

An Internship Report on
“NetMart – An Online Shopping Portal”

A project with
Webwide IT Solutions LLP

Submitted to
Vishwakarma Institute of Technology, Pune

In partial fulfilment of requirements for

Bachelor of Technology

In
Electronics and Telecommunication Engineering

By
Akash Bhadange
(GR NO. 12010665)

Under the guidance of
Rahul Pharande
(Chief Executive Officer at Webwide IT Solutions LLP, Pune)

Prof. (Dr.) Vaishali Jabade
(Department Faculty mentor)



Academic Year: 2023-24 Sem II
Bansilal Ramnath Agarwal Charitable Trust'
Vishwakarma Institute of Technology, Pune – 37
(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology, Pune – 37

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)



Certificate

This is to certify that the internship report titled, **NetMart – An Online Shopping Portal at Webwide IT Solutions LLP, Pune** submitted by **Akash Prakash Bhadange** (GR NO. 12010665) is a record of bonafide work carried out by him under guidance of Industry mentor **Mr. Rahul Pharande** and College mentor **Prof. Dr Vaishali Jabade** in partial fulfillment for the award of the Degree of Final Year Bachelor of Technology in **Electronic and Telecommunication**.

Signature

Industry Mentor

Mr. Rahul Pharande
(CEO, Webwide IT Solutions LLP)

Signature

Faculty Mentor

Prof. Dr Vaishali Jabade

Signature

Internship Coordinator

Prof. Dr. Ashwini Pulujkar

Signature

Head of the Department

Prof. (Dr.) Medha Wyawahare

Date: May 2024

Place: Pune

Acknowledgement

I wish to express my gratitude to Prof. (Dr.) **Rajesh Jalnekar Director, VIT Pune** for providing the facilities of the Institute and for his encouragement during the course of this work. I also express my deep gratitude to **Prof. (Dr.) Medha Wyawahare**, the Head of the Department of Electronics and Telecommunication Engineering, VIT Pune for his guidance and support.

I would like to thank my industry mentors **Mr. Rahul Pharande** and project lead **Mrs. Sushilkumar Shintre** and the **employees of Webwide IT Solutions LLP** for providing me with guidance and help on every step of the way during the course of this internship and for imparting me with invaluable knowledge and teaching me the etiquettes of a professional employee.

I would also like to gratefully acknowledge the enthusiastic supervision of my internship guide, **Prof. Dr. Vaishali Jabade** for his continuous, valuable guidance, patience, constant care, and kind encouragement throughout the internship work that made me present this internship report in an efficient manner.

Finally, I wish to thank my family members and my friends who have always been very supportive and encouraging.

Akash Bhadange

Date: May 2024

Internship offer letter

webwide

Webwide IT Solutions LLP
A1-301, Laxmiganga, Near Suncity, Anand Nagar Pune 51
+919822864888 / +919923797490

GSTN : 27ACFW1247R1ZM

Date: January 15, 2024

To,
Name: Akash Prakash Bhadange
Address: J-10 State Bank Nagar, Bibwewadi, Pune
City:
Pin: 411037
College Name: BRAC's Vishwakarma Institute of Technology
Department: DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION

Dear Akash Prakash Bhadange

We are pleased to offer a position of intern to you. Your internship will be for Six months starting from January 15, 2024

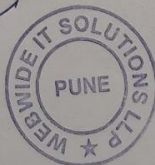
Position: Internship
(UI UX Development)
Start Date: January 15, 2024
End Date: June 15, 2024

For the position your major duties will be include:

1. Understand development language with our video tutorials and lectures.
2. Detail Study of web application
3. Understand project
4. Deliver successful project

Reporting Address: Webwide IT Solutions LLP
A1-701, 702, Mahalaxmi Nagar, opp. RMD College
Pune - 411 058
9923797490

Thank you,
With best regards,
Sushilkumar Shintre



Email : contact@webwideit.solutions

Web : www.webwideit.solutions

INDEX

INDEX.....	4
LIST OF FIGURES	6
LIST OF ABREVIATION.....	7
ABSTRACT	8
CHAPTER 1	9
1. Introduction to Web Development Technologies	9
1.1 The Significance of HTML5, CSS3, and JavaScript.....	10
1.2 Bootstrap: Empowering Responsive Design.....	11
1.3 Challenges.....	12
1.3.1 Cross-Browser Compatibility	12
1.3.2 Responsive Design and Mobile Optimization.....	12
1.3.3 Security Vulnerabilities.....	13
1.3.4 Performance Optimization.....	13
1.4 Problem Statement.....	14
1.4.1 Defining the Problem Statement.....	14
1.5 Aim and Objectives.....	15
CHAPTER 2.....	16
2. Literature Review.....	16
2.1 Literature Review - Responsive Web Design.....	17
2.2 Literature Review - PHP for Web Applications.....	18
2.3 Literature Review – PHP + MySQL.....	19
CHAPTER 3.....	22
3. Contribution.....	22
3.1 Bootstrap Framework.....	22
3.1.1 Enhancing Web Interface with Bootstrap Jumbotron.....	22
3.1.2 Bootstrap Alerts.....	23
3.1.3 Bootstrap Dropdowns for Enhanced User Experience.....	24
3.1.4 Bootstrap Media Objects in Web Design.....	25
3.1.5 Bootstrap Carousel Plugin.....	25
3.2 Integration of HTML and CSS Components in Web Development.....	26
3.2.1 Implementing HTML Tables for Data Presentation.....	26
3.2.2 HTML Forms: Building Interactive User Interfaces.....	28
3.2.3 Creative Potential of HTML5 Canvas.....	29
3.2.4 Optimizing User Interface: Essential CSS Components.....	29
3.3 Enhancing User Experience with JavaScript.....	31
3.4 Utilization of PHP and MySQL in Web Development.....	34
3.4.1 Backend Development with PHP.....	34
3.4.2 User Registration and Login Functionality.....	34
3.4.2.1 GET vs. POST: When to Use Each Method.....	35

3.4.2.2 PHP cookies.....	36
3.4.2.3 PHP sessions.....	38
3.4.3 Dynamic Content Generation.....	39
3.4.4 Database Interaction.....	40
3.4.5 Database Design and Schema Creation.....	40
3.4.6 Query Optimization and Performance Tuning.....	41
3.4.7 Security and Access Control.....	42
3.4.8 Integration of PHP and MySQL.....	43
CHAPTER 4.....	45
4 Experimentation.....	45
4.1 Methodologies.....	45
4.1.1 Prototyping.....	45
4.1.2 A/B Testing.....	46
4.1.3 Usability Testing.....	47
4.1.4 Performance testing.....	49
4.1.5 Cross-Browser and Device Testing.....	50
4.2 Experimentation Tools.....	50
4.3 Experimentation Findings.....	50
CHAPTER 5.....	52
5 Results.....	52
5.1 Performance Metrics.....	52
5.2 Impact on Project Objectives.....	52
5.3 Netmart - An Online Shopping Portal.....	53
5.4 DiscussHub Forum Website.....	
CHAPTER 6.....	57
6 Conclusion.....	57
REFERENCES.....	60

LIST OF FIGURES

Figure 1 Frontend Technologies.....	9
Figure 2 Backend Technologies.....	10
Figure 3 Bootstrap in web development.....	11
Figure 4 Different Types of Bootstrap Alerts.....	23
Figure 5 Types of Bootstrap Dropdowns.....	24
Figure 6 Bootstrap Media Object Example.....	25
Figure 7 HTML Components List.....	26
Figure 8 HTML Table Example.....	27
Figure 9 Cascading Style Sheets.....	30
Figure 10 Uses of JavaScript.....	31
Figure 11 PHP and MySQL in Web Design.....	34
Figure 12 GET and POST in PHP.....	35
Figure 13 Working of Cookie in PHP.....	36
Figure 14 PHP Session.....	38
Figure 15 Normalization in DB.....	41
Figure 16 Structure of Index in DB.....	42
Figure 17 Security in MySQL.....	43
Figure 18 Prototyping Steps in Web Development.....	45
Figure 19 A/B Testing.....	46
Figure 20 Web Design Usability.....	47
Figure 21 NetMart – Homepage.....	53
Figure 22 NetMart – Products.....	54
Figure 22 NetMart – Shoping Cart.....	54
Figure 24 NetMart – Admin Page.....	55
Figure 25 DiscusHub – Homepage.....	55
Figure 26 DiscusHub – Threads.....	56

LIST OF ABBREVIATIONS

HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
SQL	Structured Query Language
PHP	Hypertext Preprocessor
DB	Database
IoT	Internet of Things
XSS	cross-site scripting
CSRF	cross-site request forgery
API	Application programming interface
AJAX	Asynchronous JavaScript and XML
ER	Entity-Relationship
CRUD	Create, Read, Update, Delete
UI	USER INTERFACE
UX	USER XPERIENCE
KPIs	Key performance indicators
CDNs	content delivery networks
CES	Customer E-commerce Satisfaction

ABSTRACT

This internship report provides a comprehensive exploration of web development practices, focusing on the integration of PHP, MySQL, and SQL technologies. As a web developer, the objective was to gain a deep understanding of these technologies and utilize them to build robust and scalable web applications. The report begins with an in-depth analysis of PHP, highlighting its role as a server-side scripting language in dynamic web development. Various PHP methods and techniques were explored to enhance website performance, including caching mechanisms, code structuring, and error handling strategies. Additionally, the report delves into database management, examining MySQL and SQL for efficient data storage and retrieval. Database design principles such as normalization, indexing, and query optimization were studied to ensure optimal database performance and scalability.

In addition to backend technologies, the report addresses key frontend components essential for crafting engaging user experiences. Concepts such as HTML5, CSS, JavaScript, and Bootstrap were explored to design responsive and visually appealing interfaces, seamlessly integrated with backend functionalities. Throughout the internship, emphasis was placed on adopting industry best practices and standards in both backend and frontend development. Performance optimization techniques, security considerations, and code maintainability were prioritized to deliver high-quality web solutions. Moreover, collaborative tools and version control systems were utilized to streamline project management processes and facilitate efficient collaboration among team members.

This report serves as a testament to the immersive learning experience gained during the internship, offering valuable insights into the intricate workings of web development and its synergy between backend and frontend technologies. By leveraging a combination of backend and frontend skills, the goal was to create innovative and impactful web solutions that address the evolving needs of users in the digital landscape.

CHAPTER 1

1 Introduction to Web Development Technologies

The internet has transformed the way we communicate, access information, and conduct business. From its humble beginnings as a network of interconnected computers, the World Wide Web has evolved into a vast ecosystem of websites, applications, and digital services that permeate nearly every aspect of our lives. The evolution of the web has been driven by advancements in technology, standards, and user expectations. Early websites were static, consisting mainly of text and images displayed in a linear fashion. However, with the advent of web technologies such as HTML, CSS, and JavaScript, developers gained the ability to create dynamic and interactive web experiences. The rise of social media, e-commerce, and cloud computing further fueled the growth of the web, enabling users to connect, collaborate, and transact online with unprecedented ease. Web development technologies form the foundation upon which modern websites and applications are built. These technologies encompass both frontend and backend components, each serving a specific purpose in the development process.

Frontend technologies, such as HTML, CSS, and JavaScript, are responsible for creating the user interface and experience of a website. HTML (HyperText Markup Language) provides the structure and content of web pages, while CSS (Cascading Style Sheets) controls the presentation and styling of these pages. JavaScript adds interactivity and dynamic behavior to web pages, enabling features such as animations, form validation, and real-time updates.



Figure 1 Frontend Technologies

(Source: <http://surl.li/tqaho>)

On the backend, technologies like PHP, MySQL, and SQL handle data storage, retrieval, and processing. PHP (Hypertext Preprocessor) is a server-side scripting language used for generating dynamic web content and interacting with databases. MySQL is a popular relational database management system (RDBMS) used for storing and managing structured data, while SQL (Structured Query Language) provides the syntax for querying and manipulating this data.

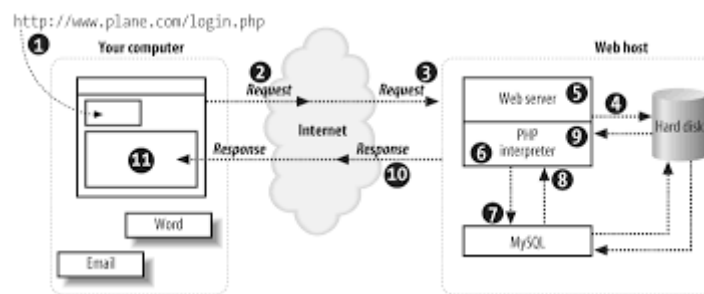


Figure 2 Backend Technologies

(Source: <http://surl.li/tqahr>)

1.1 The Significance of HTML5, CSS3, and JavaScript

HTML5, CSS3, and JavaScript represent the latest advancements in frontend web development, marking a significant leap forward in the capabilities and possibilities of web design and interactivity. HTML5 revolutionizes web development by introducing new semantic elements that provide a more meaningful and accessible structure to web pages. Additionally, HTML5 incorporates native support for audio and video playback, canvas for drawing graphics, and geolocation for location-based services, enriching the user experience and expanding the scope of web applications. Similarly, CSS3 empowers developers with a plethora of new features and techniques for styling and layout, enabling them to create visually stunning and responsive designs that adapt seamlessly to different screen sizes and devices.

Moreover, JavaScript emerges as a powerhouse in frontend development, fueled by its extensive ecosystem of frameworks and libraries. From popular frameworks like React, Angular, and Vue.js to libraries such as jQuery and lodash, JavaScript offers developers a rich toolkit for building dynamic and interactive web applications. These frameworks and libraries provide pre-built components, state management solutions, and routing capabilities, enabling developers to streamline development workflows and deliver cutting-edge user experiences that rival the performance and responsiveness of native desktop and mobile apps.

1.2 Bootstrap: Empowering Responsive Design

Bootstrap, developed by Twitter, has emerged as a cornerstone of modern frontend development, offering a comprehensive toolkit for building responsive and mobile-first web designs. Its grid system provides a flexible and customizable layout structure, allowing developers to create fluid and adaptive designs that seamlessly adapt to various screen sizes and resolutions. Furthermore, Bootstrap's extensive collection of reusable components, including navigation bars, buttons, forms, and modals, enables developers to quickly prototype and build sophisticated user interfaces with minimal effort. In addition to its layout and component capabilities, Bootstrap comes equipped with a wide range of built-in utilities and helper classes that streamline common frontend tasks and enhance development efficiency. From typography utilities for controlling font sizes and styles to spacing utilities for managing margins and padding, Bootstrap empowers developers with a robust set of tools for fine-tuning the visual appearance and behavior of their web applications. Moreover, Bootstrap's focus on accessibility and browser compatibility ensures that web applications built with the framework are accessible to users of all abilities and perform consistently across different web browsers and devices. Its adherence to responsive design principles and best practices enables developers to deliver a seamless and intuitive user experience that delights users and drives engagement.



