Rachit Khanal

MODERN RESPONSIVE AND INFORMATIVE CORPORATE WEBSITE

Developing Informative Website for Akaladevi Trading Concerns.

MODERN RESPONSIVE AND INFORMATIVE CORPORATE WEBSITE

Developing Informative Website for Akaladevi Trading Concerns.

Rachit Khanal
Bachelor's Thesis
Spring 2020
Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Information Technology

Author: Rachit Khanal
Title of the bachelor's thesis: Modern Responsive and Informative Corporate
Website.
Supervisor: Kari Laitinen
Term and year of completion: Spring 2020 Number of pages: 34 + 3 appendices

---

Akaladevi Trading Concern, ADTC, is a hardware company based in Nepal. It has many branches and works in different sectors. With the demand of time, and in the age of Internet, ADTC realize website as a vital part of the business as it informs about the services to the costumers and also builds the brand. As part of experimentation ADTC wants to enter in the game with strong informative website. Responsiveness of the website is a vital part, as the consumer uses the services with both smart-phones and computers with different screen-sizes.

Initial meetings were set between both parties and project requirements were discussed. As, part of thesis, user and owner stories were noted in details. Basic mock-ups and wireframe were made several times before reaching the final layout. Balsamiq was used for developing mock-ups, Bootstraps was used for developing responsiveness with the addition of media-queries. A web-design was prepared using the basic HTML, CSS and Bootstraps. In the second phase, the code were isolated and divided into components. The project was transformed in the React. Clean and dynamic code was achieved following the component architectures and React principles.

An informative and modern looking website was developed as a part of the project, while keeping the responsiveness of the site intact. The project is further carried out into hosting phase.

Keywords: Web Development, Modern Web Technologies, Responsiveness, React, Bootstraps, HTML, CSS.

CONTENTS

VOCABULARY

| Terms	Definition
BOOTSTRAP	Front-end framework used for developing responsive web-application
CDN	Content Delivery Network
CSS	Cascading Style Sheet
CSS3	Latest version of CSS, version third
DevTools	Developer Tools
DNS	Domain Name Server
HTML	Hyper Text Mark-up Language
HTML5	Latest version of HTML
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
JS	JavaScript
NPM	Node Package Manager
PC	Personal Computer
SEO	Search Engine Optimization
TCP	Transmission Control Protocol
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience

1 INTRODUCTION

Akaladevi Trading Concern (ADTC) is the reliable name in its locality. It was founded in 1995 as a hardware company. With the change of time and demand of the country, the company has branched and increased its scope in many areas. One of the prominent sector is plumbing. The Company sells the modern day plumbing supplements, equipment and most importantly services related to plumbing.

In the modern days, having the best product and service is not everything for a corporate. Clients prefer or tend to use the service provider whose information is up to date and readily available on the Internet. Clients like to compare the services between the companies via a website. The website these days is not just for providing information but it also builds the trust in the brand.

In this thesis, the modern-day corporate level responsive website is built keeping the user interface and user experiences as a priority. The website will be dynamic meaning that it can be used by any dimension of the computer screen, tablets, mobile phone while maintaining user experiences and without compromising the functionalities.

2 MODERN WEB APPLICATION

To understand how web works, one can start with the connection between a server and a client, which can be taken as computers. Clients are devices that are connected to the Internet for consuming the information such as a phone or a laptop connected to the Internet for consuming multimedia. A server can be a computer which is responsible for storing the web contents. When a client computer requests the web content, a copy is generated from the server to the client's computer. Figure 1 demonstrates the connection between clients and server.

requests

CLIENT

responses
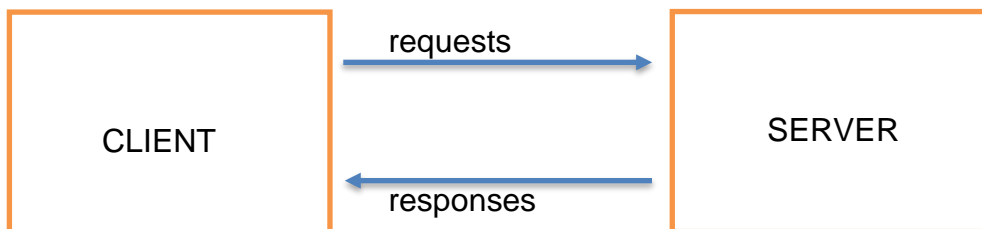
SERVER

FIGURE 1. Client and Server Interaction.

