.

6. Requirements:

We update the Requirements and its wiki page according to the feedbacks in several times, both in terms of visuals and its content. As the ideas pop up in the meetings, and there are some points that need clarification, we go ahead and update the related requirement. Overall, we think we are in good shape in terms of the requirements page and good at updating it according to both instructors' and team's feedback. Also, we often go back to requirements and refer to them in different parts of the project, such as Mockups and Design Diagrams. Hence, it is really important for us to keep this part up to date and well-explained.

4 Tools and Processes

The main way we deliver our project is via **our wiki page on GitHub**. Everything from Requirements to Project Plan is written in it.

- Wiki Page:

GitHub allows us to easily edit a wiki page into our repository. This wiki can include as many pages as needed, each supporting markdown, pictures and hyperlinks that can be pointed as needed. Overall GitHub is a pretty good website to use so our Requirements and Scenarios are implemented into the wiki page, as requested.

- Mock-ups:

We used **moqups.com** as it enables for multiple users to work on the same scheme. This eased our work quite a bit. However, unless the user has premium account, there is a limit for objects to be put in one scheme. This caused some problems later as we couldn't simply copy and paste, or even download the design(s). It also has minor bugs but as said, these bugs are minor and bearable.

- UML Diagrams:

- Class Diagram shows the general design of our project, including the interactions between classes. **moqups.com** was our choice to implement it. The website worked for us quite good, since it is well designed to create such graphs.
- Use Case Diagram shows how a user can interact with the project. This time, we used **Visual Paradigm** instead of moqups.com since Visual Paradigm can provide better creation of include exclude relation between functions.

- Sequence Diagram shows the interaction between classes. Once again, **moqups.com** was once again the choice as we were familiar with it at this point. Since it's main purpose is to create such diagrams, unless the 200 object limit becomes a problem, we plan to use **moqups.com** for further such diagrams.
- Project Plan:
We were requested to use **ProjectLibre** for our Project Plan. Learning how to use ProjectLibre has taken more time than the Project Plan itself. Even if the application is free to use, it is inefficient and we plan not to use it as much as possible.
- Milestones:
Including this and the second Milestone, **Overleaf** was our choice to create a LaTeX project. Just like **moqups.com**, Overleaf allows multiple users to work on one project which helps us work efficiently. It uses LaTeX, which can be hard to get used to, but the outcome is professional, as we need.

5 Member Reports

Hasan Demirkiran	<ul style="list-style-type: none">• Creating personal wiki page• Researching useful GitHub repositories• Researching Git• Editing and revising requirements• Participating in possible scenario discussions• Participating in name discussion• Preparing the Persona, User scenario, Preconditions and goals of the User Scenario 1• Attending weekly meetings• Taking notes for some weekly meetings• Editing the User scenario 1• Creating necessary issues in the group's GitHub repository• Contribution to Customer Meeting Questions• Take notes in customer review and transfer to the github• Reviewing customer review questions• Creation of some sequence diagrams• Creating Use Case Diagram specifically for Admin and Vendor• Researching APIs• Editing the Use Case Diagram after feedback• Contribution to creation of milestone report
------------------	---

Muhsin Etki	•
-------------	---

Şadi Uysal	<ul style="list-style-type: none"> • Creating wiki page,Adding new pages,Editing wiki page • Creating personal wiki page • Exploring useful GitHub repositories • Research Git • Sharing some ideas about the possible scenario ideas • Attending weekly meetings and documenting meeting (3,4,6,7) notes • Editing Github repository descriptions according to feedbacks • Discuss our scenario and creating it.(Scenario 2) • Mockup review • Creating issues according to meeting notes • Preparing customer meeting questions • Examine customer meeting answers • Editing and revising requirements according to customer meeting documentation • Review final state of requirements and scenarios • Examine our class diagram • Creating “login via username” sequence diagram • Writing the project plan and creating Gantt Chart with group • Updating sequence diagrams(Login via username,Add to List,Filter, Ban user) • Research API • Milestone report - Contribution to List and Status of Deliverables
------------	---

Ömer Cihan Benzer	<ul style="list-style-type: none"> • Create Personal Wiki Page • Schedule and Attend regular meetings • Name and Logo Discussion • Research Git and GitHub Repositories • Research AWS and API frameworks • Review Requirements, Sequence Diagrams and Project Plan • Create Slack and Discord servers to communicate • Update GitHub Repository Research Wiki Page • Update Requirements: Non-Functional • Design/Update scenario_3 and its mock-ups • Design UML Class Diagram: User based classes • Update UML Class Diagram • Create Gantt Chart Template • Regularly Update RAM Diagram • Milestone Report: Member Reports, Tools and Processes • Overall small improvements, reorganization, typos etc..
----------------------	--

Ahmet Bilal Uçan	<ul style="list-style-type: none"> • Creating the wiki page • Creating personal Wiki page • Attending weekly meetings • Participated in name and logo choice • Researching Git and GitHub repositories • Preparing questions for customer meeting • Attending customer meeting • Proof-reading customer meeting report • Creating and commenting on some issues • Revising non-functional requirements • Contributing to the Scenario 3 • Milestone Report - Deliverable Table
---------------------	--

Beste Göger	<ul style="list-style-type: none"> • Creating personal Wiki page • Attending weekly meetings • Documenting meeting notes • Creating issues • Name and logo discussion • Creating first version of Functional Requirements • Preparing Scenario 2 and initializing mockup • Reviewing scenario and mockup • Contribution to Customer Meeting Questions • Review of Requirements • Creating Use Case Diagram specifically for Guest User and Customer, and Review • Milestone report - contribution to Executive Summary and reviewing the report
-------------	---

	<ul style="list-style-type: none"> ● Editing wiki page ● Creating personal wiki page ● Researching useful GitHub repositories ● Researching Git ● Preparing, editing and revising requirements ● Creationg the glossary for requirements ● Participating in possible scenario discussions ● Participating in name discussion ● Preparing the logo of "Bazaar" ● Attending weekly meetings ● Creation of Scenario 1 discussion ● Preparing the mockup for Scenario 1 ● Editing the mockup according to the feedbacks ● Creating necessary issues in the group's GitHub repository ● Reviewing customer review questions ● Creation of "search" and "filter" sequence diagrams ● Review of 6 sequence diagrams ● Researching APIs ● Contribution to creation of milestone report
Alcan Unsal	

Halil Ibrahim Orhan	<ul style="list-style-type: none"> • Creating wiki page, Adding new pages, Editing wiki page • Creating personal wiki page • Research Git • Attending weekly meetings • Creating Non-Functional Requirements • Setting up customer meeting • Preparing customer meeting questions and Attending customer meeting • Editing and revising requirements according to customer meeting documentation • Review final state of requirements • Creating User Scenario • Name and logo discussion • Creating class diagram • Writing the project plan and creating Gantt Chart with group • Milestone report - contribution to Executive Summary • Research API
---------------------------	--

Mehmet Berk Kemaloglu	<ul style="list-style-type: none"> ● Creating personal Wiki page, Adding new pages, Editing wiki page ● Attending weekly meetings ● Documenting meeting notes ● Creating issues ● Creating readme ● Reviewing scenario and mockup ● Review of Requirements ● Review of Diagrams ● Preparing customer meeting questions and Attending customer meeting ● Name and logo discussion ● Writing the project plan and creating Gantt Chart with group ● Editing and revising mockup according to the feedback ● Editing and revising the project plan Gantt Chart according to the feedback ● Milestone report - contribution to Deliverables
-----------------------------	---

	<ul style="list-style-type: none"> • Creating personal wiki page • Customizing issue labels • Attending weekly meetings • Creating first issues • Exploring GitHub repositories • Research Git • Functional requirements • Completing Scenario 2 • Creating most of Mockup 2 • Review of functional requirements according to feedback • Preparing customer meeting questions • Analyze customer meeting • Review requirements, scenarios and mockups • Creating class diagram • Help with sequence diagrams • Update class diagram • Creating project plan • Revision of project plan • Research API • Milestone report - summary of individual work
Firat Bulut	

6 Project

6.1 Communication Plan

PURPOSE	WHO	WHEN	WHERE	CHANNEL
Weekly Meeting & Assignments	All Team Members	Thursday / 13:00	BM- Student Lounge	Face-to-Face
Weekly Meeting & Assignments	All Team Members	During Pandemic	Discord	Online
Tracking Issues and Tasks	All Team Members	Any Time	GitHub	Online
Progress Tracking	All Team Members	Any Time	Slack	Online
Urgent Communication	Some Team Members	Whenever Needed	WhatsApp	Online
Polls	All Team Members	Whenever Needed	Doodle, Slack	Online
Project Related Issues	All Team Members	During the Project	Designated Place	Online or Face-to-Face
Customer Meetings	Some Team Members	Whenever Needed	Designated Place	Face-to-Face
Communication with Customer	All Team Members	Whenever Needed	Piazza	Online

6.2 Requirements

GLOSSARY

Admin: A type of user who has special privileges to maintain the order of the e-commerce platform.

Customer: A person/profile who buys goods using the e-commerce platform. A customer must be logged-in and has specific privileges.

Dockerize: Contain all software in a folder such that everything the program uses is contained in that folder. This eases portability.

Guest: A type of user who is not logged-in and has the following privileges: searching products and vendors, filtering through products, reading comments.

List: A list of items on the platform that a customer is interested in but is not ready to purchase them yet.

Order Processing: A sequence of processes starting from the customer creating an order request and ending at customer receiving the order.

Shopping Cart: A list of items on the platform that contains the goods selected by a customer or a guest for purchase until the transaction is completed.

Transaction: A financial interchange of money for goods between a vendor and a customer, completion of payment for goods.

User: A person or profile who uses or operates on the e-commerce platform. A user may or may not be logged-in. It can be of type customer, vendor, guest or admin.

Vendor: A person/profile who sells goods using the e-commerce platform. A seller must be logged-in and has some specific privileges.

1. FUNCTIONAL REQUIREMENTS

1.1. User Requirements

1.1.1. Sign Up

1.1.1.1. Guests shall be able to sign up by providing their email address, name, surname and by also providing a valid username-password pair.

1.1.1.2. Guests shall be able to sign up either as a vendor or a customer.

1.1.1.3. Guests who want to register as a vendor are expected to specify the location of their store(s) through Google Maps as well as the specifications specified at 1.1.1.1..

1.1.1.4. Guests should be able to sign up with their Google account.

1.1.1.5. Users shall be able to delete their accounts.

1.1.1.6. In order to fulfill the registration process, users shall verify their email addresses and agreeing the Bazaar contract, a contract where users approve to the legal terms of the website.

1.1.2. Sign In

1.1.2.1. Customers and vendors shall be able to sign in with their email address and password.

1.1.2.2. Customers and vendors should be able to sign in with their Google account if they have signed up via their Google account.

1.1.3. Social Interactions and Communication

1.1.3.1. Customers shall be able to communicate with vendors via a direct messaging system. Vendors shall be able to receive the messages and answer the customers.

1.1.3.2. Customers shall be able to start a conversation with the vendor of a certain product from that product's page, whether it is

before customer buys the product or after the product has been ordered and delivered.

1.1.3.3. Customers and vendors shall be able to see and manage their direct messages on the ‘Messages’ page.

1.1.4. Products

1.1.4.1. Guests, admins, vendors and customers shall be able to display a product’s page including the following information: image, label, brand, price, vendor info, stock status, comments, rating.

