[BOO] Requirements Specification

Version 1.0

April 19, 2021

Table of Contents

1. Executive Summary

1.1 Project Overview

This project is about creating and building a software, specifically a web-based application that manages the activity within the Epoka University canteen. The software is called BOO (Bereqet online ordering). BOO can be used by the customers and staff/students. It covers different procedures like online ordering from the students/professors, different guests and products, price, and orders management by staff (admin or storekeeper).
The client can make the orders directly from a web browser and can be directly notified if order is ready. This is a very efficient way of buying, as it provides a better coordination between clients sellers and admin (storekeeper). Basically, the app has a special user interface for each user. Another task being managed by the app is sales management. The lists of products are updated every time a new product is added in a specific category. These lists are accessible by the customers, who can check the prices, product description and photo. The staff (sellers) can see the orders, description of them for additional changes. Sellers are those who notify if order is ready and are those who make the purchase with the customers. Extra feature is that staff can keep track of entire bills purchases by them during a working day till the decide to quit the shift. Admin is the maintainer of the entire system with feature like adding, updating, deleting product and checking their supplies. Also staff is managed by admin that can add, delete and update particular members. All purchasing record are accessible by admin where he/here can audit, manage weekly, monthly, annual balance. Admin is fully responsible for finances and entire billing records.

1.2 Purpose and Scope of this Specification

Describe the purpose of this specification and its intended audience. Include a description of what is within the scope what is outside of the scope of these specifications.

The purpose of this web app is to provide a set of solutions to the university canteen that depend on sales management and user management. In the current state, when ordering a product when it comes to management it is clear that it is very unorganized. Clients have to wait in a long queue and the employees have a very frustrated way of managing the orders since they have to deal with many people at once. This becomes uncomfortable because some orders may be overlooked by mistake or when there is no more stock of a product and someone is waiting it is timewasting and frustrating. This software will provide a very organized and comprehensive solution for these issues, which will increase the cooperation between administrators, employees and customers. It will make the life inside the canteen easier and save needed time and energy for all the parties that go through the process.

2. Product/Service Description

BOO (Bereqet Online Ordering) is a software which aims to redesign the lifecycle of the university candeen and provide new technological solutions to every step of the process. This software will be used by the administrator, storekeeper and the customer. We want to create a dynamic software, which will become useful to every user (staff/students and clients(outsiders/guests)).

The software aims to help every step of the company job, starting from simple order receiving to order status tracking, report generation, and much more. If we divide the company in two major subsections we are able to give a more detailed description of what this software provides for both parties.

➔ Company Staff (Administrator, Storekeeper, Economists)

- Real time Statistics, Sales Performance
- Order Receiving and Tracking, Reports and much more.

➔ Company Customers

- Order Making
- Order Status Checking (ready or not)

2.1 Product Context

The context of this software is related to the Epoka University. This software solution will be used by every employee of the canteen and also by the customers in order to enhance the cooperation between staff and clients, make their jobs/purchases easier and make sure they make a full use of their time and energy.

2.2 User Characteristics

There will be 3 types of users in this software:

➔ Administrator

➔ Staff (sellers)

➔ Customer ()

a) Administrator (admin/ super user)

- Add/ remove sellers.
- Add/ remove/ update product.
- Check all bills of all sellers during a day, month, entirely.
- Make query for bills and also check the inventory.
- Add/ remove bills.
- Can update his credentials.
- Admin is created since production, a hardcoded one

b) Staff (seller)

- Receive orders that are going to be prepared.
- Notify user for the preparation.
- Delete order if user doesn't come to receive it.
- Confirm the order after the purchase and automatically bill is created
- Able to check bills that are being prepared during his/her shift.
- Is able to change his/her username and password.
- Close the shift/turn at end of the day.

c) Customers (simple user, professor/student,)

- User is temporary as long his order is not purchased yet.
- User is created when he enters the website.
- User can give description to a specific buying (example "kafe shkurter , sanduic pa qep etc. ")
- User can add remove sells entity (sells are orders of the same product).
- The entire order is every particular product ordered by an user (customer)
- After confirming the bought product, needed to and a contact to notify if order is ready.
- Order is created and is assigned to a seller.

2.3 Assumptions

- ➔ It is assumed that the data generated from the system will be fully confidential and only available to the university
- ➔ It is assumed that every staff member is equipped with a smart device from where they will perform every operation they are required to. If not, the university is assumed responsible for equipping every member with the required device. These smart devices have to be capable of receiving information from web browsers. The smart devices are assumed to have a Windows Operating System.
- ➔ It is assumed further that the staff are able to use smart devices and especially the BOO web application effectively and efficiently.
- ➔ It is assumed that every customer should be limited to overview only his orders and not interfere with other customers.
- ➔ It is assumed that administrator, staff , clients have a web browser and an active internet connection.
- ➔ It is assumed that every completed order in a form of bill needs to be stored in the system for documentation and report purposes.
- ➔ It is assumed that the staff member is responsible for the progress of the orders and real time notification.
- ➔ It is assumed that the administrator is responsible for updating stock in real time.

2.4 Constraints

This system will be potentially constrained by:

- The fact that employees need to be equipped at all times with smart devices
- Customer need to provide contact to get order notification.
- The need of a fast internet connection and real time database update.
- Having every employees understand the way the system works and training them to use the web application correctly and efficiently.
- The need of a browser in every device to run the application.