2.1 Web Components

Other than the Client and Server, there are many web components that make the connection possible[1], some of which are explained below.

2.1.1 Internet Connection

The Internet is a global wide area network which makes the connection between the computer around the globe possible[2].

2.1.2 Protocol

A protocol is a set of rules which determines the correct way to function. Many protocols are used to make the connection possible.

2.1.2.1 **TCP/IP**

TCP/IP is a protocol on how to communicate or how the data is supposed to travel across the web. These were developed by the U.S military to communicate over the long distance networks. The TCP verifies if the packets are delivered. IP tracks the data movement across the nodes[3].

2.1.2.2 **HTTP**

HTTP uses the server-client model as shown in Figure 1. When a client sends the request to the server, the server responds with HTTP status code which conveys if the request is granted, if the connection is made or if the URL is valid[4].

2.1.3 DNS

Computers access the website by their IP address. An IP address consists of numbers which are hard to remember and make no sense to a human. DNS translates the domain names into IP addresses, allowing the user to write the domain name of the website[5].

2.2 Primary client side web technologies

Modern and latest client side web technologies were used to build the project, some of which are described below.

2.2.1 HTML

HTML stands for HyperText Markup Language. It is used to define the structure of the website. Hypertext refers to links that connect the web pages to one another. Markup refers to the tags or HTML elements, such as `<head></head>`.[6] Mostly HTML elements pair with opening and closing tags. Tags are used to shape the content. the content between the tags `<p> ...... </P>` is read by the browser as the content. The whole tags with the content can be called an element, which is shown in Figure 2.
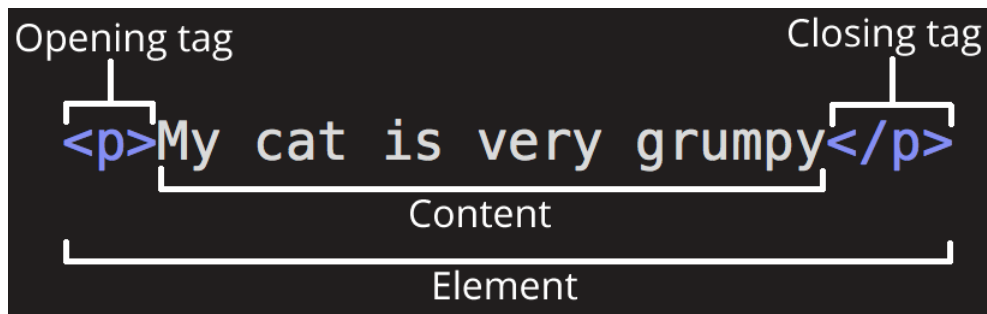
FIGURE 2. HTML element. [6]

The <!DOCTYPE html> represents the document to be HTML5. <html> is considered as the root element, <head> stores the meta data, <title> specifies the title of the document, <body> contains the web page data for the user interface. Below Figure **3** is a simple HTML code which outputs a heading, a paragraph and an image. The output of the HTML file is shown in Figure 4.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <meta charset="utf-8">
5  <title>Page Title</title>
6  </head>
7  <body>
8
9    <h1>Hello I am a Heading</h1>
10   <p>Hello I am a paragraph.</p>
11   <img src="https://www.oamk.fi/images/Logot/Suomi-toimisto-tulostettava-png-
   rgb/Toimistokayttoon_Suomeksi-06.png" height="100px" width='200px' alt="My test
   logo">
12
13   <!-- logo Image source: https://www.oamk.fi/fi/oamk/medialle/logopankki -->
14
15 </body>
16 </html>
```

FIGURE 3. HTML Document

Hello I am a Heading

Hello I am a paragraph.

OΛMK
OULUN AMMATTIKORKEAKOULU

FIGURE 4. Response of the HTML Document.

3.1.1 CSS

CSS stands for cascading style sheet. It is used for styling the web page. It allows each HTML element to be styled uniquely. CSS can be in-line as shown in Figure 5, which directly references the style of the HTML element. It can be written in the separate file as in Figure 6 and linked to the main file or in a same file as in Figure 7.

