JOURNAL OF DIGITAL SYSTEM DEVELOPMENT

https://e-journal.uum.edu.my/index.php/jdsd

How to cite this article:
Brett, L., Zeigler, A. & Fazea, Y. (2024). Toronto Cupcakes E-Commerce Powered by React JS Frameworks for User-Centric Design. *Journal of Digital System Developments*, *2*(2), 12-27. https://doi.org/10.32890/jdsd2024.2.2.2

TORONTO CUPCAKES E-COMMERCE POWERED BY REACT JS FRAMEWORKS FOR USER-CENTRIC DESIGN

[1]Brett Lilly, [2]Adam Zeigler & [3]Yousef Fazea

[1,2,3]Department of Computer Sciences and Electrical Engineering, One John Marshall Dr., Huntington, WV 25704

[1]Corresponding author: fazeaalnades@marshall.edu

Received: 05/04/2024 Revised: 23/07/2024 Accepted: 05/08/2024 Published: 20/10/2024

ABSTRACT

In 2013, Facebook introduced React.js to the world, and it has since gained widespread popularity as a tool for front-end development. React.js is a JavaScript library that allows developers to build user interfaces using a component-based approach. This means developers can break down complex UIs into smaller, reusable components, making it easier to manage and update the codebase. In addition, React.js utilises a virtual Document Object Model (DOM), which allows for faster rendering and better performance compared to traditional methods. This paper emphasises the vital role of user-centric design in boosting customer satisfaction and business success in e-commerce. It examines current challenges and proposes a solution, showcasing how addressing usability issues can benefit small businesses like Toronto Cupcakes by prioritising essential features and creating a responsive, mobile-friendly website prototype.

Keywords: Back-end, Front-end development. JavaScript library, Virtual DOM, React JS.

INTRODUCTION AND RELATED WORK

The literature examines the ever-changing environment of e-commerce and web-based applications, focusing on the critical role of the React.js framework. According to the Global E-commerce Forecast (Forecast, 2023) by eMarketer, around 2.56 billion people, which accounts for almost one-third of the world's population, engaged in online shopping in 2022 (Worldwide Retail E-commerce Sales, 2019). The exponential rise of the e-commerce industry is shown in its projected sales of almost $6.151 trillion in 2023, a significant increase from its 8% market share in total retail sales a decade ago (Ariella, 2022). Although 35.6% of the world's population still does not have internet access (Ariella, 2022; Kemp, 2023), the growing inclination towards online buying, driven by its flexibility, accessibility, and convenience, as highlighted in Dynata's Global Consumer Trends (Trends, 2022), emphasises the need for solid web applications (Charlton, 2019; Ward, 2023). React.js, a ubiquitous framework for server-side web apps, is essential to this transformation. The product has three main advantages: improved user satisfaction,

outstanding display quality, and simplified developer chores. React.js enhances the functionality and performance of online apps by integrating important features like routing (Bodepudi et al., 2019; Chen et al., 2019). The core application's architecture, structure, and style are maintained when the little component application applications bring this framework online (Dinku, 2022).

With its crossover application concept, the React.js framework makes combining server- and client-side websites accessible. This method allows programmers to construct robust JavaScript web applications without being concerned with the intricacies of the back end. While this approach is simple, it has to be fine-tuned for speed to meet the high expectations of e-commerce customers who want perfect online buying experiences. The second iteration of the Digitalization & Sustainability Review measures the time it takes to provide data, styles, and code to clients, emphasising how crucial it is for developers to use tactics that enhance the efficiency of React.js apps. For instance, sports-related web apps have complex tasks, such as collecting and processing many game videos and generating data during page load, which might hinder optimisation and performance. Furthermore, conventional web development approaches can produce inefficient monolithic apps; React.js's component-based design is quite different. Architectures based on cutting-edge parts also use antiquated web development methods, leading to less performant monolithic apps. React.js has revolutionised online application development by introducing substantial improvements, making it the optimal solution for creating web apps that are scalable, easy to maintain, and deliver exceptional performance. Ultimately, incorporating React.js into e-commerce and web-based apps tackles the difficulties presented by traditional technologies and capitalises on emerging patterns in online consumer behaviour. By prioritising design that focuses on the needs and preferences of users and utilising cutting-edge technological systems, businesses can improve customer happiness and attain success in the fiercely competitive e-commerce industry. This literature analysis provides a rationale for utilising React.js while emphasising earlier traditional solutions' drawbacks. This paper proceeds as follows: Introduction presented in Section 1 and Methodology in Section 2. Conclusion in Section 3.

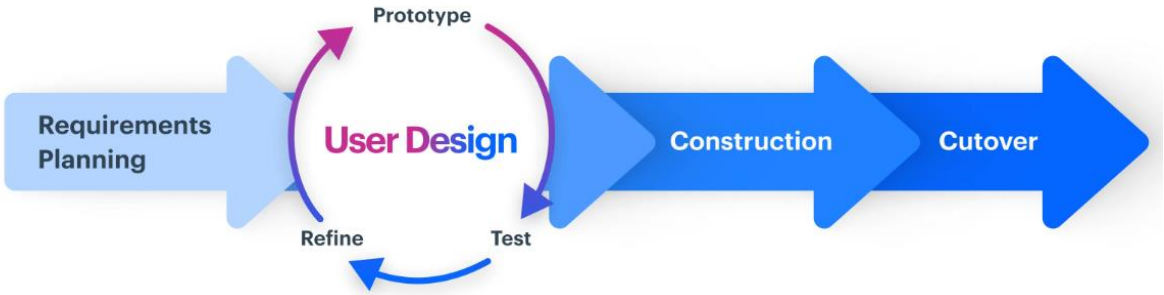
METHODOLOGY AND FRAMEWORK

The technique section outlines the specific approach used to create an e-commerce website. The Rapid Application Development (RAD) Model implements a systematic and sequential process that enables accurate prototype planning, execution, and management. This technique outlines and defines several stages: requirements planning, design, construction, and cutover. Allocating specific roles and tasks to team members reduces uncertainty and improves communication. The RAD methodology ensures quality at every level by facilitating progress tracking and stressing the need for documentation. Risk reduction, increased customer satisfaction, and cost-effectiveness are further RAD hallmarks. Therefore, it is beneficial for E-commerce endeavours to have a well-defined scope, detailed requirements, and a simplified development process to create an efficient website. Figure 1 shows the RAD approach's hierarchical development architecture. The new Toronto Cupcakes website was built using the RAD methodology, an organised approach with many important phases. The first step is to collect and record all of the requirements, which should include everything from features and functionality to performance targets and user experience standards. The next step is to create a comprehensive strategy for the design, which includes wireframes, site architecture, and user flows. This plan will also provide the groundwork for the rest of the user experience, branding, and visual design, which will be reviewed and adjusted as needed. The next step is to build the website according to the plan, test it extensively to make sure it meets all criteria, and fix any problems that come up quickly. In the verification phase, we check that everything is in order and put the website through its pace with user acceptability testing to make sure it lives up to everyone's expectations. Once the website is online and running, it will undergo continuous monitoring and maintenance to make sure it keeps working properly. Updates will be provided as necessary to improve the functionality and user experience. The redesigned Toronto Cupcakes website will closely match user

wants and expectations since the RAD technique allows for a methodical and controlled transition from need discovery to final implementation.

