CHAPTER-1

INTRODUCTION

1.1 Introduction to Project

In the ever-evolving landscape of technology, where innovation is the heartbeat of progress, Tyagi Electronics emerges as a beacon of excellence, providing a seamless and unparalleled e-commerce platform to cater to your electronic needs. Welcome to Tyagi Electronics Web Store, where cutting-edge technology meets the convenience of online shopping, delivering an immersive and comprehensive experience tailored for tech enthusiasts, professionals, and everyday consumers alike.

Navigating the Digital Frontier:

In an era where the digital realm intertwines with our daily lives, Tyagi Electronics Web Store stands at the forefront, committed to offering a diverse array of electronic products that cater to the dynamic demands of the tech-savvy generation. From state-of-the-art smartphones and laptops to home appliances, audio-visual equipment, and a myriad of electronic accessories, our web store is designed to be your one-stop destination for all things tech.

Unparalleled Product Selection:

At Tyagi Electronics, we understand that every individual has unique preferences and requirements. Therefore, our curated selection of products spans a wide spectrum, ensuring that you find the perfect match for your needs. Whether you are a professional seeking high-performance gadgets for work or a casual user looking for user-friendly and stylish electronics, our extensive catalog has something for everyone.

Seamless Shopping Experience:

We believe that your online shopping journey should be as enjoyable and effortless as possible. Tyagi Electronics Web Store boasts an intuitive and user-friendly interface, making navigation a breeze. Our website is designed to provide you with detailed product information, user reviews, and expert recommendations, empowering you to make informed decisions with confidence.

Secure Transactions and Timely Deliveries:

Your trust is our priority. Tyagi Electronics Web Store ensures the highest standards of security for your transactions, guaranteeing a safe and reliable shopping experience. With our robust logistics network, we are committed to timely deliveries, ensuring that your eagerly awaited tech treasures reach your doorstep without delay.

Customer-Centric Approach:

Beyond offering a vast array of products, Tyagi Electronics is driven by a customer-centric ethos. Our dedicated customer support team is ready to assist you at every step, addressing queries, providing guidance, and ensuring your satisfaction. We value your feedback, and it fuels our commitment to continuous improvement.

Embark on a journey through the digital marketplace with Tyagi Electronics Web Store. Discover the perfect synergy of innovation, convenience, and customer-centricity as we redefine your electronic shopping experience. Welcome to a world where technology meets seamless e-commerce, and where your electronic aspirations find a home.

Future Scope:

1. Integration of Emerging Technologies:

Tyagi Electronics Web Store envisions incorporating cutting-edge technologies such as augmented reality (AR) and virtual reality (VR) to enhance the online shopping experience. This will allow customers to virtually experience products before making a purchase decision, providing a more immersive and informed buying journey.

2. Personalization and Artificial Intelligence (AI):

The future holds exciting prospects for personalization using AI algorithms. Tyagi Electronics Web Store aims to implement advanced recommendation systems, understanding user preferences and behavior to tailor product suggestions. This level of personalization will not only streamline the shopping process but also create a more engaging and user-centric platform.

3. Enhanced Mobile Experience:

Recognizing the increasing reliance on mobile devices, Tyagi Electronics Web Store plans to optimize its platform further for mobile users. The development of a dedicated mobile application will offer a seamless, intuitive, and responsive interface, ensuring a consistent and enjoyable shopping experience across various devices.

4. Expanding Product Categories:

To meet the evolving needs of our diverse customer base, Tyagi Electronics Web Store will continue to expand its product categories. This expansion will include the inclusion of innovative tech products, home automation devices, and eco-friendly electronic solutions, providing customers with a comprehensive and up-

to-date selection.

5. Global Outreach and Multilingual Support:

Tyagi Electronics aspires to extend its reach globally, catering to a broader audience. The implementation of multilingual support will facilitate a more inclusive experience, breaking down language barriers and ensuring that customers from different regions can navigate the platform effortlessly.

6. Eco-Friendly Initiatives:

Recognizing the importance of sustainability, Tyagi Electronics Web Store aims to introduce eco-friendly initiatives. This includes partnering with environmentally conscious manufacturers, promoting the recycling of electronic waste, and offering a range of green products. This commitment to sustainability aligns with global efforts towards a greener and more responsible future.

7. Collaborations and Partnerships:

The future holds opportunities for strategic collaborations and partnerships with tech giants, startups, and industry leaders. Tyagi Electronics Web Store aims to forge alliances that bring exclusive products, special promotions, and innovative solutions to our customers, fostering a dynamic and thriving ecosystem.

8. Enhanced Security Measures:

Cybersecurity is paramount in the digital age. Tyagi Electronics Web Store will continually invest in advanced security measures to safeguard customer data and transactions. Implementing the latest encryption technologies and fraud prevention systems will ensure a secure and trustworthy online environment.

Tyagi Electronics Web Store is poised for an exciting future, driven by a

commitment to innovation, customer satisfaction, and a sustainable approach. As technology advances, our platform will evolve to meet and exceed the expectations of our discerning customers, ushering in a new era of electronic commerce.

1.2 Introduction to Technology

HTML is one of the cornerstones of web development, as developers use it to structure a website's content. Essentially, it's the code that dictates where all the website elements are and how they appear to the user.

Still, learning HTML can be valuable even if you aren't pursuing a career in web development. Below, we'll explore the benefits of learning HTML to help youHTML stands for Hyper Text Mark-up Language to help you decide if it's right for you.

1. HTML is the standard mark-up language for creating Web pages.
2. HTML describes the structure of a Web page.
3. HTML consists of a series of elements.
4. HTML elements tell the browser how to display the content.
5. HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.
6. All HTML documents must start with a document type declaration.
7. The HTML document itself begins with <html> and ends with </html>.
8. The visible part of the HTML document is between <body> and </body>.

Importance of HTML

- **HTML can help to build own Unique website:**

  HTML allows us to customize our website or build our theme from scratch. It also allows us to create and organize the elements of our website like images, texts, or forms.

- **No longer wait for IT team:**

  With HTML we could improve and fix errors as we find them.

- **HTML helps to understand how to make things more accessible:**

  Semantic HTML tags form the basis for good accessibility practices, as they allow screen readers to determine where different elements are on a web page and how they are formatted.

  Learning how to make our website accessible is crucial. Not only does it help maximize the audience, but it will also help demonstrate that the company values inclusivity.

- **Knowing HTML helps to collaborate more effectively with technical teams:**

  Learning HTML is a great step if we are looking to collaborate more effectively with our technical teammates. Basically, being familiar with HTML helps ensure that technical teammates are on the same page. This helps save time and streamline communications.

- **The design can shine with HTML**

  By leaving the basics of HTML, we can create mock-up designs to illustrate our ideas to our clients. This will also help us to save time.

- **HTML skills will open up more income generating opportunities:**

  HTML is a valuable skill – even for non-technical professionals. Front-end engineers are highly in demand, and HTML is a core component of their skillset.

Characteristics of HTML

Following are important characteristics of HTML.

- HTML is the simplest language that can be easily understood and modified.
- It gives the flexibility to design the web pages so that a structured display can be possible for all listed documents on the web pages.
- Formatting tags could be used for effective presentations in the web portal, and it could possibly because of HTML.
- Multiple links can be added to the web page to redirect to other pages using listed links easily.
- The most important part of HTML is, it can be displayed in Macintosh, Windows, and Linux and support all environments. Html is platform independent.
- For an attractive look for our web pages, Sounds, Graphics and videos can also be added in HTML.

Introduction to CSS

Cascading **S**tyle **S**heets, fondly referred to as **CSS**, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows us to apply styles to web pages.

More importantly, CSS enables us to do this independent of the HTML that makes up each web page. It describes how a webpage should look: it prescribes colours, fonts, spacing, and much more. CSS lets developers and designers define how it behaves, including how elements are positioned in the browser.

While html uses tags, CSS uses rulesets. CSS is easy to learn and understand, but it provides powerful control over the presentation of an HTML document.

Advantages Of CSS

- **CSS saves time:** CSS can be Written once and reuse the same sheet in multiple HTML pages.
- **Easy Maintenance:** To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
- **Search Engines:** CSS is considered a clean coding technique, which means search engines won't have to struggle to "read" its content.
- **Superior styles to HTML:** CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Offline Browsing:** CSS can store web applications locally with the help of an offline cache. Using this, offline websites can also be viewed.

CSS Syntax

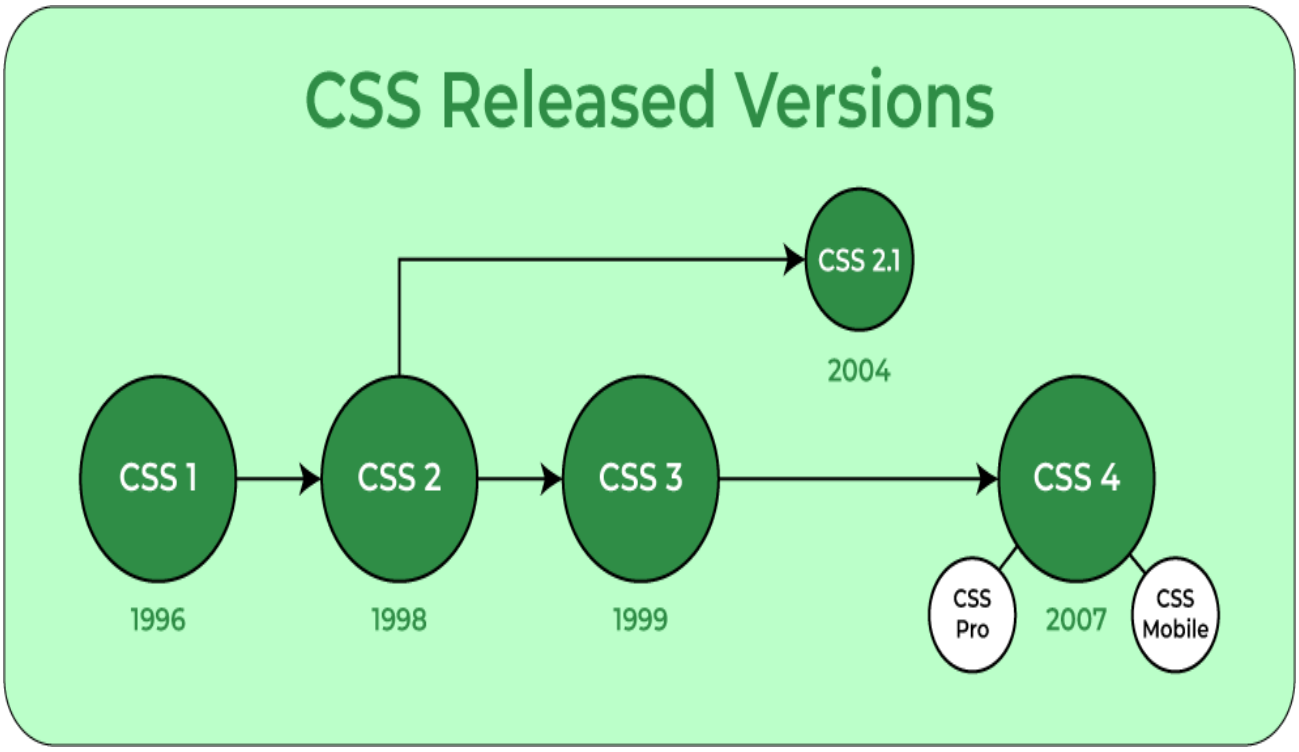
CSS comprises style rules that are interpreted by the browser and then applied.

to the corresponding elements in your document.

A style rule set consists of a selector and declaration block.

- Selector -- h1 Declaration -- {color: blue; font size: 12px;}
- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a CSS property name and a value, separated by a colon.

  For Example:
- Color is property and blue are value.
  Font-size is property and 12px is value.
- CSS declaration always ends with a semicolon, and braces declaration blocks are surrounded by curly.



1.3 Introduction to JavaScript

JavaScript is a lightweight, cross-platform, and interpreted compiled programming language which is also known as the scripting language for webpages. It is well-known for the development of web pages, many non-browser environments also use it. JavaScript can be used for client-side developments as well as Server-side developments. JavaScript is both imperative and declarative type of language. JavaScript contains a standard library of objects, like Array.

, Date, and Math, and a core set of language elements like operators, control structures, and statements.

- **Client-side:** It supplies objects to control a browser and its Document Object

  Model (DOM). Like if client-side extensions allow an application to place elements on an HTML form and respond to user events such as mouse clicks, form input, and page navigation. Useful libraries for the client-side

are AngularJS, ReactJS, Vue JS and so many others.

- **Server-side:** It supplies objects relevant to running JavaScript on a server. Like if the server-side extensions allow an application to communicate with a database and provide continuity of information from one invocation to another of the application, or perform file manipulations on a server. The useful framework which is the most famous these days is node.js.
- **Imperative language** – In this type of language we are mostly concern about how it is to be done. It simply control the flow of computation. The procedural programming approach, object, oriented approach comes under this like a sync await we are thinking what it is to be done further after a sync call.
- **Declarative programming** – In this type of language we are concern about

  how it is to be done, basically here logical computation require. Here main goal is to describe the desired result without direct dictation on how to get it like arrow function do.

  JavaScript can be added to your HTML file in two ways:

- **Internal JS:** We can add JavaScript directly to our HTML file by writing the code inside the <script> tag. The <script> tag can either be placed inside the <head> or the <body> tag according to the requirement.
- External JS: We can write JavaScript code in other file having an extension.js and then link this file inside the <head> tag of the HTML file in which we want to add this code.

Syntax:

Example:

```
<!DOCTYPE html>
<html Lang="end">
<head>
<title>
Basic Example to Describe JavaScript
</title>
</head>

<body>
<!-- JavaScript code can be embedded inside
head section or body section -->
<script>
console.log ("Welcome to Referral Hub");
</script>
</body>

</html>
```

Output: The output will display on the console. Welcome to Referral Hub

History of JavaScript:

It was created in 1995 by Brendan Each while he was an engineer at Netscape. It was originally going to be named Live Script but was renamed. Unlike most programming languages, the JavaScript language has no concept of input or output. It is designed to run as a scripting language in a host environment, and free is up to the host environment to provide mechanisms for communicating with the outside world. The most common host environment is the browser.

Features of JavaScript:

According to a recent survey conducted by Stack Overflow, JavaScript is the most popular language on earth.

With advances in browser technology and JavaScript having moved into the server with Node.js and other frameworks, JavaScript is capable of so much more. Here are a few things that we can do with JavaScript:

- JavaScript was created in the first place for DOM manipulation. Earlier websites were mostly static, after JS was created dynamic Web sites were made.
- Functions in JS are objects. They may have properties and methods just like another object. They can be passed as arguments in other functions.
- Can handle date and time.
- Performs Form Validation although the forms are created using HTML.
- No compiler is needed.