2.5 Dependencies

- Users need to have a strong internet connection in order to run effectively.
- Customer should provide contact number to be notified.
- The order needs to satisfy product stock availability.
- The costumer need to have at least 1 product in bag in case to make an order.
- Sellers receive orders in turs, depending to the customer who is currently ordering.
- Bills, purchases track for a sellers is available only for current working day.
- The system is also dependent on the browser performance.

3. Requirement

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_01	The software should have different view for each user.	View for client that should be smartphone prototype, view for the sellers and view for the administrator both computer size prototype.			
BR_02	Administrator and staff accounts should be secured with basic authentication(log in)	Data will be saved hashed in DB.			
BR_03	Administrator can add, update and delete product information.	Admin can add product according to the given categories or can add new categories also every product is editable and deleted by admin. Validation is important for them.			
BR_04	Administrator can add, delete, update staff members (sellers)	At first time a seller is created by admin also only admin can delete. Validation must be passed.			
BR_05	Administrator audit the entire inventory.	Check sold items according to the date (annually monthly, weekly, daily), name, category, division			
BR_06	Administrator should make custom query in inventory	For example: admin can add range of dates, can add item name or category for search.			
BR_06	Administrator can check bills, also may cancel any of them	Admin has a detailed view of the bills also hard printed views. Cancelation is needed because wrong bills may be saved.			
BR_07	Administrator should update his/ her credentials.	Validation of data must be passed.			
BR_08	Client is saved temporally.	As long as has an open order not purchases is open. After that client is deleted in form DB			
BR_09	Client can freely chose the product as he/ her want.	Search should made handy using division cantina/ cafeteria, categories or direct search using name			

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_10	Client can add custom description for prepared product in cantina.	Each client may need to remove some ingredient of product or add some other.			
BR_11	Bag should be created at user view.	At the bag the user see entire order with different product. Add more amount form there, remove any of them or decrease amount.			
BR_12	User can not send the order without adding the contact number.	The only requirement is the contact number for notifying the order preparation.			
BR_13	A copy of order is saved locally in user device after order confirmation.	This is needed to show that he is the user with that order or just showing the notification message by the cantina.			
BR_14	Seller has a panel that waits for the incoming order.	There might be 2-3 sellers working at a time so the orders should go one at a seller other at the other seller and repeated the same logic.			
BR_15	Seller notify the client when order is ready.	Message is sent to the client using the contact number, notifying that your order with x id is ready.			
BR_16	Seller confirm the order only if client purchase it.	After confirmation bill is created and saved in DB.			
BR_17	Seller delete the order if the client don’t purchase the order.				
BR_18	Seller is responsible in viewing all his daily bills.	After closing the turn no bill can be seen.			
BR_19	Seller close the turn and all bills are remove.				
BR_20	Sellers can update their credentials.	Confirmation is made after completing the necessary validation.			

3.2 Non-Functional Requirements

3.2.1 User Interface Requirements

Every user is going to access the software using the browser using every device that access the internet. There will be 3 different independent interfaces for each user. Client interface will have smartphone resolution that can be adaptive for each prototype. Other 2 interface will full HD (1920 x 1080) resolution for computer base usage.

Three main interfaces are:

- **Client site**

Ways that use can access the site:

- Using the QR code that redirect to the website link.
- Typing the website in browser
- There is no need for login

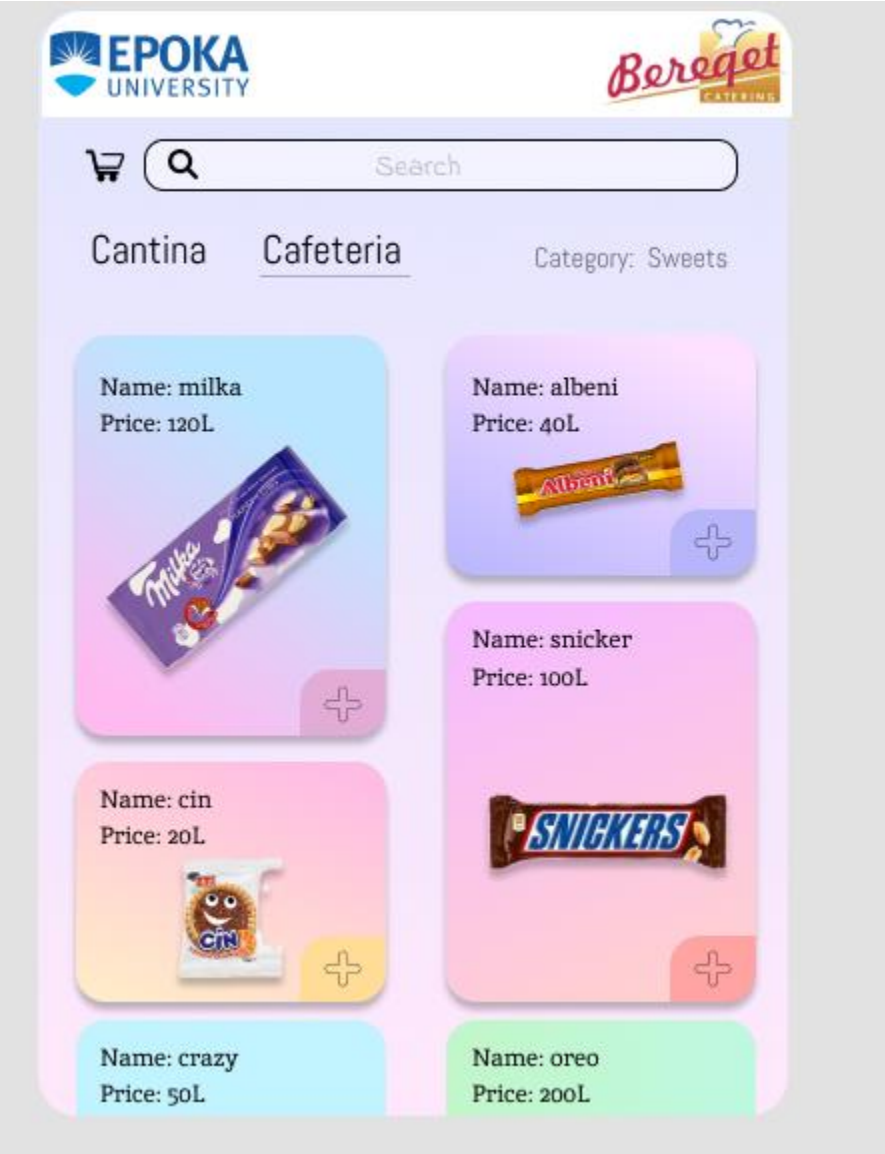
- ***This is the first page of the website (is the view of division and category selection)***

