metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**ISTINYE UNIVERSITY**

**FACULTY OF ENGINEERING AND NATURAL SCIENCE**

**COMPUTER ENGINEERING DEPARTMENT**

**CAPSTONE PROJECT 1**

**FINAL REPORT**

22 January 2023

**PROJECT TITLE**

E-Commerce Web Site and Mobile Application

**PROJECT MEMBERS**

Enes Taha Öz – 180722052

Ekin Şanlı – 180722015

**ADVISOR**

Asst. Prof. Buse Yılmaz

**ABSTRACT**

E-commerce is a term which refers to creating artificial environment that build a bridge between seller and customer through web sites, mobile applications.

Our project is about creating well-structured web site and mobile application by caring high quality user experience, in order to provide e-commerce service to the end user. At the end of the project, our aim is to gain more experience about software development and create easily usable, comprehensive e-commerce web site and mobile application.

**TABLE OF CONTENTS**

[LIST OF FIGURES 6](#_Toc125319997)

[INTRODUCTION 1](#_Toc125319998)

[LITERATURE REVIEW 2](#_Toc125319999)

[MATERIAL AND METHODS 3](#_Toc125320000)

[APPLICATION 3](#_Toc125320001)

[TESTS AND RESULTS 4](#_Toc125320002)

[CONCLUSION 5](#_Toc125320003)

[FUTURE WORK 6](#_Toc125320004)

[REFERENCES 7](#_Toc125320005)

LIST OF FIGURES

[Figure 1: Simple Microservice Design 4](#_Toc125320159)

INTRODUCTION

Because of the fact that human being have been suffering from COVID-19 pandemic, problem that is conventional shopping especially in crowded places being risky raised and therefore, their approach to the shopping have changed from the conventional shopping to the online shopping and these circumstances create a need for online shopping applications and a new term e-commerce.

Because of raised and continuing problem of being risky of conventional shopping because of pandemic and therefore changed shopping habits, we have decided to develop e-commerce web and mobile application.

We believe that constructing a not complicate to use e-commerce application starts from well-thought project basis. In the front-end side, we will construct project basis in a way that core components that builds UI of the applications hides the complication of the back-end of the project and serves easy usage. In the back-end side, we construct our application in a manner caring about responsiveness and speed in the whole of the project. After creating our basic, core mechanisms for our application, we will continue build new mechanisms without losing the main ideas of the project by using these core mechanisms in a cumulative manner.

We decided to choose this problem to solve because e-commerce is one of the best growing areas in the software development and hopefully, making use of e-commerce applications more basic to the general user will increase this growing rate more.

LITERATURE REVIEW

Anam Bhatti and et al claimed that: “In this article determined the ecommerce trends in coronavirus predicament as well as how imminent progress in e-commerce that might affect consumer behavior in future.” [1]

As it can be seen above that the relationship between e-commerce and coronavirus has been stated in other studies also.

For instance, one of the e-commerce web site is of MediaMarkt that is company selling technological products. And UI of the MediaMarkt site sometimes can be very complicated. Also web site has not responsive design which is to design UI responding various devices which have different screen resolutions, by changing layout of UI. [2] Our aim is to simplify such complicated situations and provide responsive design.

MATERIAL AND METHODS

First of all, we will explain the terms which are frequently used in the rest of the report. Front-end term used to indicate the UI side of the web project. All of the development done for front-end side is accomplished so as to construct UI of the web project. At the other side, back-end refers to the operations performed at the back side of the UI, such as user authentication, in order to provide usability of the web application.

In order to make collaboration more effective and provide version control, we have considered to use Git [3] version control system and GitHub [4].

We have decided to use the PostgreSQL DBMS and JavaScript language and Node.js run time environment and Express.js framework in order to develop our back-end for web site and mobile application. [5] [6] [7] We will be using microservices architecture to constitute our back-end.

Moreover, we are going to use React, which is one of the JavaScript libraries that is used for generating UI, with Styled Components, which is one of the modern styling methods in the web environment for styling the components created with React. [8] [9] We will be also using React Router in order to achieve client-side routing. [10]

For State Management in React, we will be using Redux Toolkit and so we can handle complicated state changes more easily. [11]. We have selected React Native library to develop cross platform e-commerce mobile application. [12]. Lastly, we have decided to use Jest testing framework in order to testing our back-end and front-end of our applications. [13]

We have found important to mention that some new materials can be added to handle specific problems during the project development and some materials mentioned here can be excluded according to the project development process.