Applications of JavaScript:

Web Development: Adding interactivity and behavior to static sites JavaScript

was invented to do this in 1995. By using AngularJS that can be achieved so easily.

Web Applications: With technology, browsers have improved to the extent that a language was required to create robust web applications. When we explore a map in Google Maps then we only need to click and drag the mouse. All detailed view is just a click away, and this is possible only because of JavaScript. It uses Application Programming Interfaces (APIs) that provide extra.
power to the code. The Electron and React is helpful in this department.

Server Applications: With the help of Node.js, JavaScript made its way from client to server and node.js is the most powerful on the server-side.

Games: Not only in websites, but JavaScript also helps in creating games for leisure. The combination of JavaScript and HTML 5 makes JavaScript popular in game development as well. It provides the EaseJS library which provides solutions for working with rich graphics.

Smartwatches: JavaScript is being used in all possible devices and applications. It provides a library PebbleJS which is used in smartwatch applications. This framework works for applications that require the internet for its functioning.

Art: Artists and designers can create whatever they want using JavaScript to draw on HTML 5 canvas, and make the sound more effective also can be used p5.js library.

Machine Learning: This JavaScript ml5.js library can be used in web development by using machine learning.

Mobile Applications: JavaScript can also be used to build an application for non-web contexts. The features and uses of JavaScript make it a powerful tool for creating mobile applications. This is a Framework for building web and mobile apps using JavaScript. Using React Native, we can build mobile applications for different operating systems. We do not require to write code for different systems. Write once use it anywhere!

Limitations of JavaScript:

- **Security risks:** JavaScript can be used to fetch data using AJAX or by manipulating tags that load data such as <img>, <object>, <script>. These attacks are called cross site script attacks. They inject JS that is not the part of the site into the visitor's browser thus fetching the details.

- **Performance:** JavaScript does not provide the same level of performance as offered by many traditional languages as a complex program written in JavaScript would be comparatively slow. But as JavaScript is used to perform simple tasks in a browser, so performance is not considered a big restriction in its use.
- **Complexity:** To master a scripting language, programmers must have a thorough knowledge of all the programming concepts, core language objects, client and server-side objects otherwise it would be difficult for them to write advanced scripts using JavaScript.
- **Weak error handling and type checking facilities:** It is weakly typed

  language as there is no need to specify the data type of the variable. So wron g type checking is not performed by compile.

Why JavaScript is known as a lightweight programming language?

JavaScript is considered as lightweight due to the fact that it has low CPU usage, is easy to implement and has a minimalist syntax. Minimalist syntax as in, it has no data types. Everything is treated here as an object. It is very easy to learn because of its syntax similar to C++ and Java.

A lightweight language does not consume much of your CPU's resources. It

doesn't put excess strain on your CPU or RAM. JavaScript runs in the browser even though it has complex paradigms and logic which means it uses fewer resources than other languages. For example, NodeJs, a variation of JavaScript not only performs faster computations but also uses less resourc es than its counterparts such as Dart or Java.

Additionally, when compared with other programming languages, it has less in-built libraries or frameworks, contributing as another reason for it to be lightweight. However, this brings it a drawback that we need to incorporate external libraries and frameworks.

Is JavaScript compiled or interpreted or both?

JavaScript is both compiled and interpreted. In the earlier versions of JavaScript, it used only the interpreter that executed code line by line and shows the result immediately. But with time the performance became an issue as interpretation is quite slow. Therefore, in the newer versions of JS, probably after the V8, JIT compiler was also incorporated to optimize the execution and display the result more quickly. This JIT compiler generates a bytecode that is relatively easier to code. This bytecode is a set of highly optimized instructions. The V8 engine initially uses an interpreter, to interpret the code.

CHAPTER 2

(SOFTWARE DEVELOPMENT LIFECYCLE)

- Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.
- SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.
- SDLC is the acronym of Software Development Life Cycle. It is also called as Software Development Process.
- SDLC is a framework defining tasks performed at each step in the software development process.
- ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.

SDLC Models

There are various software development life cycle models defined and designed which are followed during the software development process. These models are also referred as Software Development Process Models". Each process model follows a Series of steps unique to its type to ensure success in the process of software development.

Following are the most important and popular SDLC models followed in the industry –

Waterfall Model

Iterative Model Spiral Model

SDLC Stages

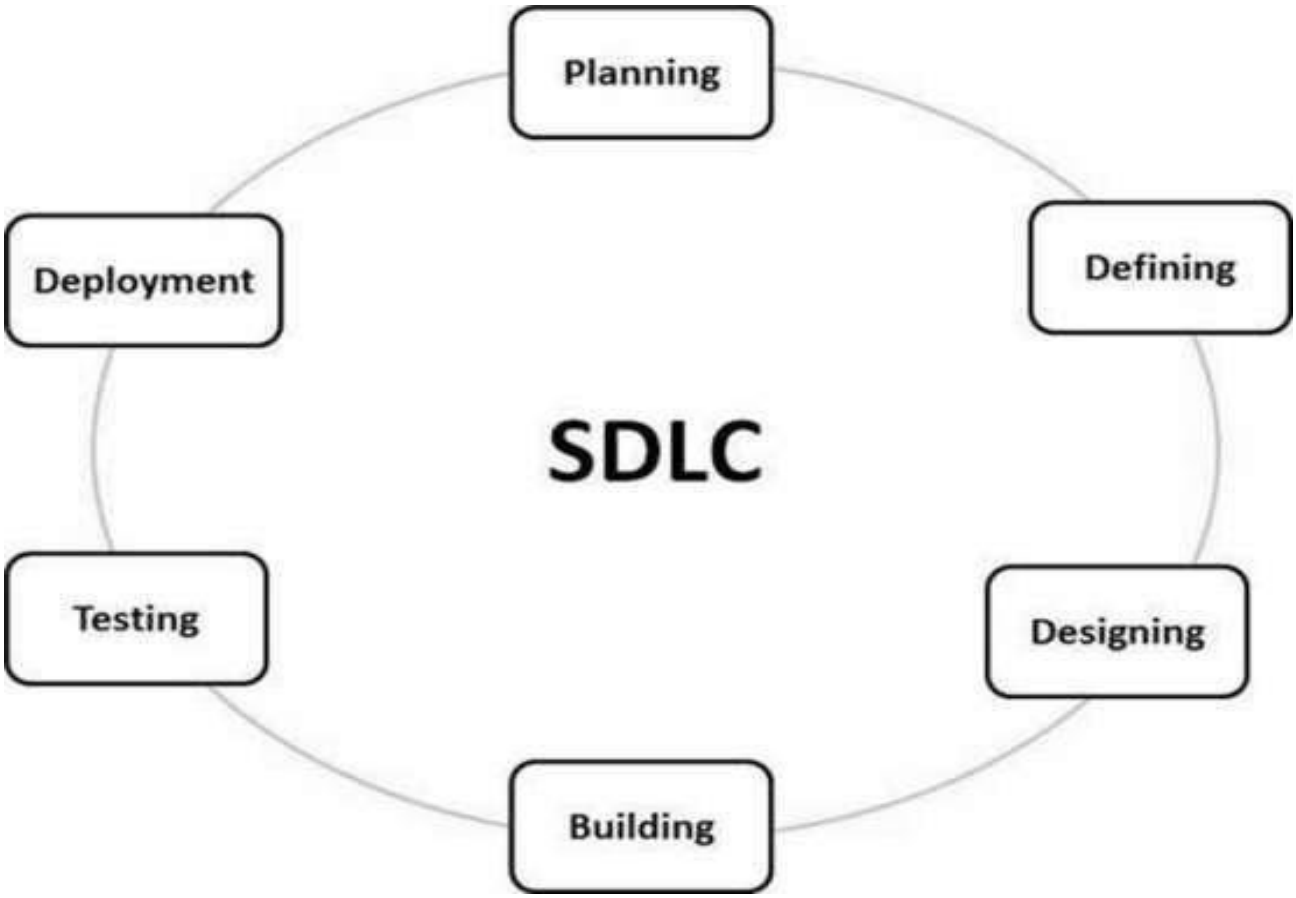


FIGURE 2.1

The following figure is a graphical representation of the various stages of a SDLC

2.1 Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

2.2 Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an **SRS (Software Requirement Specification)** document which consists of all the product requirements to be designed and developed during the project life cycle.

2.3 Designing the Product

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget

and time constraints, the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

2.4 Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design. Is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

2.5 Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

2.6 Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per thebusiness strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT - User acceptance testing).

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

2.7 Feasibility Study

A feasibility study is an assessment of the practicality of a proposed project or system. It is an important step in the planning process that helps determine whether the project is viable and worth pursuing.

Conducting a feasibility study for an informative website can help you assess the potential success and profitability of the site, as well as identify any potential challenges or risks.

Here are some key considerations to include in a feasibility study for an informative website

- Target audience: Identify who the website will be aimed at and what type of information they are looking for.
- Competition: Research other websites that provide similar information and assesshow your website will differentiate itself from the competition.
- Resources: Determine the resources (e.g., time, money, expertise) required to develop and maintain the website.
- Revenue potential: Consider potential revenue streams, such as advertising, subscriptions, or e-commerce sales, and assess their feasibility.
- Legal and regulatory considerations: Make sure the website complies with relevant laws and regulations, including those related to privacy and data protection.
- Technical requirements: Determine the technical infrastructure and capabilities

needed to support the website, including hosting, platform, and content management system.

By conducting a thorough feasibility study, you can make informed decisions about the feasibility and viability of your informative website project.

2.8 Gantt Chart

Activities	15 Aug to 20 Aug	21 Aug to 25 Aug	26 Aug to 1 Sept	2 Sept to 18 Sept	19 Sept to 28 Sept	29 Sept to 10 Oct	11 Oct to 18 Oct	19 Oct to 5 Nov
Problem Analysis								
Analyzing the existing system								
Hardware and software selection								
Database design								
Coding								
Testing								
Evaluation								
Implementation								

FIG 2.2

CHAPTER 3
(SYSTEM MODEL)

3.1 Spiral Model

The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

Spiral Model - Design

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

IDENTIFICATION

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase.

This phase also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral, the product is deployed in the identified market.

DESIGN

The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals.

3.2 Building Phase

The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback.

Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to the customer for feedback.

The software development process enters the following iteration based on the customer evaluation and then adopts a linear strategy to execute the consumer feedback recommendations. Throughout the software's lifespan, revisions along the spiral continue to be made.

3.3 Risk Analysis

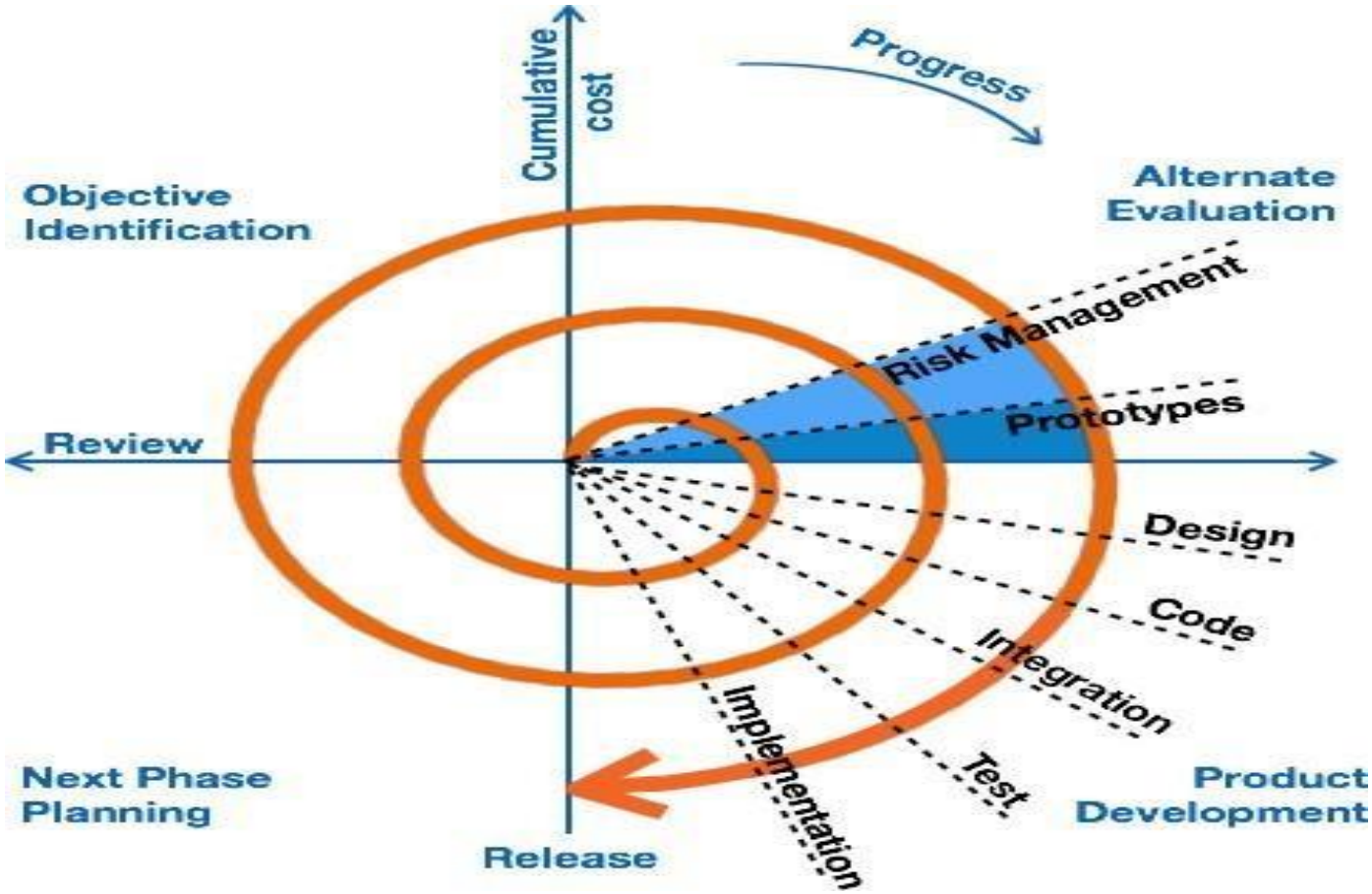
Risk Analysis includes identifying, estimating, and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback.

Identification, estimation, and monitoring of management and technical risks, such as schedule slippage and cost overrun, are all part of the risk analysis

process. At the conclusion of the first iteration, the customer analyses software and offers comments after testing the build.