Elements of the page are:

- Logo of Epoka University and Bereqet. These will be present in every page
- 2 Division arrange in a nav bar.
- Categories for each division that continue horizontally.

User enter the site and the first page is this. He/ her can select the division and categories will be displayed horizontally. Client can easily scroll and find the want category. After selection other page will be displayed.

- ***Chose product page***



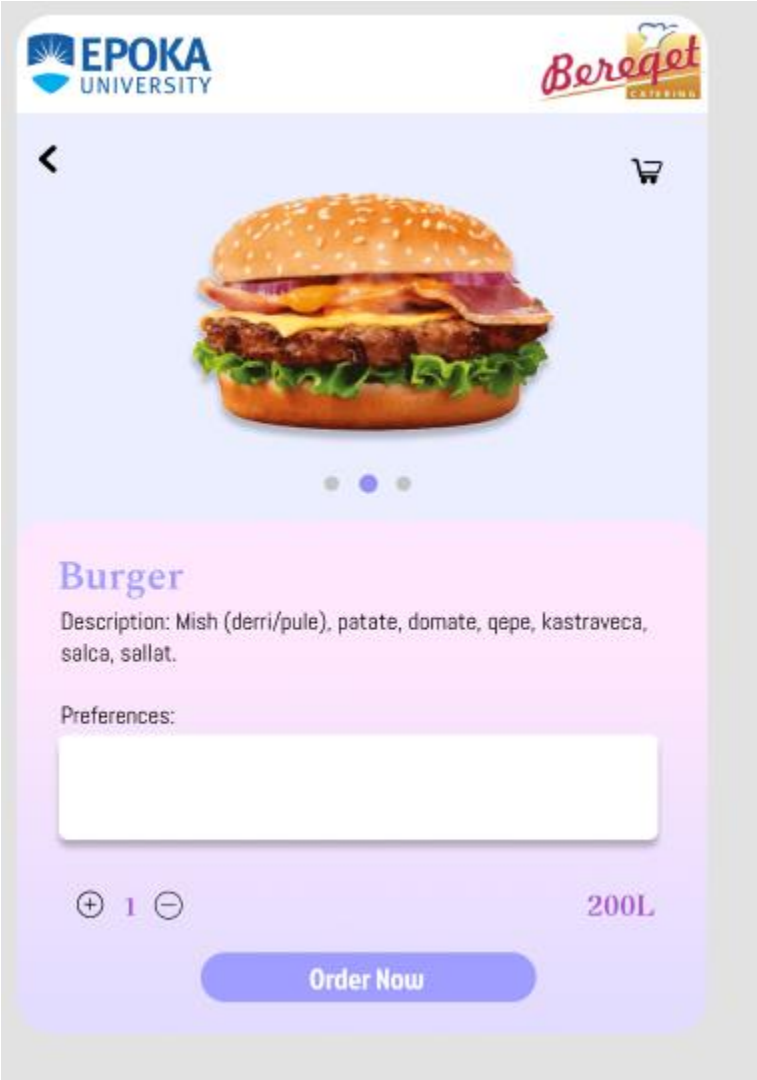
Element of this page are:

- Common header and footer logo and other information.
- Bag icon that present the bag of ordered product.
- The search bar that find product by name.
- Nav bar which define the division.
- Select list at the corner near the nav bar that define the category of the product.

- Product that are arranged in div using grid system.
- Plus icon in every product.

User can open the bag view page clicking the bag icon. The search bar is needed to search product by name. Navigation bar define the division and by clicking it will display by default the first category of the division. Category can be changed using the category select list. Then are listed well arrange product for defined category and client can continue scrolling for finding the wanted product. By clicking the product, the product description panel will open. Client can add directly the product by clicking the plus icon. Every extra clicking of the same product will increase its amount in bag.