APPLICATION

We have started to develop user login logic for both as web application user interface and as back-end service. In front-end side user credentials are handled by using forms in React. The user information is sent over Axios [14], which is a library used for network requesting, and uses XMLHttpRequests [15] in JavaScript, to our user login microservice. Redux have used to provide state management so that state changings in React can be more obvious and be more trackable. We have also created the “login” route in the React Router. Because of the fact that our focus is primarily in the logic of the UI, we couldn’t take many steps on the styling of the UI. In the back-end side, we have used JSON Web Tokens (JWT) [16] in order to provide authentication of the users. As it can be seen from Figure 1, we have constructed API Gateway to handle requests. For login and user operation, database which includes user information, and user services are created. Also, product services and database about products have started to be developed for future.

Graphical user interface, text, application

Description automatically generated

Figure 1: Simple Microservice Design

TESTS AND RESULTS

In the front-end side, Jest was used to provide unit testing for React Components. Because we care about user experience, we are sensitive to provide comprehensive testing environment in the front-end side. Many of the times, we encounter problems about state updates of the components however by using React DevTool and Redux DevTool, we could observe the working logic of the UI clearer and therefore we got rid of these problems more easily. In the back-end side, we haven’t done comprehensive testing yet. We have only tested basic user database operations and some of the features of the user service such as adding user, deleting user, changing user credentials.

CONCLUSION

Consequently, we have design our back-end microservice architecture and have developed our login logic for both front-end and back-end of our web application by using several libraries mentioned above. And also, we have created some of the databases for our back-end. Even though we are at the beginning of the styling of the UI, we have progressed some steps on it. As mentioned above, thanks to the development tools of React and Redux, we could observe state changes more clearly and therefore could fix bugs more easily.

FUTURE WORK

In the back-end side, we are planning to expand our back-end microservices and to integrate new microservices such as order microservice, cart microservice etc. to the our microservice architecture in consistency with our planned design and by taking care of performance. In the front-end side, we are planning to develop main page, cart logic, product page, payment page etc. and create related routes. And later on, we will start to develop our mobile application which have similarities between UI of our web application and uses same microservices with it.

REFERENCES

|  |  |
| --- | --- |
| [1] | A. Bhatti, H. Akram, H. M. Basit, A. U. Khan, S. M. R. Naqvi and M. Bilal, "E-commerce trends during COVID-19 Pandemic," *International Journal of Future Generation Communication and Networking,* vol. 13, no. 2, pp. 1449-1452, 2020. |
| [2] | "MediaMarkt - Avrupa'nın 1 Numaralı Elektronik Perakendecisi," MediaMarkt, 2013. [Online]. Available: https://www.mediamarkt.com.tr. [Accessed 19 January 2022]. |
| [3] | "Git," Git, [Online]. Available: https://git-scm.com/. [Accessed 19 January 2022]. |
| [4] | "GitHub: Let's build from here · GitHub," GitHub, Inc., [Online]. Available: https://github.com/. [Accessed 19 January 2022]. |
| [5] | "PostgreSQL: The world's most advanced open source database," PostgreSql, [Online]. Available: https://www.postgresql.org/. [Accessed 19 January 2022]. |
| [6] | Node.js Contributors, "About | Node.js," OpenJS Foundation, [Online]. Available: https://nodejs.org/en/about/. [Accessed 19 January 2022]. |
| [7] | "Express - Node.js web application framework," OpenJS Foundation, [Online]. Available: https://expressjs.com/. [Accessed 19 January 2022]. |
| [8] | "React – A JavaScript library for building user interfaces," Meta Platforms, Inc., [Online]. Available: https://reactjs.org/. [Accessed 19 January 2022]. |
| [9] | [Main page of the web site], [Online]. Available: https://styled-components.com/. [Accessed 19 January 2022]. |
| [10] | "Home v6.5.0 | React Router," Remix Software, Inc., [Online]. Available: https://reactrouter.com/en/main. [Accessed 19 January 2022]. |
| [11] | Redux documentation authors, "Getting Started | Redux Toolkit," [Online]. Available: https://redux-toolkit.js.org/introduction/getting-started. [Accessed 19 January 2022]. |
| [12] | "React Native · Learn once, write anywhere," Meta Platforms, Inc., [Online]. Available: https://reactnative.dev/. [Accessed 19 January 2022]. |
| [13] | "Jest · Delightful JavaScript Testing," Meta Platforms, Inc., [Online]. Available: https://jestjs.io/. [Accessed 19 January 2022]. |
| [14] | M. Zabriskie, "Axios," The Axios Project, [Online]. Available: https://axios-http.com/. [Accessed 22 January 2023]. |
| [15] | "XMLHttpRequest Standard," WHATWG, 25 November 2022. [Online]. Available: https://xhr.spec.whatwg.org/#interface-xmlhttprequest. [Accessed 22 January 2023]. |
| [16] | M. Jones, M. J. Bradley, P. I. N. Sakimura and N. , Internet Engineering Task Force, May 2015. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7519. [Accessed 22 January 2023]. |