The following illustration is a representation of the Spiral Model, listing the activities in each phase.

Based on the customer evaluation, the software development process enters the next iteration and subsequently follows the linear approach to implement the feedback suggested by the customer. The process of iterations along the spiral continues throughout the life of the software.



3.4 Spiral Model Working

The Spiral Model is widely used in the software industry as it is in sync with the natural development process of any product, i.e. learning with maturity which involves minimum risk for the customer as well as the development firms.

The following pointers explain the typical uses of a Spiral Model −

When there is a budget constraint and risk evaluation is important.

- For medium to high-risk projects.
- Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
- Customer is not sure of their requirements which is usually the case.
- New product line which should be released in phases to get enough customer feedback.
- Significant changes are expected in the product during the development cycle.

3.5 Model Pro's and Con's

The advantage of spiral lifecycle model is that it allows elements of the product to be added in, when they become available or known. This assures that there is no conflict with previous requirements and design.

This method is consistent with approaches that have multiple software builds and releases which allows making an orderly transition to a maintenance activity. Another positive aspect of this method is that the spiral model forces an early user involvement in the system development effort.

On the other side, it takes a very strict management to complete such products and there is a risk of running the spiral in an indefinite loop. So, the discipline of change and the extent of taking change requests is very important to develop

and deploy the product successfully.

The advantages of the Spiral SDLC Model are as follows

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts canbe developed earlier which helps in better risk management.

The disadvantages of the Spiral SDLC Model are as follows

- Management is more complex.
- End of the project may not be known early.
- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex.
- Spiral may go on indefinitely.
- Large number of intermediate stages requires excessive documentation.

CHAPTER 4
(RISK AND RISK MANAGEMENT)

4.1 Risk

The majority risks that are encountered during the development of a project are

Cost risk

Schedule risk Performance risk Strategic risks Operational risk

Cost Risk

- There are risks that will cost the project money
- The project is too risky from a cost aspect
- There is a risk that project costs could go higher than planned
- There is a risk that the cost of the project will be lower than planned

Schedule Risk

Schedule risk in project management is a critical aspect of risk to consider when executing a capital project. Project timeline delays often occur due to poor planning as it pertains to scope development and project execution strategies. Therefore, managing schedule risk depends highly on a contractor's ability to accurately and effectively develop project scope and a project execution strategy during front-end planning.It can be both costly and frustrating for project owners when a capital project is delayed. Delays can happen for reasons beyond an owner's control. Weather, climate, and unexpected site conditions can set a project back by days or weeks. However, more often than not, delays occur because of poor front-end planning at the beginning of the project.

Some of the most common schedule risks in project management include:

- **Poor scope definition:** Delays can occur when the project's scope isn't clearly defined or the project execution plan is inadequate. It's important to define the scope as much as possible during the first stage of the project to ensure project success.
- **Change in project scope:** Anytime a plan is changed, and a change order is required, time is needed to assess, review, and implement the change.
- **Project complexity:** When the project has a high degree of complexity, there is an increased likelihood of items being missed during scope development. This oversight can cause multiple change orders to occur during project execution. Change orders often cause schedule delays as well as cost overruns.
- **Insufficient communication:** A lack of communication with project stakeholders and contractors can tremendously affect the success of your project. Inadequate communication can result in:
    1. Limited or no buy-in and dedication to the project from owner management.
    2. Failure to correctly interpret stakeholder expectations on what is considered project success. potential rework can lead to fabrication and construction delays.
- **Inaccurate engineering estimates:** Underestimating the cost of the project at the beginning can lead to cost overruns. If it's found during the design phase that more material and equipment quantities are needed to execute the project, the schedule will inevitably be expanded. More quantities lead to longer installation times as well as more required labor. Allowing the construction team to participate in the design efforts helps to ensure the schedule stays on track.

- **Performance Risk**

This project risk is not the fault of any one party, which makes it especially daunting. Performance risk is simply the risk that the project won't produce the results and benefits outlined in the project specifications. Even if you keep costs within budget and stick to the schedule, performance risk can mean that you've lost time and money on a project that ultimately did not deliver.

- **Strategic Risk**

Strategic risk refers to the internal and external events that may make it difficult, or even impossible, for a project to achieve their objectives and strategic goals. These risks can have severe consequences that impact the whole project in the long term.

- **Operational Risk**

Operational risk summarizes the uncertainties and hazards that project creatorA faces when it attempts to do its day-to-day activities. A type of business risk, it can result from breakdowns in internal procedures, people and systems—as opposed to problems incurred from external forces, such as political or economic events, or inherent to the entire market or market segment, known as systematic risk.

4.2 Risk Management

Project risk management is the process of identifying, analyzing and responding to any risk that arises over the life cycle of a project to help the project remain on track and meet its goal. Risk management isn't reactive only; it should be part of the planning process to figure out the risk that might happen in the project and how to control that risk if it in fact occurs.

- **FOR COST -** Project expenses should be pre planned before you start creating a project.

- **FOR SCHEDULE -**Project creating task must be divide into multiple phases so that the project work can be done in scheduled time and each phase work must be done in a given time period.
- **FOR PERFORMANCE -**Multiple Testing should be done on the project time to time to enhance the performance of the project.
- **FOR STRATEGY-**Preparing a Strategy plan before starting a project helps a project from upcoming risks.
- **FOR OPERATIONAL -**A back-up plan must be prepared for every risk aproject can face during its development.
- **Design errors:** If a construction team encounters an error in the design, additional time will be needed to revamp and reassess.

CHAPTER 5
(SYSTEM ANALYSIS)

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem solving technique that improves thesystem and ensures that all the components of the system work efficiently to accomplish their purpose.

System Analysis mainly focuses on:

- Systems
- Processes
- Technology

5.1 System

The word System is derived from Greek word System, which means an organized relationship between any set of components to achieve some common cause or objective.

A system is “an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal.”

Constraints of a System

- A system must have three basic constraints

- A system must have some **structure and behavior** which is designed to achieve a predefined objective.
- **Interconnectivity** and **interdependence** must exist among the system components.
- The **objectives of the organization** have a **higher priority** than the objectives of its subsystems.

5.1.1 Processor(s)

The processor is the element of a system that involves the actual transformation of input into output.

It is the operational component of a system. Processors may modify the input either totally or partially, depending on the output specification.

As the output specifications change, so does the processing. In some cases, input is also modified to enable the processor for handling the transformation.

5.1.2 HARDWARE

Hardware refers to the external and internal devices and equipment that enable you to perform major functions such as input, output, storage, communication, processing, and more. There are two types of computer hardware: external and internal. External hardware devices include monitors, keyboards, printers, and scanners, whereas internal hardware devices include motherboards, hard drives, and RAM.

5.1.3 SOFTWARE

Software is a set of instructions, data or programs used to operate computers and execute specific tasks. It is the opposite of hardware, which describes the physicalaspects of a computer. Software is a generic term used to refer to applications, scripts and programs that run on a device. It can be thought of as thevariable part of a computer, while hardware is the invariable part.

The two main categories of software are application software and system software. An application is software that fulfills a specific need or performs tasks. System software is designed to run a computer's hardware and provides a platform for applications to run on top of.

Other types of software include programming software, which provides the programming tools software developers need; middleware, which sits between system software and applications; and driver software, which operates computerdevices and peripherals.

Early software was written for specific computers and sold with the hardware it ran on. In the 1980s, software began to be sold on floppy disks, and later on CDs and DVDs. Today, most software is purchased and directly downloaded over the internet. Software can be found on vendor websites or application service provider websites

SOFTWARE AND HARDWARE USED FOR OUR PROJECT

1. HARDWARES

- **RAM –** 8GB
- **SSD –** 256 GB
- **PROCCESSOE –** Apple M2 Silicon

2. SOFTWARES and LANGUAGES

- **VS Code**
- **HTML/CSS**
- **JAVASCRIPT**
- **PHP**

Chapter 6
(PROJECT REASONING)

6.1 PRIMARY REASON TO CHOOSE THE PROJECT

- Cost-Effective: Employee referral programs are a cost-effective way of recruiting new talent. Compared to other methods such as job boards or recruitment agencies, employee referrals are relatively cheaper since you only need to pay the referring employee a bonus or commission.
- Higher Quality Candidates: Employee referrals can result in higher quality candidates as they are already familiar with the company culture and have been vouched for by an existing employee. These candidates are more likely to fit in with the team and stay with the company longer.
- Faster Hiring Process: Employee referrals can speed up the hiring process since referrals are typically pre-screened and pre-qualified. This can save time and resources for both the HR team and the hiring manager.
- Improved Employee Engagement: Employee referral programs can improve employee engagement as employees feel more connected to the company and its mission.
- Sure, here are some additional reasons to choose a project called "Depth" based on employee referral programs:
- Diverse Candidates: Employee referral programs can help increase diversity in the workplace. Since employees often have networks that differ from those of the HR team, referrals can bring in candidates from diverse backgrounds, leading to a more inclusive workforce.
- Positive Impact on Retention: Employees who refer their friends or family members to the company are more likely to feel invested in the success of those

new hires. This can lead to increased job satisfaction, and ultimately, higher employee retention rates.

- Enhanced Employer Branding: An effective employee referral program can improve the company's employer branding. Employees who are happy with their jobs and the company are more likely to speak positively about their experiences, which can help attract new talent.
- Easy to Track: Employee referral programs are easy to track and measure, allowing HR teams to identify which employees are making successful referrals and reward them accordingly. This can also help identify areas for improvement in the referral process.
- Scalable: Employee referral programs can be scaled up or down depending on the company's hiring needs. For example, during periods of rapid growth, increasing the bonus for successful referrals can help attract more candidates.

6.2 OBJECTIVE OF PROJECT

The specific objectives of this project

- Increasing the number of quality hires through employee referrals
- Reducing recruitment costs and time-to-hire
- Improving employee engagement and retention rates
- Increasing diversity in the candidate pool
- Enhancing the company's employer branding and reputation as a great place to work
- Developing a standardized referral program that can be scaled and replicated

across different departments and locations.

- To achieve these objectives, the project may involve various activities, such as developing clear referral guidelines, creating an effective communication plan to promote the program, designing and implementing a reward system for successful referrals, tracking and measuring program success, and providing training and support to employees to help them identify and refer suitable candidates.

6.3 SCOPE OF THIS PROJECT

- Defining the program structure: The scope of the project may include defining the structure and guidelines of the employee referral program. This may include determining the types of positions eligible for referrals, the referral bonus or incentive structure, and the process for submitting referrals.
- Developing communication and promotion strategy: The project may involve developing a communication and promotion strategy to inform employees about the referral program and encourage their participation. This may include creating marketing materials such as brochures, posters, and email templates.
- Establishing tracking and reporting mechanisms: To ensure the program's success, the project may involve establishing tracking and reporting mechanisms to monitor the referral process's progress. This may include creating a database to store referral information, tracking the status of referrals, and measuring program effectiveness.
- Creating training and support resources: The project may involve developing training and support resources to help employees identify and refer suitable candidates. This may include providing training on how to assess candidate fit, how to communicate the program to their network, and how to submit referrals.
- Evaluating and adjusting the program: The scope of the project may also include

evaluating the program's success and adjusting it accordingly. This may involve analyzing referral data to identify areas for improvement, soliciting feedback from employees, and making changes to the program's structure or incentives.

free Overall, the scope of the project would be to create a comprehensive and effective employee referral program that can help the company attract high-quality talent and improve its recruitment process while also enhancing employee engagement and retention rate.

Chapter 7

(SYSTEM DESIGN)