- ***Product description page*** *(have 2 different instances depending on product division)*



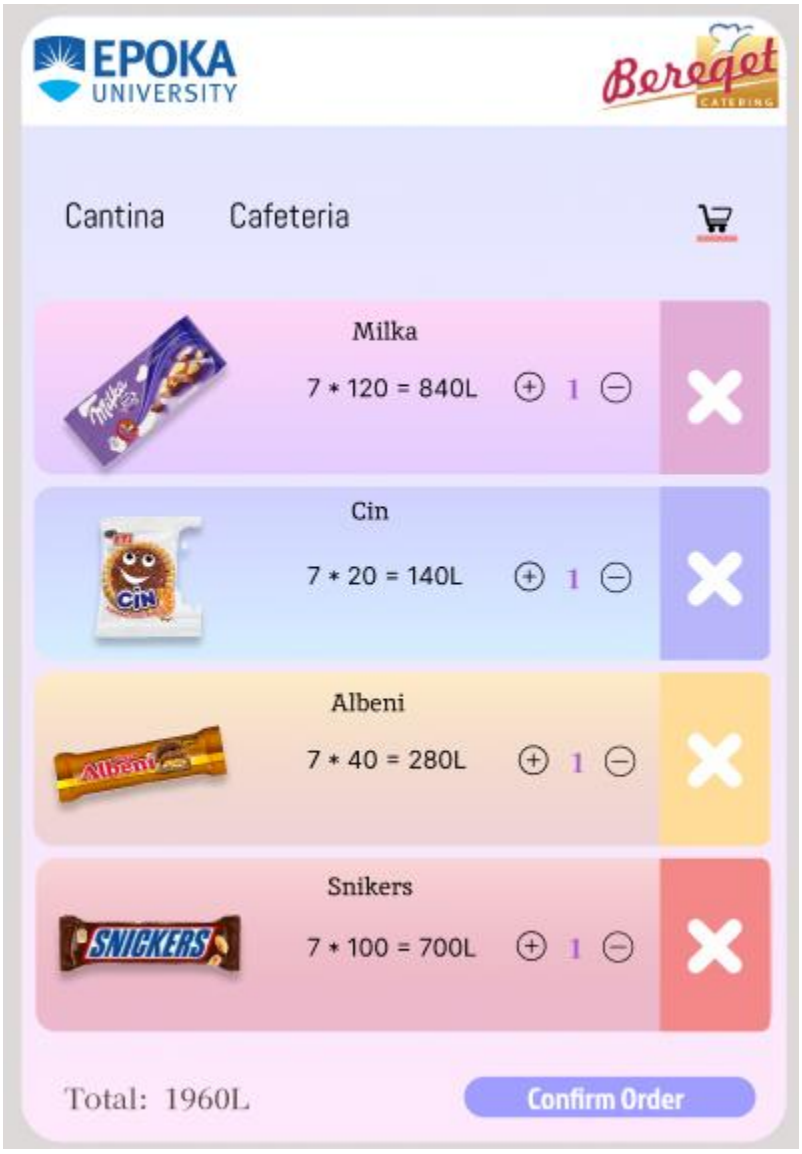
Element of this page are:

- Turn back arrow
- Bag icon
- Product image
- Product name and description.
- Preferences text area

- Price and amount (input number)
- Order Now button

Use can turn back to the product selection page using the arrow. Bag icon open the bag of ordered products. There is a preview of product given by the image also name and description are given. There is an extra text area for preferences if the products belong to cantina. There, client can add custom preferences like adding ore removing ingredients. This field is not present in the product belonging to the cafeteria. Client can chose the amount using plus or minus in the amount input component. Lastly by typing Order now product is added to the bag and after clicking site is redirected to the Bag page.

- ***Bag page***



Element of this page are:

- Custom header and footer
- Nav bar with division and the bag icon
- Products grid structure that contain image name, amount the price
- X icon
- Total
- Confirm order button

Here in the bag are all products ordered by the client. On top is nav bar, by clicking the division sends to the product selection page for the specified division. On the product grid are given the price of the product, amount, name image also an X icon. Client can remove product by clicking the X and can increase or decrease the amount using plus and minus. At the end order is confirmed by pressing confirm order button. A prompt is show that asks for the contact number than order is confirmed.