Figure 1

An Example of a Chart Represented in A Shaded Pattern



DESIGN AND DEVELOPMENT

Before we could begin building the new Toronto Cupcakes website, our team set out to identify all of the requirements. We separated the requirements into two groups: those that deal with the system's functionality and what it must be able to do and those that deal with the website's operation and the provision of its services, including any applicable policies. Two tables were created from the process's results: Table 1 presented the functional requirements of the project, whereas Table 2 delineated the non-functional requirements. The priority and level of relevance of each need are indicated by the same scale in both tables:

- M: Mandatory requirements (essential functions the system must fulfil)
- D: Desirable requirements (functions that are preferred to be included)
- O: Optional requirements (functions the system may include)

Table 1

Functional Requirements

No.	Requirement ID	Requirement Description	Priority
	NTCW_01	Homepage	
1	NTCW_01_01	The website must have a homepage to greet visitors.	M
2	NTCW_01_02	The homepage should include links to the register page and the login page (in addition to those found in the header).	D
3	NTCW_01_03	The homepage may contain a carousel slider. Images should showcase the company's actual products (i.e., pictures of cupcakes, not packaging).	O
	NTCW_02	Register	
4	NTCW_02_01	The website must have a register page where users can create a new account.	M

5	NTCW_02_02	The registration form must include – at a minimum – fields for username, password, email, first name, and last name.	M
6	NTCW_02_03	The register page should include a link to the login page.	D
7	NTCW_02_04	If something goes wrong when registering a new user, an error message should be printed to indicate that a failure has occurred.	D
8	NTCW_02_05	Users should be redirected to the login page whenever a new account is created successfully.	D
	NTCW_03	Login	
9	NTCW_03_01	The website must include a form that can be used to sign into existing accounts.	M
10	NTCW_03_02	The login page should include a link to the register page.	D
11	NTCW_03_03	If invalid login credentials are provided, an error message should be printed, and the user should be prompted to try again.	D
12	NTCW_03_04	Successful logins should redirect users to either the product page (if the user is a regular customer) or the admin page (if the user is an admin).	D
	NTCW_04	Shop	
13	NTCW_04_01	The website must have a shop page. This shall display the various products being sold by Toronto Cupcakes.	M
14	NTCW_04_02	Customers must have the ability to click an “Add to Cart” button to append a product to their shopping cart from the shop page. It must also be possible to remove a product that was previously added to the cart by clicking a “Remove from Cart” button.	M
15	NTCW_04_03	It should not be possible for customers to create multiple duplicate entries for Product X if their shopping cart already contains Product X.	D
	NTCW_05	Shopping Cart	
16	NTCW_05_01	To make the shopping experience more convenient, the website must include a shopping cart similar to other major online retailers.	M
17	NTCW_05_02	Customers shall be able to adjust the quantity of each product in their shopping cart. They must also have the ability to remove a product from the shopping cart (this option shall be separate from the “Remove from Cart” button available from the shop page).	M
18	NTCW_05_03	Both the subtotal for each line (unit price * qty) and the overall total (sum of all lines) must be dynamically calculated.	M
19	NTCW_05_04	The shopping cart may support an option to clear all of its contents at once.	O

	NTCW_06	Checkout	
20	NTCW_06_01	When users place their order, they should be brought to a checkout page to review details, enter payment information, and confirm the transaction.	D
21	NTCW_06_02	Guest checkout may or may not be possible. If not, users shall be required to log into an account before placing an order.	D
	NTCW_07	Contact	
22	NTCW_07_01	The website must have a page that provides customers with the company's contact information.	M
23	NTCW_07_02	The contact page must include Toronto Cupcake's business hours, physical address, phone number, and email address.	M
24	NTCW_07_03	To give customers a better sense of where the company is located (as well as its delivery radius), the contact page may contain a map.	O
	NTCW_08	Admin	
25	NTCW_08_01	The website should have an admin page where authorised personnel can perform various tasks. Examples include removing existing products (e.g., when a cupcake has to be discontinued because the ingredients used to make it are no longer attainable), updating prices (e.g., when the company offers a sale), or adding new products to the shop page.	D
	NTCW_09	About	
26	NTCW_09_01	The website should include an About page where visitors can learn more about the owner and the company's origins.	D
	NTCW_10	Special Occasions	
27	NTCW_10_01	The website may feature a page for customers who are planning special occasions (e.g., weddings, anniversaries, graduations, birthdays, corporate events, etc.). It should include referrals to various partners (e.g., photographers, florists, catering services, etc.) and combine the content of three existing pages ("Corporate Events", "Occasions", and "Resources") into one.	O
	NTCW_11	FAQs	
28	NTCW_11_01	The website may include a page for frequently asked questions. Although helpful, this isn't a critical function. We can always add it to the website later.	O

Table 2

Non-Functional Requirements

No.	Requirement ID	Requirement Description	Priority
	NTCW_12	Security	
29	NTCW_12_01	The website's database must be secured with a username and password. Neither shall be exposed to the outside world.	M
30	NTCW_12_02	User passwords must be encrypted. Even if the database is compromised, it must not be trivial for a user's account to be hijacked.	M
31	NTCW_12_03	Users must be incapable of signing into accounts without entering valid credentials.	M
32	NTCW_12_04	Regular customers must not be able to access the admin page.	M
33	NTCW_12_05	By default, all users created via the register form shall be customers (non-admin accounts).	M
	NTCW_13	Usability	
34	NTCW_13_01	The website should be easy to navigate and easy to use.	D
35	NTCW_13_02	The layout of the website should be responsive and mobile-friendly.	D
	NTCW_14	Reliability / Resiliency	
36	NTCW_14_01	User input shall be validated to ensure consistency with what the data is meant to represent. For example, email addresses must have the same format as test@email.com, phone numbers must conform to the E.164 standard, and product quantities must be numeric. Invalid input must be prevented, sanitised, and/or rejected.	M
37	NTCW_14_02	When creating a new account via the register form, the username, password, and email fields shall be required (These cannot be null in the database).	M
	NTCW_15	Performance	
38	NTCW_15_01	All website components should load within a reasonable timeframe (<2s).	D
	NTCW_16	Availability	
39	NTCW_16_01	Sign-in should not be required to view the shop page. Whilst an account may eventually be required to complete an order, we want everyone to be capable of seeing what the company has to offer. Greater accessibility encourages more sales.	M
	NTCW_17	Maintainability	