7.1 ER DIAGRAM

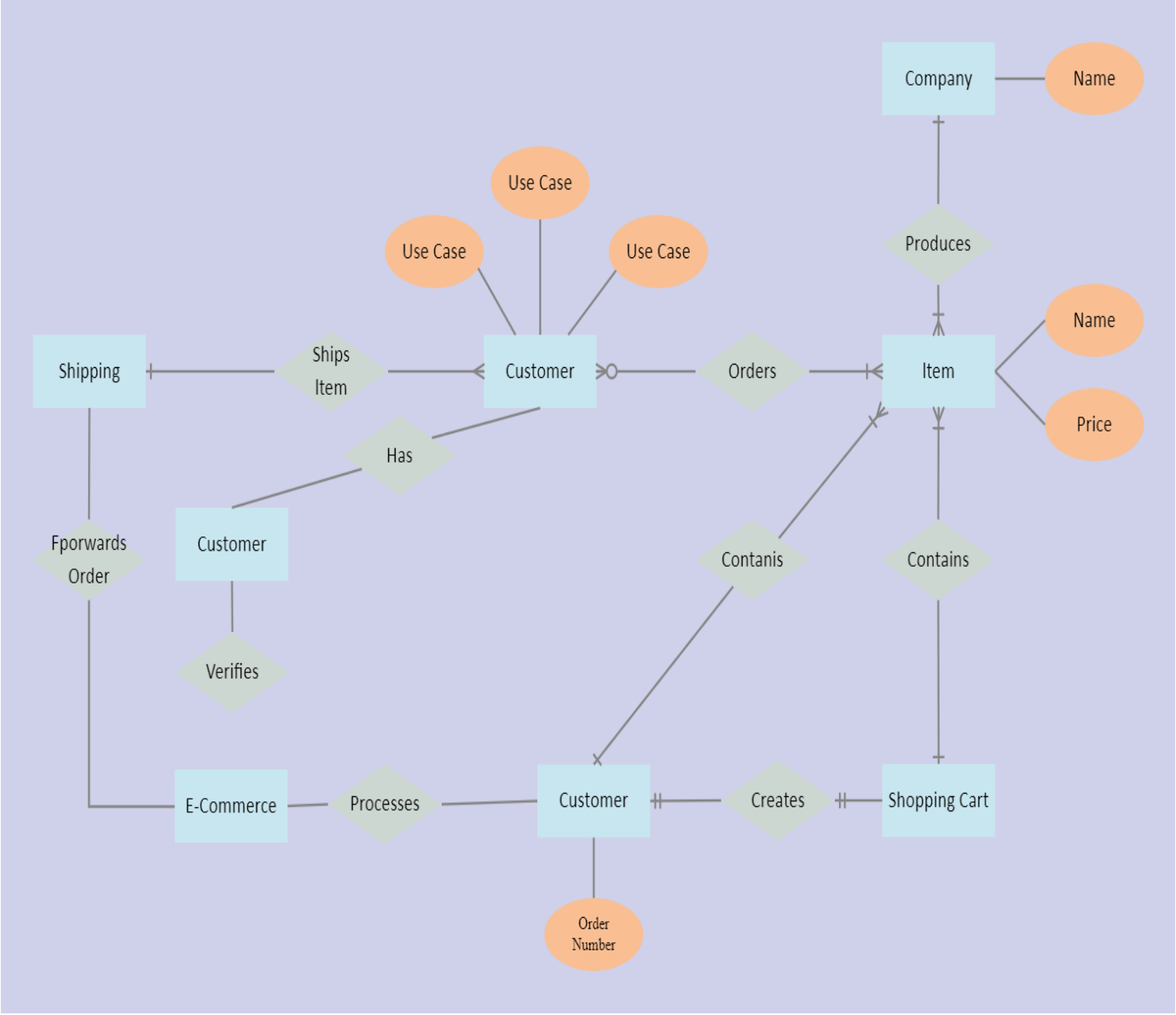


FIG 7.1

LEVEL 0 DFD OF PROJECT

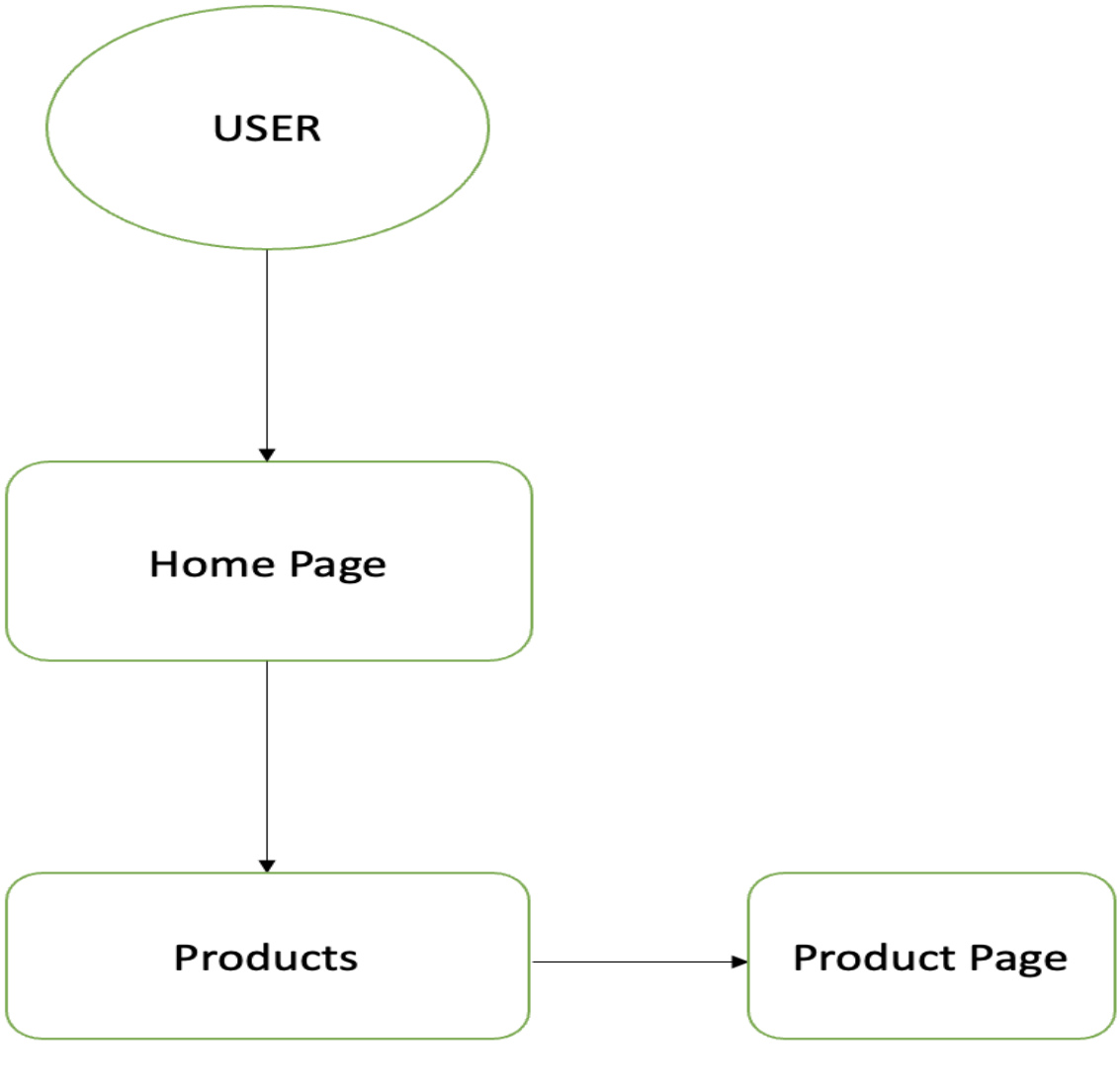


FIG 7.2

LEVEL 1 DFD OF PROJECT

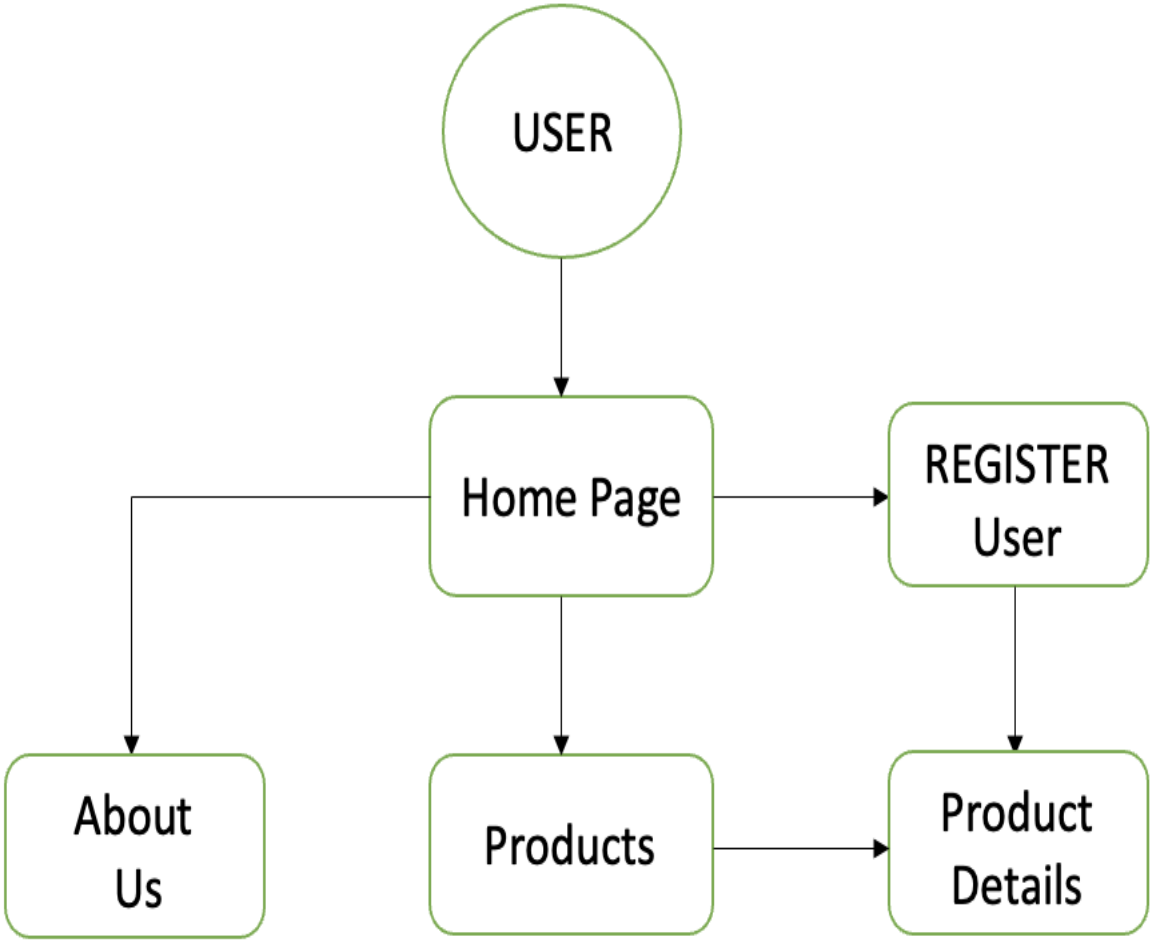


FIG 7.3

LEVEL 2 DFD OF PROJECT

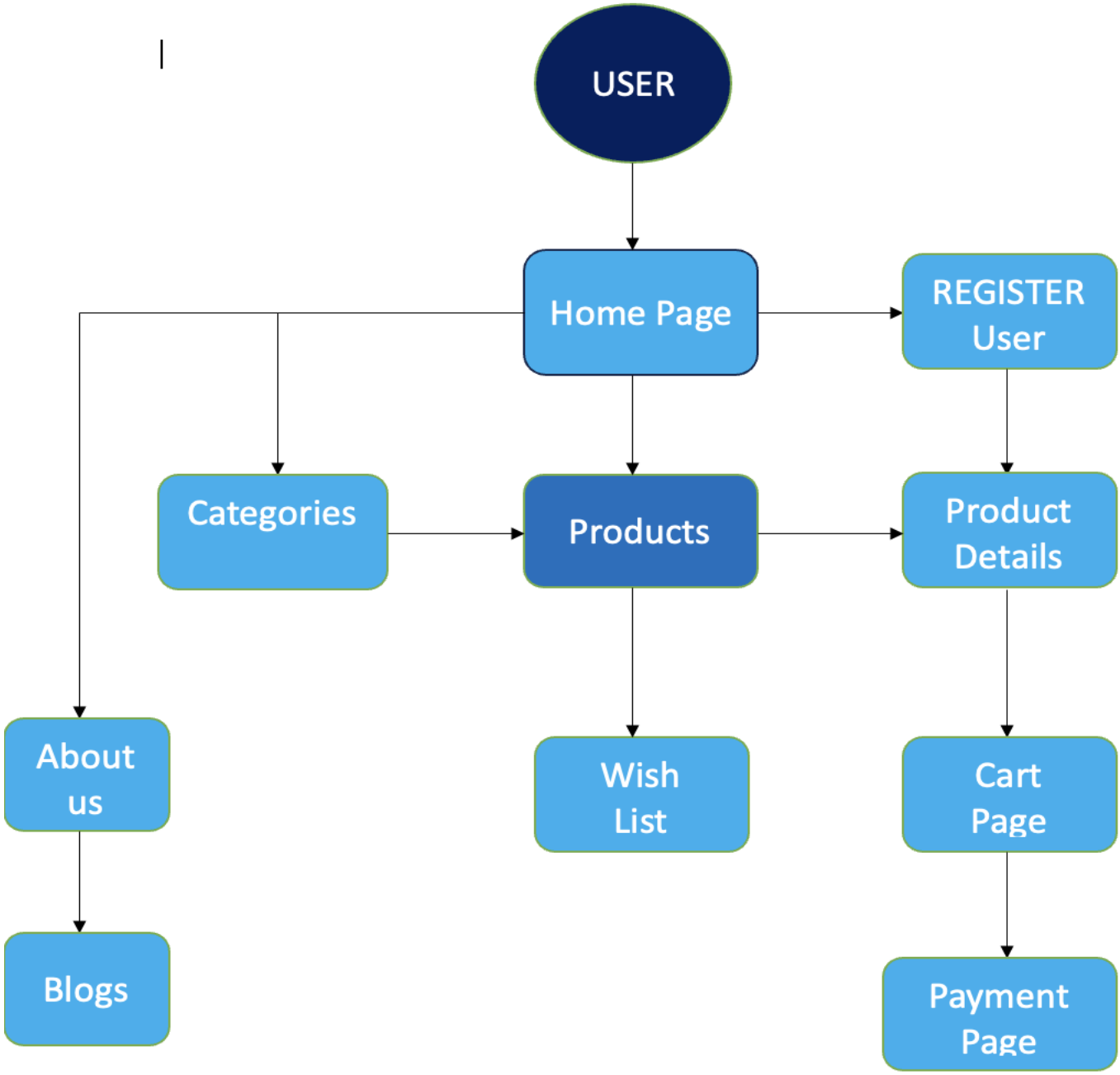
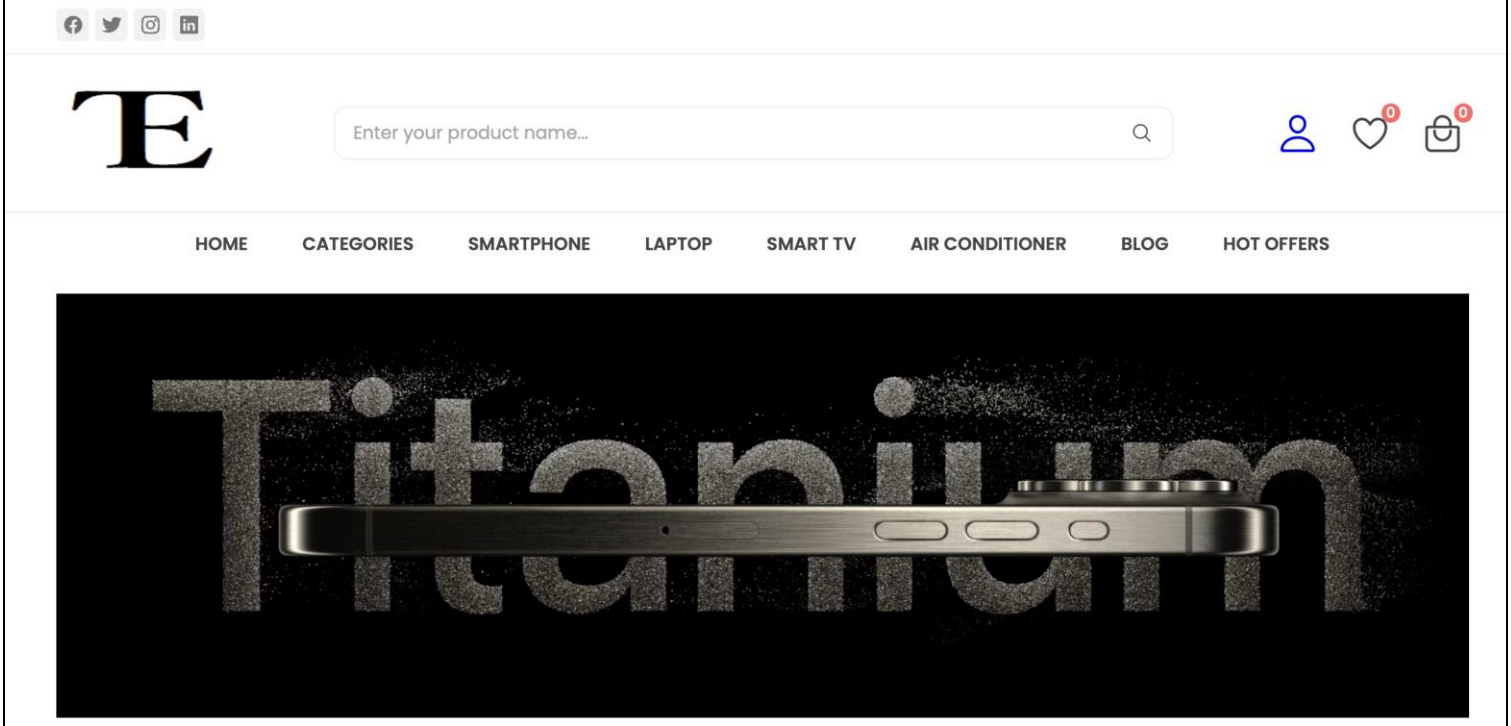


