# TapTaze E-Commerce Website

by

**Mecit Melih HOCAOĞLU - 2020555033**

**Muhammed Mustafa SAVAR - 2020555403**

**Department of Computer Engineering**

**Çukurova University**

**10.05.2024**

# Abstract

TapTaze is a fully responsive general daily groceries e-commerce website aimed to serve as a user-friendly, convenient, modern shopping site for daily grocery needs of individuals. It is aimed to be as easy-to-use as possible with little to no confusing interfaces and also with a simple interface of modern and friendly design. Each aspect of design, development and scripting on both ends of the stack(front-end and back-end) are handmade from scratch.

# Contents

# 1- Introduction

E-Commerce is a vast virtual plane with almost infinite opportunities and lots of fields to compete in, consisting of a great number of aspects of commercial attributes such as marketing, promotion, visuals and so on. With its worldwide value is estimated to grow $47.73 trillion by 2030 and new competitors arise day by day, it can be foreseen to be a long-durable industry.

TapTaze is planned to be a real-life e-commerce website project in the industry of grocery sales. It is designed both for easy use and usefulness on the field. The website is fully responsive both for desktop and mobile devices. It consists of a market page displaying the grocery goods and also filtering options on the site header. Single product page mechanics such as greater image view of product, product description, add to cart mechanics, other products display are included also. There are both user and admin logins of the website and a shop cart. Shop cart displays the user's cart products. It allows users to remove each product separately. It also has a promotion code section that applies discounts on the total price. Admin panel is simple and easy to use with most of the vitals of the website can be controlled through.

# 2- Technologies and Tools Used in the Project

### For Design and Visuals:

- Adobe Photoshop
- Coolors Color Palette Generator

### For Front-End:

- HTML
- CSS
- React + Vite
- Font Awesome and Google Fonts services
- Axios
- Formik

### For Back-End:

- Node.Js
- Express.Js
- MongoDB
- Mongoose
- JSON Web Token (JWT)
- Bcrypt.Js
- Cors
- Postman
- Resend

# 3- Getting Started

First step to take was to plan our progress and which tools that we are going to use. The development process is separated into two aspects: Front-End and Back-End. Mecit Melih HOCAOĞLU is responsible of front-end development and Muhammed Mustafa SAVAR is of back-end. Name “Taptaze” came to our mind for its relevance in grocery terms, referring to the freshness of the products.

After the main idea was in shape, we decided to create a color palette relevant to our idea and what we came up with is shown in the figure below.