Figure 3 Bootstrap in web development

(Source: <http://surl.li/tqahx>)

1.3 Challenges

While web development offers immense opportunities for innovation and creativity, it also presents a myriad of challenges that developers must navigate to deliver successful projects. In this chapter, we explore some of the key challenges faced by web developers and strategies for overcoming them.

1.3.1 Cross-Browser Compatibility

One of the most significant challenges in web development is ensuring cross-browser compatibility. With the multitude of web browsers available, each with its own rendering engine and quirks, ensuring that a website looks and functions consistently across different browsers can be a daunting task. Developers often encounter issues related to CSS rendering, JavaScript compatibility, and HTML layout discrepancies, which can lead to unexpected behavior and usability issues for users.

To address this challenge, developers employ various techniques such as progressive enhancement, feature detection, and browser testing. Progressive enhancement involves starting with a basic version of the website that works across all browsers and then progressively adding advanced features for modern browsers. Feature detection allows developers to detect browser capabilities and tailor the user experience accordingly, while comprehensive browser testing across different devices and browsers helps identify and resolve compatibility issues before deployment.

1.3.2 Responsive Design and Mobile Optimization

With the proliferation of smartphones and tablets, ensuring that websites are optimized for mobile devices has become essential. Responsive design techniques enable websites to adapt their layout and content dynamically based on the user's screen size and device orientation, providing a seamless user experience across desktops, tablets, and smartphones. However, designing and developing responsive websites presents its own set of challenges. Developers must consider factors such as viewport sizes, touch interactions, and network constraints when optimizing websites for mobile devices. Balancing performance and visual fidelity on smaller screens while maintaining consistency with the desktop experience requires careful planning and execution.

To overcome these challenges, developers leverage frameworks like Bootstrap and CSS media queries to create responsive layouts and fluid designs that scale gracefully across different devices. Additionally, performance optimization techniques such as lazy loading images, minification, and caching help improve the loading speed and responsiveness of mobile websites, enhancing the overall user experience on smartphones and tablets.

1.3.3 Security Vulnerabilities

Security is a paramount concern in web development, as websites are frequently targeted by malicious actors seeking to exploit vulnerabilities for nefarious purposes such as data theft, fraud, and unauthorized access. Common security vulnerabilities in web applications include cross-site scripting (XSS), SQL injection, cross-site request forgery (CSRF), and insecure authentication mechanisms. To mitigate these risks, developers must adhere to secure coding practices and adopt security measures such as input validation, output encoding, parameterized queries, and authentication and authorization controls.

1.3.4 Performance Optimization

Performance optimization is another significant challenge in web development, as users expect fast-loading websites and responsive interactions. Factors such as page load times, rendering speed, and server response times can impact the perceived performance of a website and influence user satisfaction and engagement.

To improve website performance, developers employ various techniques such as code minification, image optimization, caching, and content delivery network (CDN) integration. Minifying CSS, JavaScript, and HTML files reduces file sizes and improves loading times, while optimizing images for the web helps reduce bandwidth usage and speed up image delivery. Moreover, leveraging browser caching and CDNs allows developers to store static assets closer to users, reducing latency and improving load times for users accessing the website from different geographic locations.

1.4 Problem Statement

In this section, we delve into the specific problem statement addressed by this internship project. By identifying and articulating the challenges and opportunities within the scope of web development, we lay the groundwork for developing targeted solutions and strategies to address them effectively.

The field of web development is dynamic and ever-evolving, driven by advancements in technology, changes in user behavior, and shifts in industry trends. As such, web developers constantly face a myriad of challenges and opportunities as they strive to create compelling, functional, and user-friendly web experiences. One of the primary challenges in web development is the need to balance aesthetic appeal with functionality and performance. While users expect visually appealing designs and intuitive interfaces, they also demand fast-loading websites and seamless interactions across devices and platforms. Achieving this delicate balance requires careful planning, strategic decision-making, and meticulous attention to detail throughout the development process. Additionally, web developers must contend with an increasingly diverse and fragmented landscape of devices, browsers, and screen sizes. Ensuring cross-browser compatibility and responsive design across desktops, tablets, and smartphones poses significant technical challenges, as each platform may have its own set of rendering quirks and limitations.

1.4.1 Defining the Problem Statement

Against this backdrop, the problem statement for this internship project revolves around the following key objectives:

- 1. Developing Robust and Scalable Web Solutions:** The primary objective is to develop web solutions that are robust, scalable, and capable of meeting the evolving needs of users and stakeholders. This entails implementing best practices in frontend and backend development, optimizing performance, and ensuring compatibility across devices and browsers.

- 2. Addressing Cross-Browser Compatibility and Mobile Optimization:** Another focus area is to address the challenges of cross-browser compatibility and mobile optimization. This involves adopting responsive design techniques, leveraging frameworks like Bootstrap, and conducting thorough testing to ensure consistent and seamless experiences across different

platforms.

3. Mitigating Security Risks and Vulnerabilities: Security is a top priority, and efforts will be directed towards mitigating security risks and vulnerabilities in web applications. This includes implementing secure coding practices, performing regular security audits, and staying informed about emerging threats and vulnerabilities.

4. Enhancing User Experience and Engagement: Finally, the project aims to enhance user experience and engagement by creating intuitive, user-friendly interfaces and optimizing website performance. This involves incorporating user feedback, conducting usability testing, and iteratively refining the user interface to meet the needs and preferences of target audiences.

1.5 Aim and Objectives

The aim of this internship project was to develop comprehensive web development skills and expertise by tackling real-world challenges and opportunities in the field. Through hands-on experience, collaboration with industry professionals, and guided mentorship, the project aims to equip participants with the knowledge, skills, and confidence to excel as web developers in a competitive and rapidly evolving landscape.

The project aims to enhance technical proficiency in frontend and backend web development technologies, including HTML5, CSS3, JavaScript, PHP, MySQL, SQL, and Bootstrap. By mastering these technologies, participants will gain the skills and knowledge needed to develop robust, scalable, and user-friendly web solutions.

A key objective is to address the challenges of cross-browser compatibility and mobile optimization. This involves adopting responsive design techniques, leveraging frameworks like Bootstrap, and conducting thorough testing to ensure consistent and seamless experiences across different platforms. Security is a top priority, and efforts will be directed towards mitigating security risks and vulnerabilities in web applications. This includes implementing secure coding practices, performing regular security audits, and staying informed about emerging threats and vulnerabilities. Another objective is to enhance user experience and engagement by creating intuitive, user-friendly interfaces and optimizing website performance. This involves incorporating user feedback, conducting usability testing, and iteratively refining the user interface to meet the needs and preferences of target audiences.

CHAPTER 2

2 Literature Review

In this chapter, we review existing literature and research related to web development technologies, practices, and challenges. By synthesizing insights from academic studies, industry reports, and professional publications, we aim to gain a deeper understanding of the current state of web development and identify trends, best practices, and areas for further exploration.

The research paper [4] provides a comprehensive overview of the evolution of web design from its inception in the 1990s to the present day, emphasizing the significant changes driven by the proliferation of various online devices. It highlights the importance of responsive design in ensuring that web content is displayed in a readable and comprehensible manner across a wide range of devices, including desktops, hybrid computers, mobile devices, and wearables. The paper underscores that the primary goal of web design is not merely aesthetics but also effective web content transfer, regardless of the device used to access the content. Additionally, the research paper offers insights into future trends in web design, providing a glimpse into what can be expected in terms of advancements and changes in design practices. Overall, the paper serves as a valuable resource for understanding the historical development of web design, the importance of responsive design, and potential future directions in the field.

Ajax is a new web application development concept introduced in 2005, focusing on asynchronous interaction to enhance user experience and application responsiveness. The traditional web development model is compared to Ajax, highlighting Ajax's ability to reduce server burdens, improve user experience by minimizing page refreshes, and enhance web application interaction and availability. The research paper [5] emphasizes the significant advantages of Ajax technology, showcasing its direct, sensitive, interactive, and rich features compared to traditional methods, making it the mainstream technology in web application development. The paper specifically applies Ajax to website pass registration, demonstrating how it enhances user registration experience, enthusiasm, and functionality significantly, proving the superiority of the Ajax web application development model over traditional approaches.

The paper [6] addresses the increasing data rates in wireless networks, highlighting radio

spectrum usage as a critical issue. Spectrum sharing is acknowledged as a cost-effective solution for this issue, leading to the introduction of the concept of the Internet of Spectrum Devices (IoSDs). It discusses the development of a cloud-based architecture for IoSD over future wireless networks, aiming to facilitate efficient spectrum sharing and management.

E-commerce is rapidly expanding globally, surpassing traditional brick-and-mortar businesses. Developing countries are also witnessing a surge in e-commerce usage, although there is room for further growth. The paper [7] outlines the planning process for developing an e-commerce website, including use case determination, domain modeling, and architectural patterns. Emphasis is placed on front-end and back-end development, with a focus on relational database design. Codeigniter framework in PHP is utilized to support the MVC pattern for application development. database schema includes tables for categories, brands, products, advertisements, and user registration. Dynamic web pages are created using client and server-side scripts, with local hosting for testing before professional deployment. Measuring IS success is crucial in the IS field, with researchers extending models like the DeLone and McLean IS success model for e-commerce success measurement. Previous studies focused on e-commerce success models, such as Molla and Licker's extension of the DeLone and McLean model to include CES as a dependent variable. Challenges in evaluating e-commerce success led to the need for a new theoretical model integrating various factors proposed in prior studies, emphasizing the importance of financial performance as a dependent variable [8].

2.1 Literature Review - Responsive Web Design

The use of interactivities and real-time feedback in e-learning platforms has been shown to enhance user engagement and motivation, facilitating better learning outcomes. Combining HTML, CSS, and JavaScript in the implementation of e-learning websites can significantly increase online learning interactivities and real-time feedback, with JavaScript playing a crucial role in this process. Multimedia technology integration in e-learning platforms allows users to access learning materials from any location, promoting flexibility and accessibility in education. The research [9] conducted in this study focused on English underachievers at universities of science and technology, aiming to investigate how an E-learning website could serve as a remedial teaching aid to enhance vocabulary, phrases, and listening skills among

students.

The paper [10] discusses the importance of web page design in a world where web pages can be generated automatically. It questions the necessity of discussing web page design when technical writers can save documents as HTML and work with multimedia experts, emphasizing the significance of good web design principles. It highlights the organization of the paper, focusing on the movement from user input through forms to web responses, the use of multiple servers for web pages, and the relationship between browser structure and language definition. The paper stresses the need for platform independence, browser speed, and protocol flexibility to enhance user experience. It also mentions the differences between web site design and traditional paper design, the constraints of typefaces, and the upcoming presentation of the complete web site with specific URLs for access.

The paper [11] discusses the implementation of Media Queries, Bootstrap responsive navigation, and streaming technology for responsive web design. It provides a rational analysis of responsive web development in the current stage, focusing on adapting to various internet terminal resolutions. Implemented to create switchable navigation for small screens, supporting touch screens and CSS3 transition effects. Introduced meta tags in the head section to define viewport settings for mobile browsers, ensuring proper scaling and resolution adjustment.

2.2 Literature Review - PHP for Web Applications

The paper [12] discusses using feature analysis and heterogeneous aspects to enhance modularization in PHP-based web applications. It introduces the AOWP framework for PHP programs to demonstrate the implementation of identified application features through heterogeneous aspects. It highlights that even developers without prior knowledge of aspect-oriented programming can benefit from the advantages of aspect-oriented development, emphasizing the importance of feature analysis and the radical use of heterogeneous aspects in incremental web application development. It emphasizes the advantages of aspect-oriented programming in developing web applications incrementally, showcasing the benefits of feature analysis and the use of heterogeneous aspects to enhance modularization and organization of PHP-based web applications. The paper [13] addresses common programming errors in web applications, focusing on issues like SQL command neutralization, cross-site request forgery,

and missing authorization. It introduces a novel approach to address these challenges by demonstrating how their technique outperforms existing tools in handling vulnerabilities. It introduces a new approach for bug discovery in PHP web applications, emphasizing precise modeling of aliasing, taint analysis, and path-sensitivity to detect vulnerabilities more effectively than existing tools. It demonstrates the limitations of the Pixy tool in handling PHP code fragments and illustrates how their proposed method effectively addresses these shortcomings.

Developers with a solid understanding of core PHP concepts will find PHP frameworks easier to grasp, emphasizing the importance of core PHP knowledge in enhancing web development skills and enabling flexibility in framework selection. The decision to use a PHP framework or develop from scratch should consider project size and complexity, as frameworks are more beneficial for large, complex applications, while smaller projects may not fully utilize their advantages. It highlights the comparison of features among 23 PHP frameworks, including ORM, Code Generator, Template Engine, and CRUD Generator, to aid developers in understanding their suitability for projects [14].

The paper [15] compares PHP frameworks like Symfony2 and PhalconPHP for web development. The paper suggests expanding web system capabilities using web services like SOAP, XML-RPC, and REST, enabling machine collaboration and enhancing system functionality. Phalcon and Symfony2 were compared, with Phalcon showing superior performance, aiding developers in selecting the most efficient framework for web projects.

2.3 Literature Review – PHP + MySQL

The paper [16] focuses on the development of a website named ZhuoQi using PHP and MySQL, which are popular tools in web development due to being free and open-source. It discusses the choice of PHP as the development language over ASP.NET and JSP due to its cost-effectiveness for small projects and the absence of the need for paid development platforms like ASP.NET and JSP. The paper highlights the importance of database security, especially in websites where data integrity is crucial. It emphasizes the significance of implementing MySQL security measures to prevent data loss, duplication, or unauthorized access, which could lead to severe consequences.