1.1.4.2. Customers shall be able to comment on the products they have ordered.

1.1.4.3. Customers shall be able to rate the products they have ordered.

1.1.5. Categories

1.1.5.1. Guests, admins, vendors and customers shall be able to discover the categories.

1.1.5.2. Guests, admins, vendors and customers shall be able to search for products in the given category and filter/sort a category.

1.1.6. Search

1.1.6.1. Guests, admins, vendors and customers shall be able to search for products by their name, label or brand.

1.1.6.2. Guests, admins, vendors and customers shall be able to filter searched products based on several properties: average customer review, brand, vendor, price range.

1.1.6.3. Guests, admins, vendors and customers shall be able to sort searched products based on several properties: bestsellers, newest arrivals, price, average customer review, number of comments, ratings.

1.1.6.4. Guests, admins, vendors and customers shall be able to search for vendors by vendors’ name and location.

1.1.7. Lists

1.1.7.1. Customers shall be able to create private lists.

1.1.7.2. Customers shall be able to name, edit and delete a list.

1.1.7.3. Customers shall be able to add a product to a list.

1.1.7.4. Customers shall be able to search for products in a list and filter/sort the lists.

1.1.8. Cart

1.1.8.1. Customers shall be able to add products to their cart.

1.1.8.2. Customers shall be able to remove the products from their cart.

1.1.8.3. Customers shall be able to change the amount of products

in their cart.

1.1.9. Buying

1.1.9.1. Customers shall be able to buy the products in their cart by proceeding to the ‘Payment’ page.

1.1.9.2. Customers shall provide valid payment information and shipping address in order to complete the payment.

1.1.9.3. Customers should be able to save their payment and shipping information for future use. These information are private, can be seen only by the customer.

1.1.9.4. Customers should be able to select their address as the shipping address.

1.1.10. Vendor Profile

1.1.10.1. Vendors shall have a profile page that provides all sufficient information about the vendor: name, address(es), products, rating, comments made to their products.

1.1.10.2. Guests, admins, vendors and customers shall be able to see all the products of the vendor in his profile.

1.1.10.3. Guests, admins, vendors and customers shall be able to reach a certain product page via the vendor’s profile.

1.1.11. Selling

1.1.11.1. Vendors shall be able to put their products on their profile by providing all the information which is given to the customers and which is displayed on the product’s page.

1.1.11.2. Vendors shall be able to receive an order with all sufficient information: customer name, address and product IDs.

1.1.12. Orders (Customers)

1.1.12.1. Customers shall have an ‘Orders’ page. On this page, customers shall be able to follow their orders.

1.1.12.2. Customers shall be able to see their active orders with the following information: order id; products with their image, their label, their price, and the vendors; total price; order time; estimated delivery time.

1.1.12.3. Customers shall be able to cancel their active orders.

1.1.12.4. Customers shall be able to see their delivered orders with such details:

* Order ID, date and current status.

* Product ID, name, label, price, picture, vendor etc.

1.1.12.5. Customers shall be able to return their delivered orders when an approved reason is specified.

1.1.13. Orders (Vendors)

1.1.13.1. Vendors shall have an ‘Orders’ page where vendors shall be able to see their active and delivered orders with such details:

- * Order ID, date and current status.
- * Product ID, name, label, price, picture, vendor etc.

1.1.13.2. Vendors should be able to modify the order status as ‘in preparation’, ‘cancelled’, ‘shipped’ and ‘delivered’.

1.1.13.3. Vendors should be able to cancel an order during the order processing stage in case of a problem.

1.1.13.4. Vendors shall be able to communicate with the platform admins about a certain order.

1.1.14. Notification

1.1.14.1. Customers shall be able to choose to get notifications for price changes of a certain product.

1.1.14.2. Customers shall be able to set alerts for a certain price of a certain product. Customers shall be notified if the price of that product goes below the chosen amount.

1.1.15. Admin Panel

1.1.15.1. Admins shall have an admin panel to administrate the platform where he can:

- * ban users.
- * delete comments.
- * communicate with other users if needed.
- * see ongoing transactions(Orders)

1.1.15.2. Admins shall be able to communicate with vendors about orders.

1.2 System Requirements

1.2.1. Searching/ Filtering/ Sorting

1.2.1.1. The system shall let guests, admins and customers search for products by the properties: label, brand, price.

1.2.1.2. The system shall let guests, admins, vendors and customers search for vendors the properties: name, location.

1.2.1.3. The system shall support semantic search, which is used for finding semantically similar products and vendors based on the context information.

1.2.1.4. The system shall let guests, admins, vendors and customers filter the searched products or the products of a certain category. The filtering shall be based on several properties:

- * Average customer review.

- * Brand.
- * Vendor.
- * Price range.

1.2.1.5. The system shall let guests, admins, vendors and customers sort the searched products or the products of a certain category. The sorting shall be based on several properties: bestsellers, newest arrivals, price, average customer reviews, number of comments, rating.

1.2.2. Transaction

1.2.2.1. The system shall support transactions from customers' banking account to the sellers' account.

1.2.2.2. The system shall provide a secure medium for the transactions.

1.2.3. Recommendation

1.2.3.1. The system shall be able to give recommendations to customers based on their search and order history.

1.2.4. Notification

1.2.4.1. The system shall be able to notify the customers for the price changes of products that they wish to get notified.

1.2.4.2. The system shall be able to notify the customers if a product they ordered is out of stock or the order is cancelled for any other reason.

1.2.4.3. The system shall be able to notify the vendors if an active order to buy their product is cancelled by the customer.

1.2.4.4. The system shall be able to notify the vendors if admins report a problem about selling.

1.2.5. Interactions

1.2.5.1. The system shall let customers and vendors communicate via direct messaging mechanism.

1.2.5.2. The system shall provide a messaging mechanism for vendors to communicate with admins about an order or a product.

2. NON-FUNCTIONAL REQUIREMENTS

2.1. Availability

2.1.1. The platform shall work as a web site via modern web browsers.

2.1.2. The platform should be available for use as an Android application.

2.1.3. The platform should be in English.

2.1.4. The platform should be deployed on Amazon EC2 server

2.2. Reliability

- 2.2.1.** Unless intentionally shut down, the system shall be able to run 24/7.
- 2.2.2.** The platform shall use a dedicated server, be portable and dock-erized.
- 2.2.3.** The server shall be able to scale if necessary.
- 2.2.4.** The server shall additionally backup user data every 24 hours.

2.3. Security

- 2.3.1.** All website interactions shall use SSL.
- 2.3.2.** All money transactions shall use 3-D Secure.
- 2.3.3.** All user data, especially username-password shall be encrypted.
- 2.3.4.** Admins shall be able to keep track of transaction in case of a problem.

2.4. Response

- 2.4.1.** The server should respond to every request under 5 seconds.
- 2.4.2.** The server should be able to handle up to 100 requests every minute.

2.5. Privacy

- 2.5.1.** The platform should follow the rules defined by GDPR and KVKK in terms of collecting and storing any personal information, contact information, copyrighted contents and everything related to these paradigms.
- 2.5.2.** The platform should follow the standards introduced by the World Wide Web Consortium (W3C), as well as the W3C Activity Streams Protocol.
- 2.5.3.** Users should be able to read privacy policy and they must accept it for using the platform.

6.3 Scenarios and Mockups

6.3.1 Scenario 1: Vendor Selling an Item



Persona:

- Erdal Bozkurt
- 54 years old
- He has an electronic sales house
- He is very enthusiastic and anxious about not providing service
- He has an apprentice in his store who learns the job from him

Preconditions:

- He has already used different e-commercial websites before but he decided to switch platforms because he is not very pleased about the application he used.

- He likes that it is very easy to communicate with the admins in this platform.
- There is no limits in terms of selling products in this application, and Mr. Erdal thinks this is the most important feature in the application.
- His worker has a lot of free time. In his free time, he searched the internet and looked for the best e-commerce application and he found "bazaar" at last.
- He is already registered to the application.

Goals:

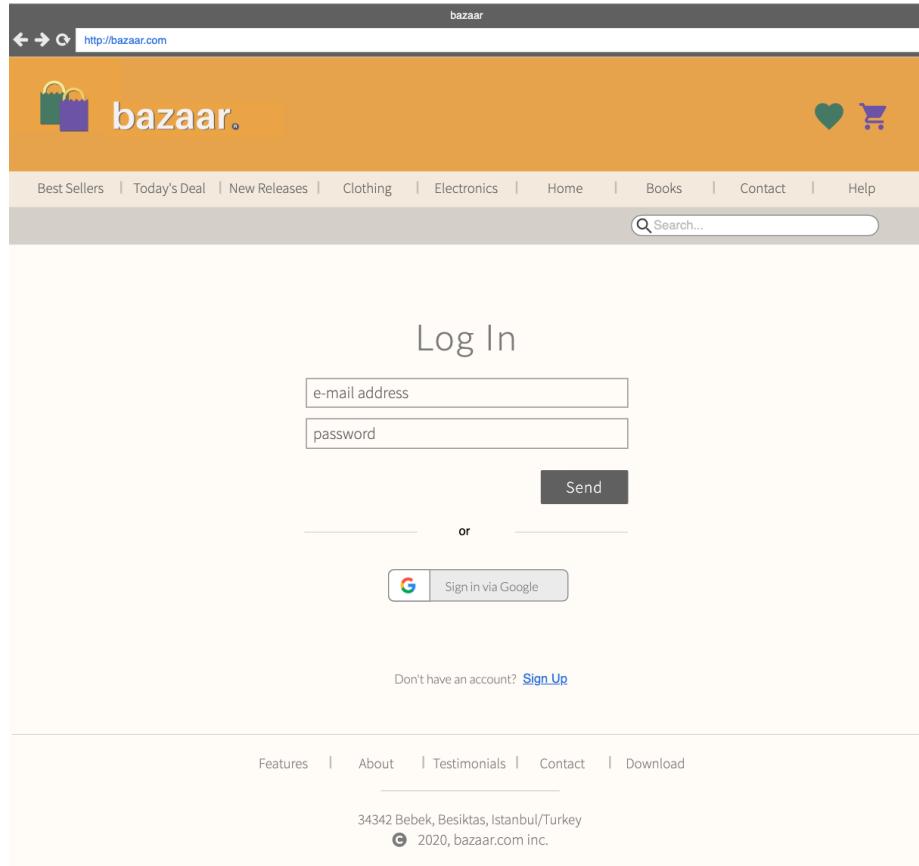
- His store is not that large, therefore online sales is crucial for him to increase his selling volume.
- He wants his apprentice's free time to be put to good use.