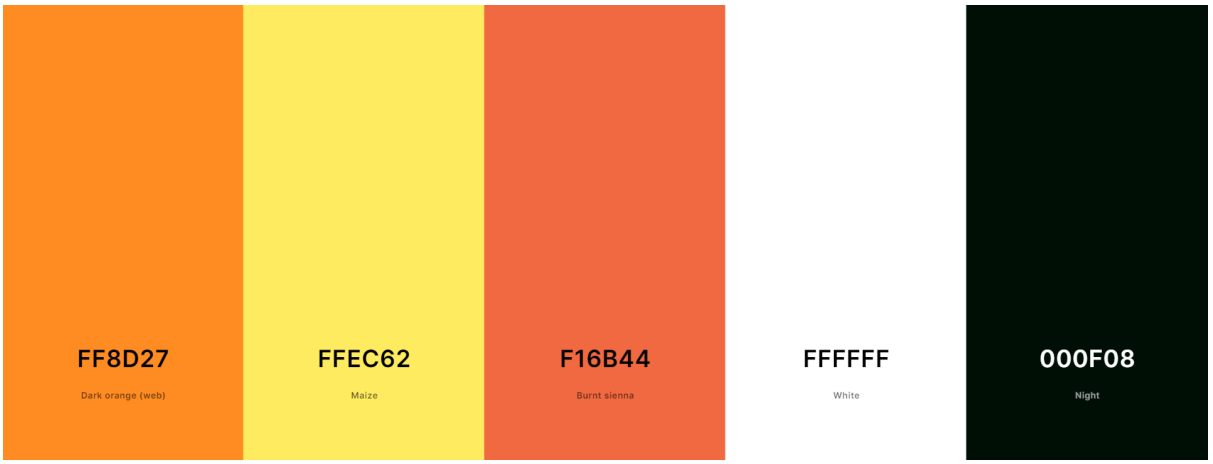



Figure 1: Color Palette

The next step was to design the characteristics of our website. It is a grocery site and has to have characteristics resembling the groceries, grocery shopping or shopping in general. Logo would summarize all these the best. Figure below is the logo we came up with designing in Adobe Photoshop.


Figure 2: Taptaze Logo

# 4- Development

We started out with sharing the workload. As mentioned before, the development is divided into two: front-end and back-end. Research was done and we came up with our technologies and tools to be used. Two separate GitHub repositories were created for both ends of the stack planned to be consolidated into one single repository at the end of the project.

## 4.1 Front-End Development

First, the environment was set to develop the front-end part of the project. React + Vite project was set up using Node Packet Manager. Folders were created for components, pages and context.

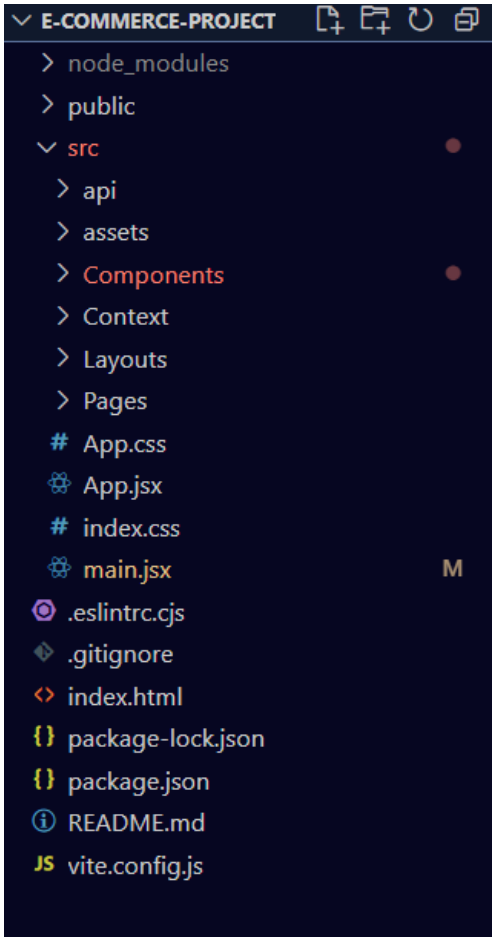



Figure 3: Document Layout

First component to create was the global header component of the website, namely Navbar.jsx consisting of three parts: logo, navigation menu and utilities(sign in button and cart). Here is the view of the header component:


Figure 4.1 and 4.2: Header and Admin Panel Header

The other components of the main page were a hero banner component at the topmost part of the body; navigating the user to filter all products. The rest of the main page consists of a popular products component an offers call to action component, and a new products component.

The other characteristic component footer consists of two main sections: main footer and a sub-footer. Main footer is a flexbox of column orientation. Main footer section has a logo and a quick menu to navigate through the site. Sub-footer contains the copyright and author information of the website.