The paper [17] has presented the design method and key techniques of the web-based examination system, as stated in the conclusion. The web-based examination system is suitable for conducting exams for different courses and types, offering efficiency and convenience for both students and teachers. The web-based examination system was constructed using PHP and MySQL, as detailed in the abstract. Passwords in the system are encrypted using the md5() function before being stored in the database. It is suggested that using the crypt(str [,salt]) function for encryption would enhance system security. The paper [18] discusses optimizing MySQL database systems for efficient data access without impacting server performance. It emphasizes the importance of database design in enhancing system performance and introduces the use of MySQL VIEW function for process optimization. Testing results showed that the virtual tables implemented in MySQL, such as 'viewpublikasi,' significantly improved query processes in information systems by speeding up data retrieval without the need for repeated queries. The paper [19] provides an overview of PHP and MySQL features for modern web projects, highlighting PHP's popularity due to its economy, scalability, and compatibility. It also discusses the importance of user authentication and personalization mechanisms in commercial web projects, along with an analysis of PHP properties. The paper suggests exploring the use of PHP and MySQL in education, particularly in developing and deploying student forums, to further enhance interactive learning experiences. Additionally, it hints at the potential for studying the performance evaluation models and analytics platforms influenced by new circumstances, such as the COVID-19 pandemic, to adapt to changing technological landscapes and user needs.

The paper [20] highlights a performance comparison of Node.js, Python-Web, and PHP, showing Node.js as lightweight and efficient for I/O intensive websites, while PHP is suitable for small to medium-scale applications, and Python-Web is developer-friendly for large web architectures. It emphasizes the use of benchmark and scenario tests to evaluate the technologies, with Node.js outperforming PHP in high concurrency situations, PHP struggling with large requests, and Python-Web not suitable for compute-intensive websites. The paper [21] discusses the challenges in PHP web development due to mixing business logic with database queries and presentation tags, emphasizing the importance of PHP frameworks for scalability and maintenance. It highlights the comparison of Cake PHP and CodeIgniter, suggesting Cake PHP for enterprise applications and CodeIgniter for small to medium-sized

applications, while also mentioning Symfony and Laravel as suitable frameworks for different project scales. The study does not delve into the detailed technical aspects of how the performance testing of the frameworks was conducted, which could affect the reproducibility and robustness of the findings.

The paper [22] presented an online service for conducting numerical experiments with coded apertures, aiming to provide researchers with a platform to explore a wide range of coded apertures effectively. The developed service allows users to conduct an unlimited number of experiments with various types of coded apertures, such as analyzing the behavior of coded apertures when encoding and decoding noisy flat images. Users can save reports on their experiments, compare results with previous experiments, and perform statistical analyses on the outcomes. The service is expected to find applications in research centers and laboratories involved in coded aperture-related developments, particularly in fields like gamma astronomy and gamma visors.

CHAPTER 3

3 Contribution

This chapter delineates the contribution made by this internship project within the realm of web development. By adhering to a systematic methodology and employing pertinent algorithms, alongside the utilization of block diagrams and circuit diagrams for visualization, this project significantly advances the understanding and implementation of web development practices.

3.1 Bootstrap Framework

Bootstrap, an essential frontend framework developed by Twitter, is pivotal to modern web development. Its comprehensive set of pre-designed components, styles, and JavaScript plugins streamline the development process, enabling developers to create responsive and visually appealing web interfaces with ease.

One significant contribution of this internship project lies in its exploration and utilization of Bootstrap for frontend development. By leveraging Bootstrap's grid system, reusable components, and built-in utilities, the project achieved remarkable advancements in creating consistent and user-friendly web layouts across various devices and screen sizes. Moreover, Bootstrap facilitated the implementation of responsive design principles, ensuring that web applications adapt seamlessly to different viewport sizes and orientations. This responsiveness enhances user experience by providing optimal readability and navigation, regardless of the device used to access the application.

3.1.1 Enhancing Web Interface with Bootstrap Jumbotron

A Jumbotron, also known as a hero unit, is a prominent, large-scale element typically found at the top of a webpage. It's designed to capture the user's attention and convey important information or messages. Jumbotrons often feature a bold title, descriptive text, and sometimes a call-to-action button. Bootstrap, a popular front-end framework, provides a pre-styled Jumbotron component that makes it easy for developers to create visually appealing hero sections without having to write extensive CSS from scratch.

Jumbotrons are commonly used in various scenarios, including:

- **Introduction:** They can introduce the website or highlight its main purpose.
- **Promotions:** Jumbotrons are effective for promoting products, services, or special offers.

- Events: They can be used to announce upcoming events or conferences.
- Calls to Action: Jumbotrons often feature prominent buttons or links encouraging users to take specific actions, such as signing up for a newsletter or exploring a product.

3.1.2 Bootstrap Alerts

Bootstrap alerts are another handy component provided by the Bootstrap framework. They allow developers to display important messages, notifications, or alerts to users in a visually appealing and user-friendly manner. Bootstrap alerts are simple components used to convey important information to users. They typically appear as colored boxes with text and an optional close button. Alerts can be used for various purposes, such as displaying success messages, warnings, errors, or informational messages.

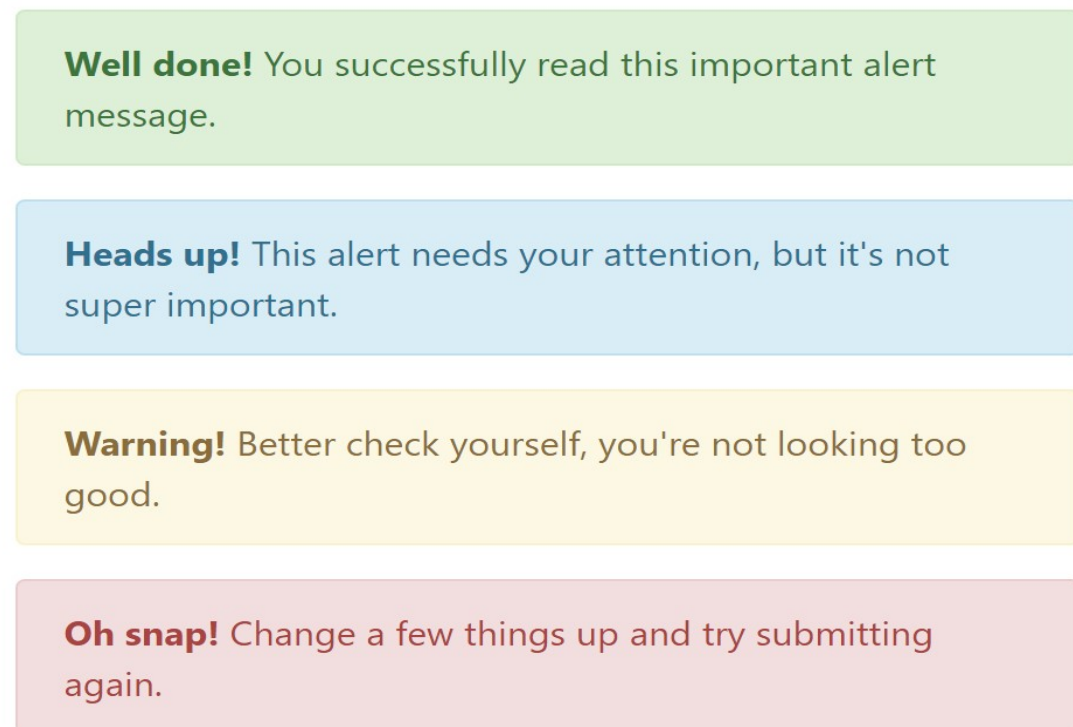


Figure 4 Different Types of Bootstrap Alerts

(Source: <https://shorturl.at/bwR17>)

Bootstrap alerts are versatile and easy-to-use components for displaying important messages or notifications to users. With their variety of styles and optional dismissal functionality, they help enhance the user experience and communicate information effectively on websites or web applications.

3.1.3 Bootstrap Dropdowns for Enhanced User Experience

Bootstrap Dropdowns provide a convenient and customizable way to present a list of options to users. They typically consist of a button or link that, when clicked, reveals a dropdown menu containing selectable items. Dropdowns are commonly used for navigation menus, form inputs, and various interactive elements on websites and web applications.

Key Features of Bootstrap Dropdowns:

- **Customization:** Bootstrap Dropdowns can be customized extensively to match the design and functionality requirements of your project. You can change the appearance, size, alignment, and behavior of dropdowns using Bootstrap's CSS classes and JavaScript components.
- **Accessibility:** Bootstrap ensures that Dropdowns are accessible to all users, including those who rely on assistive technologies like screen readers or keyboard navigation. Proper HTML structure and ARIA attributes are utilized to enhance accessibility.
- **Responsive Design:** Dropdowns are designed to be responsive, meaning they adapt gracefully to different screen sizes and devices. Whether viewed on a desktop, tablet, or smartphone, Dropdowns maintain usability and readability.
- **Variety of Options:** Bootstrap provides various types of Dropdowns, including basic Dropdowns, Dropdown buttons, split Dropdown buttons, and dropdowns with headers or dividers. This flexibility allows developers to choose the most suitable dropdown style for their specific use case.

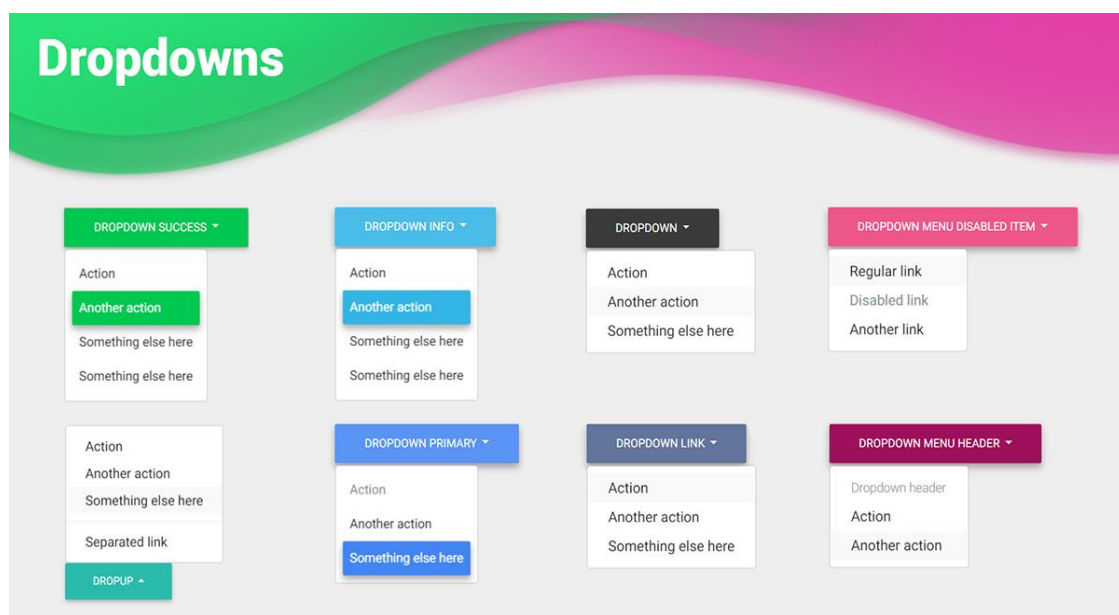


Figure 5 Types of Bootstrap Dropdowns

(Source: <https://shorturl.at/bwR17>)

3.1.4 Bootstrap Media Objects in Web Design

Bootstrap Media Objects provide a flexible way to align media (such as images or videos) alongside textual content in a consistent layout. They consist of an image or video thumbnail on one side and text content on the other. Media Objects are responsive and can adapt to different screen sizes, making them suitable for use in various contexts.

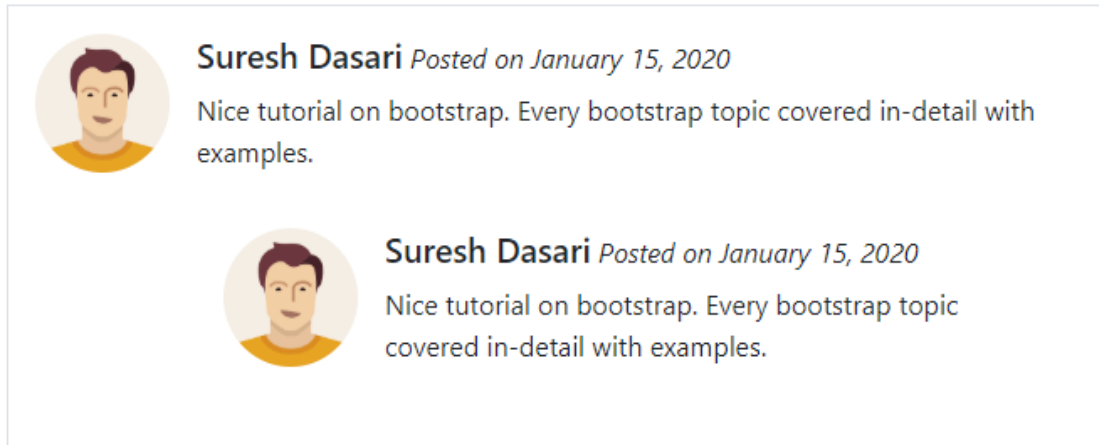


Figure 6 Bootstrap Media Object Example

3.1.5 Bootstrap Carousel Plugin

The Bootstrap Carousel Plugin provides a flexible and customizable carousel component that allows users to cycle through a series of slides. Each slide can contain images, videos, text, or any other HTML content. Carousels can be set to automatically transition between slides or controlled manually by the user. Carousels created with the Bootstrap Carousel Plugin are responsive by default, meaning they adapt smoothly to different screen sizes and devices. This ensures an optimal viewing experience across desktops, tablets, and smartphones.

The plugin supports various slide transition effects, such as slide, fade, and zoom. Developers can choose the desired transition effect to create visually engaging carousels that capture users' attention. Bootstrap Carousel provides built-in navigation controls, including previous and next arrows, as well as optional indicators to indicate the current slide. These controls allow users to navigate through the carousel easily. Carousels can be configured to automatically transition between slides at a specified interval. This autoplay feature is useful for creating self-playing slideshows or presentations. The Bootstrap Carousel Plugin offers a wide range of customization options, allowing developers to customize the appearance and behavior of carousels to fit their specific design requirements. Customization options include slide interval, animation speed, and more.