User Story:

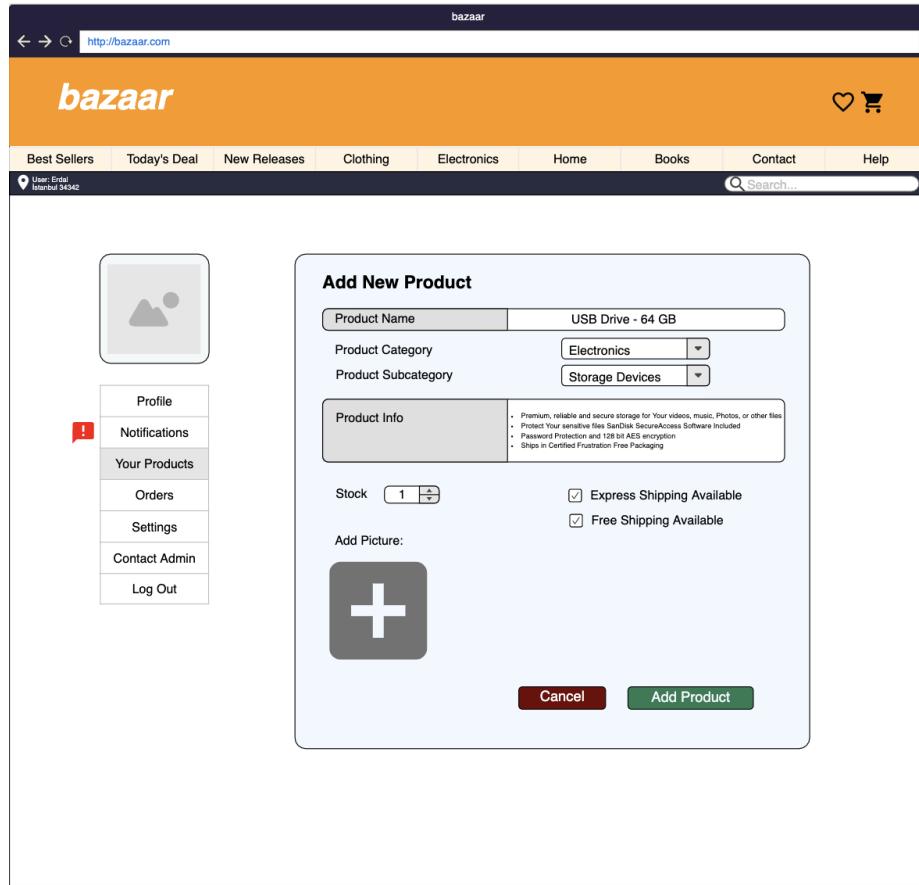
Before Erdal found out about this platform, he was hanging with his friends, Ismail Keskin and Mecnun Atay. His friends asked if he's selling triple sockets, but sadly, he wasn't and he suddenly got nervous about not being able to make a potential profit. Later, his friends kept asking him if he sells other kinds of electronic products that -again- he was not selling. After seeing that people were interested in electronic appliances, he decided to convert his grocery store to electronics store. Then, his apprentice came with a brilliant idea: selling products online. They weighed the pros and cons of having an online store, and decided that selling products online will make more profits than having a physical store. Today, Erdal has the capacity to sell to more than 300 people per day. There are a lot of perks to using "bazaar", such as easily updating product stock information, maintaining product information and healthy communication with the admins.

Scenario:

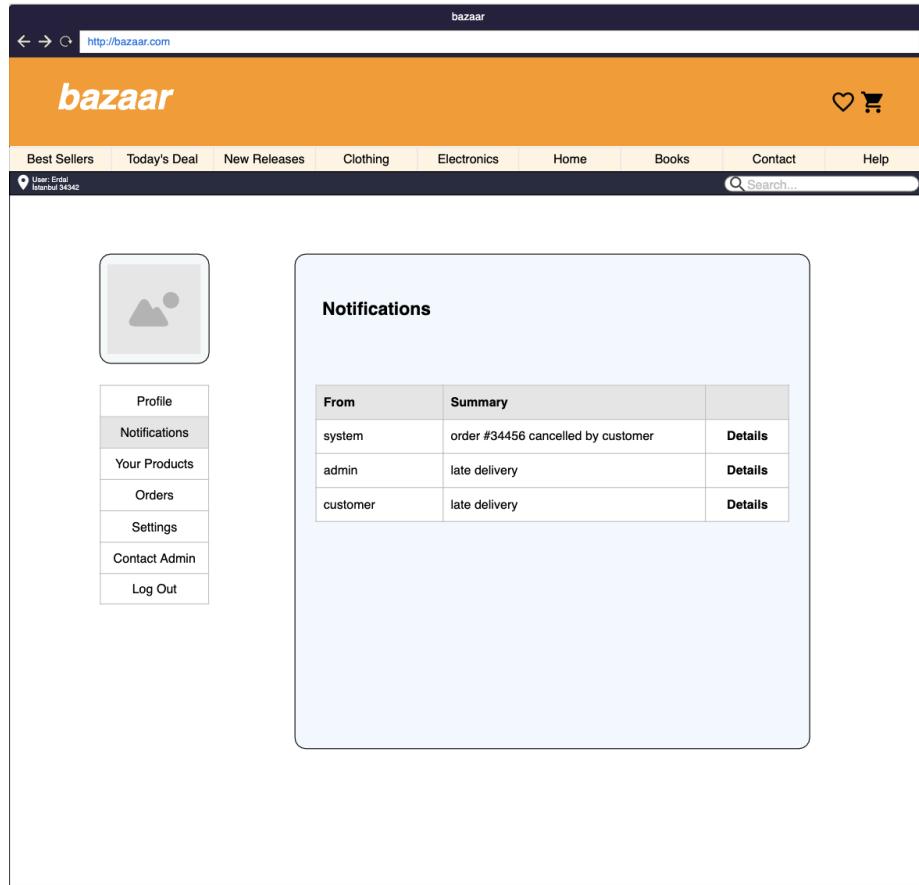
1. Erdal entered the platform, "bazaar".
2. He confronted the sign-in page.



3. He signed in with his Google account.
4. He entered to his own profile and added a new product that he bought newly from the wholesaler.



5. He saw a notification and clicked on it.



6. He realized that the order having id 34456 is cancelled by a customer.
7. He realized that the order having id 34457 is requested by a customer and he changed the order status to "in preparation".

The screenshot shows a user profile on the left and an 'Order Details' modal on the right.

User Profile (Left):

- Profile picture: A chef with a mustache.
- Menu items (vertical):
 - Profile
 - Notifications
 - Your Products
 - Orders
 - Settings
 - Contact Admin
 - Log Out

Order Details Modal (Right):

Order Details:
Order #: 34457

Products:

Product Id	Product Name	Price	Quantity	In Stock
25589	AirPods Case	TL 899.90	1	Yes
23588	PowerBank	TL 69.90	2	Yes

Order Status: in preparation

Shipment Details ▾

Buttons: Cancel (red), Save (green)

Features | About | Testimonials | Contact | Download

34342 Bebek, Besiktas, Istanbul/Turkey
© 2020, bazaar.com inc.

- He realized that there is another order with id 34459, but the requested item in stock was defected. Therefore, he canceled the order.

Order Details

Order #: 34459

Products:

Product Id	Product Name	Price	Quantity	In Stock
25592	iPhone XS Silicone Case	TL 899,90	1	No
23588	PowerBank	TL 69,90	2	Yes

Order Status:

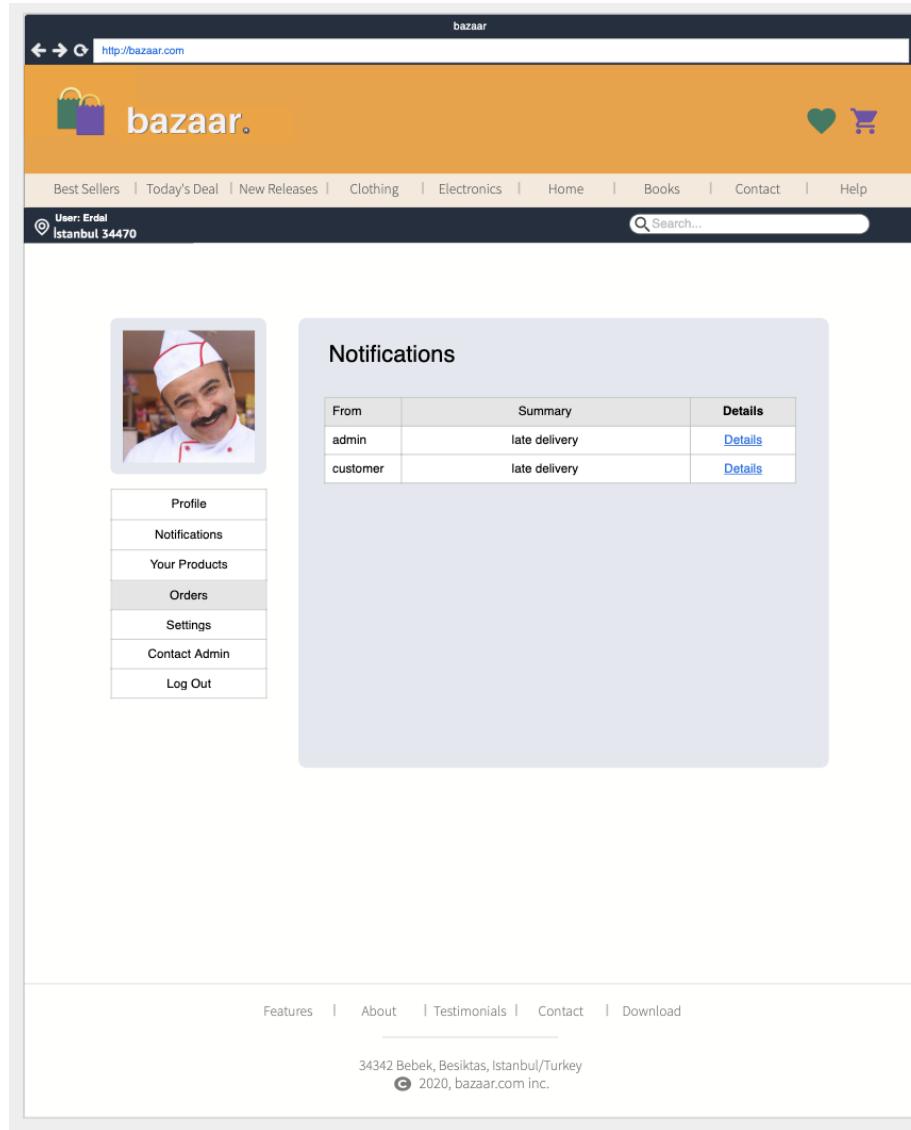
Shipment Details ▾

Cancel **Save**

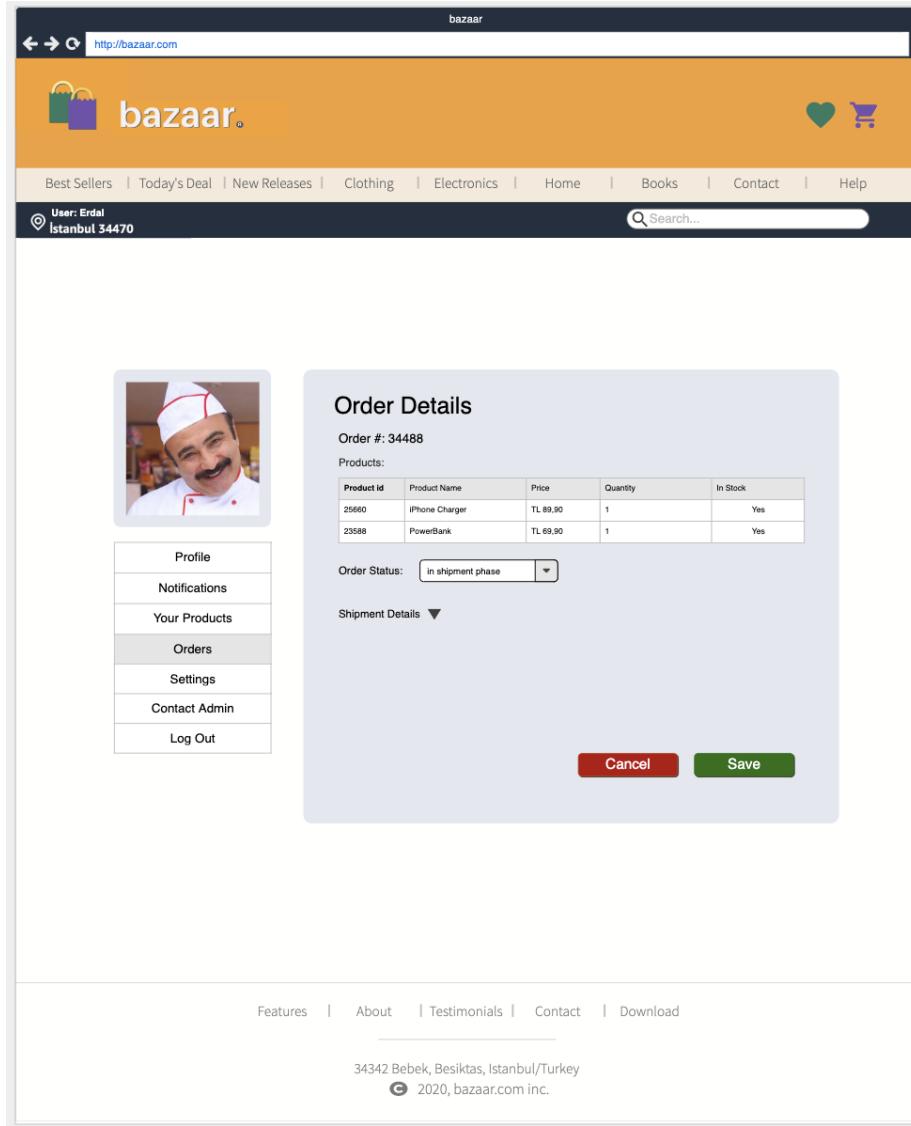
Features | About | Testimonials | Contact | Download

34342 Bebek, Besiktas, Istanbul/Turkey
© 2020, bazaar.com inc.

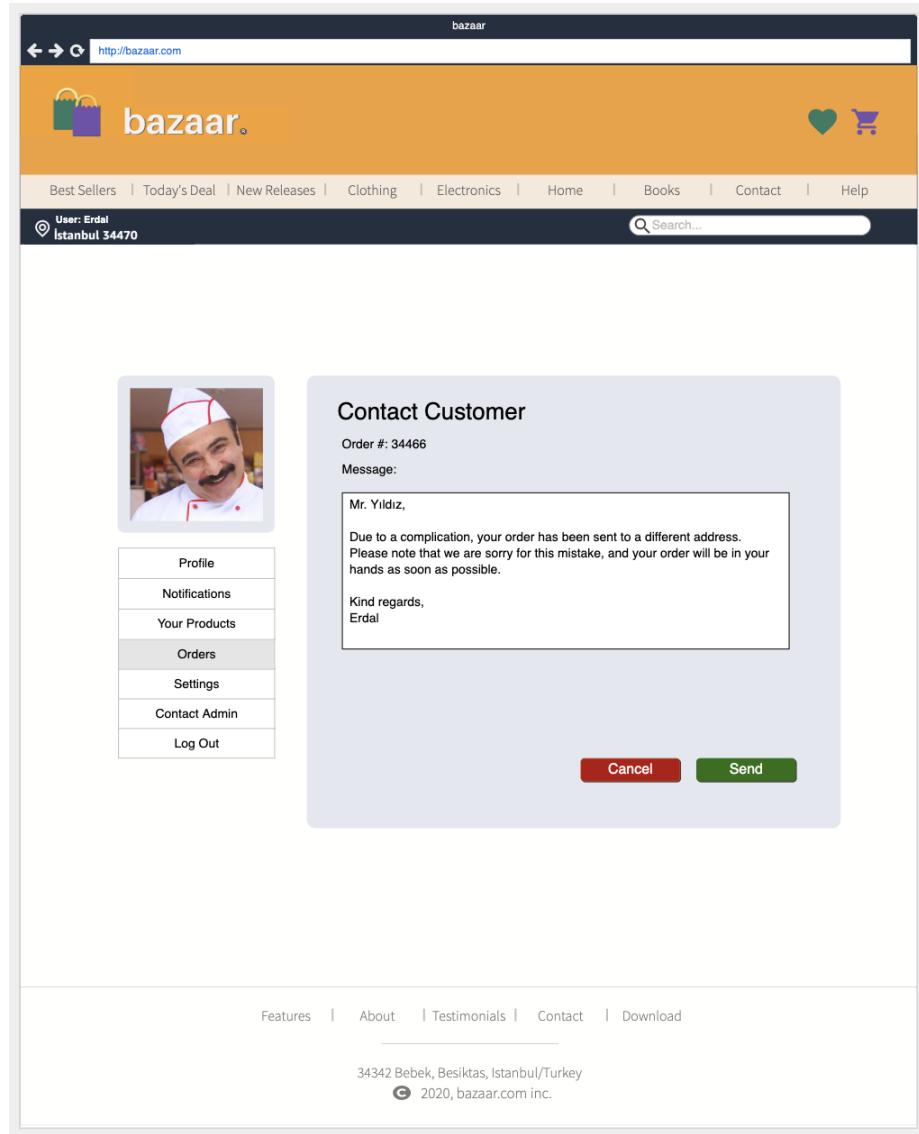
- He saw a notification coming from the admin and clicked on the notification. The notification read that the product has not been delivered to the customer.



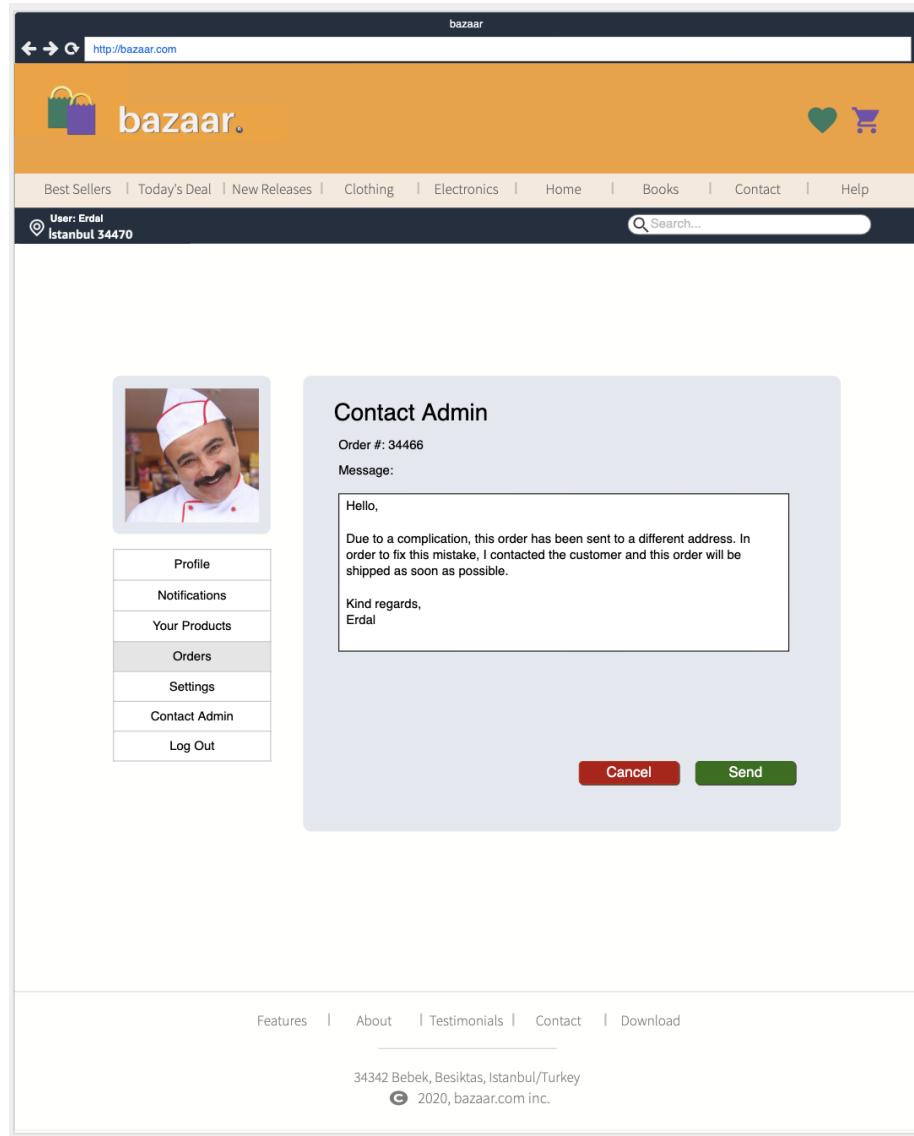
10. He clicked on the order status page and saw that the order is still in shipment phase.



11. He also received a message from the customer that says the shipment is late by 10 days.
12. Then he checked the shipping company's page and realized that he have sent the product to the wrong address.
13. He answered the customer's direct message, explaining the situation and saying that the order will be shipped as soon as possible.



14. He sent the same message to the admin as well.



15. Then, he shipped the new product to the right address.

Acceptance Criteria:

- 1.1.2.2 Guests shall be able to sign up either as a vendor or a customer.
- 1.1.3.1. Customers shall be able to communicate with vendors via a direct messaging system. Vendors shall be able to receive the messages and answer the customers.
- 1.1.3.3. Customers and vendors shall be able to see and manage their direct messages on the 'Messages' page.

- 1.1.11.1. Vendors shall be able to put their products on their profile by providing all the information which is given to the customers and which is displayed on the product's page.
- 1.1.14.1. Vendors shall have an 'Orders' page. In this page, vendors shall be able to see their active orders with the following information: order id; products with their image, their label/id, their price; total price; order time.
- 1.1.14.2. Vendors should be able to modify the order status as 'in preparation', 'cancelled', 'shipped' and 'delivered'.
- 1.1.14.3. Vendors shall be able to cancel an order during the order processing stage in case of a problem.
- 1.1.14.4. Vendors shall be able to see their delivered orders with the information of the orders: order id; products with their image, their label/id, their price; total price; order time.
- 1.1.14.5. Vendors shall be able to communicate with the platform admins about a certain order.
- 1.2.4.3. The system shall be able to notify the vendors if an active order to buy their product is cancelled by the customer.
- 1.2.4.4. The system shall be able to notify the vendors if admins report a problem about selling
- 1.2.5.1. The system shall let customers and vendors communicate via direct messaging mechanism.
- 1.2.5.2. The system shall provide a messaging mechanism for vendors to communicate with admins about an order or a product.
- 2.1.1. The platform shall work as a web site via modern web browsers.
- 2.1.3. The platform should be in English.
- 2.3.4. Admins can keep track of transaction in case of a problem

6.3.2 Scenario 2: Customer Buying an Item

Persona:

- Cansu Serbest
- 26 years old
- Lives in Sarıyer, İstanbul
- Job: Consultant

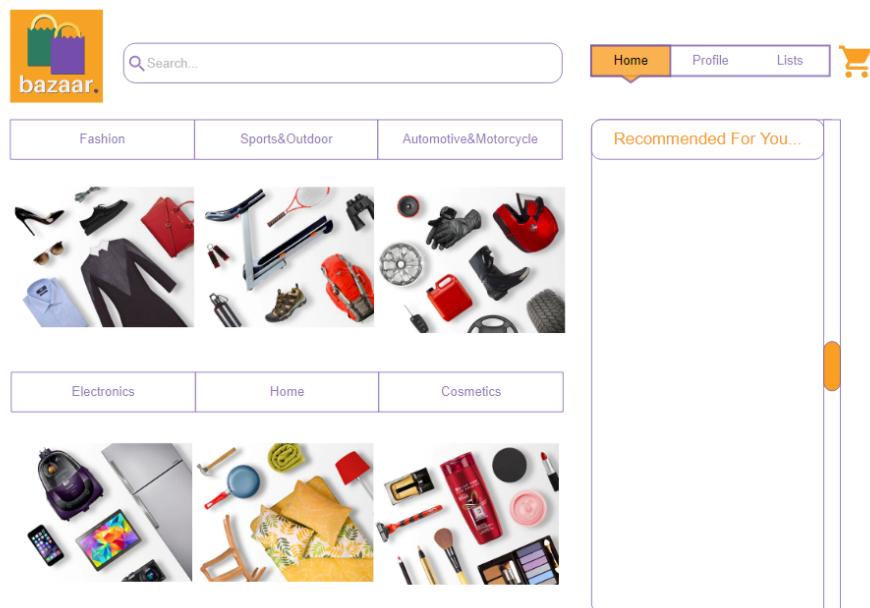
- Financial situation: middle - high
- Need: Wireless headphones
- Story: Cansu enters her house after a long day. On the way home, her headphone cable tangled up multiple times, she got worked up over it. So, she finally decides to buy a wireless one. She always works late on weekdays and on weekends she doesn't prefer to go to the shopping mall. She decides to visit her favorite shopping website 'Bazaar' where she hopes to find a good wireless headphone.

Preconditions:

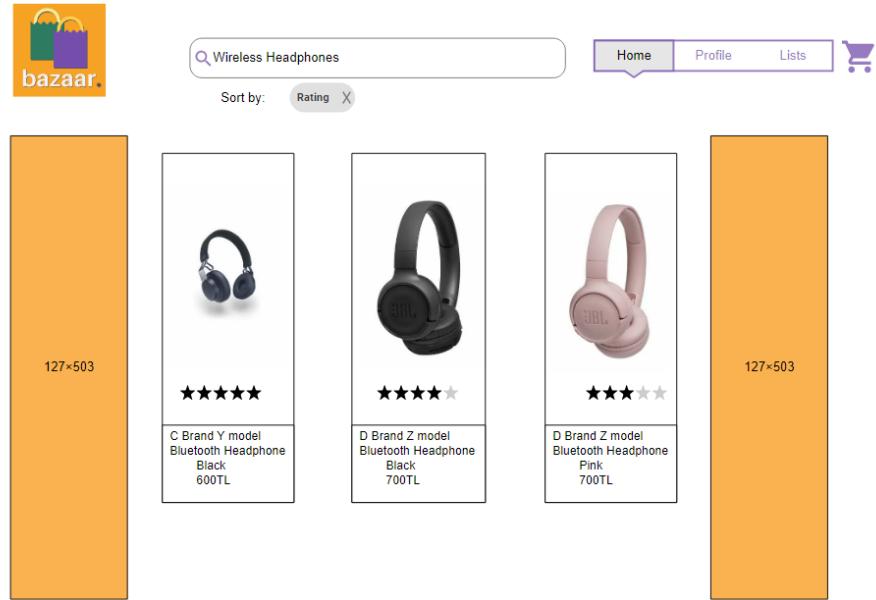
- Cansu uses a web browser and she is already signed in.
- Cansu knows the features of the website since she is an old and active user.

Scenario:

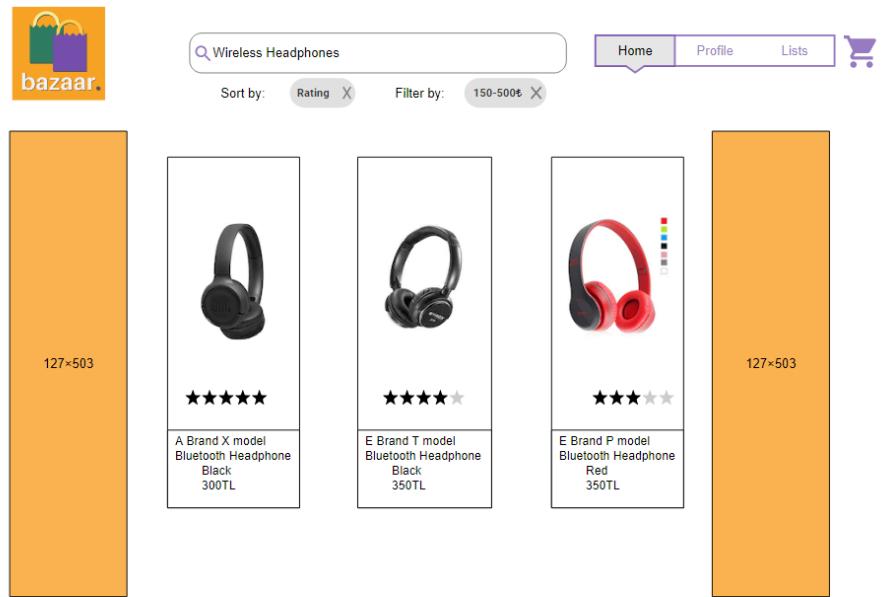
1. Cansu opens the homepage of "Bazaar".



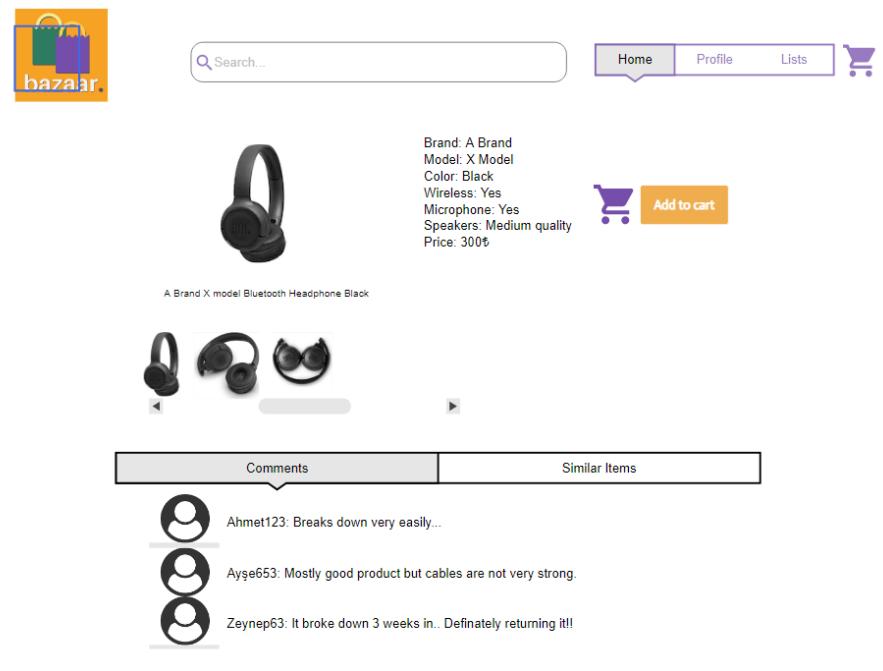
2. Cansu searches for 'wireless headphones' on the site. Then, as always, she sorts the items by their ratings. Now she can see all the headphones sorted by their ratings from high to low.



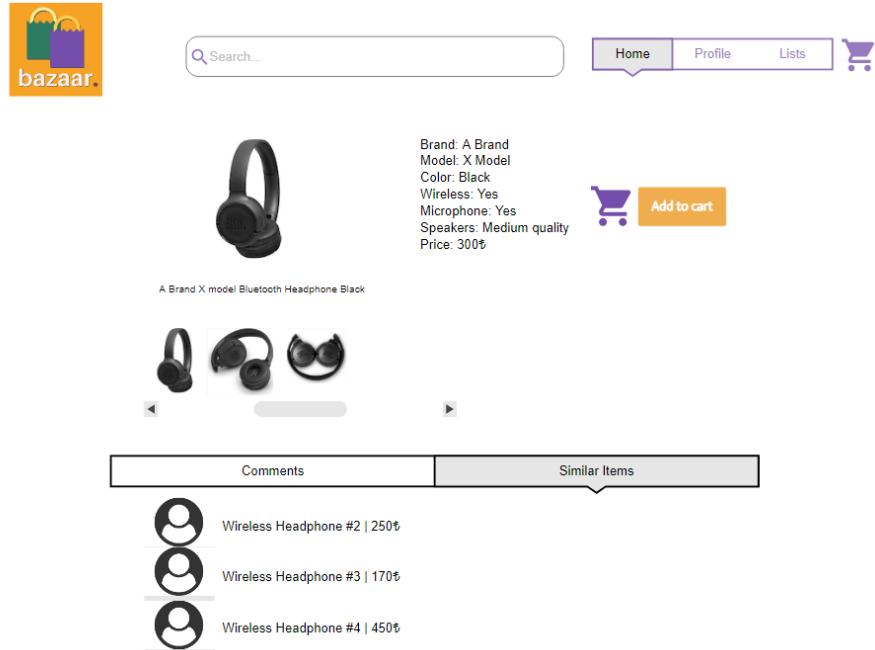
3. But now, she notices that the top-rated items are too expensive for her budget. So she decides to add a price range to her search. Satisfied with the search results this time, she begins to examine the items.



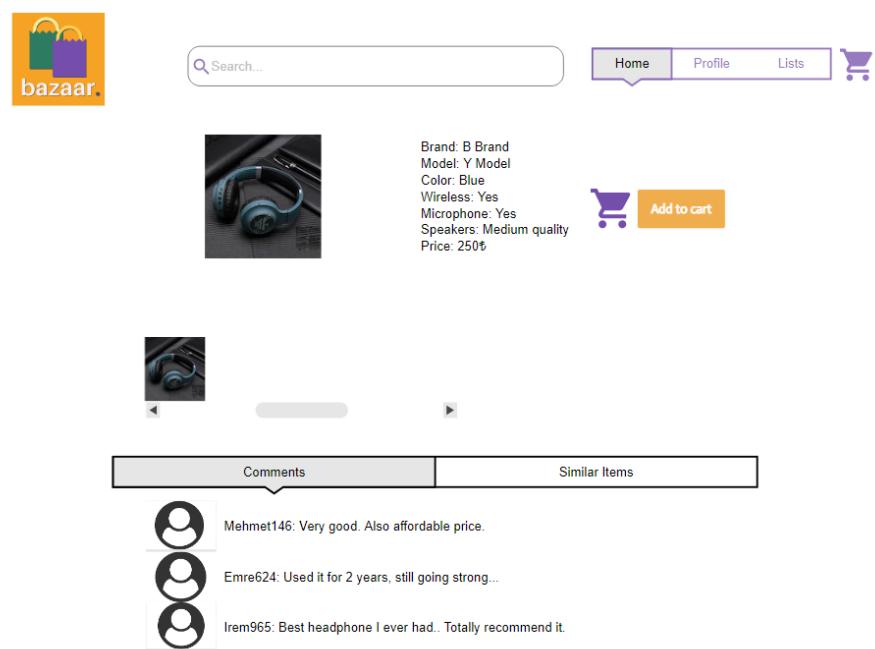
-
4. She likes one of them and clicks on it. When she looks at the reviews of the item, she finds out that the item breaks down rather easily. So she decides not to buy it.