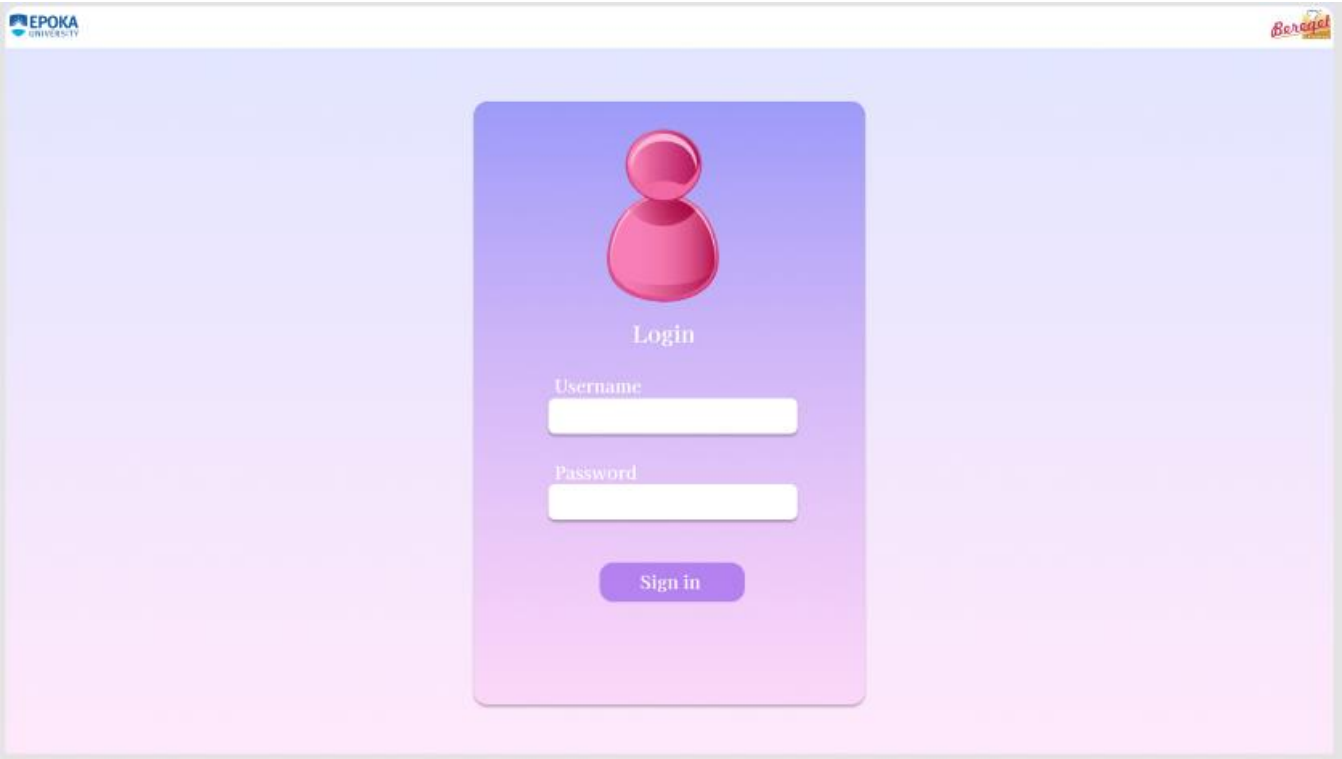
- **Seller site**

Ways that use can access the site:

- Seller enter the site by typing site name and adding /user (made this way not to interfere with user interface).
- Seller should login first to enter the site.

Pages:

- ***Login page***

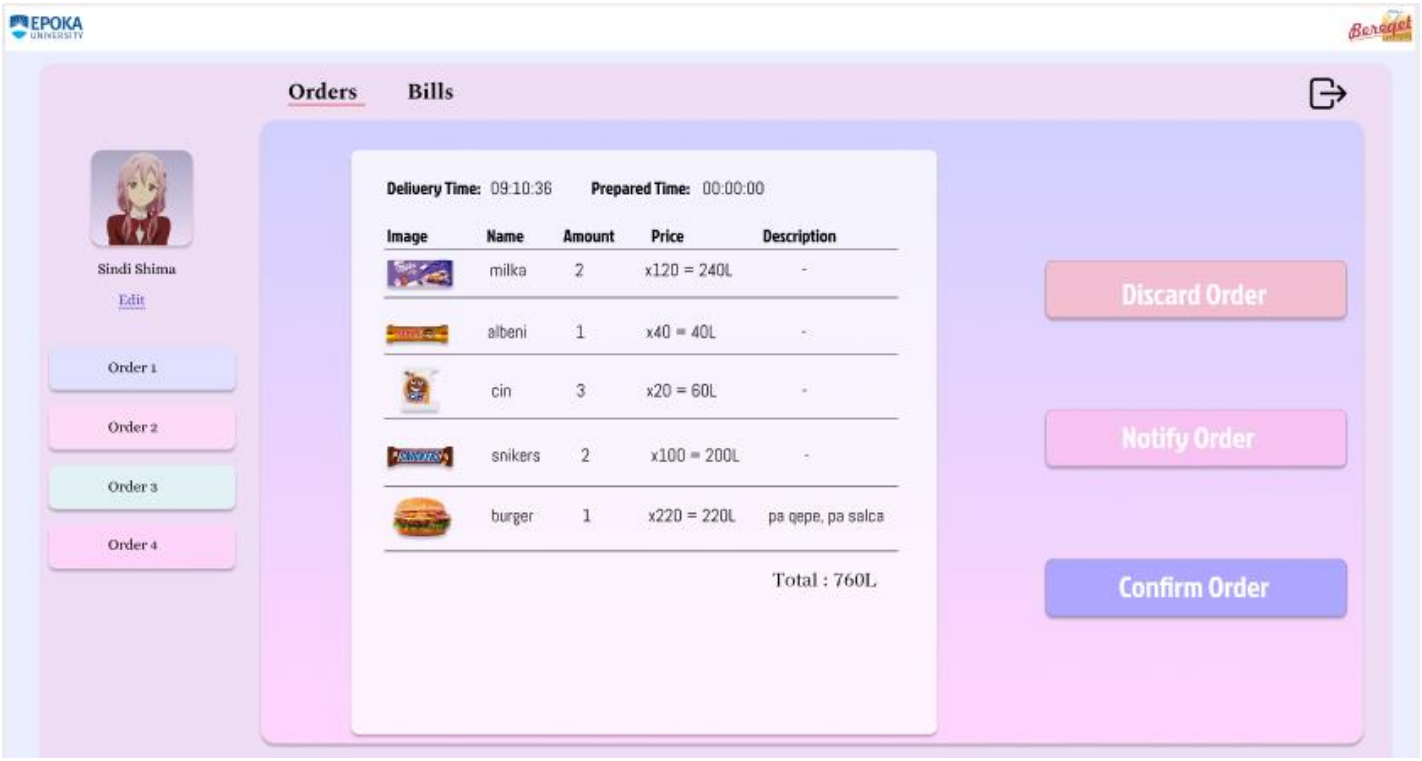


Element of this page are:

- Custom header and footer
- Enter username input field
- Enter password input filed
- Sign in

Seller type the username and the password and press the sign in button to enter the website. If validation and the credentials are not correct, seller can not enter.

- ***Order page (main page)***



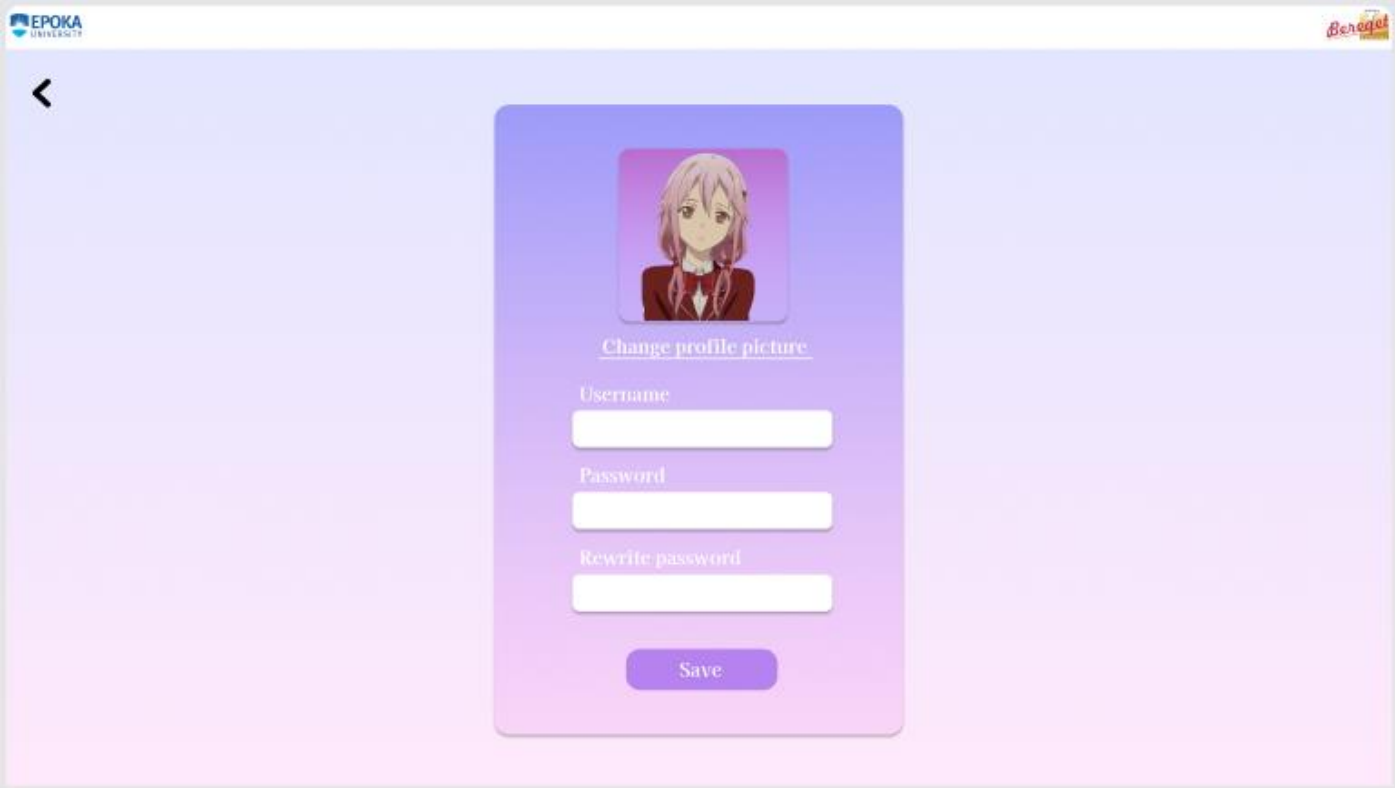
Element of this page are:

- Custom header and footer
- Nav bar
- Photo and name
- Edit link
- List orders
- Order panel
- Log out icon
- Discard, Notify and Confirm order Buttons

At the top is the nav bar that contain orders and bill. By clicking the bills, bills page will open. At the right corner is the log out icon, which is used to log out. At other corner is the seller image and the name. Lower by clicking the edit link the seller update page is opened. There are order list arranged in a scroll panel. By clicking the order, a detailed order is show that contain all products to be prepared for that order. Seller click the notify button to notify the client that his/ her order is ready. Seller click the confirm order when client has purchased the order. After that

order is removed from order list and saved as a bill. Seller click the discard order button when client do not purchase the order and again order is removed from the list but not saved as bill.