3.2 Integration of HTML and CSS Components in Web Development

Within the scope of this internship project, HTML5 played a fundamental role in shaping the structure and content of web pages. Leveraging the latest features and elements offered by HTML5, the project aimed to enhance the semantic clarity, accessibility, and interactivity of web interfaces. HTML5 introduced semantic elements such as `<header>`, `<nav>`, `<main>`, `<section>`, `<article>`, and `<footer>`, which provide a clearer and more meaningful structure to web documents. By appropriately incorporating these elements, the project enhanced the accessibility and SEO-friendliness of web pages, improving usability and search engine rankings.



Figure 7 HTML Components List

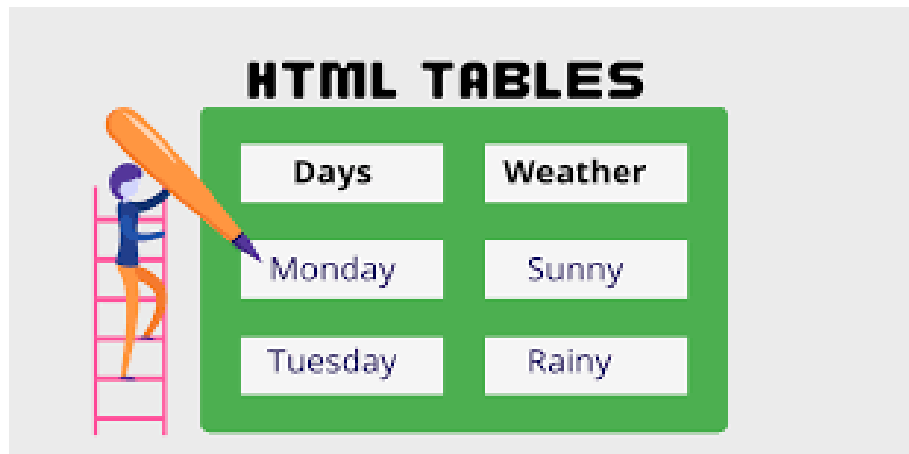
(Source: <https://rb.gy/32vlss>)

HTML5 introduced native support for multimedia content, including `<audio>` and `<video>` elements, eliminating the need for third-party plugins such as Adobe Flash. These elements were utilized to embed audio and video files directly into web pages, enhancing the richness and interactivity of the user experience.

3.2.1 Implementing HTML Tables for Data Presentation

HTML tables are fundamental components for organizing and presenting tabular data on web pages. Within the context of this internship project, HTML tables were utilized to

effectively display structured information in a clear and accessible manner.



Days	Weather
Monday	Sunny
Tuesday	Rainy

Figure 8 HTML Table Example

(Source: <https://shorturl.at/ejlqO>)

Several key features and practices related to HTML tables were employed:

1. Basic Table Structure:

HTML tables consist of rows (`<tr>`) and columns (`<td>`) arranged in a grid-like structure. The project adhered to a standard table layout, with each row representing a distinct data entry and each column containing specific attributes or values associated with the entry.

2. Table Headers:

Table headers (`<th>`) were used to define column headings, providing context and clarity to the data presented in each column. By using `<th>` elements within the first row (`<tr>`) of the table, the project ensured that column headers were visually distinguished from data cells and properly aligned with their respective columns.

3. Table Caption:

HTML tables allow for the inclusion of a table caption (`<caption>`) to provide a brief description or title for the table. The project utilized table captions to provide context and summarize the content of the table, enhancing its accessibility and usability for users.

4. Table Styling and Formatting:

CSS (Cascading Style Sheets) was employed to apply styling and formatting to HTML tables, including adjustments to cell padding, border thickness, text alignment, and background colors. By carefully designing the visual presentation of tables, the project aimed to improve readability and aesthetics while maintaining consistency with overall design themes.

5. Responsive Tables:

To ensure optimal viewing experiences across different devices and screen sizes, the

project implemented responsive table designs using CSS media queries and viewport units. Responsive tables dynamically adjust their layout and appearance based on the available screen space, allowing users to comfortably view and interact with table content on smartphones, tablets, and desktops.

6. Accessibility Considerations:

Accessibility features such as ``scope``, ``headers``, and ``aria-label`` attributes were incorporated into HTML tables to enhance accessibility for users with disabilities. These attributes improve screen reader compatibility and navigation, ensuring that all users can access and interpret table content effectively.

3.2.2 HTML Forms: Building Interactive User Interfaces

HTML forms are essential components of web development, facilitating user interaction and data submission on websites and web applications. Within the scope of this internship project, HTML forms were utilized extensively to collect user input, process user requests, and enable various interactive functionalities.

HTML forms are defined using the `<form>` element, which encapsulates input fields, buttons, and other form elements. The project adhered to a structured form layout, organizing form elements logically and intuitively to guide users through the input process. HTML offers a wide range of input field types (`<input>`), including text fields, password fields, email fields, number fields, checkboxes, radio buttons, and dropdown menus. Each input field type serves a specific purpose, such as collecting textual input, numeric input, or selecting options from a list. The project selected appropriate input field types based on the nature of the data being collected and the user's input expectations. Labels (`<label>`) were associated with input fields using the `for` attribute to provide descriptive text and context for each input field. Properly labeled input fields improve form accessibility and usability by helping users understand the purpose of each field and providing visual cues for completing the form. Form submission is triggered by user interaction, typically through a submit button (`<button type="submit">`). Upon submission, form data is sent to a server-side script for processing or handled client-side using JavaScript. The project implemented form submission mechanisms to capture user input and initiate relevant actions or processes based on the submitted data. HTML5 introduced built-in form validation features, including required fields, pattern matching, and input type validation. Additionally, custom JavaScript validation functions were implemented to perform more complex validation logic, such as checking for valid email formats, password strength, or

input length constraints. Form validation ensures that user input meets specified criteria before submission, reducing errors and improving data accuracy.

3.2.3 Creative Potential of HTML5 Canvas

HTML5 Canvas is a powerful element that allows for dynamic rendering of graphics, animations, and interactive visualizations directly within web pages. Within the context of this internship project, HTML Canvas was employed to create engaging and interactive graphical elements to enhance user experience and provide dynamic content. Here's how HTML Canvas Graphics were utilized:

Drawing Operations: HTML Canvas provides a comprehensive set of drawing operations, including lines, shapes, curves, text, and images. The project utilized these drawing operations to create custom graphics and visual elements tailored to the specific requirements of the web applications. For example, canvas drawing operations were employed to render charts, graphs, diagrams, and custom illustrations directly within the web page.

Animation: HTML Canvas enables smooth and fluid animations through the use of JavaScript and requestAnimationFrame API. Animation effects were implemented to add dynamism and interactivity to the web applications, such as animated transitions, scrolling effects, and interactive elements. These animations enhance user engagement and provide a more immersive browsing experience.

Interactive Elements: HTML Canvas can respond to user input events, such as mouse clicks, mouse movements, and keyboard inputs, allowing for the creation of interactive elements and interfaces. Interactive features were incorporated into the canvas graphics to enable user interaction, such as draggable elements, clickable buttons, and interactive charts. These interactive elements encourage user engagement and facilitate intuitive navigation and exploration of content.

3.2.4 Optimizing User Interface: Essential CSS Components

Within the framework of this internship project, various CSS components were instrumental in shaping the visual presentation, layout, and responsiveness of web interfaces. These CSS components, in conjunction with HTML and JavaScript, contributed to creating visually appealing and user-friendly web applications.

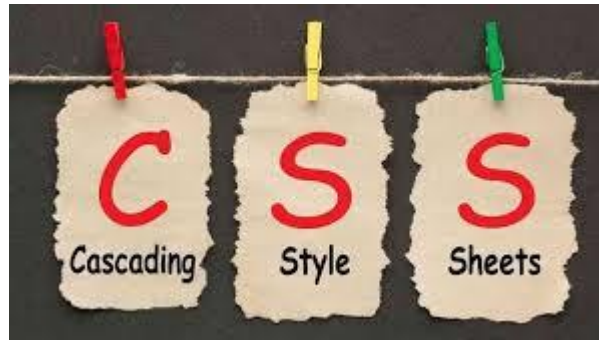


Figure 9 Cascading Style Sheets

(Source: <https://shorturl.at/rFIW3>)

Here are some of the key CSS components utilized:

- **Grid System:** CSS grid systems, such as those provided by frameworks like Bootstrap or custom grid layouts, were employed to create responsive and well-structured layouts. Grid systems allowed for the easy organization of content into rows and columns, facilitating consistent alignment and spacing across different viewport sizes.
- **Flexbox:** Flexbox, a CSS layout model, was utilized to create flexible and dynamic layouts that adapt to varying screen sizes and content proportions. Flexbox enabled the easy alignment, distribution, and ordering of elements within containers, providing greater control over the arrangement of content.
- **Typography:** CSS typography properties, including font family, size, weight, line height, and spacing, were carefully adjusted to ensure optimal readability and aesthetic appeal. Consistent typography styles were applied across the web applications to maintain visual coherence and brand identity.
- **Color Schemes:** Custom color schemes and palettes were defined using CSS to establish a cohesive visual identity and enhance the overall aesthetics of the web interfaces. Color choices were made based on considerations such as brand guidelines, accessibility standards, and user preferences.
- **Responsive Design:** CSS media queries were employed to implement responsive design principles, ensuring that web interfaces adapt seamlessly to different screen sizes and device orientations.
- **Animation and Transitions:** CSS animation and transition properties were utilized to add dynamic and interactive elements to the web interfaces. Animation effects, such as fades, slides, and rotations, were applied to enhance user engagement and provide visual feedback for user interactions.

- **Component Styling:** CSS was used to style various UI components, including buttons, forms, navigation menus, cards, and modals, to create consistent and visually appealing elements throughout the web applications. Custom styles and effects were applied to enhance the usability and aesthetics of these components.
- **Accessibility Enhancements:** CSS was leveraged to implement accessibility enhancements, such as focus outlines, contrast adjustments, and text resizing options, to ensure that web interfaces are accessible to users with diverse needs and preferences.

By leveraging these CSS components and best practices, the internship project succeeded in creating visually appealing, responsive, and accessible web interfaces that meet the needs and expectations of users while maintaining consistency with modern design trends and standards.

3.3 Enhancing User Experience with JavaScript

In the development of an online shopping portal, JavaScript played a pivotal role in enhancing interactivity, managing client-side functionalities, and facilitating a seamless user experience. Throughout the project, various JavaScript techniques and frameworks were leveraged to implement dynamic features and optimize the functionality of the portal.

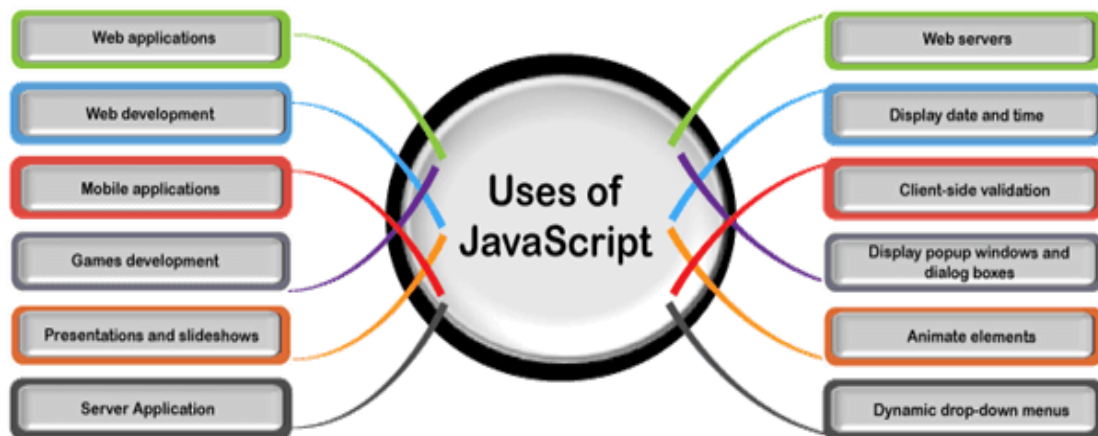


Figure 10 Uses of JavaScript

(Source: <https://rb.gy/411x6i>)

Here are some key aspects of JavaScript usage within the context of the online shopping portal:

1. User Interactivity:

JavaScript was utilized to create interactive elements and functionalities that engage users and facilitate smooth navigation throughout the shopping portal. This includes implementing features such as dropdown menus, product carousels, image galleries, and interactive forms to enhance user engagement and streamline the shopping experience.

2. Dynamic Content Loading:

JavaScript was employed to dynamically load content, such as product listings, product details, and user reviews, without requiring full page reloads. This asynchronous loading of content enhances the responsiveness of the portal and reduces page load times, providing a more seamless browsing experience for customers.

3. Shopping Cart Management:

JavaScript played a crucial role in managing the shopping cart functionality, including adding products to the cart, updating quantities, removing items, and calculating subtotal and total prices. Client-side validation was implemented to ensure that customers can easily manage their cart contents and proceed to checkout with confidence.

4. Form Validation:

JavaScript was utilized to perform client-side form validation to ensure that customer inputs, such as shipping addresses, payment details, and promotional codes, are entered correctly and meet specified criteria. Real-time feedback and error messages were provided to guide customers in completing the checkout process accurately.

5. AJAX Requests:

Asynchronous JavaScript and XML (AJAX) requests were utilized to communicate with the server-side backend of the shopping portal without requiring full page reloads. This enabled seamless retrieval and submission of data, such as product information, user authentication, and order processing, enhancing the responsiveness and efficiency of the portal.

6. User Authentication and Authorization:

JavaScript was employed to implement user authentication and authorization mechanisms, allowing customers to log in, register new accounts, and manage their profiles securely. Authentication tokens and session management techniques were utilized to ensure that users have access to their personalized account information and preferences.

7. Integration with External APIs:

JavaScript was used to integrate with external APIs, such as payment gateways, shipping calculators, and inventory management systems, to enable seamless transactions and order fulfillment processes. This integration allowed for real-time updates and synchronization of data between the shopping portal and external services.

8. Error Handling and Logging:

JavaScript error handling techniques were implemented to capture and log errors that occur during customer interactions with the portal. Detailed error messages and stack traces were logged to facilitate debugging and troubleshooting, ensuring that any issues are promptly identified and resolved to maintain the reliability and stability of the portal.

By leveraging these JavaScript functionalities and best practices, the online shopping portal was able to provide customers with a rich and immersive shopping experience, facilitating easy navigation, secure transactions, and efficient order management. JavaScript's versatility and flexibility were instrumental in enhancing the functionality and usability of the portal, ultimately contributing to its success in meeting the needs and expectations of customers.