```
1 <h1 style="color:blue;text-align:center;">This is a heading</h1>
2 <p style="color:red;">This is a paragraph.</p>
```

FIGURE 5. Inline CSS

```
1 <link rel="stylesheet" type="text/css" href="mystyle.css">
```

FIGURE 6. Importing the CSS from external file.

```
1
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <style>
6 body {
7    background-color: yellow;
8 }
9
10 h1 {
11    color: maroon;
12    margin-left: 40px;
13 }
14 </style>
15 </head>
16 <body>
17
18 <h1 style="color:red;text-align:center;">I am a heading</h1>
19 <p style="color:yellow;">I am a paragraph.</p>
20
21 </body>
22 </html>
23
```

FIGURE 7. Internal CSS

CSS works in a predictable way, it consists of the selectors and declaration. As shown in Figure 8 the HTML tag, <h1> is the selector, which specifies, where the style is targeted. There can be many declarations each separated by a semicolon, which declare the unique style to be applied. The heading element is going to possess a font size of 12 pixels and a text colour as blue.[7]

Selector   Declaration   Declaration

h1   { color:blue; font-size:12px;}

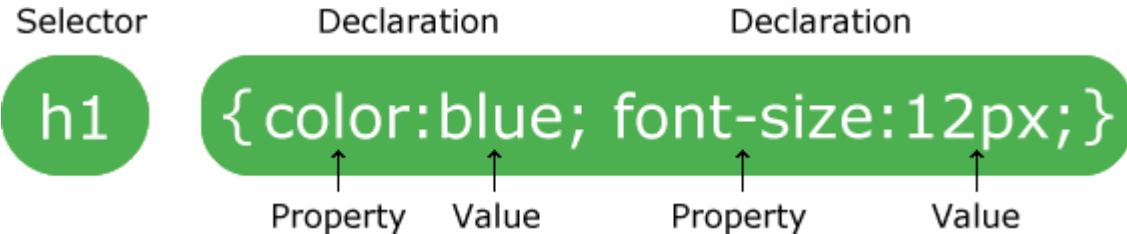Property Value   Property   Value

FIGURE 8. CSS Selector and Declaration block.[7]

3.1.2 Bootstrap

Bootstrap is an open-source front-end framework which cuts the development time as it readily facilitates the design templates and functional components out of the box. Templates make working with design components, such as forms, buttons, modals, much easier. Most importantly Bootstrap adds responsiveness to the website.

Bootstrap can be installed using the npm, using the command as $ npm install bootstrap , $ gem install bootstrap -v 4.4.1 or by the using the CDN as shown in the Figure 9.[8]