5. She explores the bottom of the product page, where she knows she can find similar products.



6. She checks another promising item from the list of similar products.



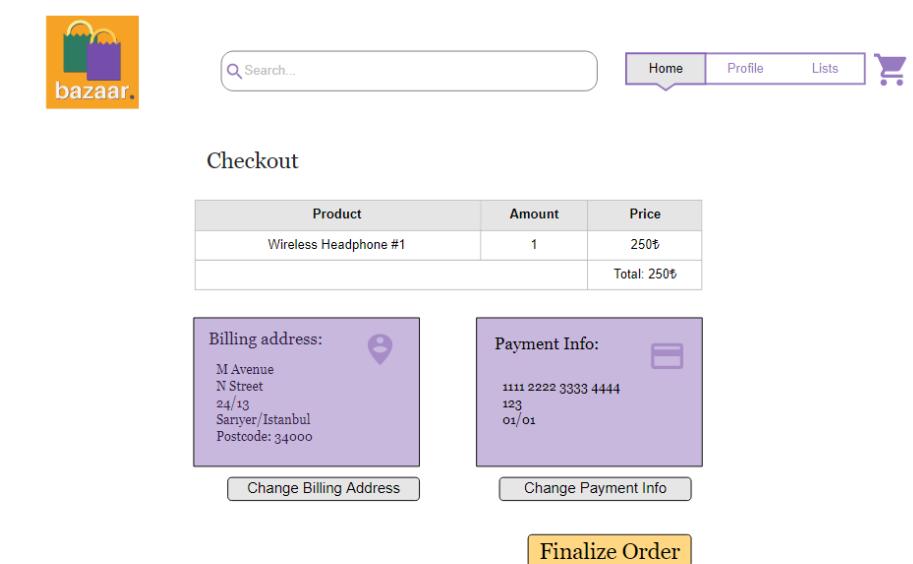
7. After reading all the good reviews, she decides to buy the item this time, since it has positive customer feedback and also is affordable. She adds the item to her cart and proceeds to the checkout.

The screenshot shows a shopping cart page with the following elements:

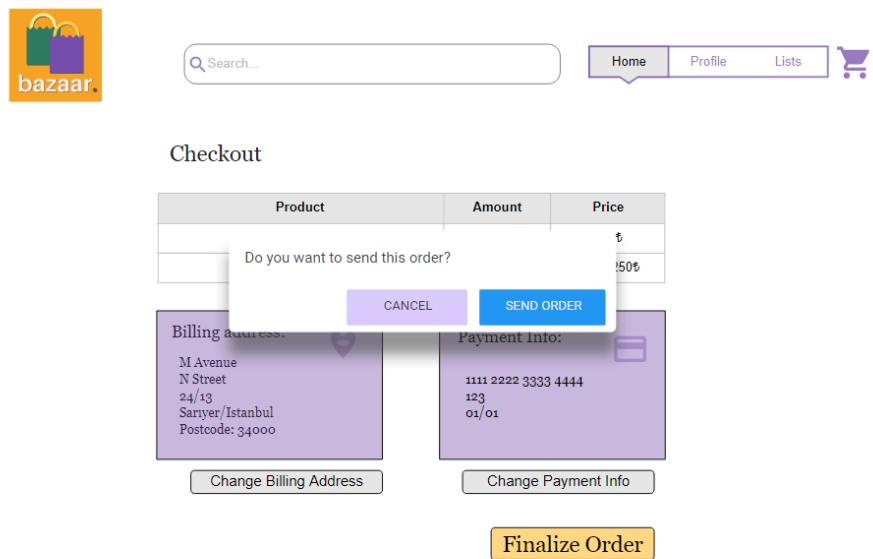
- Header:** A logo for "bazaar." featuring two bags (one green, one purple) above the word "bazaar." in white.
- Search Bar:** A search bar with a magnifying glass icon and placeholder text "Search...".
- Navigation:** A horizontal menu with "Home" (highlighted in purple), "Profile", "Lists", and a shopping cart icon.
- Cart Summary:** A section showing a shopping cart icon and the text "My Cart". Below it, a message says "You have 1 Item(s) in your cart."
- Table:** A table showing the item in the cart.

Product	Amount	Price	Actions
Wireless Headphone #1	1	250\$	Remove / Edit
- Buttons:** Two buttons at the bottom: "Continue Shopping" (orange) and "Proceed to Checkout" (green).

-
8. She is a frequent user of the site, so her payment and shipping information is already provided.



9. Before finalizing her order, she checks the information one last time and finally, she sends the order.



Acceptance Criteria:

- 1.1.6.1. Customers shall be able to search for products by their name, label or brand.
- 1.1.6.2. Customers shall be able to filter searched products based on several properties: average customer review, brand, vendor, price range.
- 1.1.6.3. Customers shall be able to sort searched products based on several properties: bestsellers, newest arrivals, price, average customer review, number of comments, ratings.
- 1.1.4.1. Customers shall be able to display a product's page including the following information: image, label, brand, price, vendor info, stock status, comments, rating.
- 1.2.1.3. The system shall support semantic search, which is used for finding semantically similar products and vendors based on the context information.
- 1.1.8.1. Customers shall be able to add products to their cart.
- 1.1.9.1. Customers shall be able to buy the products in their cart by proceeding to the 'Payment' page.
- 1.1.9.2. Customers shall provide valid payment information and shipping address in order to complete the payment.

6.3.3 Scenario 3: Customer Cancelling an Item



Persona:

- Clark Kent
 - The Customer
 - 75 years old, lives in Metropolis.
 - He is the identity Superman chose to fit in with Human Society.
 - Stores his secret clothing compressed in a secret pouch, this harms the cape often.
 - He has the Herculean strength thanks to protein powders and strength workout.
- Edna Mode
 - The Vendor.
 - Fashion Designer.
 - Far From Cities so she uses websites to sell stuff.
 - Hates capes but still sells them for some reason.

Preconditions:

- Clark has been using the website but not the mobile app.
- Edna uses the both website and mobile app often.
- Edna enables Bazaar mobile to send notifications.
- The cape is still in preparation but not on the way.

Goals:

- Clark realizes that he does not need new cape so he needs to cancel the order.
- Edna needs to be notified that the transaction is cancelled.

Customer Story:

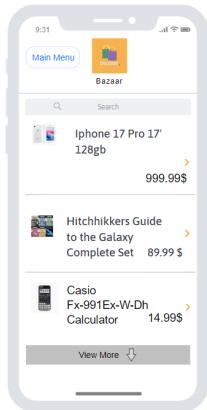
- Clark ripped his old cape and needs a new one. While he is on the way to his job, he used Bazaar to buy one. Later that day, he realised that he had a spare one and doesn't need another one. He proceeds to cancel his order but he has no internet connection but his phone. Thankfully Bazaar has a mobile application so he can use it to cancel his order. Luckily, his cape is still in preparation so he can easily cancel the order.

Vendor Story:

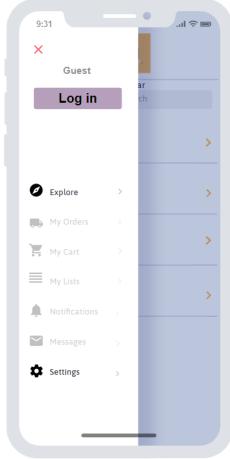
- Edna had a bad day, all her customers were rude to her today. She saw the notification on the Bazaar mobile app and opened it. She got a bit happy as she saw Clark decided not to buy a cape. Superheroes should not use cape, she muttered. She didn't know that Clark had a spare cape.

Scenario:

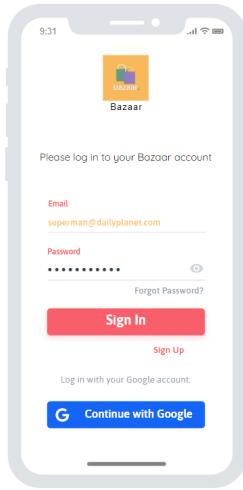
- Customer Section:
 1. Clark enters the mobile application Bazaar.



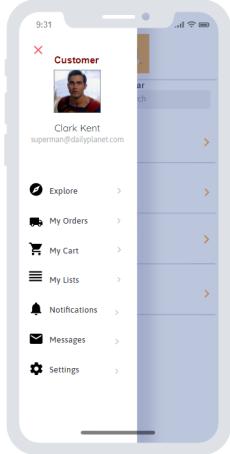
2. Clicks on the "Main Menu" button



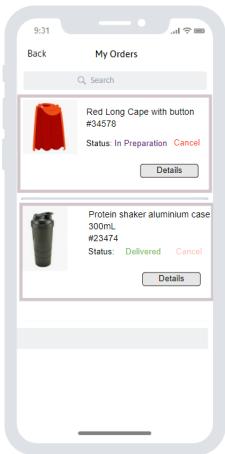
3. He clicks on the login button.



4. He enters his username and password, logs into his account.



5. He clicks on the My Orders tab.



6. He then proceeds to cancel the order 34578, Red cape, clicks on the cancel button.



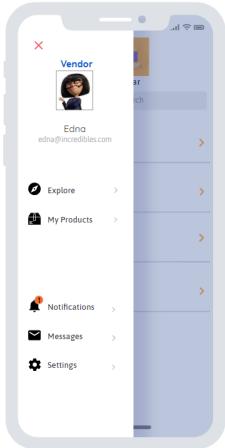
7. He confirms the cancellation.



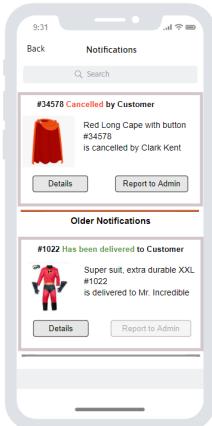
8. Cancellation request confirmed by the mobile application Bazaar.
9. Regarding this situation, the vendor was notified.
10. Money return transactions have been made successfully.

- Vendor Section:

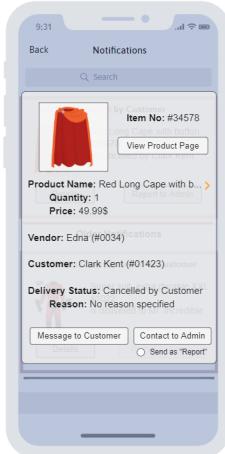
1. Cancellation notification has been sent to the vendor, Edna's phone got a notification.
2. Edna opens the app and clicks on the notification.



3. Edna clicks on the "Details".



4. Now that Edna is informed about the cancelled transaction, she is going to cancel her cargo that she was going to send to Clark, She informs Clark via "Message to Customer".