40	NTCW_17_01	MySQL shall be used to create the website's database.	M
41	NTCW_17_02	The website shall be made using a mixture of HTML, CSS, JavaScript, and React.	M

The workflow process or use case of a system can be visualised with the help of activity diagrams. Figure 2 shows the various ways a person can access a website, such as signing up, logging in, or continuing as a guest. Customers have the option to sign up using an email verification, use their existing login information, or submit their details as a guest. Problems with each route are handled independently, such as when there is a mistake with the visitor information or the email format. After a user completes one path, they can proceed with their purchase by entering shipping information or selecting in-store pickup. Designers and developers may better comprehend and control user engagement processes with the assistance of this activity diagram.

Figure 2

Login, Register, and Guest Activity Diagram

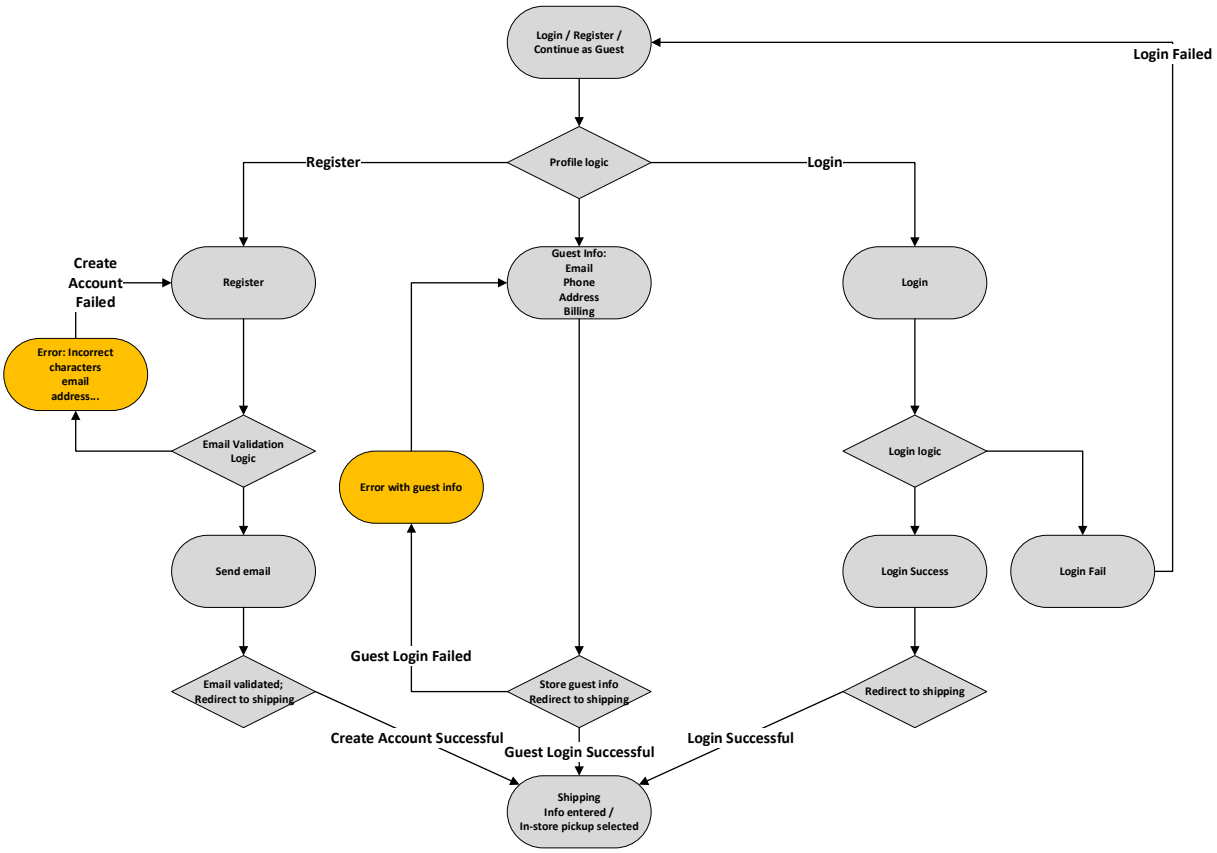
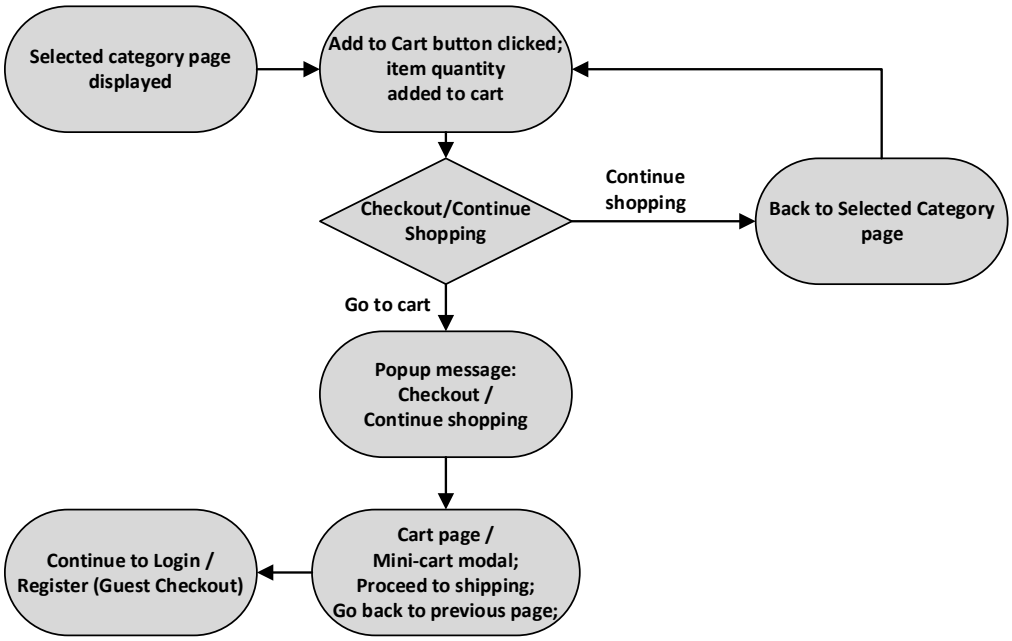