3.4 Utilization of PHP and MySQL in Web Development

During my internship, I had the opportunity to extensively utilize PHP and MySQL for various aspects of web development projects. This section highlights the key areas where PHP and MySQL were employed and the skills I gained through these experiences.



Figure 11 PHP and MySQL in Web Design

(Source: <https://rb.gy/yrmdyt>)

3.4.1 Backend Development with PHP

PHP served as the primary scripting language for server-side development in the projects I worked on. I collaborated with the development team to create dynamic web applications and interactive features using PHP. Some of the tasks I undertook include:

- Implementing user authentication and authorization systems.
- Processing form data and handling file uploads securely.
- Integrating third-party APIs to fetch and process external data.
- Creating RESTful APIs for seamless interaction between the frontend and backend components.
- Optimizing code for performance and scalability.

Through these tasks, I enhanced my proficiency in PHP programming, gaining insights into best practices for maintaining clean, efficient, and secure code.

3.4.2 User Registration and Login Functionality

Implemented HTML forms for user registration and login, utilizing PHP to process form submissions. Validated user input to ensure data integrity and prevent common security vulnerabilities such as SQL injection and cross-site scripting (XSS) attacks. Utilized PHP's `password_hash()` function to securely hash and store user passwords in the database, enhancing

security. Implemented PHP sessions to track user authentication status across multiple pages, providing a seamless login experience.

3.4.2.1 GET vs. POST: When to Use Each Method

HTTP supports multiple request methods, but two of the most used methods in web development are GET and POST. Each method serves a specific purpose and has its advantages and limitations. Understanding when to use GET and when to use POST is crucial for designing efficient and secure web applications.



Figure 12 GET and POST in PHP

(Source: <https://rb.gy/nfbifs>)

GET Method:

1. Data Retrieval: GET requests are primarily used for retrieving data from the server. When a user requests a web page or resource using a GET request, the parameters are encoded in the URL's query string.

2. Idempotent Operations: GET requests should be idempotent, meaning that multiple identical requests should produce the same result. They should not have any side effects on the server or the application state.

3. Bookmarking and Caching: Since GET requests are visible in the URL, they are suitable for bookmarking and caching. Users can easily bookmark a URL or share it with others.

4. Limited Data: GET requests have limitations on the amount of data that can be sent. Parameters are appended to the URL, and there is a practical limit to the length of a URL that browsers can handle.

POST Method:

1. **Data Submission:** POST requests are used for submitting data to the server. When a user submits a form or uploads a file, the data is sent in the body of the request, making it suitable for transferring large amounts of data.

2. **Security:** POST requests are more secure than GET requests for sensitive data. Since the data is not visible in the URL, it is less susceptible to being intercepted or cached by intermediaries.

3. **Complex Data:** POST requests can handle complex data structures, such as JSON or XML, which may not be easily represented in a URL query string.

4. **Non-Idempotent Operations:** POST requests can perform non-idempotent operations, such as creating, updating, or deleting resources on the server. These operations may modify the application state or have side effects.

When to Use Each Method:

- **Use GET:** When you want to retrieve data from the server without modifying the application state, such as fetching a web page or querying a database.
- **Use POST:** When you need to submit sensitive data, transfer large amounts of data, or perform non-idempotent operations that modify the application state.

3.4.2.2 PHP cookies

PHP cookies are a fundamental aspect of web development for maintaining stateful information across multiple page requests. Cookies are small pieces of data stored on the client's computer by the web browser. They are sent back to the server with subsequent requests, allowing websites to remember user preferences, track users' activities, and maintain stateful information.

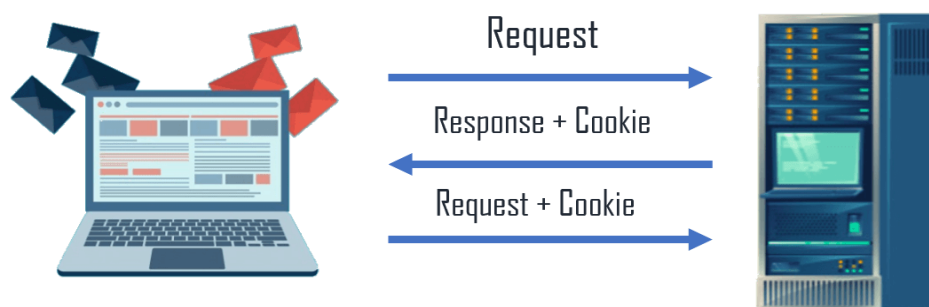


Figure 13 Working of Cookie in PHP

(Source: <https://t.ly/7L9vt>)

PHP provides built-in functions to work with cookies:

1. Set Cookie: The ``setcookie()`` function is used to set a cookie. It takes parameters such as the cookie name, value, expiration time, path, domain, and secure flag.

2. Accessing Cookies: Cookies sent by the client are accessible through the ``$_COOKIE`` super global array.

3. Expiration: Cookies can have an expiration time set using the ``setcookie()`` function. If not specified, cookies are stored until the browser session ends.

4. Deleting Cookies: To delete a cookie, you can set its expiration time to a past value. This instructs the browser to remove the cookie.

5. Security: Cookies can be manipulated by users, so avoid storing sensitive information in cookies directly. Instead, store a unique identifier and keep sensitive data on the server-side.

6. Limitations: Browsers have limitations on the number and size of cookies they can store for a particular domain. Exceeding these limits may cause older cookies to be overwritten or discarded.

Use Cases:

- User Authentication: Cookies can be used to store session tokens or user identifiers.
- Remember Me Functionality: Cookies can store persistent login tokens to remember users across sessions.
- Tracking User Preferences: Cookies can store user preferences such as language, theme, or layout.
- Shopping Carts: Cookies can store items added to a shopping cart for a user's session.

Considerations:

- Privacy: Respect user privacy by clearly communicating cookie usage in your privacy policy and providing options for users to manage cookies.
- Security: Avoid storing sensitive information in cookies and always validate and sanitize cookie data on the server-side to prevent security vulnerabilities.

3.4.2.3 PHP Sessions

PHP sessions provide a way to preserve data across subsequent HTTP requests from the same user. They are essential for maintaining stateful information and managing user authentication in web applications. Sessions are a server-side mechanism for storing user-specific data during a user's interaction with a web application. Unlike cookies, which are stored on the client-side, session data is stored on the server. Each session is identified by a unique session ID, which is typically stored in a cookie or passed via URL parameters.

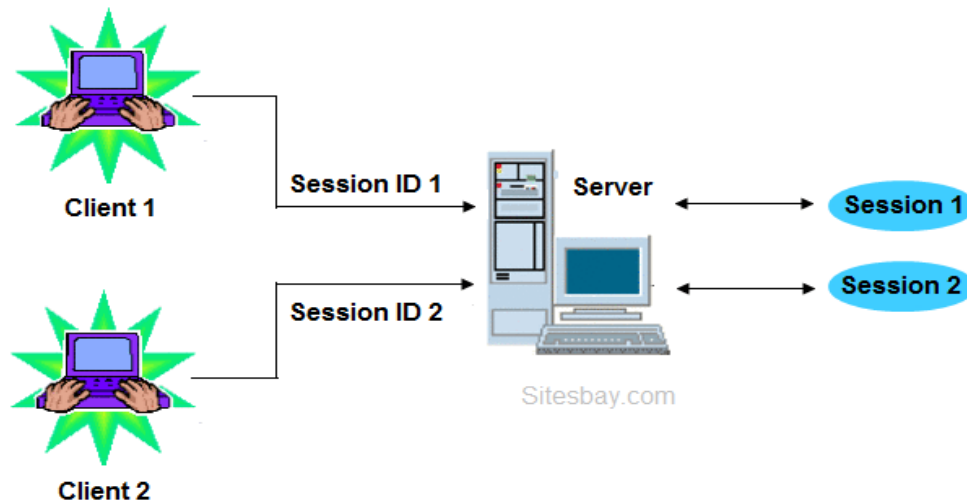


Figure 14 PHP Session

(Source: <https://tinyurl.com/4xv5u3t3>)

PHP provides a simple and efficient way to work with sessions:

1. **Session Start:** To start a session in PHP, you use the `session_start()` function. This function must be called before any output is sent to the browser.
2. **Accessing Session Data:** Once the session is started, you can store and access data using the `$_SESSION` superglobal array.
3. **Session ID:** PHP automatically generates a unique session ID for each user session. This ID is passed between the server and the client either via cookies or by appending it to URLs.
4. **Session Lifetime:** By default, PHP sessions last until the user closes their browser. However, you can adjust the session lifetime using configuration settings in `php.ini` or by using the `session_set_cookie_params()` function.
5. **Session Destruction:** Sessions can be explicitly destroyed using the `session_destroy()` function. This removes all session data associated with the current session.

Considerations:

- **Security:** Protect session data from unauthorized access and manipulation. Use HTTPS to encrypt session data in transit and avoid passing sensitive information via URL parameters.
- **Session Hijacking:** Implement measures such as session regeneration (`session_regenerate_id()`) to mitigate session hijacking attacks.
- **Session Storage:** Sessions consume server resources, so ensure that your server is configured to handle the expected number of concurrent sessions efficiently.

By leveraging PHP sessions effectively, you can create dynamic and interactive web applications that maintain user state across multiple requests while ensuring security and scalability.

3.4.3 Dynamic Content Generation

Server-side Scripting: PHP served as the backbone for generating dynamic content on the server side. Leveraging its robust features, I utilized PHP to dynamically create and deliver content such as product listings, forum threads, and user profiles. This allowed for seamless interaction with the server, enabling users to access up-to-date information without the need for page reloads.

Template Integration: Integrating PHP with HTML templates proved instrumental in streamlining the development process. By incorporating PHP code directly into HTML files, I created reusable components that could be easily customized and scaled across the application. This approach not only enhanced code maintainability but also boosted development efficiency, as it minimized the need for redundant coding tasks.

Conditional Logic: PHP's powerful conditional statements, including if-else and switch, played a crucial role in customizing content based on various factors such as user actions and application state. Leveraging these constructs, I implemented logic to dynamically adjust the displayed content, ensuring a tailored user experience. Whether it was displaying personalized recommendations or handling user authentication, PHP's conditional logic capabilities proved indispensable in creating dynamic and responsive web applications.

3.4.4 Database Interaction

MySQLi Extension: Leveraged PHP's MySQLi extension to seamlessly interact with MySQL databases. This robust extension provided a reliable means of establishing connections to databases and executing queries, thereby facilitating efficient database interaction within PHP applications.

Prepared Statements: Implemented prepared statements as a key security measure when executing SQL queries. By utilizing parameterized queries, I mitigated the risk of SQL injection attacks, ensuring that user input was safely handled and preventing unauthorized access to the database. Prepared statements also enhanced performance by allowing for the reuse of query execution plans.

Transaction Management: Utilized PHP's transaction management capabilities to maintain data integrity and consistency within the database. By encapsulating multiple database operations within transactions, I ensured that complex processes, such as order processing and forum post submissions, were executed atomically. This helped prevent data corruption and ensured that operations either completed successfully in their entirety or were rolled back in case of errors, maintaining the integrity of the database state.

3.4.5 Database Design and Schema Creation

Entity-Relationship Model: Utilizing the principles of the Entity-Relationship (ER) model, I meticulously designed MySQL database schemas to accurately represent the application's data structure. By identifying entities, attributes, and relationships between them, I crafted a comprehensive schema that efficiently stored and managed application data.

Normalization: Adhering to normalization principles, I meticulously structured the database schemas to minimize data redundancy and ensure data integrity. By decomposing tables into smaller, related entities and organizing them into normalized forms, I eliminated anomalies and redundancy, resulting in efficient and maintainable database structures that were optimized for data integrity and consistency.

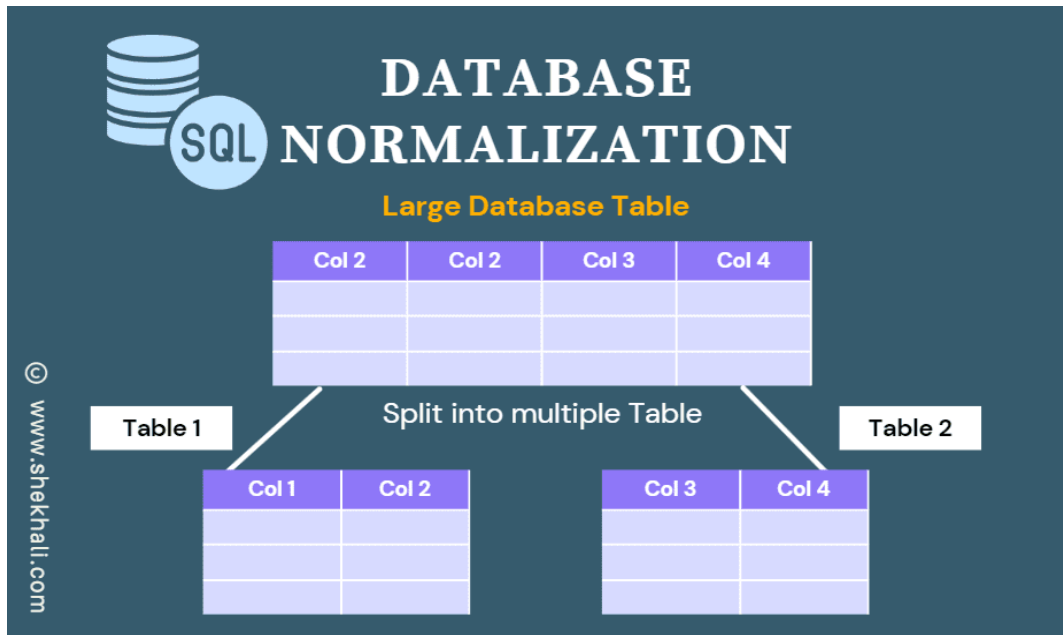


Figure 15 Normalization in DB

(Source: <https://tinyurl.com/5anvkytf>)

Indexing: Leveraging the indexing capabilities of MySQL, I strategically applied indexes to tables to enhance query performance and expedite data retrieval. By creating indexes on frequently queried columns or tables, I optimized database performance, facilitating faster data access and retrieval operations. This proactive approach to indexing proved instrumental in improving the overall responsiveness and efficiency of the database, particularly for applications with high data access demands.