- ***Seller update page***

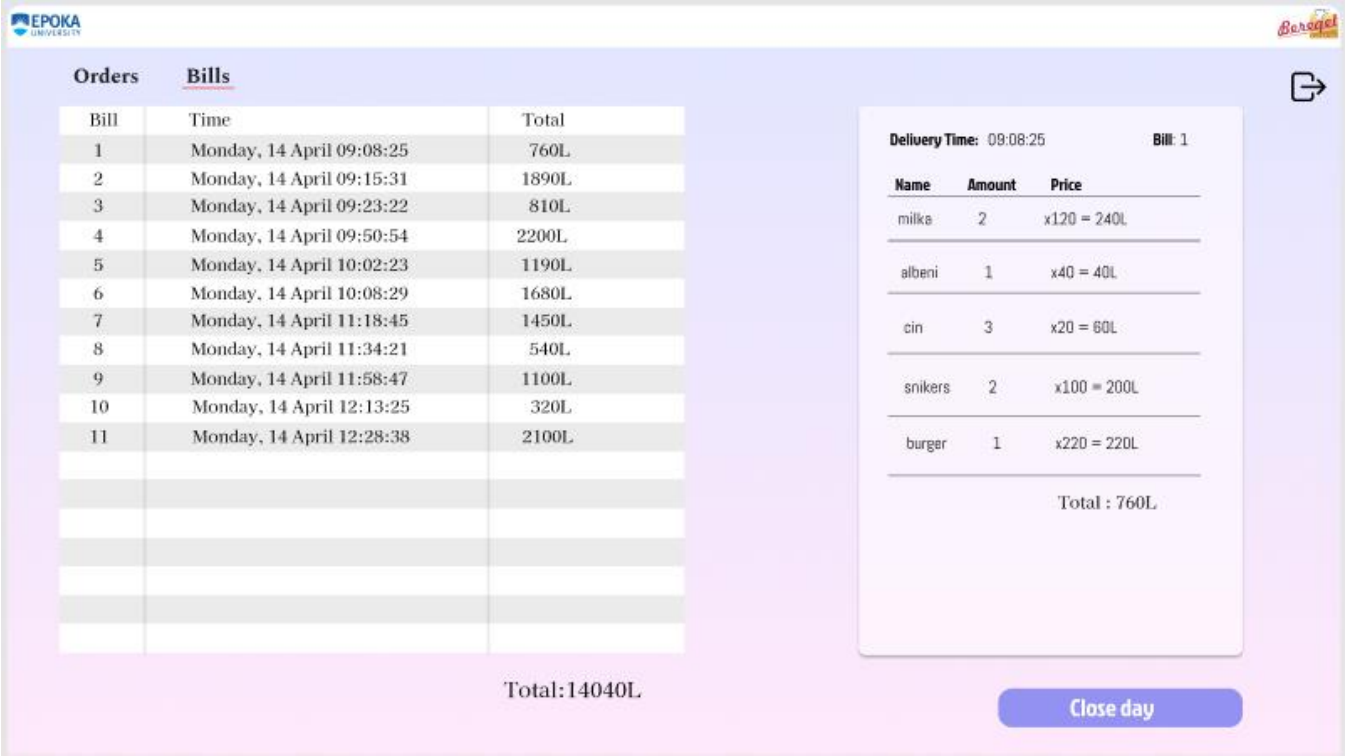


Element of this page are:

- Custom header and footer
- Turn back arrow
- Edit profile picture link
- Username, Password, Rewrite Password input field
- Save Button

Seller can turn back using the arrow icon if do not want to save change. Picture is edited using the change profile picture link. Seller complete the given input links and if data entered are valid seller is updated successfully. After updating, the seller should log in again from the beginning.

- ***Bills page***



Element of this page are:

- Custom header and footer
- Nav bar
- Log out icon
- Bills list
- Detailed view panel
- Total incomes during day
- Close day button

As explained previously the nav bar is used to pas from a page to other and icon used to log out. At the left is the list of all bills purchased by seller during a working day. Bills are visible as long as are purchased in working day. Each bill is clickable and when one is clicked, detailed view of the bill is shown. At the bottom the total incomes for today are visible. Seller end the turn by clicking Close day button. After clicking this button seller is log out and if he/ her want to login again a new working day start.

- **Administrator site**

  Ways that use can access the site:
  - Site can be accessed by typing the website url and adding /admin.
  - Admin should login first to enter site.

//TODO:

3.2.2 Usability

- Learnability
  - All users will be able to use the system less than a couple of hours because system is very user friendly and intuitively.

- Efficiency
  - System will provide user a very fast performance and users will be able to perform every operation very fast.

- Accessibility
  - Client will access the ordering site by scanning the QR code or just accessing website name through the browser.
  - Seller will need to and a “/seller” at the URL of the website. They need to enter their username and password credential to have access to their site.
  - Admin will add a “/admin” at the URL of the website. And site can be access by providing correct username and password credentials.

- Effectiveness
  - The Software will provide every user (client, seller, admin) features in a very efficient way without burdening the system and causing latency.
  - The Software will help cantina work flow to increase efficiency and make the ordering process more flexible.