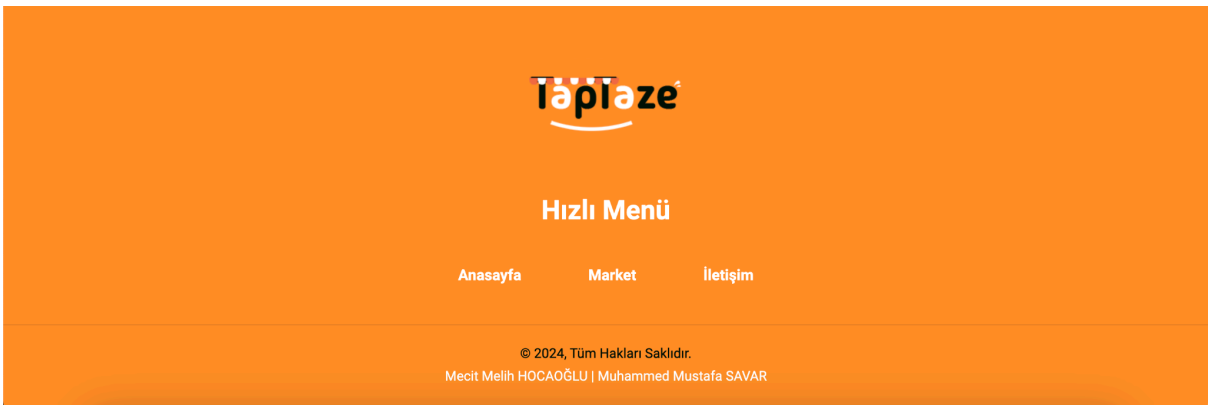



Figure 5: Footer

Next, the login page was created as a cardbox containing a form of two input elements which can be switched into a sign in component consisting of three input fields. The information acquired is sent to the back-end, processed and authorized accordingly.

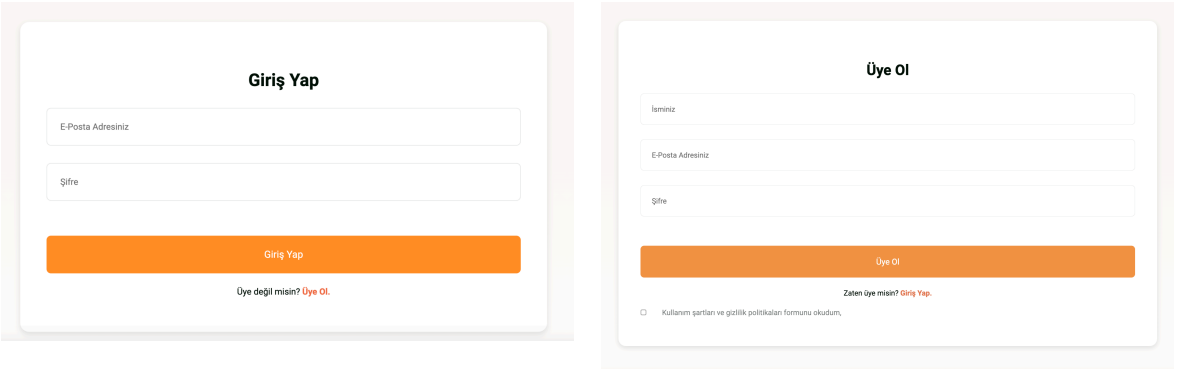



Figure 6.1 - 6.2: Login Component

A component to view the selected product in a separate page was needed so the single product page was created to do so. It is a component with two side by side(*.single-product-left* and *.single-product-right*) and containing other components underneath it(description box and related products). The left side displays the large image of the product while the right side displays the relevant information of the product such as product name, product old price and new price, short product description, and an add to cart button with a counter that can adjust the quantity of the product that is going to be added to cart.

There are also two extra components underneath the single product component which are the product description box and related products components that display products other than the selected product that is displayed currently.