3.4.6 Query Optimization and Performance Tuning

Query Analysis: Conducted thorough analysis of SQL queries generated by PHP scripts to pinpoint potential bottlenecks or areas of suboptimal performance. By scrutinizing query execution plans and identifying inefficient query patterns, I identified opportunities for optimization to enhance overall system performance.

Index Optimization: Employed advanced techniques to optimize database indexes and query execution plans, with the aim of minimizing query response times and optimizing resource utilization. This involved strategically creating, modifying, or removing indexes to better align with the application's query patterns and data access requirements, thereby improving database performance.

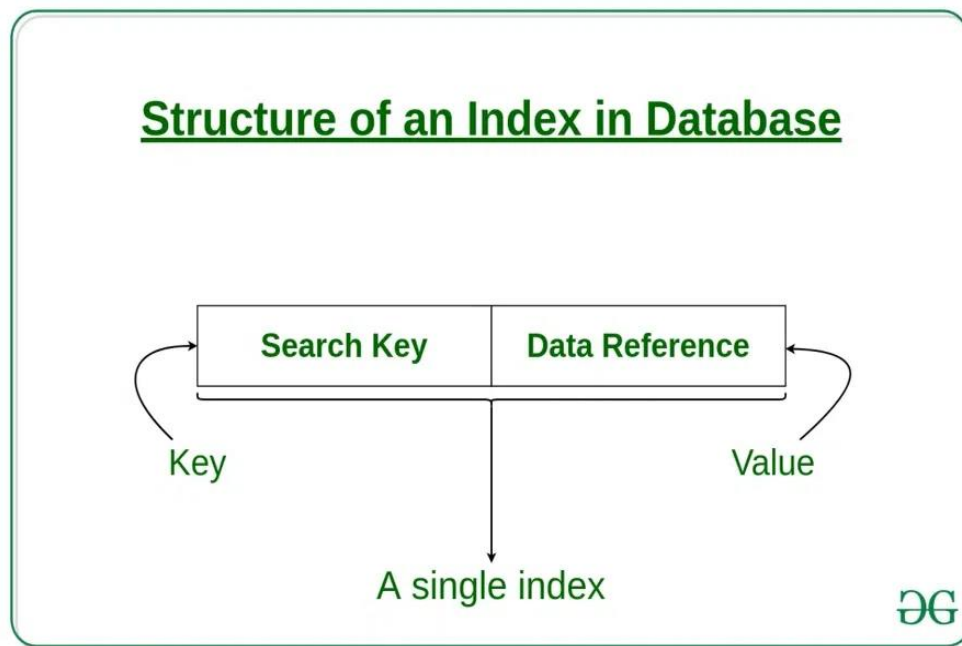


Figure 16 Structure of Index in DB
(Source: <https://tinyurl.com/yvn8sumu>)

Database Profiling: Leveraged MySQL's profiling tools to monitor and analyze query execution times, resource consumption, and other performance metrics. By profiling database activity, I gained valuable insights into query performance, identifying areas for improvement and resource allocation. This proactive approach allowed me to fine-tune database configurations and optimize query execution strategies to achieve optimal system performance and responsiveness.

3.4.7 Security and Access Control

User Privileges: Implemented stringent access control measures within MySQL by defining user accounts and privileges, ensuring that only authorized users could access sensitive data or perform specific database operations. By assigning granular permissions based on roles and responsibilities, I enforced access control policies to safeguard the integrity and confidentiality of the database.

Data Encryption: Prioritized data security by employing encryption techniques to safeguard sensitive information stored in the database. Utilizing industry-standard encryption algorithms, I encrypted sensitive data at rest to prevent unauthorized access and ensure compliance with data privacy regulations. This proactive approach to data encryption helped mitigate the risk of data breaches and unauthorized disclosure of confidential information.

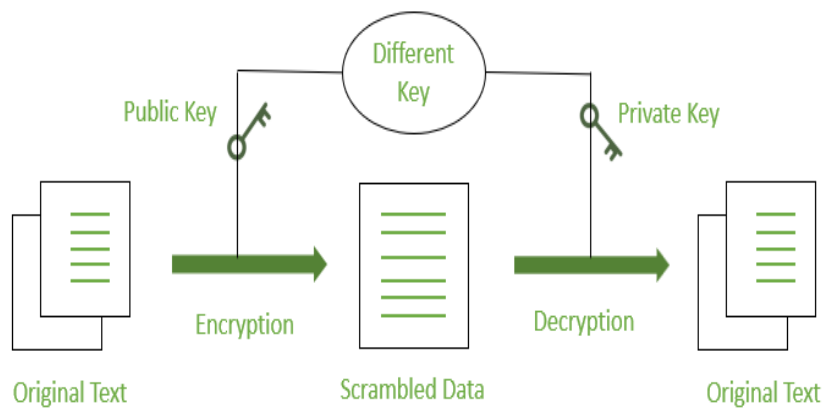


Figure 17 Security in MySQL
(Source: <https://tinyurl.com/ywb8pk3p>)

Auditing and Logging: Implemented robust logging mechanisms within MySQL to maintain a comprehensive audit trail of database activities. By logging relevant events and actions, I enabled forensic analysis and facilitated compliance with security standards and regulations. This logging infrastructure provided visibility into database activities, allowing for the detection of suspicious behavior and the timely response to security incidents, thereby enhancing overall database security posture.

3.4.8 Integration of PHP and MySQL

During my internship, a significant achievement was the seamless integration of PHP and MySQL, enabling the development of robust and dynamic web applications. By harnessing PHP for server-side scripting and MySQL for data storage and retrieval, we successfully delivered feature-rich applications that fulfilled project requirements. This integration encompassed several key components:

Secure Connections: We prioritized the establishment of secure connections between PHP scripts and MySQL databases. By implementing best practices for database connectivity, including encrypted connections and user authentication mechanisms, we ensured the confidentiality and integrity of data exchanged between the server-side scripts and the database.

Dynamic Data Rendering: Leveraging PHP's capabilities, we dynamically retrieved data from MySQL databases and rendered it on web pages. Through the integration of PHP and

MySQL, we enabled real-time updates and dynamic content generation, enhancing the user experience and responsiveness of the applications.

CRUD Operations: Central to our integration was the implementation of CRUD (Create, Read, Update, Delete) operations, allowing seamless interaction with the MySQL database through PHP scripts. This enabled users to perform essential data manipulation tasks, such as adding new records, retrieving existing data, updating records, and deleting unnecessary entries.

Data Validation and Sanitization: To mitigate security vulnerabilities, we meticulously handled data validation and sanitization within our PHP scripts. By validating user inputs and sanitizing data before interacting with the MySQL database, we fortified our applications against potential threats such as SQL injection attacks. This proactive approach to security ensured the integrity and reliability of the data stored and processed within the system.

Overall, the successful integration of PHP and MySQL empowered us to develop robust and scalable web applications that met our project objectives while adhering to stringent security standards.

CHAPTER 4

4 Experimentation

Experimentation forms a crucial aspect of the development process, allowing for the exploration of different approaches, techniques, and technologies to address project requirements and achieve desired outcomes. In this chapter, we delve into the experimentation phase of the project, detailing the methodologies, tools, and findings that guided decision-making and contributed to the successful development of the web applications.

4.1 Methodologies

The experimentation phase encompassed a variety of methodologies aimed at exploring and evaluating different approaches to web development, frontend and backend technologies, design patterns, and optimization techniques. The following methodologies were employed:

4.1.1 Prototyping

Prototyping served as a cornerstone of the development process, offering a tangible visualization of the envisioned web applications. Mockups and wireframes were meticulously crafted to outline the layout, structure, and functionality, providing stakeholders with a comprehensive understanding of the project's direction. Through iterative refinement, prototyping evolved in response to feedback garnered from stakeholders and rigorous usability testing.

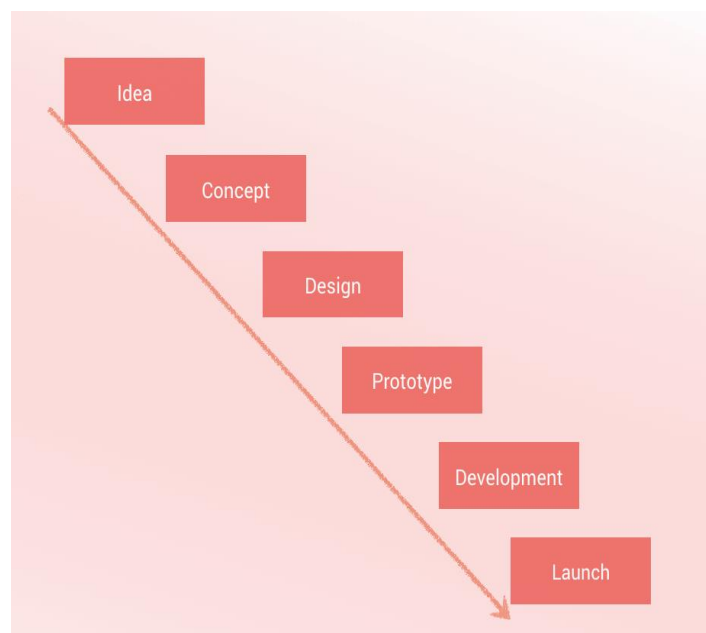


Figure 18 Prototyping Steps in Web Development

This iterative approach not only fine-tuned the aesthetics but also facilitated the alignment of the prototypes with the overarching project objectives. Stakeholder input was invaluable in shaping the user interface (UI) and user experience (UX), ensuring that the final products resonated with the intended audience. To further illustrate the evolution of the prototypes and the iterative nature of the refinement process, consider including a table or diagram showcasing the progression of mockups and wireframes. This visual representation could highlight the changes implemented at each iteration, emphasizing the responsiveness to feedback and the continuous improvement ethos ingrained within the development cycle. Additionally, incorporating screenshots or images of the initial wireframes juxtaposed with the finalized mockups could provide a compelling visual narrative of the transformation undergone by the prototypes. This comparative display would not only demonstrate the evolution of design elements but also underscore the iterative nature of the prototyping phase.

4.1.2 A/B Testing

A/B testing, also known as split testing, is a method used in web development and marketing to compare two or more versions of a webpage or application to determine which one performs better. This testing method involves creating two (or more) versions of a webpage or feature, where each version differs by one or more elements such as design, layout, content, or functionality. These versions are then presented to users at random, and their interactions and behaviors are tracked and analyzed to determine which version yields better results.

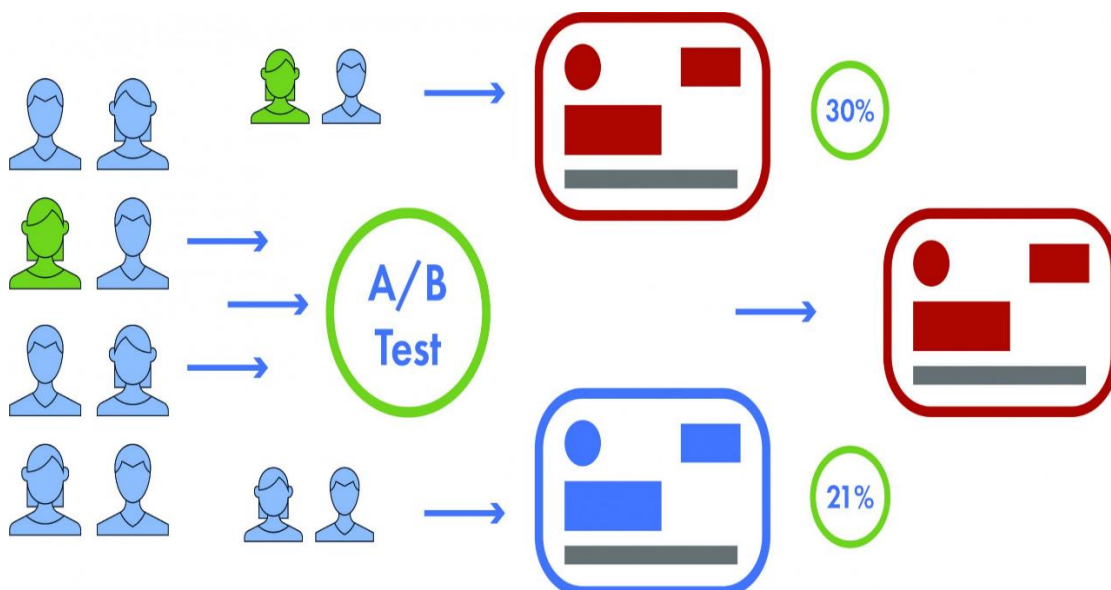


Figure 19 A/B Testing

It optimizes user experience, engagement, and conversion rates by identifying which design variations or features resonate most with the target audience. It allows developers and marketers to make data-driven decisions, thereby minimizing risks and maximizing the effectiveness of changes made to a website or application. To enhance the understanding and presentation of A/B testing results, incorporating visual aids such as charts, graphs, or tables can be highly beneficial. These visual representations can help illustrate the comparative performance of different versions more effectively and make complex data more accessible to stakeholders.

4.1.3 Usability Testing

Usability testing is a method used to evaluate the ease of use and effectiveness of a web application by observing real users as they interact with it. The primary goal is to identify any usability issues, navigation challenges, or areas for improvement that may hinder user satisfaction or task completion. Unlike A/B testing, which compares different versions of a website or feature, usability testing typically involves qualitative feedback from users rather than quantitative data analysis.



Figure 20 Web Design Usability
(Source: <http://surl.li/tqlxi>)

The process of usability testing typically involves the following steps:

1. **Participant Recruitment:** Test participants are recruited based on the target audience for the web application. These participants may include existing users, potential users, or individuals who represent the intended user demographics.

2. Test Scenario Creation: Scenarios or tasks are created that represent common user goals or actions within the web application. These tasks are designed to test specific features, workflows, or functionalities and may vary in complexity depending on the objectives of the usability test.

3. Testing Sessions: Participants are asked to complete the designated tasks while interacting with the web application. During the testing sessions, researchers observe participants' behavior, paying close attention to areas of confusion, hesitation, or frustration. Participants are encouraged to think aloud, verbalizing their thoughts, opinions, and expectations as they navigate through the application.

4. Data Collection: Feedback and observations are collected through various methods, including surveys, interviews, direct observation, screen recordings, and user session analytics. Researchers may use structured questionnaires or interview guides to gather specific insights about usability issues, user preferences, and pain points.