FIG 7.4

CHAPTER 8

(SCREEN DESIGN)

8.1 HOME SCREEN

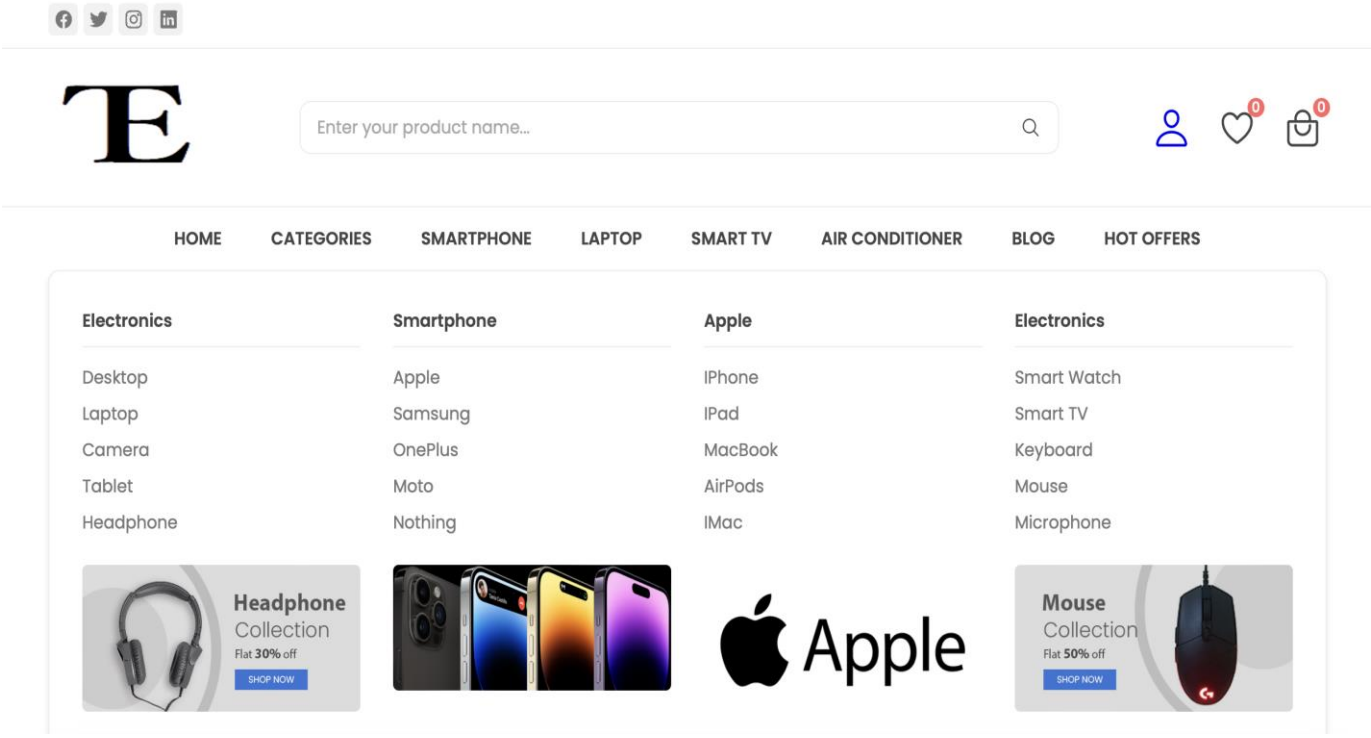


Home Screen: This is the main screen of the website that contain almost every element like Navbar, and all other sections.

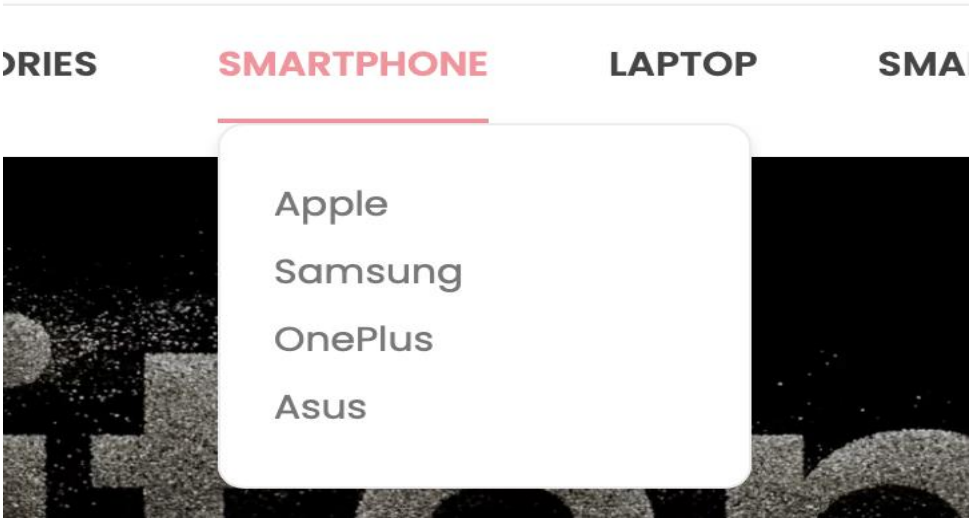
Navbar: Navbar consists of 8 elements that place on Home Screen.

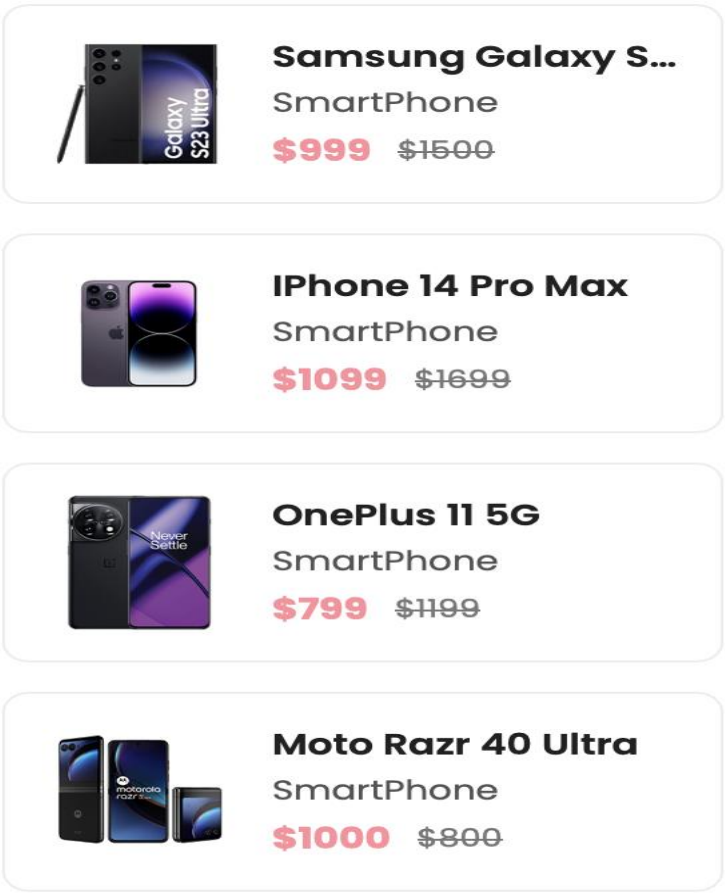
- **HOME**
- **CATEGORIES**
- **SMARTPHONE**
- **LAPTOP**
- **SMART TV**
- **AIR CONDITIONER**
- **BLOG**
- **HOT OFFERS**

CATEGORIES SECTION

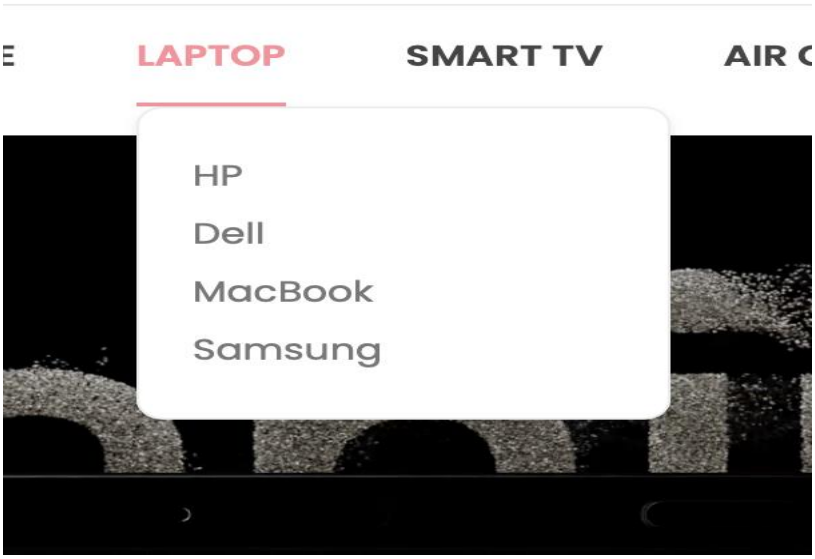


SMARTHPHONE SECTION





LAPTOP SECTION



HP Envy 15
Laptop
$1999 ~~$2500~~

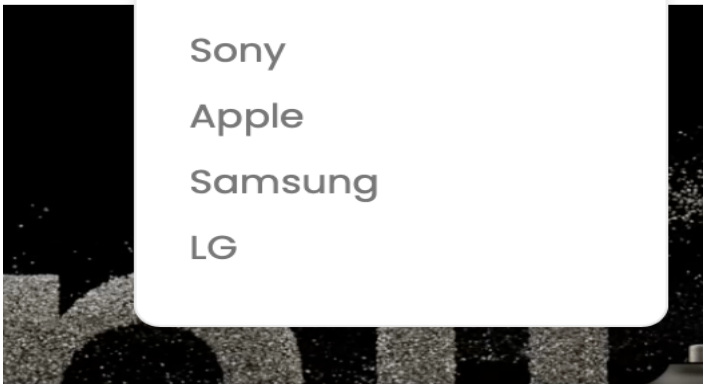
Samsung Galaxy B...
Laptop
$1200 ~~$1400~~