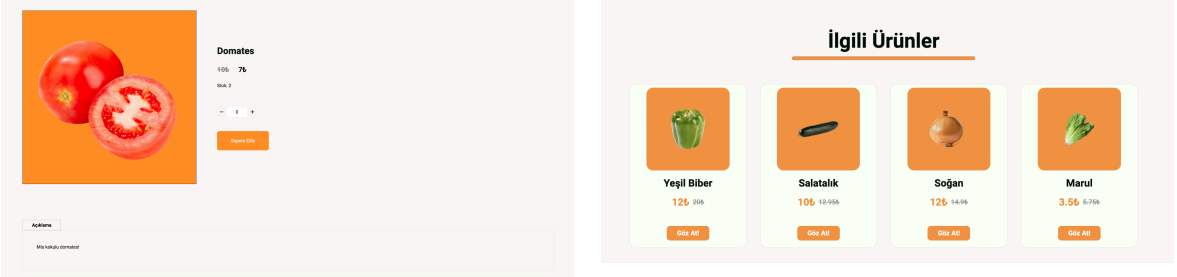



Figure 7.1 - 7.2 : Single Product Component

Cart component displays the items that are added in the cart with its thumbnail sized image, name, price, quantity, total price and a button to remove from cart.

Underneath it there is a section of a flexbox divided into two. One for final cart details such as total price of all items and a payment button and the other is an input field to apply a promotion code for discount.

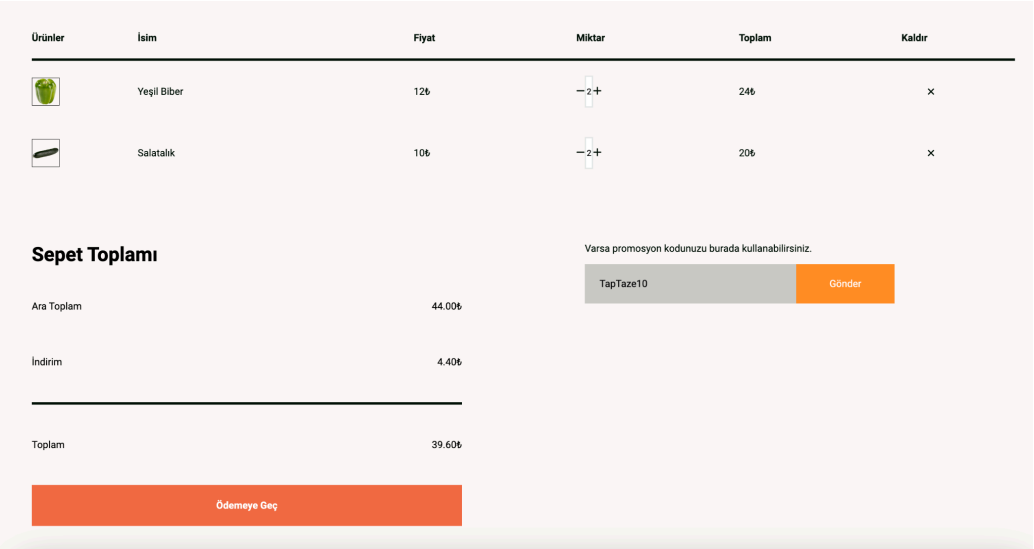



Figure 8: Product Cart

These were the main utility components in the website. The routing in the site is done using the *BrowserRouter* object. Also for data access across the components Context API was used. Here is the full list of the components in use:

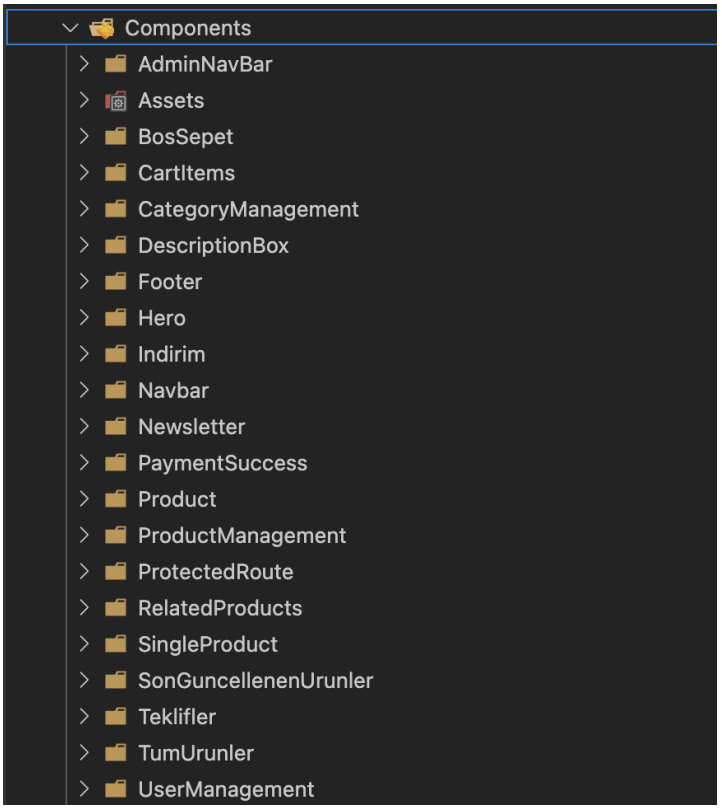



Figure 9: All Components

## 4.2 Back-End Development

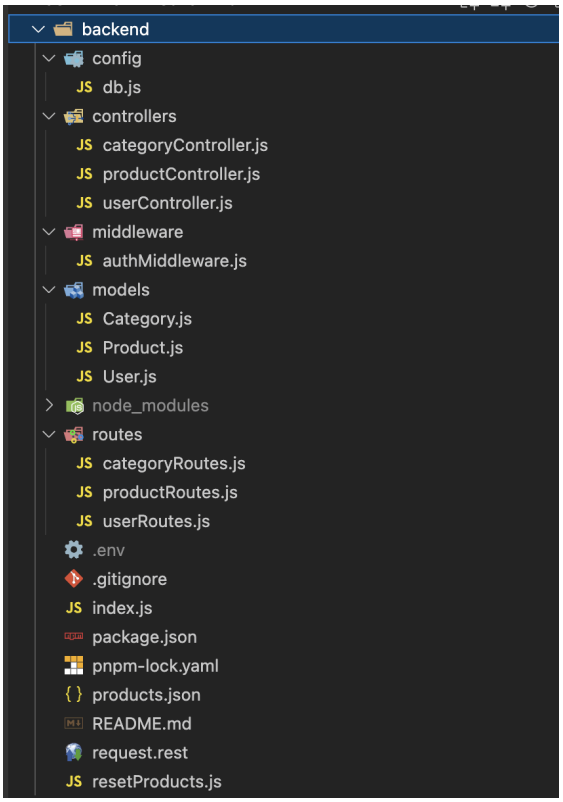



Figure 10: Backend Structure

The back-end development involved setting up a Node.js server using Express.js. MongoDB was used as the database, and Mongoose was used for object data modeling. JWT was used for authentication and authorization, while Bcrypt.js was used for hashing passwords. Axios was used for making HTTP requests, and CORS was enabled to allow cross-origin requests. Postman was used to test API endpoints, as shown in Figure 11. Resend was used to send emails. For a secure system, an .env file ( Figure 12 ) was used to store critical variables.