5. Analysis and Synthesis: The collected data is analyzed to identify common themes, patterns, and recurring issues across participants. Researchers may use qualitative analysis techniques, such as thematic coding or affinity diagramming, to organize and synthesize the findings. Key usability issues and recommendations for improvement are documented and prioritized based on their impact on the user experience.

6. Iterative Design: Usability testing results inform iterative design decisions, guiding the refinement and enhancement of the web application's interface, navigation structure, and interaction patterns. Design changes are implemented based on user feedback and validated through subsequent rounds of testing to ensure that improvements effectively address identified usability issues.

To complement the findings of usability testing and communicate insights effectively, visual aids such as heatmaps, click-tracking data, and session recordings can be valuable additions. Heatmaps visually represent user interaction patterns by highlighting areas of the interface that receive the most attention or engagement. Click-tracking data and session recordings provide detailed insights into user navigation paths, interaction sequences, and areas of confusion or frustration.

4.1.4 Performance testing

Performance testing involves assessing the speed, responsiveness, and scalability of a web application under various conditions, including different usage scenarios and load levels. The primary objective is to identify performance bottlenecks, optimize code and asset delivery, and ultimately enhance the user experience.

The process of performance testing typically includes the following steps:

1. **Define Performance Metrics:** Key performance indicators (KPIs) are identified based on user expectations and industry standards. Common metrics include page load time, time to first byte (TTFB), server response time, rendering speed, and resource utilization (CPU, memory, network bandwidth).
2. **Select Testing Tools:** Performance testing tools such as Google Lighthouse, WebPageTest, and Chrome DevTools are utilized to measure and analyze performance metrics. These tools provide insights into various aspects of web application performance and help identify areas for improvement.
3. **Create Test Scenarios:** Test scenarios are designed to simulate real-world usage conditions and load levels. Scenarios may include different user interactions, concurrent user sessions, varying network conditions (e.g., 3G, 4G, Wi-Fi), and peak traffic periods.
4. **Execute Tests:** Performance tests are conducted using the selected tools and test scenarios. The web application's performance is monitored and evaluated under different conditions to assess its speed, responsiveness, and scalability.
5. **Analyze Results:** Test results are analyzed to identify performance bottlenecks and areas for optimization. Performance metrics are compared against predefined targets or benchmarks, and any deviations are investigated to determine their root causes.
6. **Optimize Code and Assets:** Based on the findings of performance testing, optimizations are made to improve the web application's performance. This may involve optimizing code structure, reducing file sizes, leveraging browser caching, and implementing content delivery networks (CDNs) to improve asset delivery speed.
7. **Repeat Testing:** Performance testing is an iterative process, and tests may be repeated multiple times to validate the effectiveness of optimizations and ensure that

performance goals are met. As the web application evolves, ongoing performance testing helps maintain optimal performance levels.

4.1.5 Cross-Browser and Device Testing

Cross-browser and device testing ensured that the web applications functioned consistently across different web browsers, operating systems, and devices. Testing was conducted on popular browsers (e.g., Chrome, Firefox, Safari, Edge) and various devices (e.g., desktops, laptops, tablets, smartphones) to identify and address compatibility issues, layout inconsistencies, and functionality discrepancies.

4.2 Experimentation Tools

Several tools were utilized throughout the experimentation phase to facilitate prototyping, testing, debugging, and performance optimization. These tools included:

- Adobe XD and Figma for prototyping and wireframing.
- GitHub for version control and collaboration.
- BrowserStack for cross-browser and device testing.
- Lighthouse and WebPageTest for performance testing and optimization.
- Chrome DevTools for debugging, profiling, and optimizing frontend code.
- JIRA and Trello for project management and task tracking.

Each tool played a critical role in streamlining the experimentation process, enabling efficient collaboration among team members, and ensuring the quality and reliability of the web applications.

4.3 Experimentation Findings

The experimentation phase yielded several key findings that informed decision-making and influenced the direction of the project. Some notable findings include:

User Preferences:

Usability testing revealed insights into user preferences, behaviors, and pain points, guiding design decisions and feature prioritization. For example, users expressed a preference for streamlined checkout processes and intuitive navigation menus, leading to refinements in these

areas.

Performance Optimization:

Performance testing identified opportunities for optimization, such as reducing page load times, minimizing render-blocking resources, and optimizing image sizes. By addressing these performance bottlenecks, the web applications achieved faster load times and improved user experience.

Browser Compatibility:

Cross-browser testing uncovered compatibility issues on certain browsers and devices, highlighting the importance of thorough testing and ensuring consistent functionality across different platforms. Compatibility issues were addressed through targeted fixes and adjustments to CSS and JavaScript code.

Feature Effectiveness:

A/B testing provided insights into the effectiveness of different design variations and feature implementations in driving user engagement and conversion rates. Data-driven decisions were made to prioritize features that demonstrated positive outcomes and iterate on those that underperformed.

The experimentation phase of the project was instrumental in shaping the design, functionality, and performance of the web applications. Through prototyping, testing, and analysis, valuable insights were gained into user preferences, performance optimization opportunities, and feature effectiveness.

CHAPTER 5

5 Results

This chapter presents the results obtained from the development and implementation of the web applications, focusing on key performance metrics, user feedback, and the overall impact on project objectives. The results provide insights into the effectiveness of the solutions delivered and their alignment with stakeholder expectations.

5.1 Performance Metrics

Performance metrics were evaluated to assess the speed, responsiveness, and reliability of the web applications. Key metrics included:

1. **Page Load Times:** The time taken for web pages to load fully, including all assets and content.
2. **Time to Interactivity:** The time it takes for web pages to become interactive, allowing users to engage with content.
3. **Server Response Times:** The time taken for the server to process and respond to user requests.
4. **Overall Site Performance:** An aggregate measure of the overall performance of the web applications, taking into account various factors such as page load times, server response times, and resource utilization.

Performance testing tools, such as Google Lighthouse, WebPageTest, and Chrome DevTools, were utilized to measure these metrics and identify areas for optimization.

5.2 Impact on Project Objectives

The results were assessed in relation to the project objectives outlined in the earlier phases of the development process. Key project objectives included:

1. **Enhancing User Experience:** The extent to which the web applications succeeded in providing a seamless and intuitive user experience, fostering engagement and satisfaction among users.
2. **Improving Performance:** The degree to which performance optimizations and enhancements contributed to faster load times, improved responsiveness, and overall site performance.
3. **Meeting Stakeholder Requirements:** The extent to which the final deliverables met

the functional, technical, and aesthetic requirements outlined by stakeholders.

4. Achieving Business Goals: The impact of the web applications on achieving business objectives, such as increased sales, customer retention, and brand visibility.

The results were evaluated against these objectives to determine the success of the project and identify areas for future iteration and improvement.

5.3 Netmart - An Online Shopping Portal

Homepage Design: Provide an overview of the homepage design, highlighting key elements such as the navigation menu, featured products, promotional banners, and search functionality. Describe how the design promotes easy navigation and encourages users to explore further.

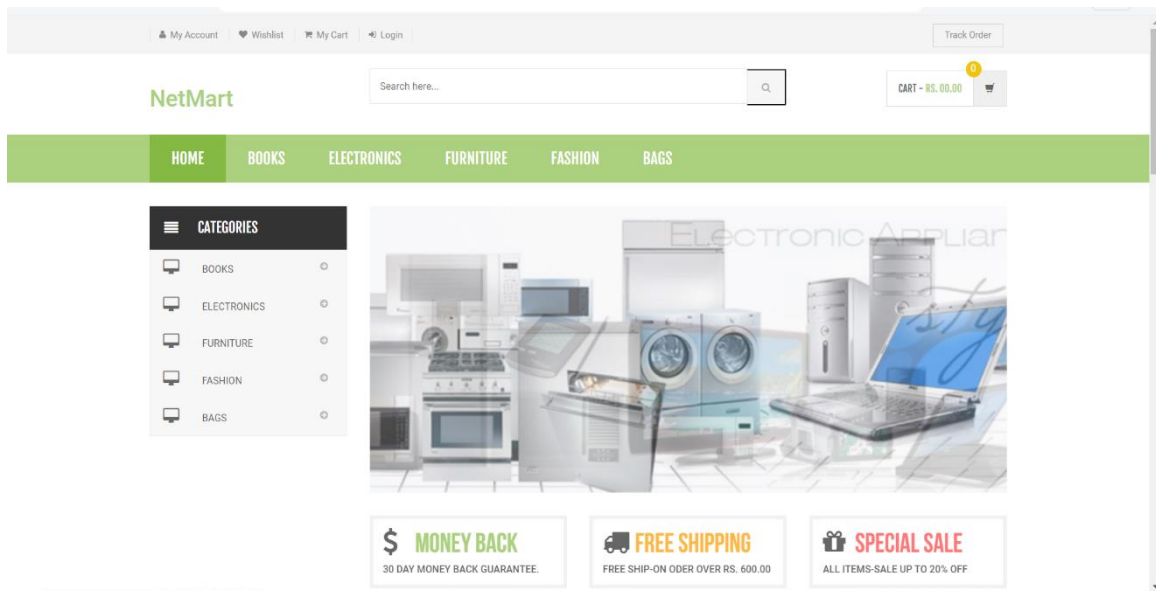


Figure 21 NetMart - Homepage

Product Details: Include screenshots of product detail pages, illustrating how product information, images, pricing, and purchasing options are presented. Describe any interactive elements or calls to action (e.g., "Add to Cart," "Buy Now") implemented to facilitate conversions.

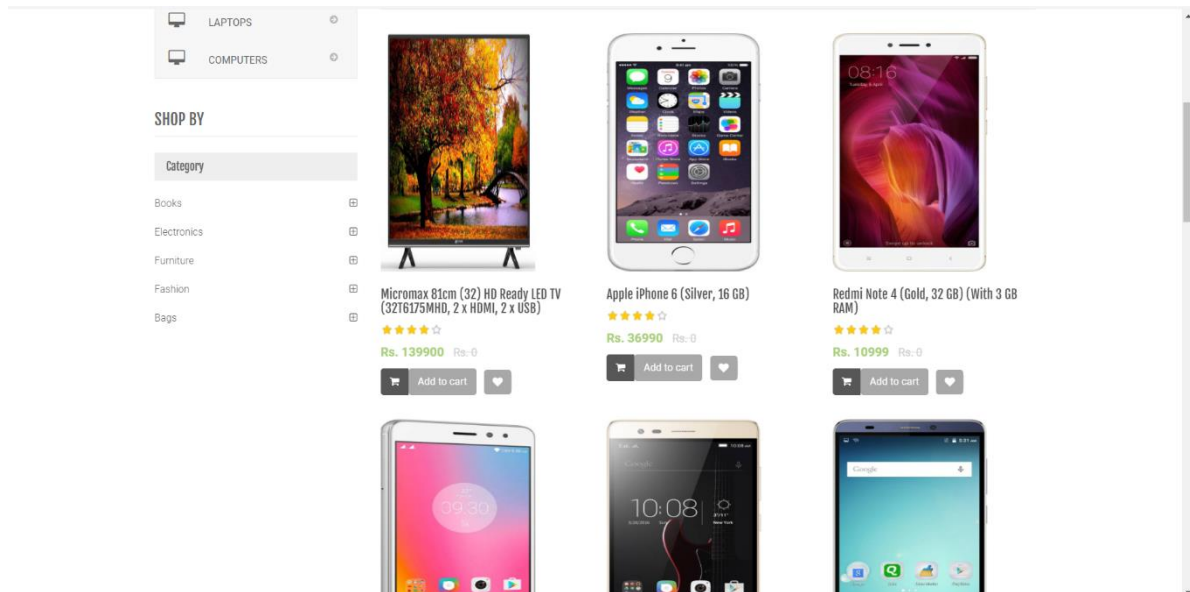


Figure 22 NetMart - Products

Shopping Cart: Display screenshots of the shopping cart page, showcasing the user's selected items, quantities, subtotal, and options for proceeding to checkout. Discuss any features implemented to modify cart contents, apply discounts, or calculate shipping costs.

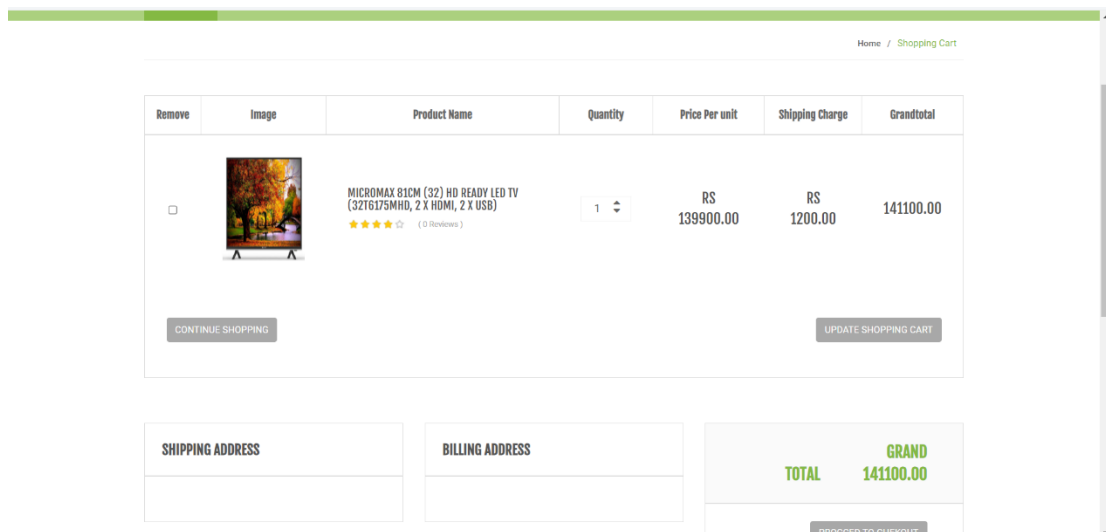


Figure 23 NetMart – Shopping Cart

The admin pages in Netmart streamline backend management, allowing administrators to efficiently add products, create categories, manage product listings, and process customer orders. These pages provide comprehensive tools for organizing inventory, updating product details, and tracking order statuses, contributing to the seamless operation of the online shopping portal.