Figure 3 (a) The diagram illustrates a standard Internet purchase process. The process begins with a user perusing products on a category page and subsequently selecting items to be added to the cart. The user is given the choice to either continue browsing for items or proceed to the checkout process. If the user decides to continue shopping, they have the option to return to the category page. Alternatively, they can choose to examine their cart, make any necessary updates, and proceed with the checkout process by either logging in or registering. The flow is specifically crafted to delineate the routes a customer can follow when traversing an e-commerce site. Figure 3 (b) The process commences with the user inputting payment information, which is then subjected to error validation and accompanied by feedback and alternative options in case any errors arise. Users verify shipment information, select delivery preferences, and redeem discounts. The checkout overview page provides a comprehensive summary of the order, allowing the user

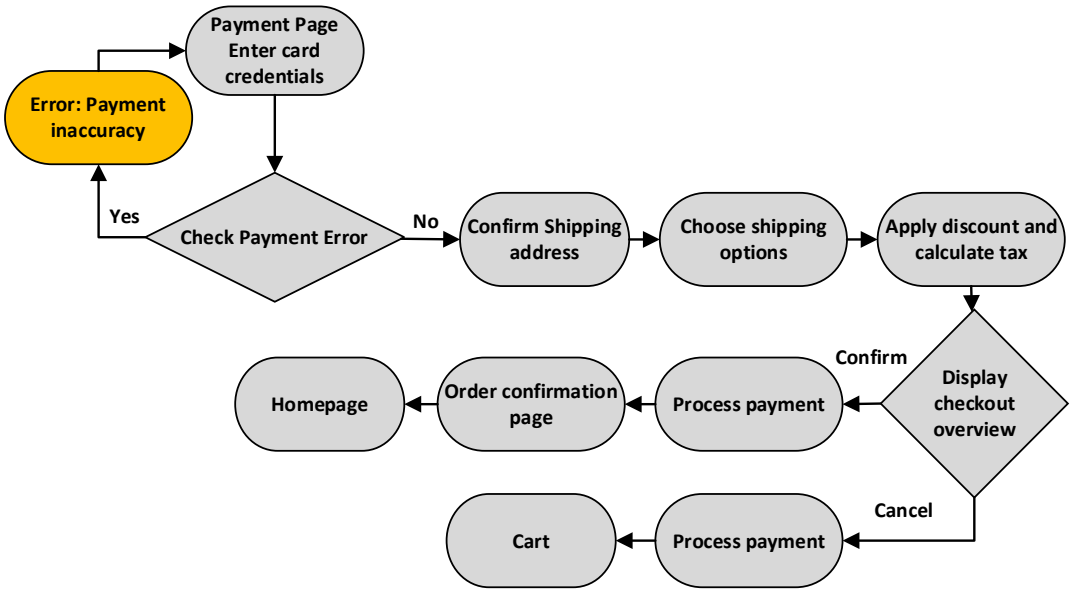
to cancel the purchase. If the user chooses to cancel, they will be redirected back to the shopping cart. Once verified, the transaction is promptly processed, the payment is securely completed, and the order details are promptly updated. Upon completion of the process, a confirmation email is dispatched to the user, who is subsequently guided to an order confirmation page and ultimately transferred to the site. This complete workflow improves the clarity and user experience during the checkout process.

Figure 3

(a) Product category



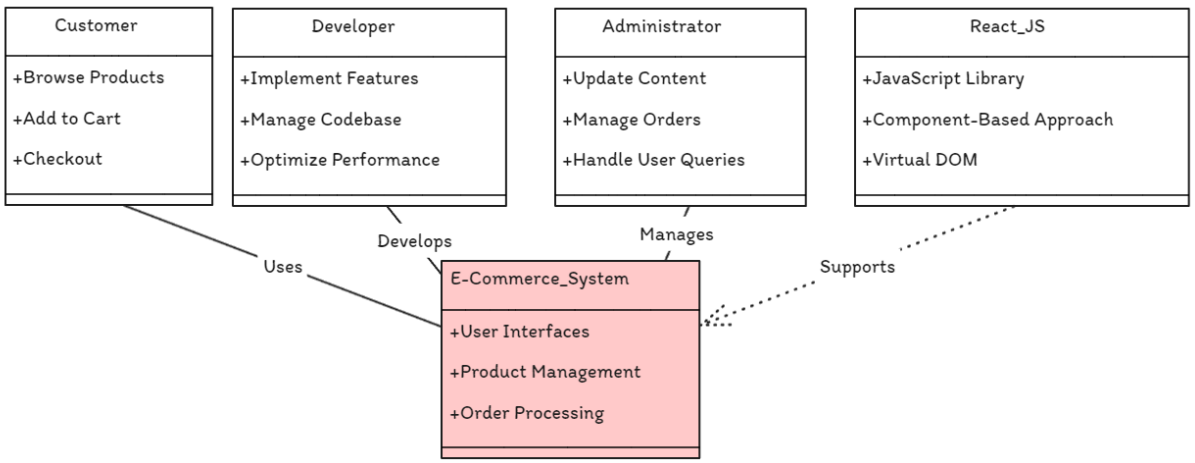
(b) Checkout



The use case diagram, as shown in Figure 4 of the e-commerce system, which is built using React.js, illustrates how various user groups interact with the system. The diagram primarily features three key actors: the Customer, Developer, and Administrator. Customers peruse, select items, and complete the purchase. Developers oversee the codebase, execute features, and enhance system performance. Administrators are responsible for updating material, overseeing orders, and addressing user inquiries.