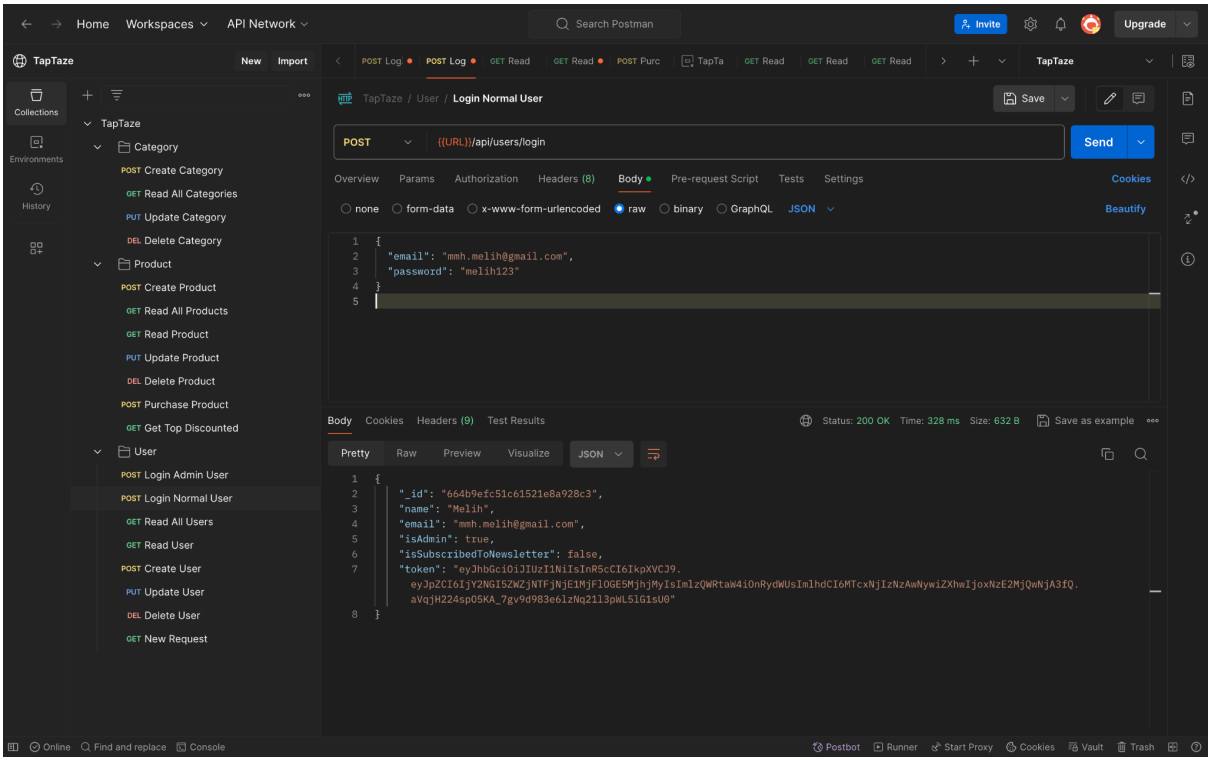



Figure 11: Postman Requests

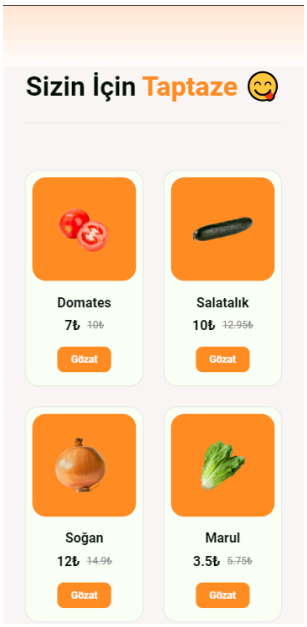

Figure 12: .env File

## Main Features Implemented:

- User authentication and authorization with JWT
- CRUD operations for products, categories, and users
- Secure password storage using Bcrypt.js
- Newsletter emails using Resend

# 5- Conclusion

TapTaze e-commerce website is created and ready for use. The visual design, website layout, scripting and back-end management are handmade. Front-end part is done using HTML for basic markup, CSS for styling, React + Vite for front-end scripting, Font Awesome for icons and Google Fonts for the fonts used in the website. Back-end part is done using NodeJS server and ExpressJS to set up the server. MongoDB for database and Mongoose for object data modeling. JWT was used for authentication and authorization and Bcrypt.JS to hash the password data. Finally, Axios was used for HTTP requests and Cors for to allow cross-origin requests. The website is functioning properly and deployed. It is in usable condition both for desktop devices and mobile devices. Here are some screenshots for both device queries:





Figures 13.1 - 13.2: Responsive Design

# 6- Links for Deployment and Review

Here is the review video YouTube link of the website:

https://www.youtube.com/watch?v=zkdGv8fMXQI

Also a deployment link of the website to examine:

https://ecommerce-project-muffafa-1.onrender.com/