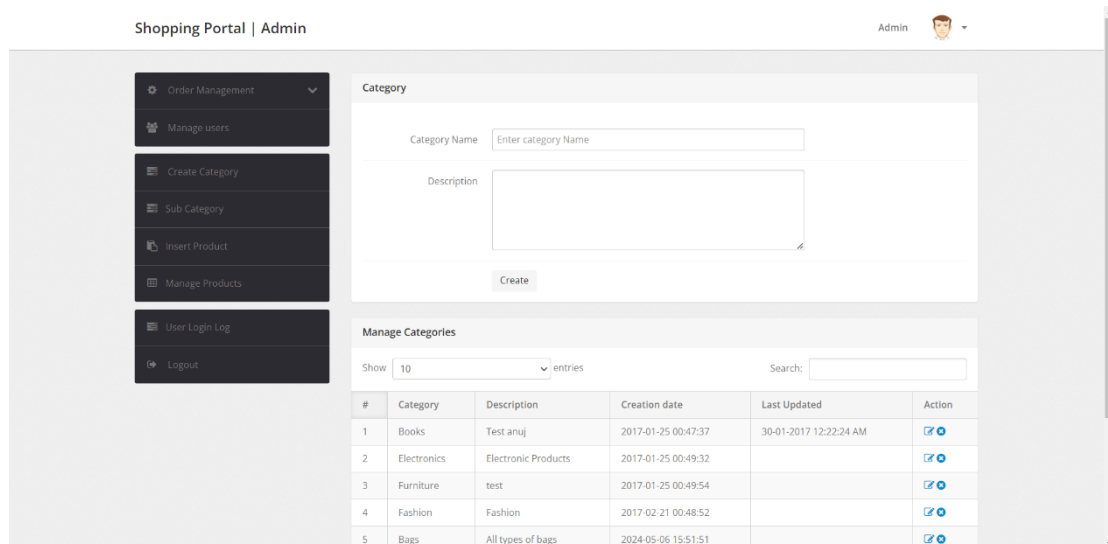


Figure 24 NetMart – Admin Page

5.4 DiscusHub Forum Website

The DiscusHub Forum Website serves as a coding forum, providing a platform for discussions on various coding topics. The homepage features sections showcasing top threads and a comprehensive list of all threads, enabling users to easily discover and engage with discussions of interest. Within each thread, users can view questions, comments, and responses from the community, fostering knowledge-sharing and collaborative problem-solving. User information, including profiles and activity history, is displayed alongside thread content, facilitating interaction and connection among forum participants.

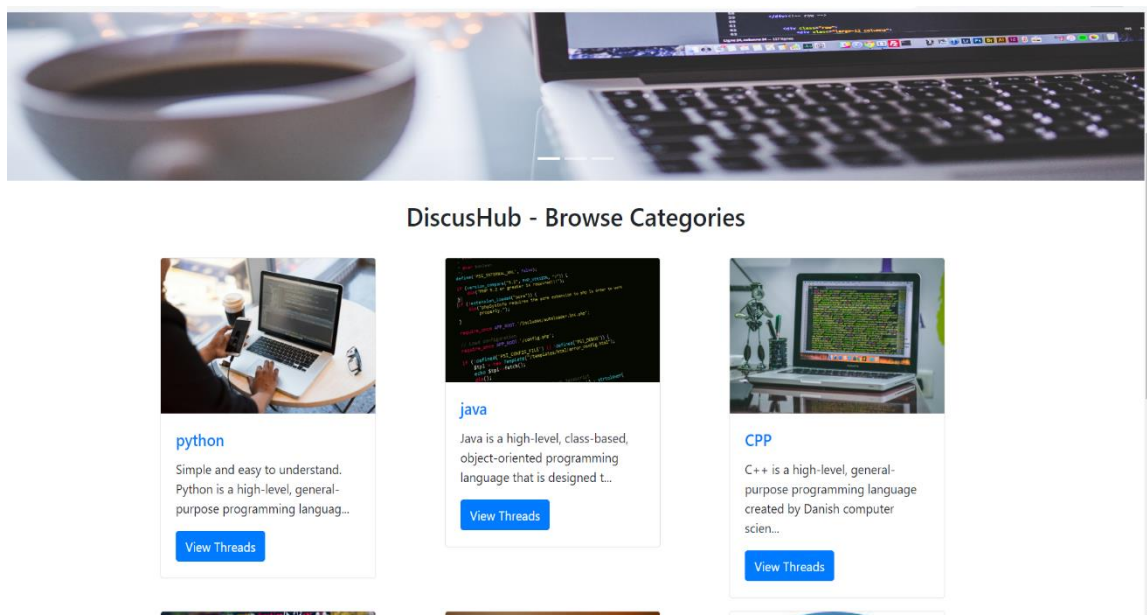


Figure 25 DiscusHub - Homepage

post. Generate posts, links or images. Do not cross post questions. remain respectful of other members at all times.

[Learn more](#)

Start a Discussion


Problem Title

Keep your title as short and crisp as possible

Elaborate Your Concern


[Submit](#)

Browse Questions



Unable to install python.
Facing issue while installing python. I was trying to install python for my major project.

Asked by: aka@gmail.com at 2024-04-02 15:49:42



Unable to use python.
Please help me.

Asked by: akak@gmail.com at 2024-04-03 14:43:07

Figure 26 DiscusHub – Threads

CHAPTER 6

6 Conclusion

The conclusion of this internship journey signifies the transition from a period of intensive learning to the commencement of a career in web development. Throughout this experience, a diverse range of technologies and methodologies has been explored and applied, paving the way for future endeavors in the field. A thorough examination of backend technologies such as PHP, MySQL, and SQL have equipped me with the skills necessary to build robust and scalable web applications. By delving into server-side scripting, database management, and performance optimization, a solid foundation in backend development has been established. Furthermore, the exploration of emerging technologies like Node.js and MongoDB presents exciting opportunities to expand upon this knowledge and leverage new tools and frameworks in future projects, ensuring adaptability and relevance in an ever-evolving technological landscape.

MySQL, as a relational database management system, offers a powerful and efficient solution for data storage, retrieval, and management. Its support for SQL (Structured Query Language) enables developers to perform complex database operations with ease, ensuring data integrity, security, and scalability. With features such as indexing, transactions, and replication, MySQL provides the foundation for building high-performance web applications that can handle large volumes of data and concurrent user interactions.

On the frontend, the exploration of HTML5, CSS3, JavaScript, and Bootstrap has enabled the creation of dynamic and visually appealing user interfaces. By mastering responsive design principles, interactive elements, and frontend frameworks, proficiency in crafting engaging user experiences has been attained. As the demand for immersive web experiences continues to grow, further exploration of frontend technologies like React.js and Vue.js will be essential to stay ahead of industry trends and deliver cutting-edge solutions that captivate and delight users. By exploring HTML5, CSS3, JavaScript, and Bootstrap, along with user experience design principles, responsive design techniques, and interactive elements, proficiency in crafting engaging and user-friendly interfaces has been honed. Through iterative design, usability testing, and feedback integration, an appreciation for the importance of user-centric design and seamless user experiences has been cultivated.

HTML5, as the latest iteration of the Hypertext Markup Language, represents a significant advancement in web development. Its introduction of new semantic elements, audio and video support, and canvas for graphics rendering has revolutionized the way web content is structured

and presented. By embracing HTML5, web developers can create more accessible, interactive, and multimedia-rich experiences for users, enhancing engagement and usability across a wide range of devices and platforms. Similarly, CSS3 has introduced a plethora of new features and capabilities that have transformed the landscape of web design. From advanced styling options such as gradients, shadows, and animations to responsive layout techniques like media queries and flexbox, CSS3 empowers developers to create visually stunning and adaptive user interfaces. By harnessing the power of CSS3, web developers can achieve greater creative freedom, improved user experiences, and enhanced brand identity through consistent and visually appealing design.

Bootstrap, developed by Twitter, has emerged as a dominant frontend framework that simplifies the process of building responsive and mobile-first web applications. With its grid system, pre-designed components, and built-in utilities, Bootstrap streamlines the development workflow and ensures consistency and coherence in design across different devices and screen sizes. By leveraging Bootstrap, developers can accelerate development timelines, reduce code duplication, and focus on delivering exceptional user experiences without compromising on performance or scalability.

Experimentation has been a central theme of this internship, driving innovation and guiding decision-making throughout the development process. Through methodologies such as prototyping, A/B testing, and usability testing, valuable insights into user preferences, performance optimization opportunities, and feature effectiveness have been gained. By leveraging tools such as Google Lighthouse, WebPageTest, and Chrome DevTools, performance metrics have been evaluated, and optimizations implemented to enhance the speed, responsiveness, and reliability of web applications.

An experiment was conducted to optimize the loading speed and overall performance of the web applications developed during the internship. Techniques such as minification of CSS and JavaScript files, image optimization, lazy loading of resources, and caching strategies were implemented. As a result, significant improvements in page load times and overall responsiveness were observed, leading to enhanced user experiences and improved search engine rankings. An experiment was undertaken to explore the concept of Progressive Web Apps (PWAs) and their potential benefits for web applications. By implementing features such as service workers, offline caching, push notifications, and home screen installation prompts, the web applications were transformed into PWAs. This resulted in improved reliability, faster

load times, and enhanced user experiences, especially on mobile devices.

The results obtained from experimentation have informed the development process, guiding the implementation of features, optimizations, and design decisions. By analyzing user feedback, performance metrics, and alignment with project objectives, a comprehensive understanding of the effectiveness and success of the solutions delivered has been achieved. Moving forward, a commitment to continuous learning, experimentation, and iteration will be essential to stay at the forefront of web development and deliver impactful solutions that meet the evolving needs of users and stakeholders. Furthermore, the internship has emphasized the significance of collaboration, communication, and project management in the development process. By working collaboratively with team members, leveraging version control systems, and utilizing project management platforms, effective teamwork and coordination have been demonstrated. The ability to communicate ideas, share knowledge, and resolve challenges collaboratively has been essential in driving the success of the internship projects.

In conclusion, the culmination of this internship represents not just the completion of projects but also the beginning of a journey toward becoming a proficient web developer. With a solid foundation in backend and frontend technologies, a commitment to experimentation and innovation, and a passion for delivering exceptional user experiences, I am excited to embark on the next phase of my career and contribute to the advancement of web development in the digital age. This internship has not only provided technical skills but also instilled a mindset of lifelong learning, adaptability, and resilience, which will continue to guide my growth and success as a web developer.

REFERENCES

- [1] H. Božiković and M. Štula, "Web design — Past, present and future," 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 2018, pp. 1476-1481, doi: 10.23919/MIPRO.2018.8400266. keywords: {Web pages;Wearable computers;web;design;progress;wearables;responsive;hybrid},
- [2] H. Wang and J. Yang, "Research and application of web development based on ASP.NET 2.0+Ajax," 2008 3rd IEEE Conference on Industrial Electronics and Applications, Singapore, 2008, pp. 857-860, doi: 10.1109/ICIEA.2008.4582637.
- [3] Q. Wu, G. Ding, Z. Du, Y. Sun, M. Jo and A. V. Vasilakos, "A Cloud-Based Architecture for the Internet of Spectrum Devices Over Future Wireless Networks," in IEEE Access, vol. 4, pp. 2854-2862, 2016, doi: 10.1109/ACCESS.2016.2576286.
- [4] S. E. Ullah, T. Alauddin and H. U. Zaman, "Developing an E-commerce website," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, India, 2016, pp. 1-4, doi: 10.1109/MicroCom.2016.7522526.
- [5] Lee, Minwoo, Hye Young Lee and Moon-Gil Yoon. "Website development strategy for e-Commerce success." The 40th International Conference on Computers & Industrial Engineering (2010): 1-6.
- [6] Y. -W. Luo, "The effect of using JavaScript and HTML to implement an E-learning web sites as a remedial teaching aid on English underachievers at universities of science and technology," The 40th International Conference on Computers & Industrial Engineering, Awaji, Japan, 2010, pp. 1-7, doi: 10.1109/ICCIE.2010.5668450.
- [7] Blyth, David S. "Web Page Design For HTML (and friends)." QUALCOMM. Accessed May 13, 2024.
- [8] Wei Jiang, Meng Zhang, Bin Zhou, Yujian Jiang and Yingwei Zhang, "Responsive web design mode and application," 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), Ottawa, ON, 2014, pp. 1303-1306, doi: 10.1109/WARTIA.2014.6976522.
- [9] Garcia, Jorge. (2008). Aspect-Oriented Web Development in PHP.
- [10] D. Hauzar and J. Kofron, "On Security Analysis of PHP Web Applications," 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops, Izmir, Turkey, 2012, pp. 577-582, doi: 10.1109/COMPSACW.2012.106.
- [11] Haris, Norhaidah & Hasim, Nurdattillah. (2019). PHP Frameworks Usability in Web Application Development. International Journal of Recent Technology and Engineering. 8. 10.35940/ijrte.C1020.1083S19.
- [12] Prokofyeva, Natalya & Boltunova, Victoria. (2017). Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems. Procedia Computer Science. 104. 51-56. 10.1016/j.procs.2017.01.059.
- [13] X. Yu and C. Yi, "Design and Implementation of the Website Based on PHP & MYSQL," 2010 International Conference on E-Product E-Service and E-Entertainment, Henan, China, 2010, pp. 1-4, doi: 10.1109/ICEEE.2010.5661595.
- [14] Ji Wen, Yang Zhang, Yong Yan and Shunren Xia, "A Web-based Examination System Based on PHP+MySQL," 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai,

2005, pp. 2882-2885, doi: 10.1109/IEMBS.2005.1617076.

- [15] K. I. Satoto, R. R. Isnanto, R. Kridalukmana and K. T. Martono, "Optimizing MySQL database system on information systems research, publications and community service," 2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Semarang, Indonesia, 2016, pp. 1-5, doi: 10.1109/ICITACEE.2016.7892476.
- [16] Sotnik, Svitlana, Volodymyr Manakov, and Vyacheslav Lyashenko. "Overview: PHP and MySQL Features for Creating Modern Web Projects." *International Journal of Academic Information Systems Research (IJASIR)* 7, no. 1 (January 2023): 11-17. Accessed May 13, 2024.
- [17] K. Lei, Y. Ma and Z. Tan, "Performance Comparison and Evaluation of Web Development Technologies in PHP, Python, and Node.js," 2014 IEEE 17th International Conference on Computational Science and Engineering, Chengdu, China, 2014, pp. 661-668, doi: 10.1109/CSE.2014.142.
- [18] Laaziri, Majida & Benmoussa, Khaoula & Samira, Khouilji & Larbi, Kerkeb. (2019). A Comparative study of PHP frameworks performance. *Procedia Manufacturing*. 32. 864-871. 10.1016/j.promfg.2019.02.295.