Figure 4

Use Case Diagram



React.js enhances user interface development by utilising a component-based approach and virtual DOM, resulting in improved system support and increased efficiency. This diagram illustrates the impact of each player on the functioning and user experience of the e-commerce platform. An e-commerce database schema is shown in this class diagram in Figure 5. User submits Order, Cart contains Cart_Item, and so on; they are some of the most important entities in the system. Attributes such as USER_ID for User and PRODUCT_ID for Product are part of every entity. Figure 6 depicts the architecture and implementation of a React.js project for an online cupcake store. The project directory in Visual Studio Code is displayed in the left panel, with the client folder being highlighted. The client folder contains different components, including ShoppingCart.js. The central terminal exhibits instructions to initiate the client-server by executing the command `npm start`, while the right panel showcases the resultant web interface, including

a grid of cupcakes. This configuration showcases the creation environment and the real-time application interface.

Figure 5

Class Diagram

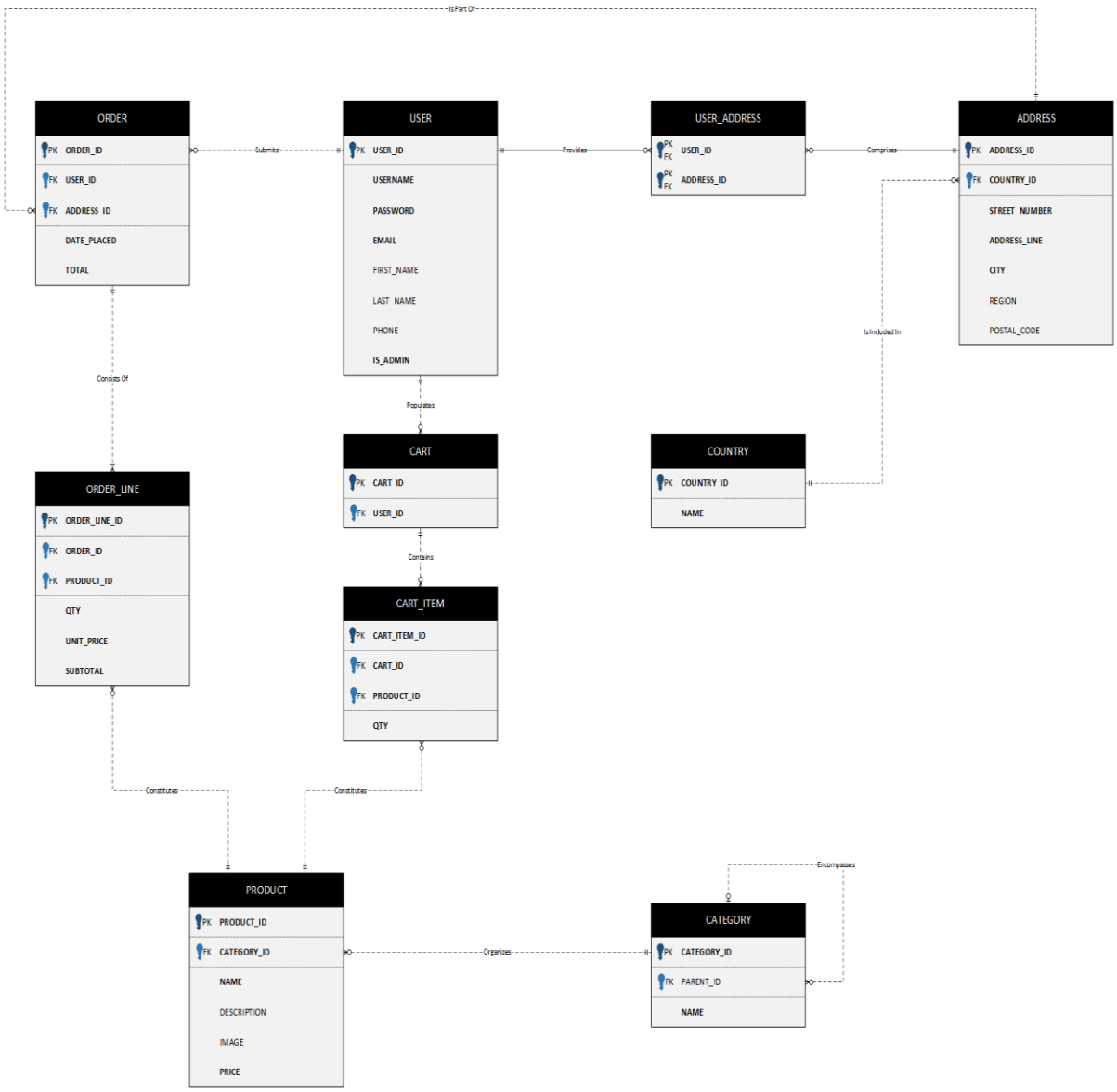


Figure 6

Architecture and implementation of a React.js project

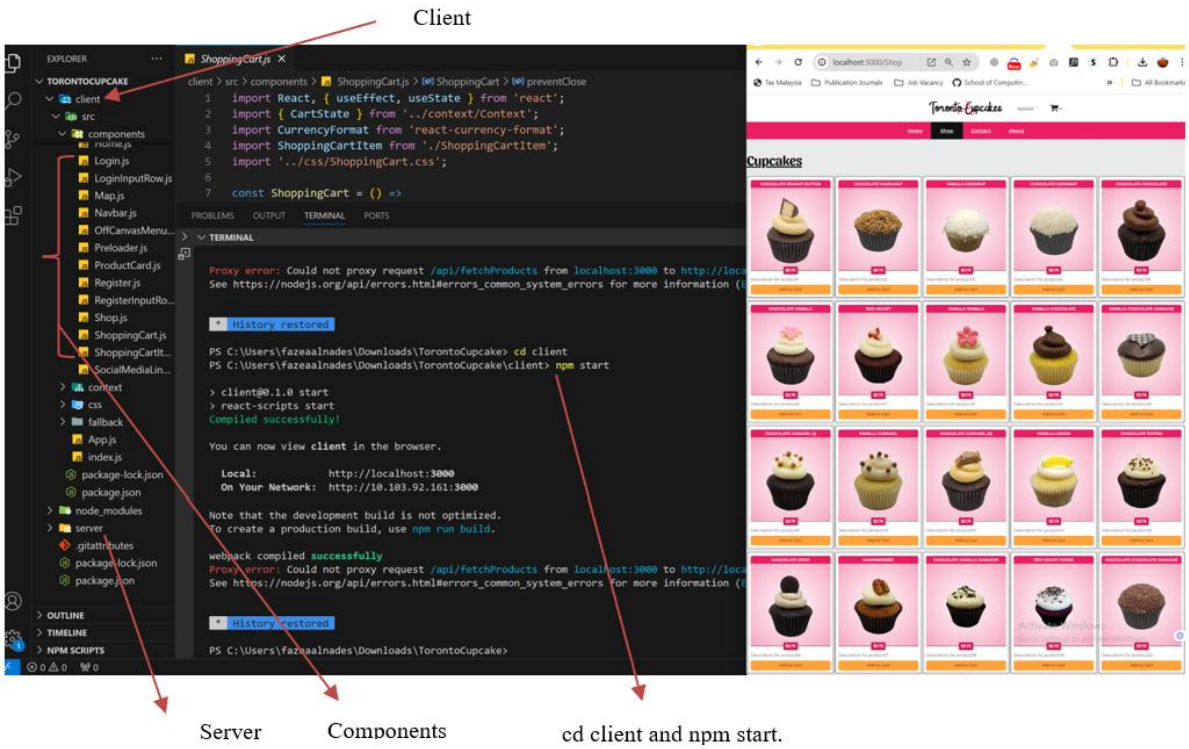


Figure 7

Login Page (With Feedback Messages)

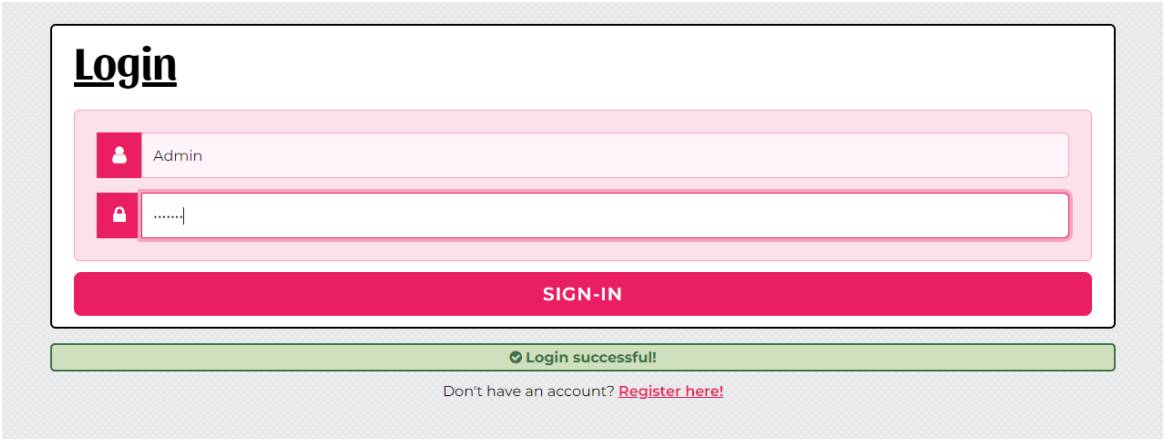
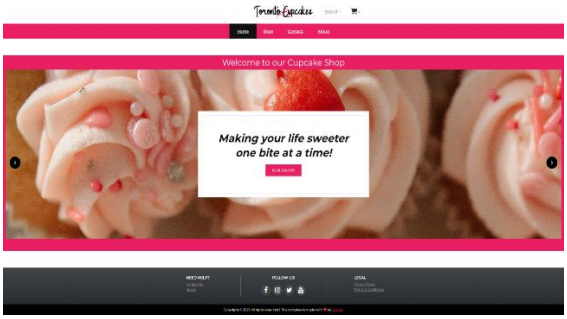


Figure 8

(a) Main menu page *(b) Shopping page*



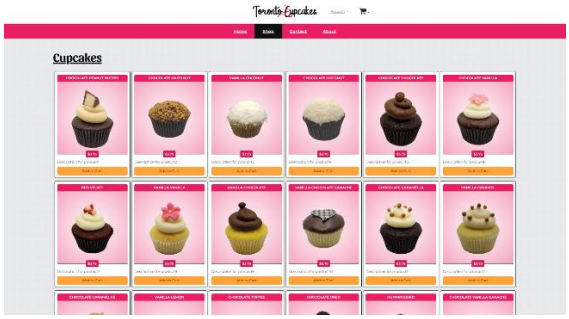
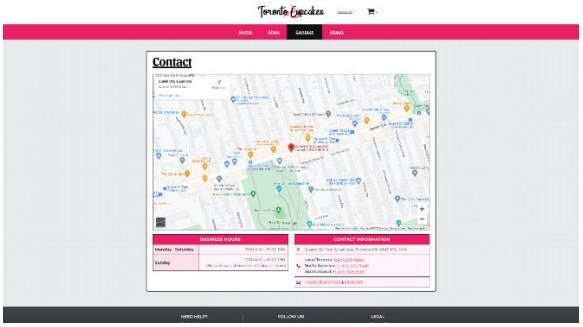


Figure 9

(a) Location and Contact Us page

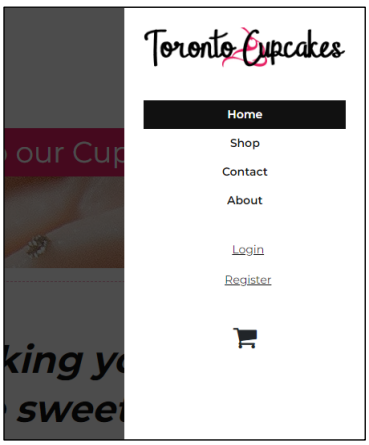


(b) About page

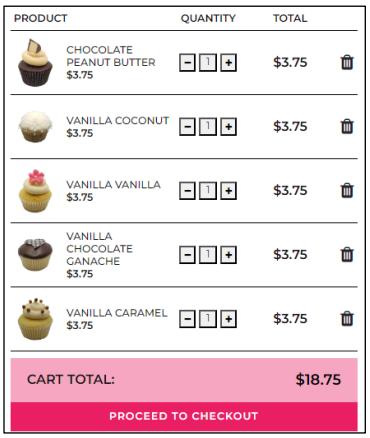
The above figures display different pages of the Toronto Cupcakes online store. Figure 7 depicts the login screen featuring feedback messages that indicate a successful login. Figures 8 (a) and 8 (b) depict the main menu page and shopping page, respectively, which showcase the website's home and product browsing interfaces. Figure 9 (a) displays the position and Contact Us page, and the About page is shown in Figure 9 (b), which provides details about the store. Figure 10 (a) shows the responsive mobile menu for the website, and Figure 10 (10) shows the shopping cart. These screenshots illustrate the graphical user interface and the process of navigating through the website.