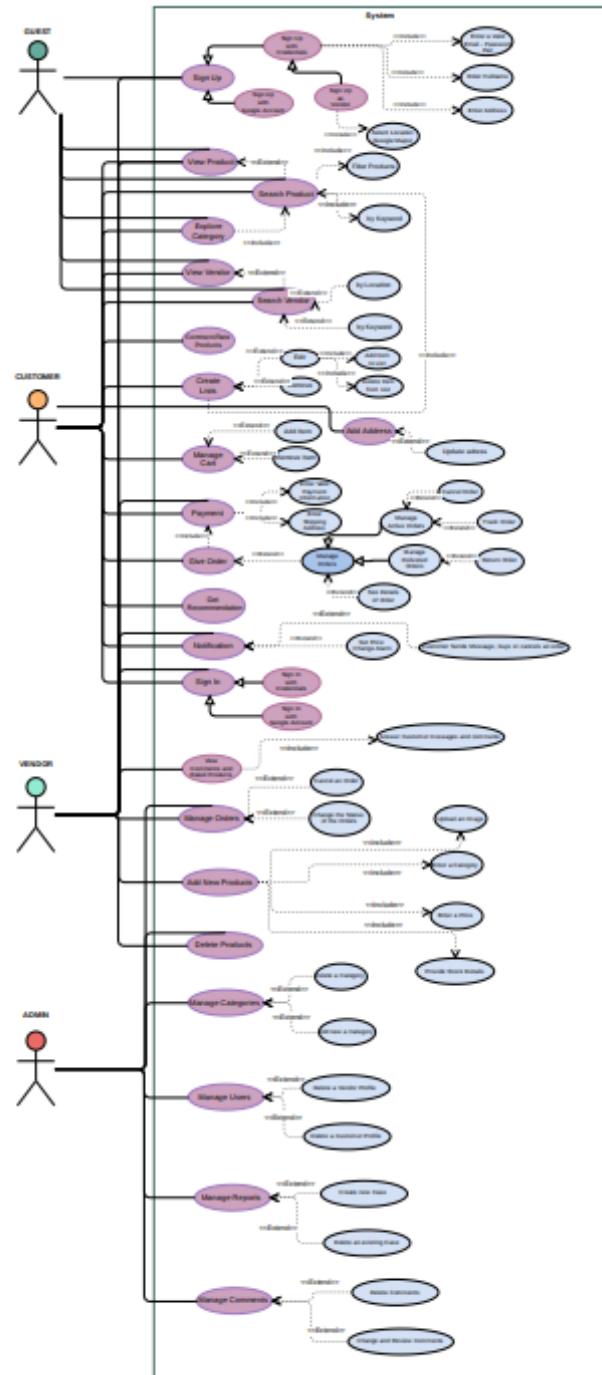


5. They message each other and happily live ever after.

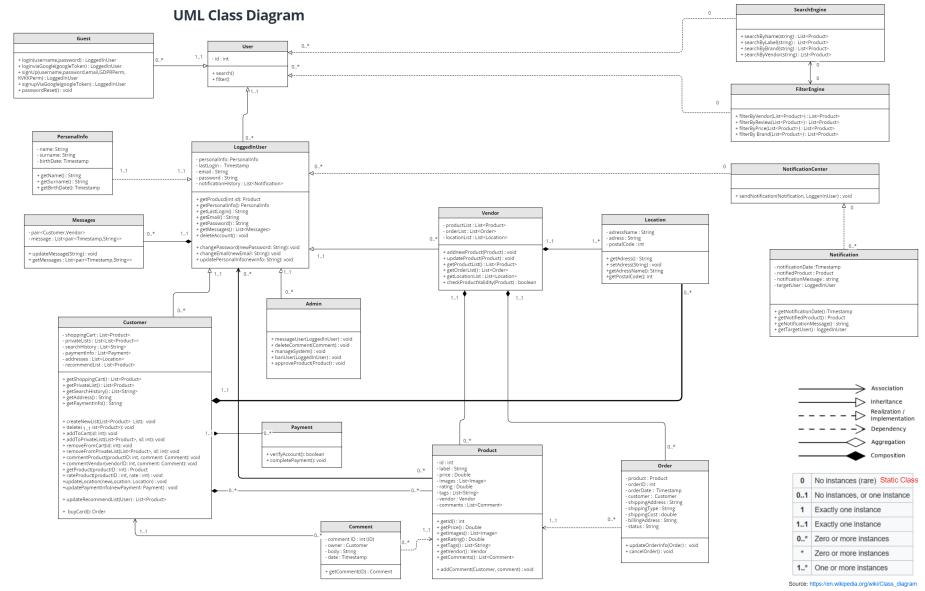


6.4 UML Software Designs

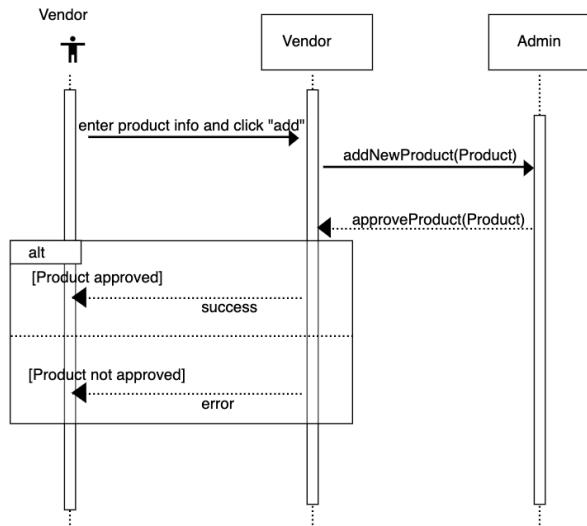
6.4.1 Use Case Diagram

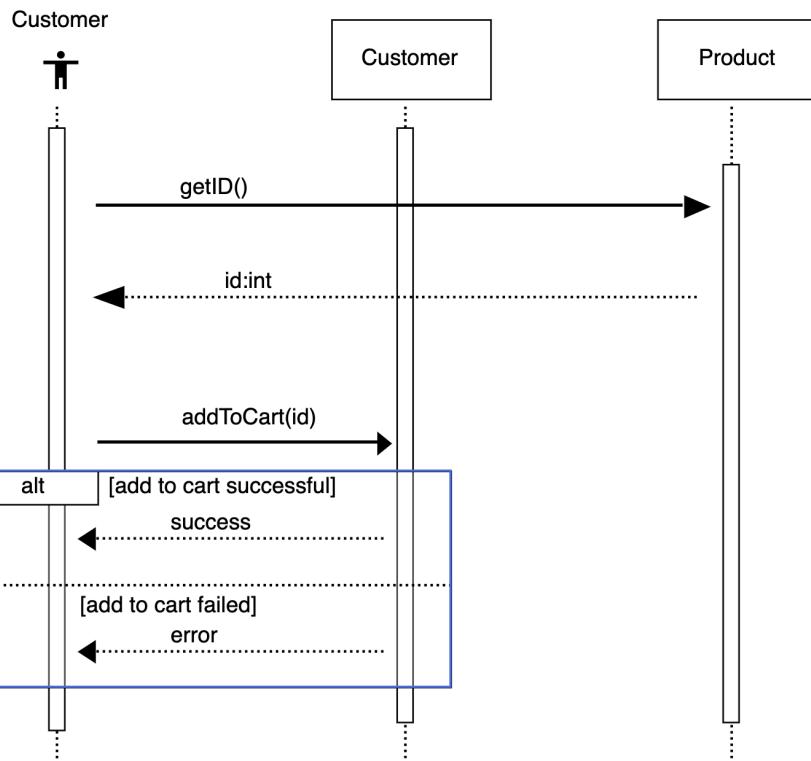


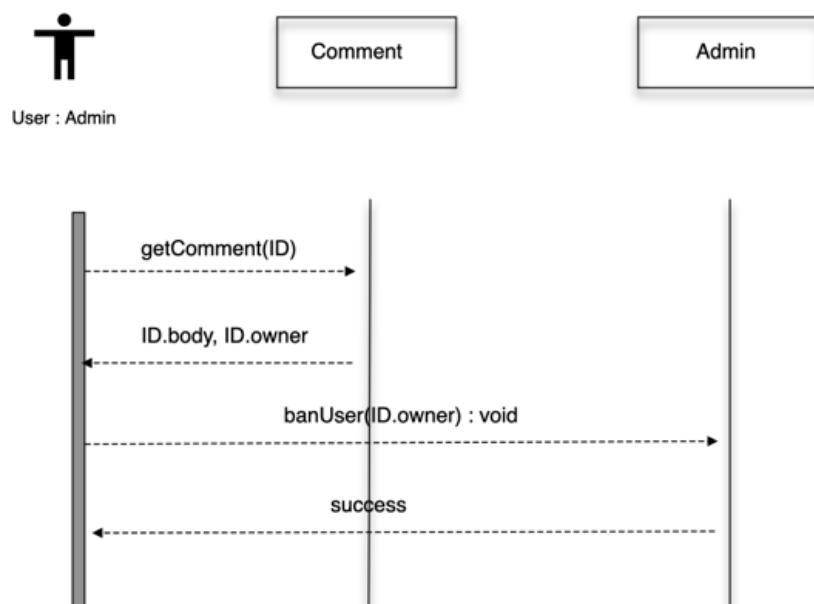
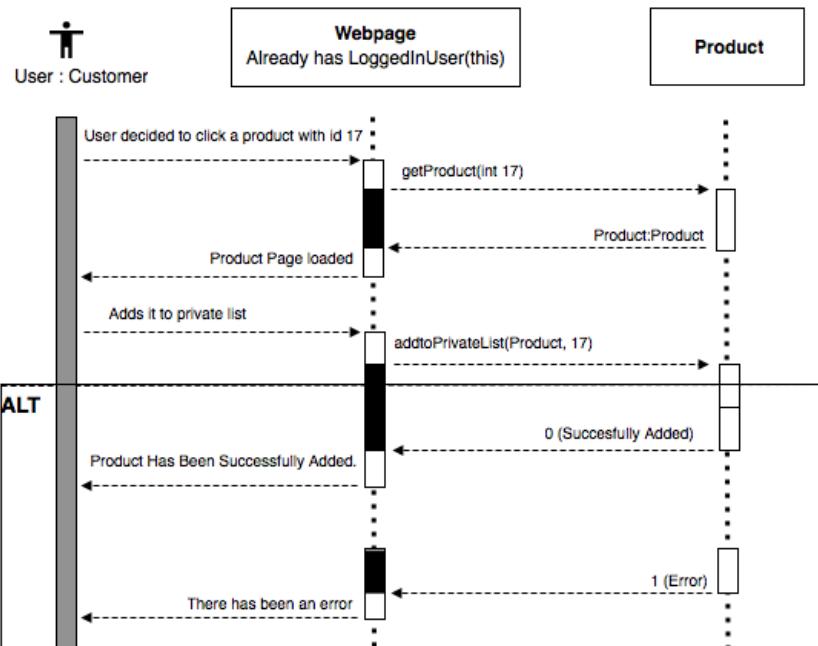
6.4.2 Class Diagram