Dell XPS 9300
Laptop
$1000 ~~$1300~~

Samsung Galaxy B...
Laptop
$1400 ~~$1700~~

SMART TV

SMART TV AIR CONDITI



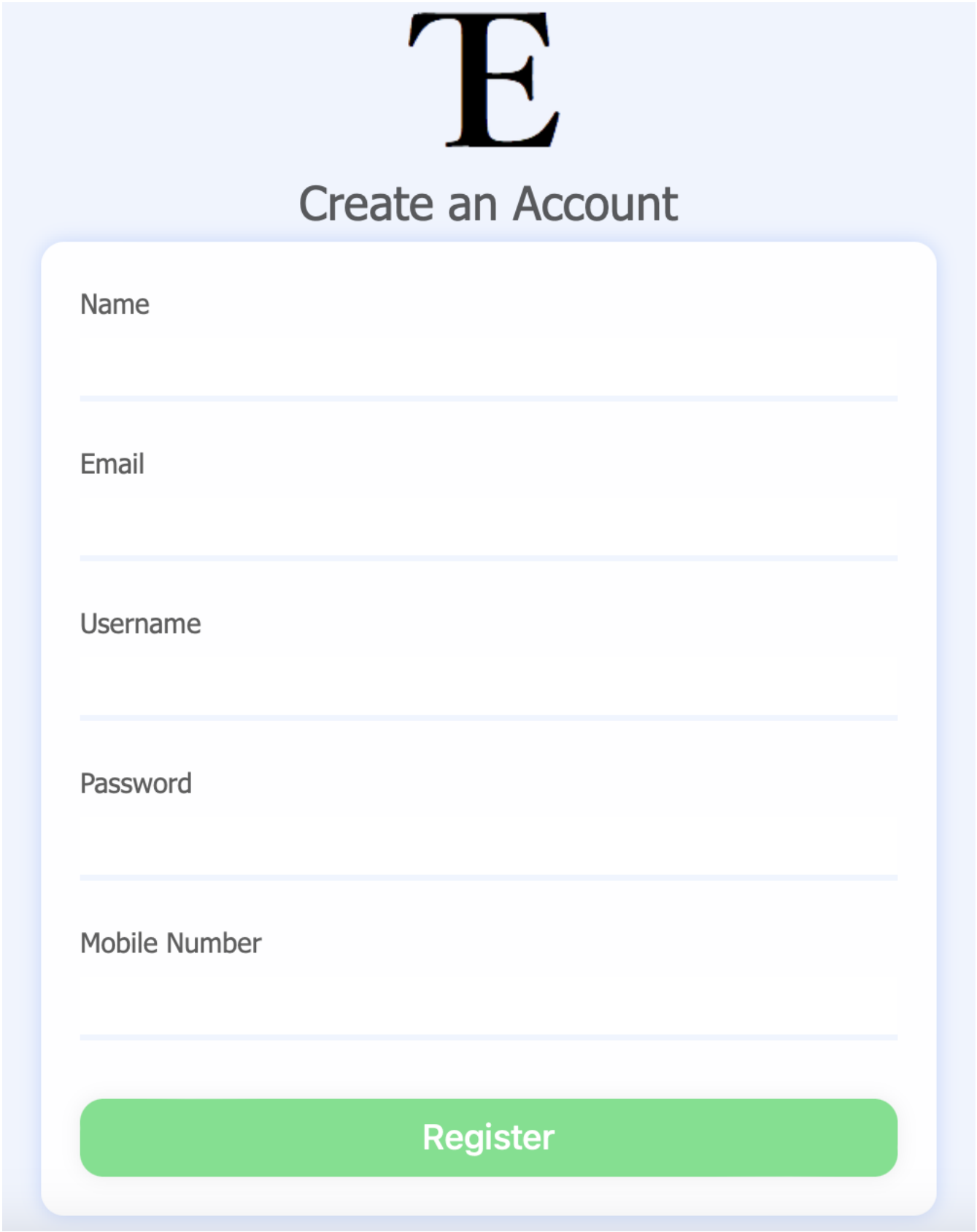
Sony Bravia 4K TV
TV
$800 ~~$1000~~

Samsung 4K TV
TV
$1300 ~~$2200~~

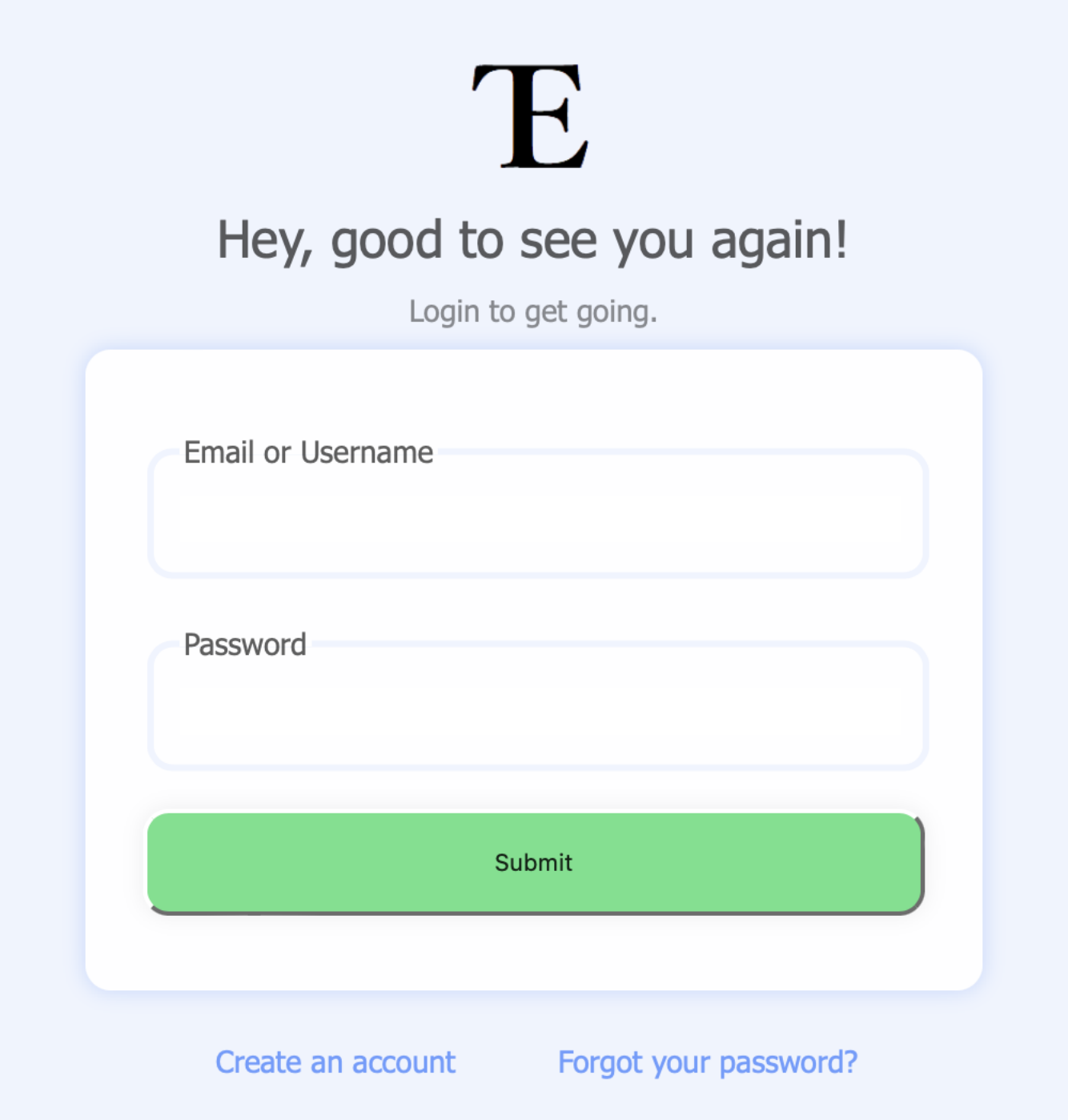
LG OLED TV
TV
$1600 ~~$2250~~

Samsung Odessey
TV
$2000.00 ~~$5000~~

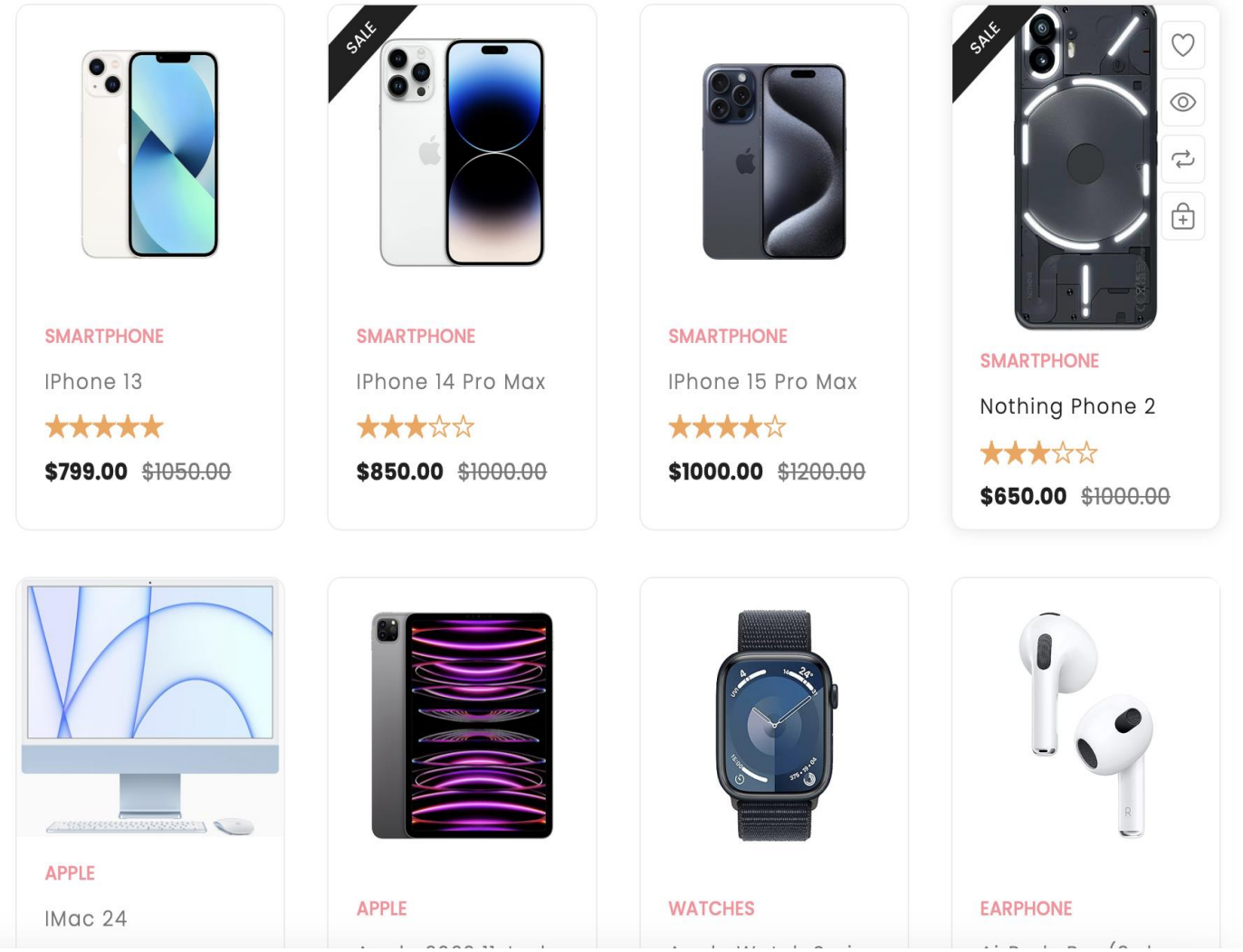
8.2 REGISTRATION PAGE



8.3 LOGIN PAGE



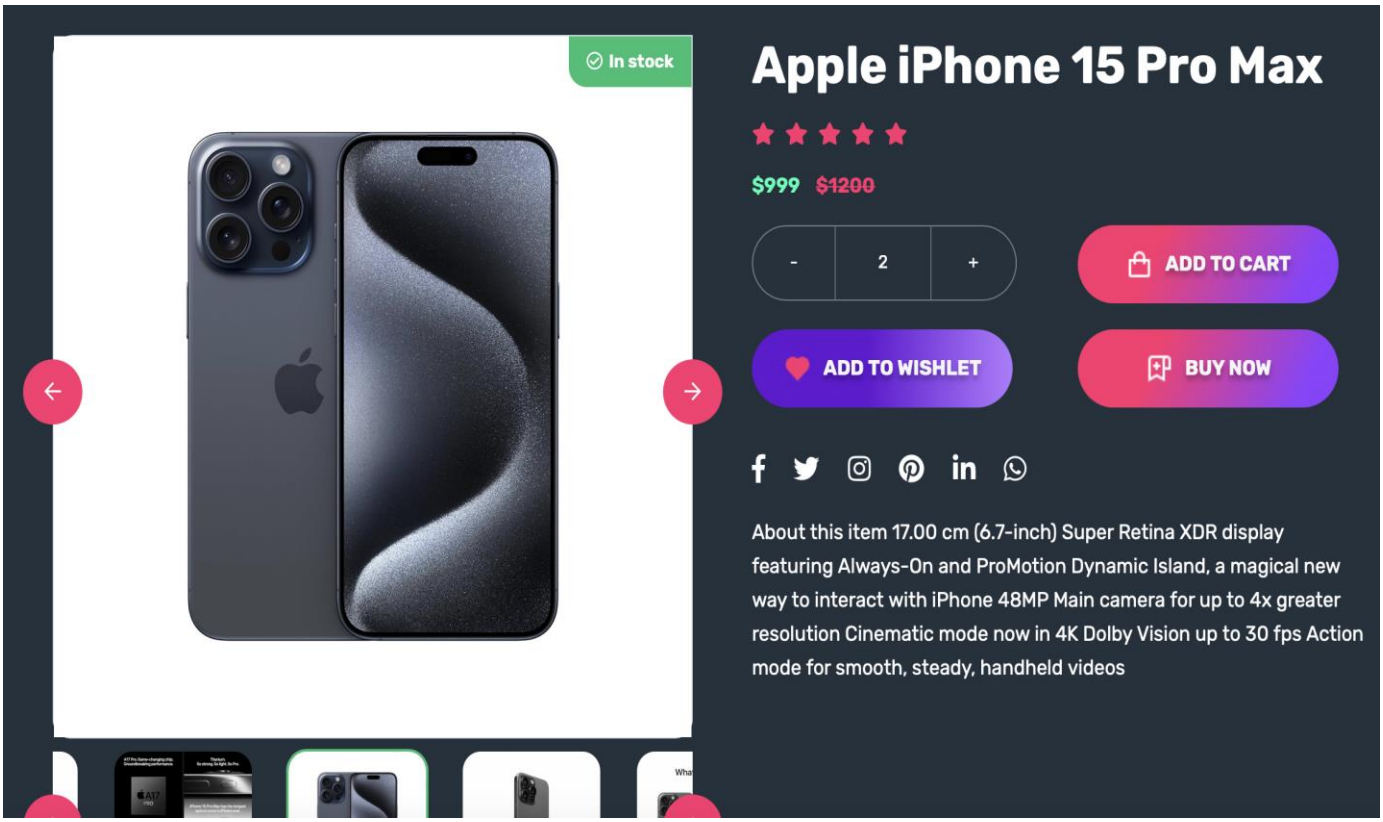
8.4 PRODUCTS



PRODUCTS PAGE: The Products Page in the Level 2 Data Flow Diagram serves as a user interface where individuals can seamlessly interact with the system to explore and manage a diverse array of products. It encompasses features such as a comprehensive product listing with details like name, description, and price, alongside search and filter options for efficient navigation. Users can access detailed product information, add items to their cart or wish list, and, if applicable, leave reviews and ratings. The administrative side facilitates product management tasks, including additions, updates, and removals. The Products Page also

integrates with the checkout process, linking users to finalizing transactions. Overall, it's a central hub for users to engage with the system's product-related functionalities, enhancing their experience and facilitating effective product exploration and management.

8.5 PRODUCT DETAILS

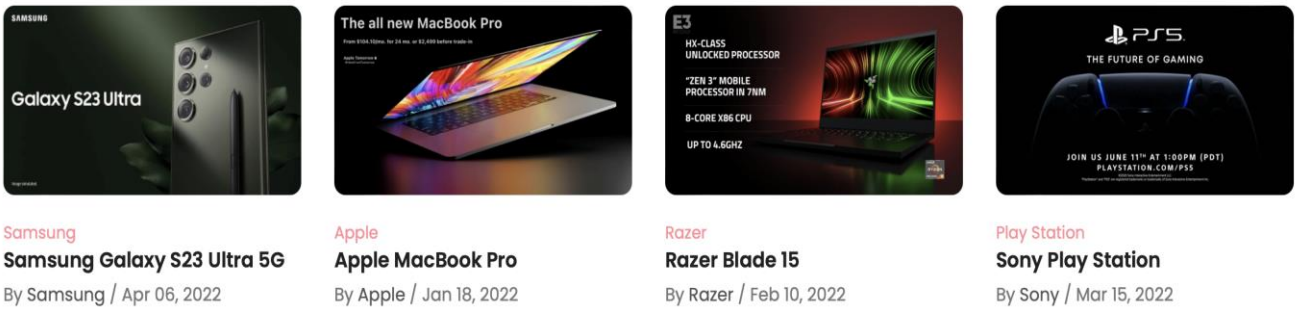


Product Detail:

The Product Detail section in the Level 2 Data Flow Diagram provides users with in-depth information about a specific product selected from the Products Page. This section typically includes a detailed view of the chosen product, presenting

specifications, customer reviews, ratings, availability status, and any associated promotions or discounts. Users can make informed decisions by examining images, reading comprehensive descriptions, and assessing the collective feedback from other users. The Product Detail interface may also offer options for users to add the item to their shopping cart directly from this view. Additionally, administrative functionalities may exist to manage and update specific product details, ensuring accuracy and relevance. The Product Detail section enhances the overall user experience by providing a comprehensive overview of individual products within the system.