```
1 //JS, Popper.js, and jQuery
2
3 <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-
   J6qa4849blE2+poT4WnyKhv5vZF5SrPoOiEjwBvKU7imGFAVOwwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>
4 <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-
   Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
5 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-
   wfSDF2E50Y2D1uUdjOO3uMBJnjuUD4Ih7YwaYd1iqfktjOUod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
```

*FIGURE 9. Bootstraps CDN for JS, Popper.js, and jQuery*

3.1.3 JavaScript

JavaScript is a cross-platform and object-oriented scripting language used for making a web page interactive. JavaScript was introduced as a front-end language but now with node.JS it can also be used in backend.

Client-Side JavaScript enables an HTML element to interact with the user events such as a mouse click, a scroll, a form input or just by calling the JavaScript with the <script> tags. This enhances the user experiences by adding behavioural features to the web page. Server-side JavaScript allows the application to communicate with a server or a database and retrieve the content as per the user request or interaction.[9]

Figure 10 represents the web page with the HTML contents and elements ready to execute the script. When the user clicks the button, the HTML content is shown as in Figure 11.
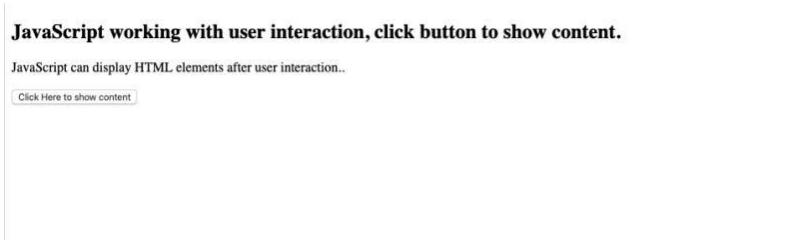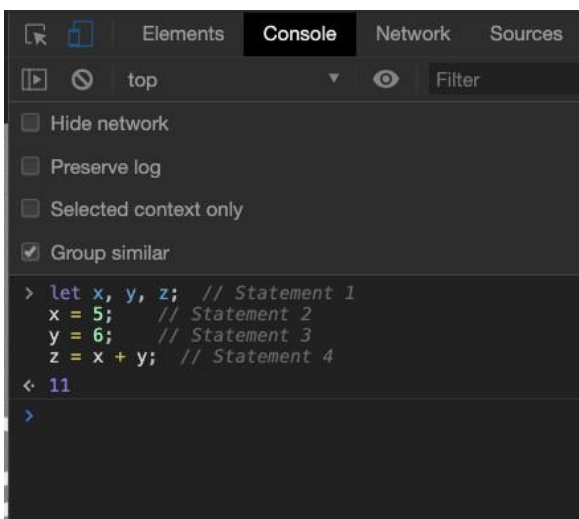

FIGURE 10. Web-application before the user-interaction.
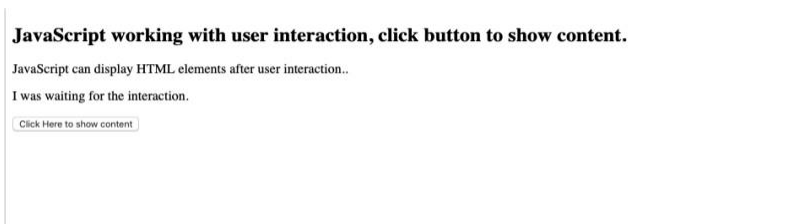
FIGURE 11. Showing the hidden content, after the user interaction.