Figure 10

(a) Mobile Menu



(b) Shopping Cart



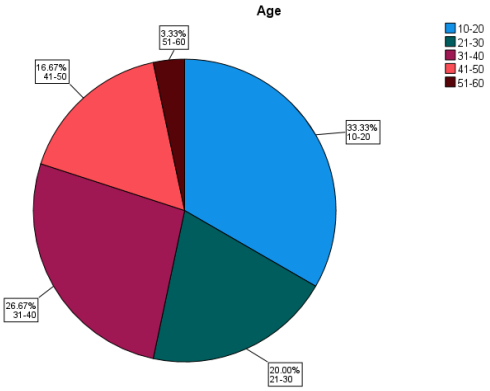
USABILITY EVALUATION

To determine whether or not the prototype was usable, information was gathered from a total of thirty different respondents. According to the data presented in Figure 11 (a), the respondents were divided into

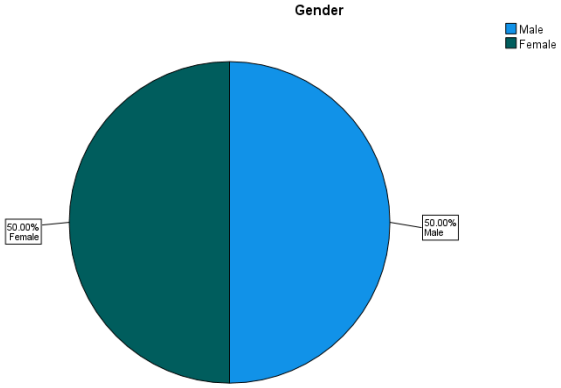
five distinct age groups. 3.33 per cent of respondents were between the ages of 10 and 20 years old (Category 1), 20 per cent of respondents were between the ages of 21 and 30 (Category 2), 26.67 per cent of respondents were between the ages of 31 and 40 (Category 3), 16.67 per cent were between the ages of 41 and 50 (Category 4), and 3.33 per cent of respondents were between the ages of 51 and 60 (Category 5). Additionally, gender was also taken into consideration. As indicated in Figure 11 (b), fifty per cent of the thirty people who responded were male, and fifty per cent were female. An evaluation of the website's navigation was carried out using a Likert scale, with a sample size of thirty individuals serving as respondents. The evaluation centred on the website's ease of navigation. As shown in Figure 12 (a), 24 respondents (80%) highly agreed that the website was easy to use, four respondents (13.3%) agreed, one respondent (3.3%) was neutral on the topic, and one respondent (3.3%) strongly opposed. These responses were given in response to the prompt.

Figure 11

(a) Age group



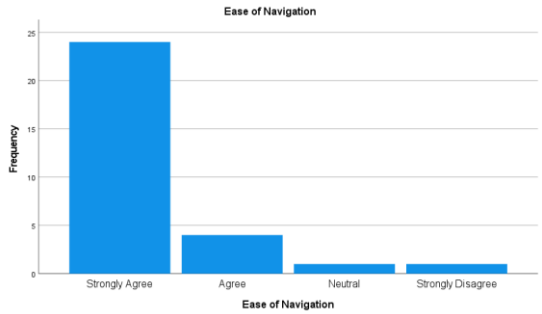
(b) Gender



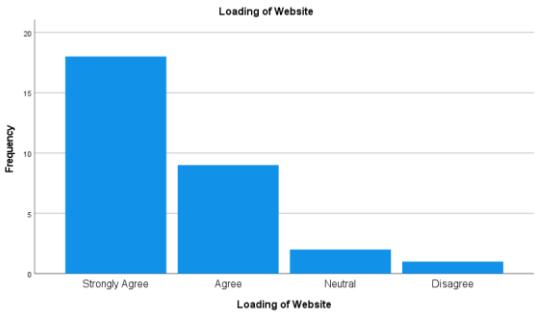
In terms of loading speed, a Likert scale was utilised to quantify the website's load times and (more importantly) the influence they have on the user experience. The sample size for this study was thirty individuals, and the question was asked about loading speed. According to Figure 12 (b), 18 respondents (60%) highly agreed that the website loaded rapidly, nine respondents (30%) agreed that it loaded quickly, two respondents (6.7%) were neutral, and one respondent (3.3%) strongly disagreed that the website loaded quickly. All of these responses are displayed in the table below.

Figure 12

(a) Ease of Navigation



(b) Loading Speed

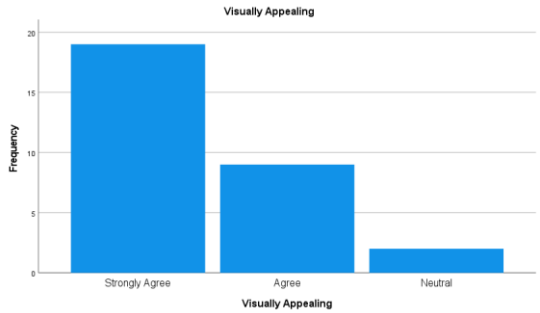


A Likert scale was utilised to evaluate the website's visual appeal. The sample size for this evaluation was thirty individuals. The results of the survey are presented in Figure 13 (a), which reveals that 19 respondents (63.3% of the total) strongly agreed that the website's appearance was beautiful, nine respondents (30%) agreed, and the remaining two respondents (6.7%) remained indifferent regarding its appearance.

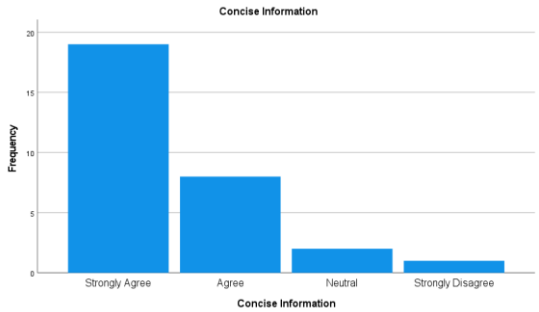
In terms of the usefulness of the material, a Likert scale was utilised to evaluate the efficacy of the website in terms of delivering information. The sample size for this evaluation was thirty individuals. As shown in Figure 13 (b), 19 respondents (63.3% of the total) highly agreed that the website successfully conveys information, eight respondents (26.7% of the total) agreed, two respondents (6.7%) were neutral, and one respondent (3.3%) strongly disagreed that the website concisely offers information.