8.6 BLOG



Samsung
Samsung Galaxy S23 Ultra 5G
By Samsung / Apr 06, 2022

Apple
Apple MacBook Pro
By Apple / Jan 18, 2022

Razer
Razer Blade 15
By Razer / Feb 10, 2022

Play Station
Sony Play Station
By Sony / Mar 15, 2022

Blog Page:

In the Level 2 Data Flow Diagram, the Blog Page to Latest Products serves as a dynamic component for disseminating information about the latest products within the system. This section allows users to stay updated on recent releases, promotions, and noteworthy additions. The Blog Page typically features engaging content such as articles, posts, or visual media that highlight key product launches, showcase unique features, and provide insights into the development process. Users can navigate through these blog entries to discover new products and gain a

deeper understanding of their functionalities. The integration of the Latest Products on the Blog Page creates a seamless bridge between product updates and user engagement, fostering a more informed and connected user community. Administrative functionalities may include content creation and management, ensuring that the latest product information is accurately and attractively presented to the audience.

CHAPTER - 9 (TESTING)

- Web testing, or web application testing, is a software practice that ensures quality by testing that the functionality of a given web application is working as intended or as per the requirements.
- Web application testing allows you to find bugs at any given time, prior to a release, or on a day-to-day basis.
- Testing is a highly important part of software development.
- Creating a top-notch web application requires a lot of testing. If functionality testing is done manually, it can become tedious and time-consuming. For this reason, many QA teams rely on automated testing to create fast, efficient, and reliable test cases for their web applications.

WEBSITE TESTING LIFECYLE

Testers use a web testing approach to evaluate apps launched using a web browser on the internet. In web page testing, QA testers carry out a set of processes before the actual testing.

REQUIREMENT GATHERING

In this phase, testers collect all requirements related to the feature and refine them and identify the gaps.

TEST PLANNING

In this phase, testers update the test plan document with a definition of the test scope and objectives, strategy, the entry and exit criteria for web testing, and an evaluation of the test estimate.

TEST CASE PREPARATION

QA testers generate test scenarios and scripts using the most appropriate automation technique considering the test strategy as a guide.

TEST EXECUTION

At this stage, testers run the defined test cases and document any deviations from the intended result.

BUGS REPORTING

If a test case fails during execution and a bug is detected, testers raise and report it using a defect tracking tool like Jira and HP ALM QC.

DEFECT RETESTING

Once the developer has fixed the defect, testers retest it and re-execute the failed test case.

TEST CLOSURE

The test cycle is closed once all defects are fixed and the web application is functioning as expected. If not, QA testers repeat the process.

9.1 TESTING METHODBLACKBOX TESTING

Black box testing is a type of software testing in which the functionality of the software is not known. The testing is done without the internal knowledge of the products. Black box testing is a technique of software testing which examines

the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a specification of requirements that is stated by the customer.

In this method, tester selects a function and gives input value to examine its functionality, and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed. The test team reports the result to the development team and then tests the next function. After completing testing of all functions if there are severe problems, then it is given back to the development team for correction.

STEPS OF BLACK BOX TESTING

- The black box test is based on the specification of requirements, so it is examined in the beginning.
- In the second step, the tester creates a positive test scenario and an adverse test scenario by selecting valid and invalid input values to check that the software is processing them correctly or incorrectly.
- In the third step, the tester develops various test cases such as decision table, all pairs test, equivalent division, error estimation, cause-effect graph, etc.
- The fourth phase includes the execution of all test cases.
- In the fifth step, the tester compares the expected output against the actual output.
- In the sixth and final step, if there is any flaw in the software, then it is cured and tested again.

TEST PROCEDURE

The test procedure of black box testing is a kind of process in which the tester has specific knowledge about the software's work, and it develops test cases to check the accuracy of the software's functionality.

It does not require programming knowledge of the software. All test cases are designed by considering the input and output of a particular function. A tester knows about the definite output of a particular input, but not about how the result is arising. There are various techniques used in black box testing for testing like decision table technique, boundary value analysis technique, state transition, All-pair testing, cause-effect graph technique, equivalence partitioning technique, error guessing technique, use case technique and user story technique. All these techniques have been explained in detail within the tutorial.

TEST CASES

Test cases are created considering the specification of the requirements. These test cases are generally created from working descriptions of the software including requirements, design parameters, and other specifications. For the testing, the test designer selects both positive test scenario by taking valid input values and adverse test scenario by taking invalid input values to determine the correct output. Test cases are mainly designed for functional testing but can also be used for non-functional testing. Test cases are designed by the testing team, there is not any involvement of the development team of software.

TEST AUTOMATION

- Test automation offloads these routine and repetitive testing tasks from humans to machines. The tests compare actual outcomes with predicted outcomes. This approach can help find bugs in specific operations and simple-use cases, like logging into your ERP system, creating a new account and doing password resets.
- By automating web application tests, testers can save time and effort on monotonous tasks. Automated tests can be run continuously or scheduled at intervals. This offloads testers from time-consuming tasks, and they can focus on exploratory testing and usability testing or other tests that require a human. perspective.
- Test automation is the use of software (separate from the software under test) to control the execution of tests. It lets software robots perform repetitive tasks and emulate end-user interaction with the system under test, to increase the range, depth, and reliability of one's quality assurance efforts.
- Still, automation in not a plug-and-play system that requires no human intervention. Automated testing requires testers to have a thorough knowledge of the software under test, as well as an "automation first" mindset.
- To be able to test software with automation, it requires testing tools that enable both technical and non-technical testers can use. We've put together this short article comparing web automation tools for web applications.

TESTING TECHNIQUES:

FUNCTIONAL TESTING

Functional testing is used to ensure that the functionality of the software OR Project works as intended for an end-user.

A single end-user can make the whole system crash in minutes, even after unit, integration, and performance tests have passed. This usually happens because the user does something the developers did not expect.

Automating tests ensures that a web application is working as intended. However, functional test automation cannot fully replace the human perspective. That's why you should always supplement your automated test cases with exploratory testing to provide a positive user experience.

There are many types of functional tests. Regression testing is one of them. Examples of functional tests on a web application that could be part of a regression testing suite could be:

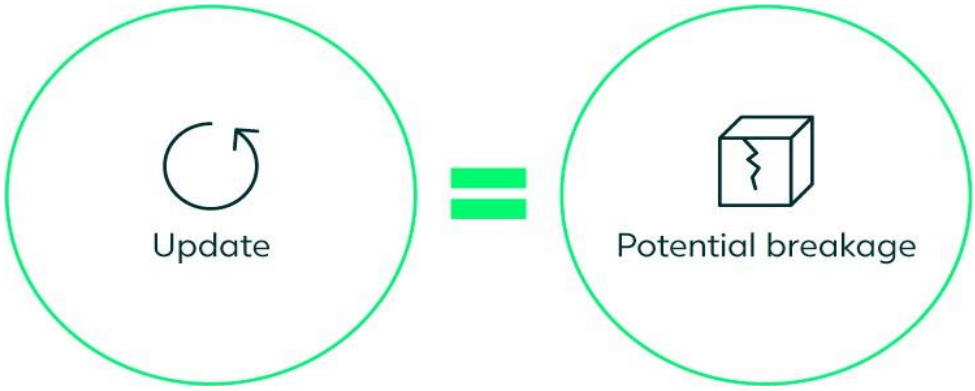
1. Checking that the login to your web application is successful.
2. Checking that the login to your web application is successful across browsers and devices.
3. Forms with multiple fields are populated with the right data.
4. The web application is interacting with external databases and syncing successfully.
5. Invoices are being sent and received with the correct information and securely.

6. The functionality of buttons across pages are working as per the requirements.

To learn even more about automated functional UI testing, you can get immediate access to this whitepaper on Functional UI testing: An introduction to codeless test automation.

REGRESSION TESTING

Regression testing can be described as “repeated functional testing”. It is used to make sure that a software’s functionality continues to work after parts of it have been modified with new code or configurations. For instance, when new features are built, regression testing ensures that old features of the software continue to work as intended.



Automated regression testing uses computer-based tools and techniques to test software that has been changed or updated.

When regression testing is automated, you can quickly and reliably run through simple scenarios and check a variety of changes to get fast feedback. This, in

turn, frees up time for testers to focus on product improvements or conduct manual exploration into more unusual cases that require special attention.

CROSS-BROWSER TESTING.

Cross-browser testing ensures that your web application is performing as expected across different browsers, both on desktop and mobile.

Browsers tend to be updated fairly frequently, meaning that by the time you are ready to deploy your application, it might not work as intended in your target browsers.

Users might have the latest browser version while using an old operating system. By automating cross-browser testing, you make sure that incompatibilities like the one just mentioned are found even before they reach an end-user.

PERFORMANCE TESTING

Performance testing, such as stress and load testing, ensures that a web application can endure extended periods of activity or peak user loads. Reaching the necessary stress conditions or load level wouldn't be feasible if done manually, therefore automation is key in proving that your application can perform in any situation.

WEB UI TESTING

Critical components of a web app are the webserver interface, database server interface, and application server interface. Web UI testing validates that all components of the web application are well-connected. Testers should use this testing method to determine whether the interaction between these servers is carried out effectively.

USABILITY TESTING

The application's user interface should adhere to industry standards regarding effectiveness and user-friendliness. It is also critical to adhere to global conventions and web standards while developing a web application. Usability testing is appropriate for apps that aim to automate manual processes.

However, testers must consider specific critical issues such as correct navigation, a site map, and avoiding over-crowded content while performing usability testing.

COMPATIBILITY TESTING

Compatibility is a critical factor to be considered while testing a web application. Compatibility testing determines the compatibility of a web app with operating systems, browsers, mobile browsing, and other printing options.

SECURITY TESTING

Security testing is a critical web app automation testing method as it determines whether data modifications or leaks are tolerable. It primarily includes various

activities such as verifying SSL, checking the CAPTCHA for automated script logins, and determining whether web files or directories can be accessed directly.

INTERFACE TESTING

This testing method ensures that the three main components of a web application which are web server, web browser and database are running harmoniously. This testing type checks whether there is any interruption while the data is being transferred. Upon that, the communication taking place between various interfaces is also thoroughly checked.

NEED OF TESTING

Compatibility across browsers

The practice of web application testing helps the applications to be compatible across all web browsers. Every user uses different browsers while surfing websites. The practice of web application testing effectively helps the apps to function the same for every user using different browsers.

Improves website performance.

The methodology of web application testing successfully helps to overcome the slow app performance. Applications with slow performance are not the choice for any user and Google even. Web app testing helps to resolve the load time delays caused due to the graphics, code, etc. and maintains a quality application.

Secures project from vulnerabilities.

Security is one of the important concerns for every application. Day-by-day the rate of website hacks is excessively increasing, so there is an immediate need for every small and large enterprises to practice the web app testing to ensure that the applications are free from the different kinds of threats.

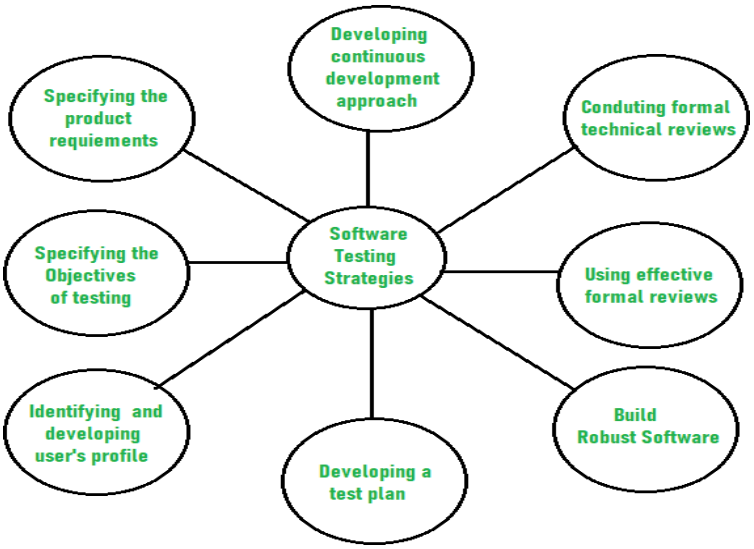
Ensures high quality of the application:

The end-to-end testing practice effectively helps in enhancing the performance and functionality of the app. And this way of testing helps to find and resolve the bugs at earliest. Thus, a high-quality app will be achieved before the release.

Reduces the Time and Cost Consumption:

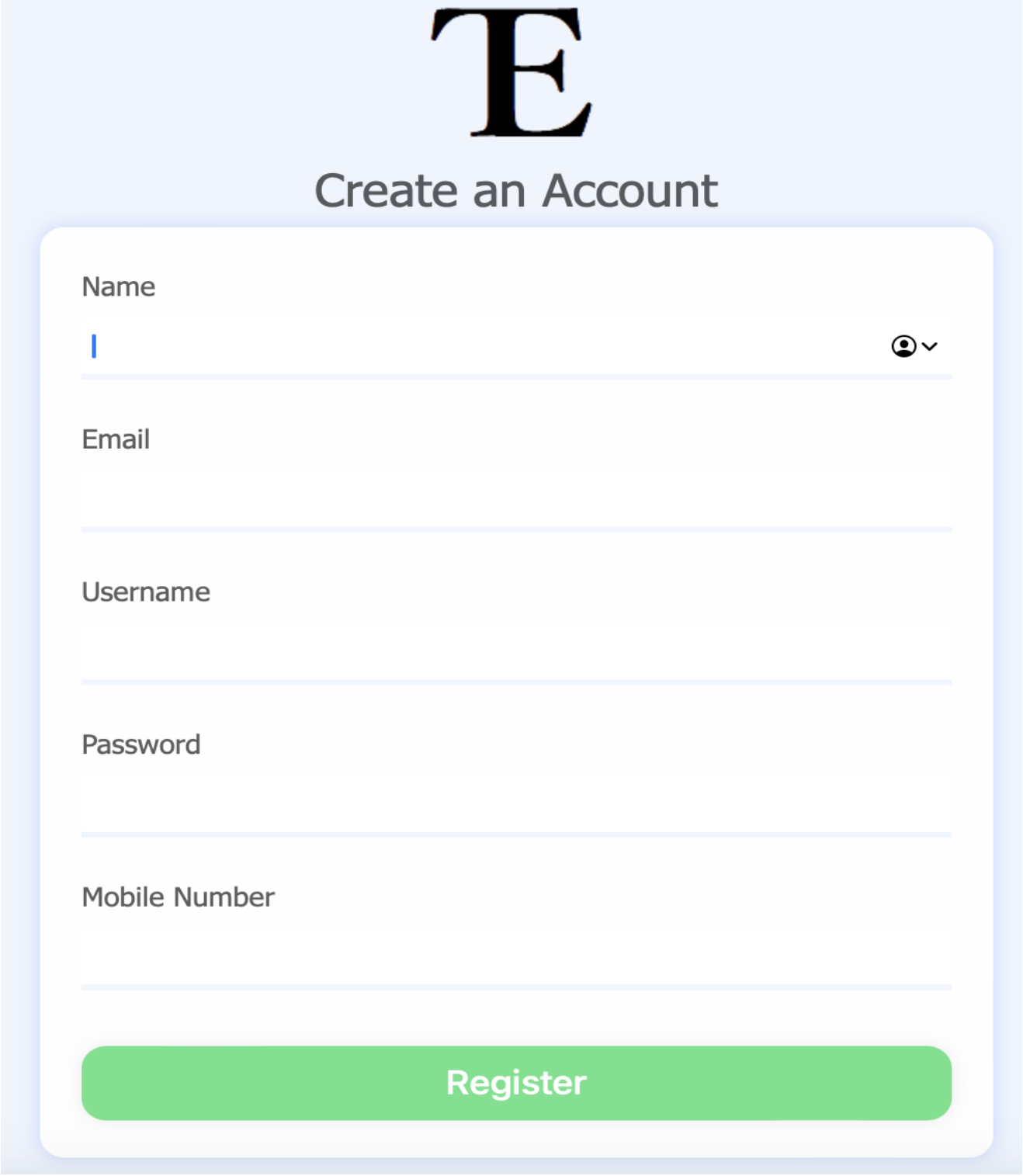
The end-to-end testing practice through automation tools reduces the occurrence of errors and the performance of repetitive test cases. This will help the enterprises to achieve quality results with reduced time and cost.