JavaScript engines are used in web browsers and this enables a web console to execute the JavaScript result. Google Chrome uses V8 Engine, which is built by Google and is open source[10].

FIGURE 12. JavaScript code in the console.

In the Figure 12, the script is run in the console of the browser, with the help of JavaScript engines, the script is executed with the correct output.

2.3 Single Page Application

Single Page Application is a modern approach which eliminates the page reloading and communication with the server time and again. This approach enhances the UX greatly by reducing the wait time between the page load.

The use of modern day framework, such as Angular[11], Ember.js[12], React [13]and Vue.js[14], has taken the approach of SPA and made the programming with JS more simplified and modern.

FIGURE 13. SPA Lifecycle[15].

Figure 13 demonstrates the lifecycle of SPA where a client computer makes an initial request with the server and receives the web contents and later communicates for data from the database.

2.4 Multi Page Application

Multi-page Application is the old or traditional approach where every change or interaction on the web page triggers the page reload. As shown in Figure 14,

every reload triggers a request to a server and a new page is sent from the server to the client computer[16].
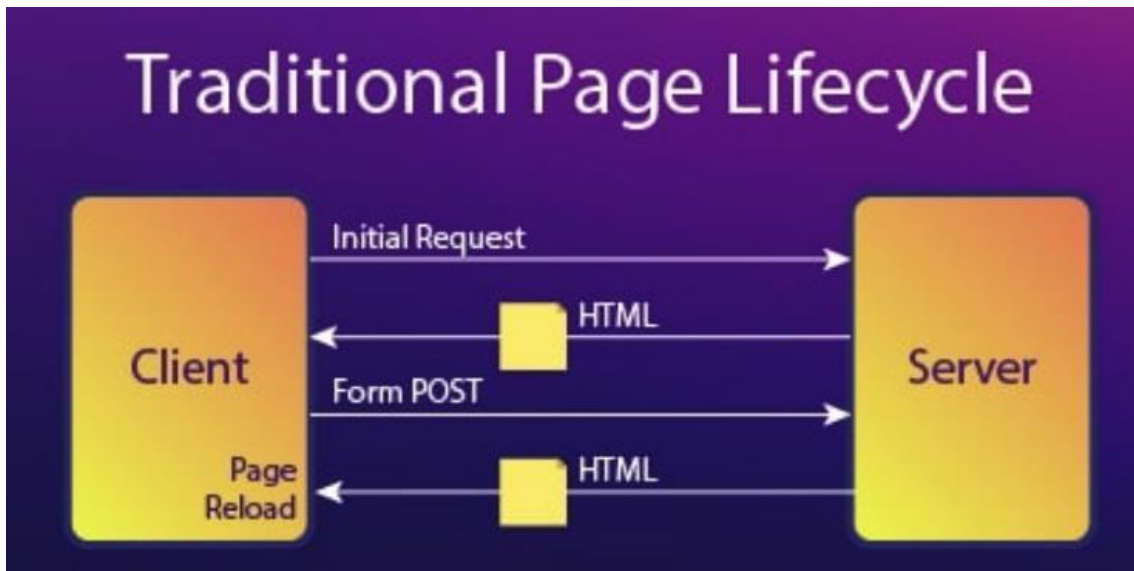

FIGURE 14. Multiple-page application lifecycle[15].

This approach makes the programming difficult as it adds many levels of complexities. Some companies still use this approach for a better SEO optimization.

3 WEB DEVELOPMENT TOOLS