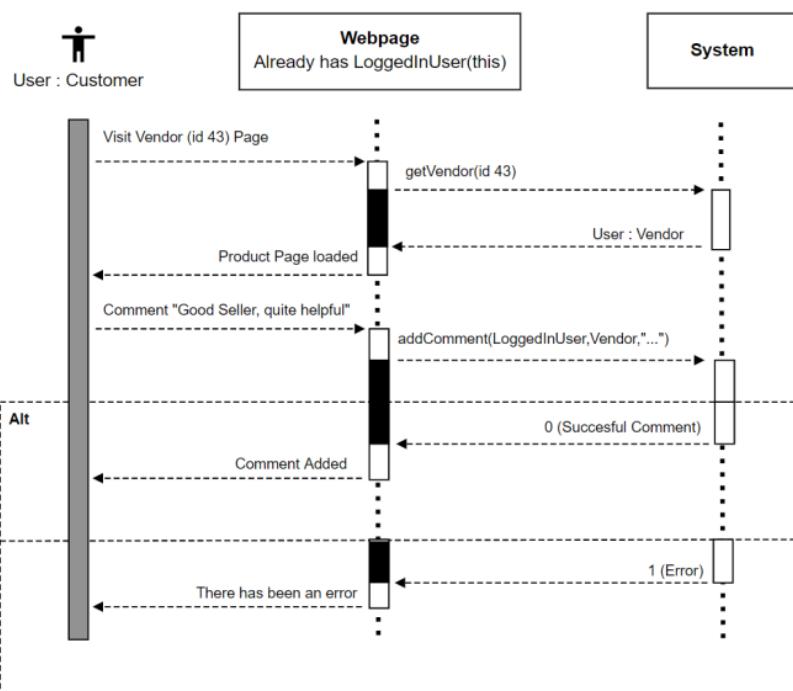
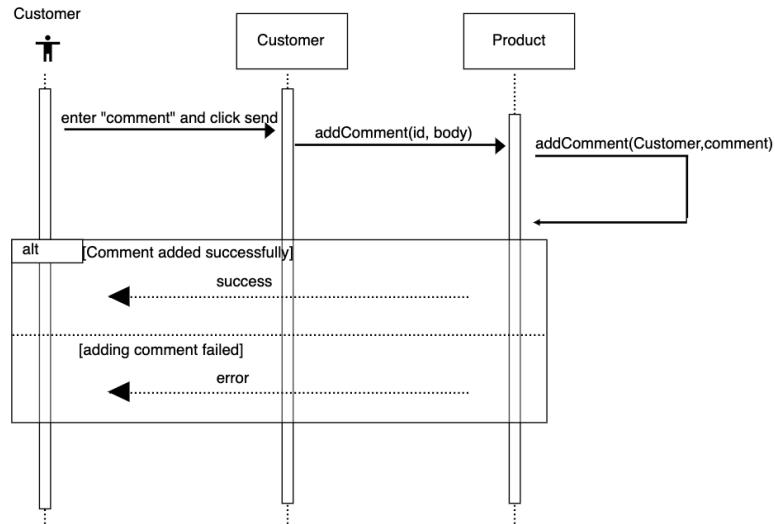


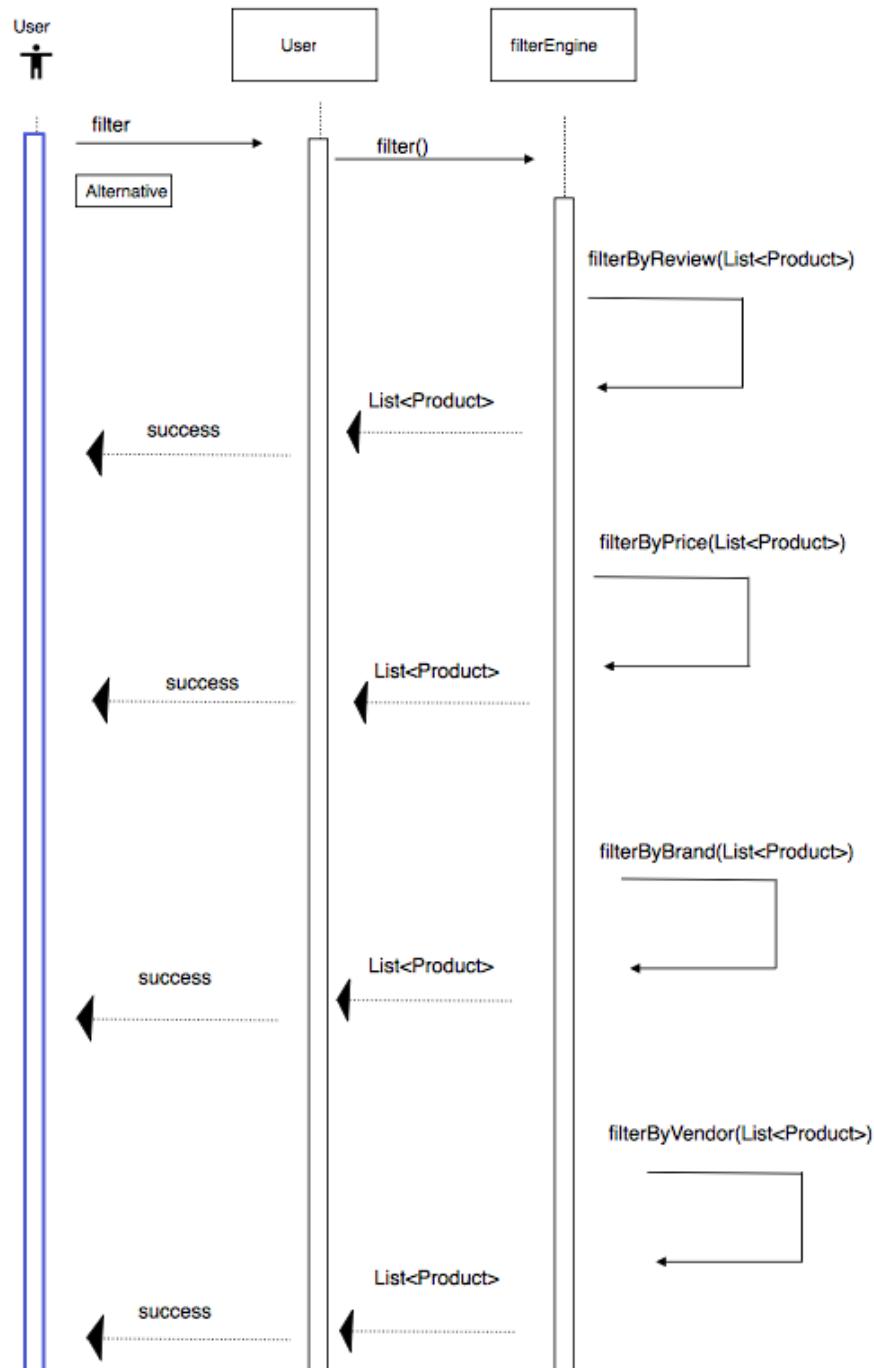
6.4.3 Sequence Diagram

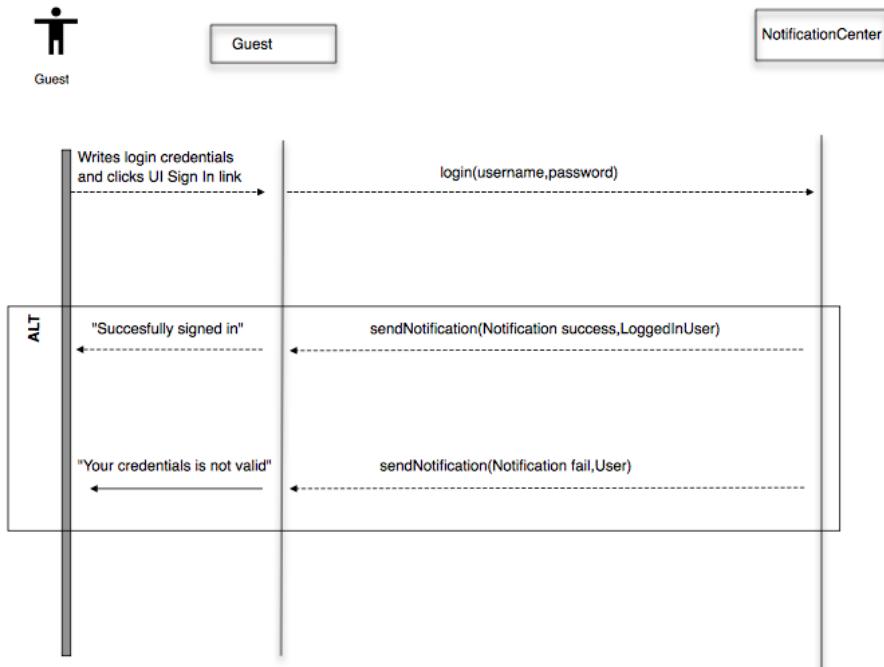


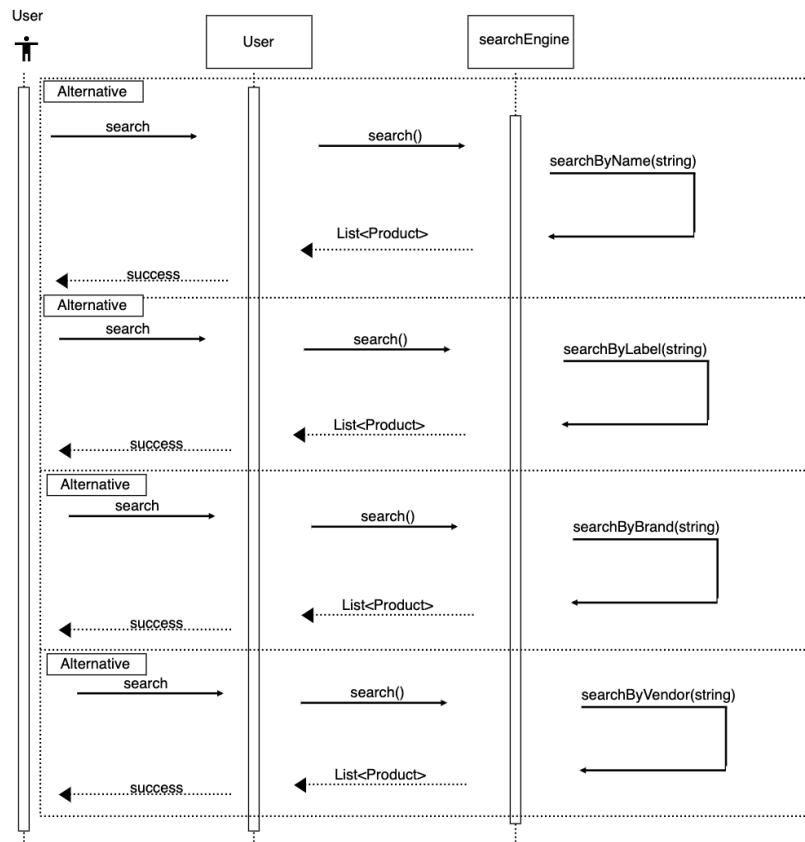


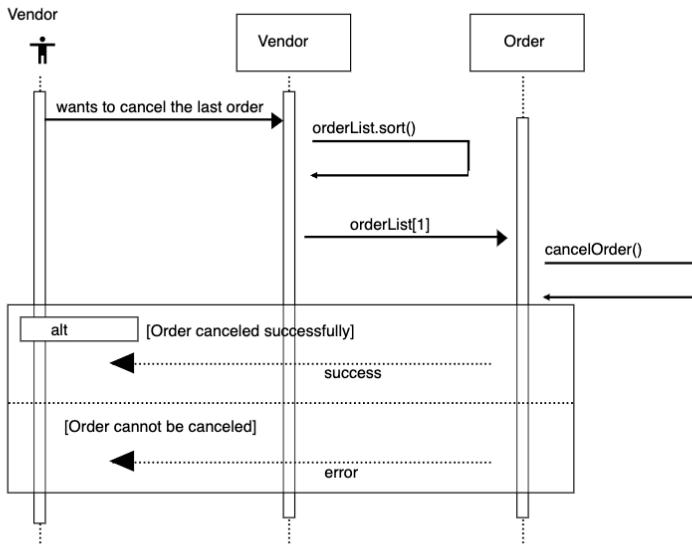












6.5 Project Plan & RAM