TESTING STRATEGIES:

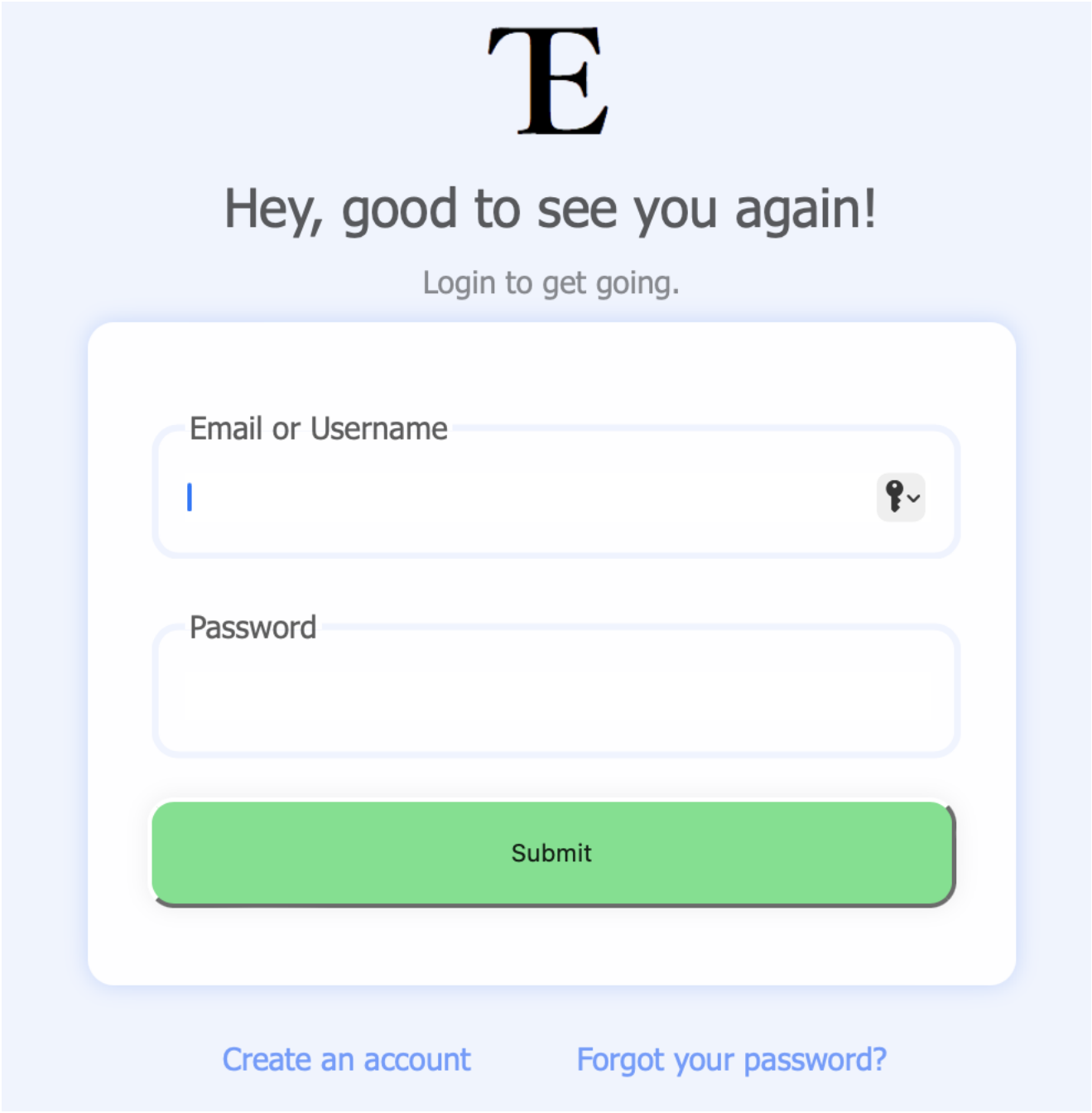


9.2 LOG-IN AND SIGN-UP

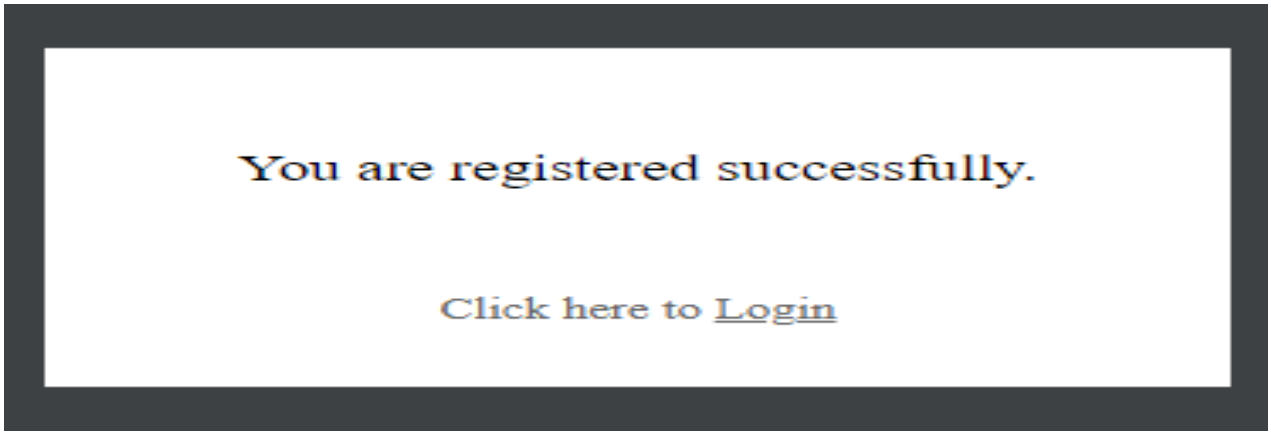
STATUS SIGN-UP



LOG-IN


Hey, good to see you again!
Login to get going.
Email or Username
Password
Submit
Create an account
Forgot your password?

DATA REGISTERTING CONFIRMATION



DATABASE

←T→	id	name	email	username	password	mobile
Edit Copy Delete	3	test2	test@123.com	admin1	$2y$10$kKffo.Ce41Mk/MHRQOIBj.UD4i88anmNR6lizjlkykm...	99990000
Edit Copy Delete	4	Devansh	tyagidevansh10@gmail.com	Devansh	$2y$10$vp/3gMkJ51vvQeDshbdHFeivDE8aYAZOWIxkZPUjCbx...	99009900
Edit Copy Delete	5	admin	admin@123.com	admin	$2y$10$j4bImnZOH6.jtBCaIEMXUeRI3hHIBGQEyHEpEG4mRYy...	999900
Edit Copy Delete	6	Shivam	mor@123.com	Leon	$2y$10$xG/3XTs6/URcLfZU8rFh4ur2pyyZA./WDWhZUWrIDU2...	999999

9.3 DATABASE-MODULE TABLE

Username	Password	Result
Incorrect	Incorrect	**Login Failure**
Correct	Incorrect	**Login Failure**
Incorrect	Correct	**Login Failure**
Correct	Correct	**Login Success**

CHAPTER 10
(LIMITATIONS)

1. **Limited Product Variety:**

- The website's initial launch may feature a limited range of products due to constraints in sourcing, cataloging, or onboarding products from suppliers.

2. **User Accessibility Challenges:**

- Accessibility features may not be fully optimized for all users, potentially limiting access for individuals with disabilities and impacting overall inclusivity.

3. **Security Concerns:**

- The project may face security vulnerabilities such as potential data breaches, especially during the initial stages of implementation, which could affect user trust.

4. **Scalability Issues:**

- The website's current architecture may have limitations in scalability, potentially hindering its ability to handle a sudden surge in traffic or a growing product catalog.

5. **Limited Payment Options:**

- Initially, the website may support a limited number of payment gateways, potentially restricting user payment preferences and impacting conversion rates.

6. **Mobile Responsiveness:**

- The website may not be fully optimized for various mobile devices, leading to suboptimal user experiences on smartphones and tablets.

7. **Incomplete Feature Set:**

- Certain desired features may not have been fully implemented due to time constraints or technological limitations, impacting the overall user experience.

8. **Dependency on Third-Party Services:**

- The project might rely on external services (e.g., payment processors, delivery services) whose limitations or downtime could affect the website's functionality.

9. **Limited Marketing Integration:**

- Integration with marketing tools and strategies may be limited, potentially affecting the effectiveness of promotional activities and user acquisition.

10. **Data Integrity Challenges:**

- Issues related to data integrity, such as inaccurate product information or inventory discrepancies, may impact the reliability of the website.

CHAPTER 11
(FUTURE ENHANCEMENTS)

1. **Expanded Product Catalog:**
   - Continuously add new and diverse products to cater to a broader audience and enhance the variety offered on the platform.

2. **Advanced Search and Filtering Options:**
   - Implement advanced search algorithms and filtering options to allow users to find products more efficiently based on specific criteria such as brand, features, or price range.

3. **Personalized User Accounts:**
   - Introduce personalized user accounts with features like order history, personalized recommendations, and saved preferences to enhance the overall user experience and encourage customer loyalty.

4. **Mobile App Development:**
   - Develop a dedicated mobile application to provide users with a seamless and optimized shopping experience on smartphones and tablets.

5. **Enhanced Security Measures:**
   - Implement advanced security protocols and regular security audits to safeguard user data, instilling trust and confidence among your customer bases.

6. **Integration with social media:**
   - Integrate social media sharing and login options to facilitate social engagement, enhance brand visibility, and attract new customers through social

networks.

7. User Reviews and Ratings Improvements:
- Enhance the user reviews and ratings system, allowing customers to provide more detailed feedback, attach images, and share their experiences with purchased products.

8. Live Chat Support:
- Introduce a live chat support feature to provide real-time assistance to customers, answering queries, and resolving issues promptly.

9. Subscription Services:
- Implement subscription services for certain products, allowing customers to subscribe for regular deliveries of consumables or products with periodic updates.

10. Loyalty Programs:
- Create a customer loyalty program with rewards, discounts, or exclusive offers to incentivize repeat purchases and customer retention.

11. Geographic Expansion:
- Explore opportunities for geographic expansion by adding international shipping options to reach a broader customer base.

12. AI-Powered Recommendations:
- Implement artificial intelligence algorithms for product recommendations based on user behavior, preferences, and past purchase history.

13. Enhanced Marketing Strategies:

- Develop and implement comprehensive digital marketing strategies, including social media campaigns, email marketing, and influencer collaborations, to increase brand awareness and attract new customers.

14. Augmented Reality (AR) Features:

- Integrate augmented reality features, allowing customers to visualize products in their own space before making a purchase decision.

CHAPTER 12
(CONTRIBUTION)

1. Increased Market Presence:

- The launch of "Tyagi Electronics" significantly contributes to expanding the market presence of your brand, establishing a strong online presence in the e-commerce sector.

2. Business Revenue Growth:

- The e-commerce platform provides a new channel for sales, contributing to the growth of business revenue by reaching a wider audience and attracting online customers.

3. Job Creation:

- The development and maintenance of the website contribute to job creation, employing individuals in roles such as web development, customer service, marketing, and logistics.

4. Improved Customer Convenience:

- "Tyagi Electronics" contributes to enhancing customer convenience by providing an online platform where users can easily browse, select, and purchase products from the comfort of their homes.

5. Digital Transformation:

- The project reflects a commitment to digital transformation within your business,

adapting to modern consumer preferences and leveraging technology for better efficiency and customer engagement.

6. **Competitive Advantage:**

- The e-commerce website contributes to gaining a competitive edge in the market by offering a convenient and accessible shopping experience, potentially setting your brand apart from competitors.

7. **Customer Access to Information:**

- "Tyagi Electronics" contributes to transparency and informed decision-making by providing customers with detailed product information, reviews, and ratings, empowering them to make confident purchase decisions.

8. **Market Reach Beyond Physical Locations:**

- The online platform allows your business to reach customers beyond physical store locations, enabling a broader market reach and attracting customers from different geographic areas.

9. **Feedback Loop for Improvement:**

- Customer interactions on the website contribute to a valuable feedback loop, providing insights into customer preferences, behaviors, and areas for improvement, which can inform future enhancements and strategies.

10. **Brand Visibility and Recognition:**

- The project contributes to increased brand visibility and recognition as "Tyagi

Electronics" becomes a familiar name in the online retail landscape.

11. **Adaptation to Changing Consumer Trends:**

- By embracing e-commerce, your business contributes to adapting to changing consumer trends and preferences, aligning with the growing demand for online shopping experiences.

12. **Accessibility for a Diverse Audience:**

- The online platform contributes to making your products accessible to a diverse audience, including those who may have limitations accessing physical stores.

13. **Data-Driven Decision Making:**

- The project contributes to data-driven decision-making processes by generating valuable analytics and insights that can be used to refine marketing strategies, optimize user experiences, and streamline operations.

CHAPTER 13
(REFERENCES)

https://www.w3schools.com/html/html_computercode_elements.asp https://www.w3schools.com/css/css_examples.asp

https://www.w3schools.com/js/js_examples.asp

https://freefrontend.com/css-text-animations/

BOOKS

HTML & CSS: Design and Build Web Sites

Learning JavaScript Data Structures and Algorithms