Figure 13

(a) Visual Appealing



(b) Content Effectiveness



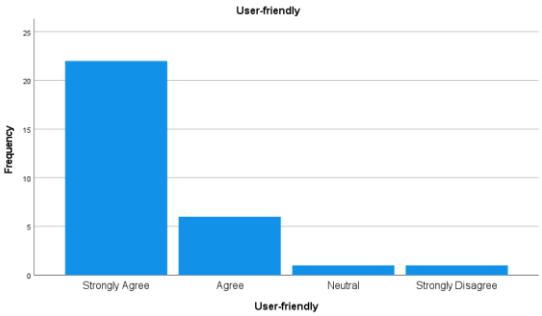
In terms of the website's user-friendliness, a Likert scale was utilised to determine the level of user-friendliness based on the responses of thirty individuals who participated in the survey. Per the findings presented in Figure 14 (a), 22 respondents (73.3% of the total) strongly agreed that the website was user-friendly, six respondents (20.0%) agreed, one respondent (3.3%) was indifferent about the website's user-friendliness, and one respondent (3.3%) strongly disagreed that the website was user-friendly.

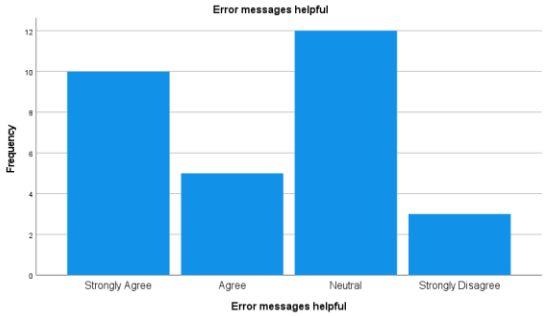
In terms of error messages, a Likert scale was utilised to explore the perceptions of the various error messages that were displayed on the website. The sample size for this investigation was thirty individuals. On the other hand, as shown in Figure 14 (b), ten respondents (33.3%) highly agreed that the website displays helpful error messages, five respondents (16.7%) agreed that they are helpful, twelve respondents (40.0%) were neutral, and three respondents (10.0%) strongly disagreed that the website displays helpful error messages.

Figure 14

(a) User-friendliness

(b) Error Messages





When all of the information that we gathered from our usability study was taken into consideration, the average score on the Likert scale was determined to be 1.5619. According to this, it appears that consumers are generally in agreement that our prototype website is simple to use, aesthetically pleasing, user-friendly, efficiently delivers information, and runs relatively well. There were a few extra comments made in response to the open-ended questions, such as "The website is not real." Expression of dissatisfaction, "No descriptions of products", "It crashed," "10/10," "Needs product descriptions," and "Some people may have visual impairments" are all possible responses. It may be deduced from the fact that the standard deviation is quite close to the meaning that a significant number of users have similar attitudes toward the website.

CONCLUSION

It is important to keep in mind that the current version of this project is still very much in the prototype stage, even though the input that was gathered from the usability study was overwhelmingly positive. Because of time limits, a great deal of compromise had to be made. The checkout page and the administrative page are the two pages that are missing the most noticeably. However, even though it is feasible to sign into an account from the login component (and this process does confirm credentials with the back-end), the present prototype does not include a method for storing the login status across other components and reflecting this information to the user on the front end. It should be emphasised that this was not "neglected" because of the difficulties; rather, our group just concentrated its efforts elsewhere to accomplish as much as we possibly could. We wanted to make sure we had a strong foundation. One may argue that support for many upcoming features is already in place. In our database, for instance, there is already a field in the USER table that is used to differentiate between consumers and administrators. The value of this variable should be sufficient to determine whether a user is authorised to view a certain component. This decision should be straightforward. In the same vein, the login status is the same. A technique that is quite like the one that we are currently utilising for the shopping cart could be described as a potential option. In addition, our database has several other tables that are pertinent to the process of checking out, such as the ADDRESS table, the ORDER table, the SHOPPING_CART table, and so on. Having said that, even though the prototype that is currently being used is primarily a proof of concept, our team has already been successful in making many enhancements to the website that was initially created for Toronto Cupcakes. Our shopping cart implementation is more convenient, navigation between pages is quicker and easier, and the style and layout of the website are mobile-friendly. These are all encouraging results from the usability study that we conducted. The resolution of these fundamental problems would be more challenging and time-consuming than the resolution of any of the aspects that are absent from the current prototype. There is little doubt that there is still room for improvement; nonetheless, it appears that progress is being directed in the appropriate direction.

ACKNOWLEDGMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

Ariella, S. (2022). 20+ Fascinating Online Shopping Statistics 2023: Online Shopping vs. In-store Shopping. www.zippia.com/advice/online-shopping-statistics/

Bodepudi, A., Reddy, M., Gutlapalli, S. S., & Mandapuram, M. (2019). Voice Recognition Systems in the Cloud Networks: Has It Reached Its Full Potential? *Asian Journal of Applied Science and Engineering*, *8*(1), 51-60.

Charlton, G. (2019). 2019 E-commerce Year in Review [Infographic]. SaleCycle. www.salecycle.com/blog/stats/2019-ecommerce-year-in-review-infographic/

Chen, S., Thaduri, U. R., & Ballamudi, V. K. R. (2019). Front-End Development in React: An Overview. *Engineering International*, *7*(2), 117-126.

Dinku, Z. (2022). React. js vs. Next. js.

Forecast, G. E.-c. (2023). As 2-Year Boom Subsides, Plenty of Bright Spots Remain. . on.emarketer.com/rs/867-SLG-901/images/eMarketer%20Global%20Ecommerce%20Forecast%20Report.pdf

Kemp, S. (2023). Digital 2023: Global Overview Report. Retrieved from datareportal.com/reports/digital-2023-global-overview-report

Kissflow, T. (2024). What is Rapid Application Development (RAD)? An Ultimate Guide for 2024. Kissflow. Retrieved May 15, 2024 from https://kissflow.com/application-development/rad/rapid-application-development/#:~:text=Rapid%20Application%20Development%2C%20or%20RAD,less%20emphasis%20on%20specific%20planning.

Trends, G. C. (2022). The New Experience Economy. www.dynata.com/content/GCT_The-New-Experience-Economy.pdf

Ward, B. (2023). 11 E-commerce Statistics & Market Size Trends 2022/23. SaleCycle. www.salecycle.com/blog/stats/ecommerce-statistics/

Worldwide Retail E-commerce Sales: eMarketer's Updated Estimates and Forecast. (2019). https://www.emarketer.com/public_media/docs/emarketer_etailwest2016_worldwide_ecommerce_report.pdf