3.2.3 Performance

1. Capacity

   The software will be responsible for all users at the same time. Back end will be responsible to make DB connection, interaction and reflect DB changes throw APIs. Front End will provide user interface to use the provided APIs. Database will be MYSQL server that can we are going to store locally in a web servers (EPOKA servers) or using cloud AWS (Amazon web server). Software will have a maximum size of 300 mega byte.

2. Availability

- The web app will be available 24/7.
- The web app will be available to all the users that have a pc or a mobile device connected to internet.
- The web app can be accessed from anywhere.
- The orders can be done only from 8 AM – 8 PM.

3. Latency

The latency of the web application will depend on:

- Internet connection strength.
- The efficiency of the algorithm, data structure used to retrieve data from database.
- Database data arrangement.
- Back end data manipulation.

3.2.4 Manageability/Maintainability

- Monitoring

The software will provide secure and reliable usage of all features for all the users. Every data entered into the system that make direct changes to Database will pass secure filters and validation. This is for security reason and for not causing BD malfunctioning. There are error handling scenario for every error caused by misusing of the website. There will be pop ups that inform about the errors and hints how to use properly.

- Maintenance
  - The entire software will be made of 2 separate project one for the back end (BE) and one for front end (FE)
  - The used rest API design patter of the back end will be MVC → Models related to DB entities and other model needed for arranging the software loges, other classes will be used as controllers to call the rest end points and other classes will be used as repository to implement database functionalities.
  - Back end will be a maven project were new dependencies can be added form pom.xml of maven.
  - Will follow the logic of Spring Boot architecture with auto configuration and dependencies injection.
  - FE will be maintained using React JS framework. Using dynamic component will increase usability and flexibly.
  - A health check service will be connected to our back end to keep track of every action logs or services.
  - Will use MYSQL server to monitor the well functioning. Later one maybe a cloud service will be better like AWS.

- Operations

  - Client enter website without any credential provided.
  - Seller log in with account created by the administrator.
  - Administrator log in with hard coded credentials since the first deployment.
  - Administrator change his/her credentials.
  - Admin add staff by creating sellers with unique credentials.
  - Amin remove sellers.
  - Admin update sellers will validated data.

- Admin add, remove product and new product
- Admin add categories.
- Admin check the inventory.
- Admin keep track of finances, product stock, supplies.
- Admin check bills and cancel them if are not saved correctly.
- Client is freely to choose any product he/ want.
- Client add custom preferences for cantina products.
- Client add amount, decrease amount, remove product at all in the bag preview.
- Before order confirmation user should provide contact number.
- Seller wait for order to be shown
- Seller notify user for order preparation.
- Seller discard order.
- Seller confirm order.
- Seller edit his/ her profile.
- Seller check his/ daily bills and incomes (only the bills that are purchased during a current working day).
- Seller close the day turn.

3.2.5 System Interface/Integration

- Network and Hardware Interfaces
  The web application will use a Wi-Fi, Ethernet or mobile data connection to connect to the internet. Since it is a web application implementing rest full webs service logic data connection and data communication will be over TCP protocol using HTTP and HTTPS.

- Cloud Deployment using cloud services.
  This is a consideration to be made later on if EPOKA server not provide access to host the website. AWS is a strong candida to host and monitor the software and DB. AWS provide secure and flexible hosting.

- Development operating system
  There will be two instances serving one (production) and the development instance for adding new features and resolving possible errors. Each developer will insatiate software locally so there will be provided specification how to settle locally for development. Developers computer operating system will take the responsibility (for every know operating system will be provided configuration to develop and test locally) .

3.2.6 Security

- Protection

  - Sensitive information will be encrypted.
  - Every information entered will be validated.
  - Action, queries, request methods will be sanitized to protect information and not to cause malfunctioning.
  - All users interfaces do not interfere with other to cause data leak (only admin is able to access data from seller)

- Authorization and Authentication

  - Admin and sellers should pass the password authentication to access the system.
  - Authorization for every user type are well defined and well secured in BE using Spring Security. No user can access different endpoint without role authorization. Only client instance don not have authorization (can be accessed by every user).
  - Cookies will help to improve the security,

3.2.7 Data Management

- All data will be stored in a MYSQL database.
- The database will be an ER model.
- Will be tables for user, seller, admin, products, sells, orders, bills, categories.
- Each table contain necessary attributes and attribute types.
- There will be 2 instances of database (one for production and one form development)
- User information will are temporary sored in DB as long as the order is active.
- Oher back up Serves is needed to keep data if server goes broke.
- All server interaction are made using data JPA and Hibernate (repositories interface that provide all database functionalities)
- Back End class entities will directly represent the database entities and form there are settled validation and necessary triggers.

3.2.8 Standards Compliance

- The software will provide well following of every legislative laws related to the users data safety and will guarantee reliable service on catering, online ordering and selling field.
- Every bill generated should be in the state format bill and every bill information should be connected with the cash register approved from the state in order for the transaction to be legal.

3.2.9 Portability

Web application that can be accessed from every device that has internet connection.
Real time order sending and orders preparation notification.
System can be adopted for very business that provide online ordering.

3.3 Domain Requirements

- BOO is a web application that provide three type of users, such as administrator, client and seller.
- All users are connected to the orders and products and all features are provided for those two keys entities.
- The main purpose is to digitalize the ordering process, to reduce queue at the cantina and to help cantina staff to provide more satisficing service for clients with less effort.