Web development is a complex development process, which needs many people and expertise from different field to collaborate. To make the development phase more effective, efficient and practical, many tools are used. These tools make the development job easier. Some of the tools used in the development of the project are discussed below.

3.1 Editor

A code editor or simply an editor is the application where the code is written. A Modern day code editor is more than just for writing code. They can run the console, show the output and suggest better coding practices. A modern day coding editor is a very important tool in the programming. VS code or Visual Studio Code[17] was used as the coding editor as it is free, open source, runs in macOS machines and is relatively fast. It is the most popular development environment.[18]

Figure 15 demonstrates the features of VS code, such as inbuilt-git, debugging window, console area and extensions.
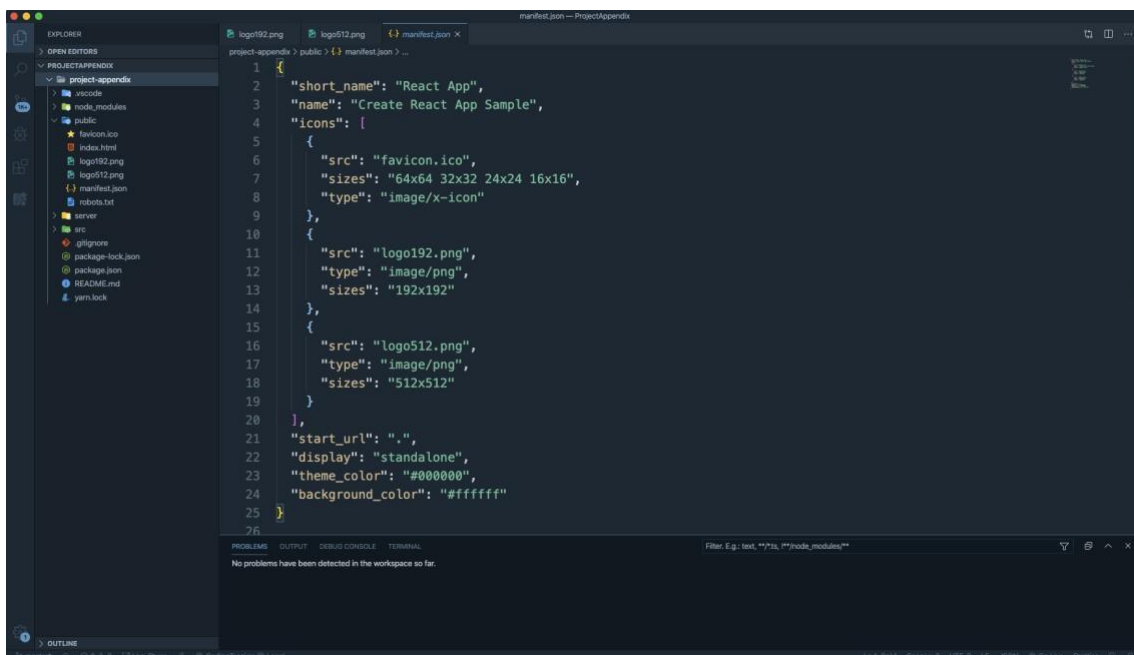

FIGURE 15. VS code in use.

3.2 Developer Tools

Chrome Developer Tools or Developer Tools are a set of developer tools in-built in the browser. With the developer tool, a developer can e.g. edit the page, diagnose the problem, debug, view and change the DOM, inspect the network activity.[19]

Figure 16 demonstrates the use of in-built Google Chrome's developer tool which is used to tweak the colour and content of the Oamk's web page.
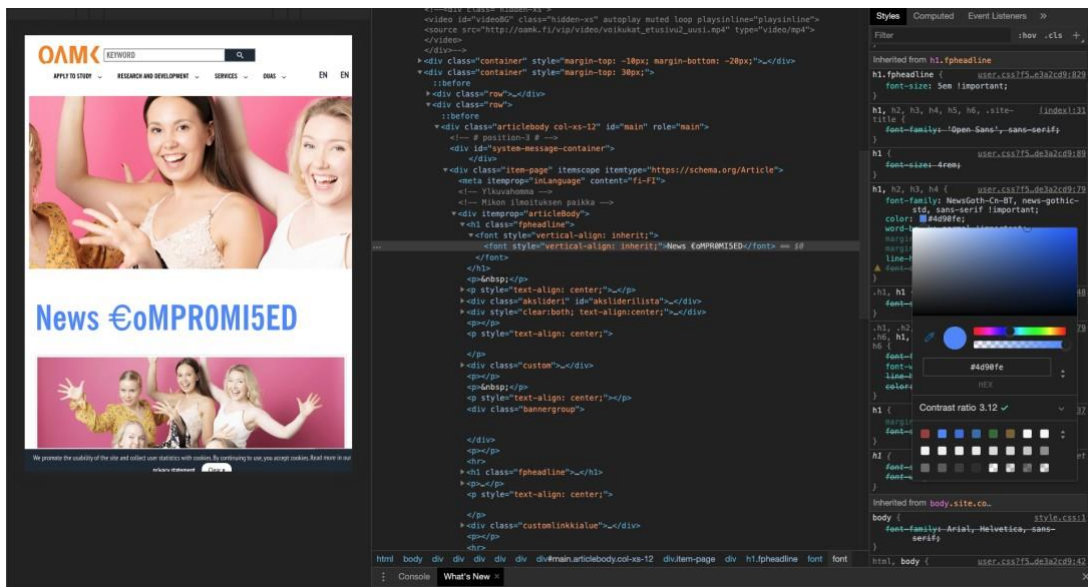

FIGURE 16. Chrome Developer tools in use.

3.3 Version Control System

A version Control System is a system where the change in file is recorded and all the versions of file are saved. When in need, the respective files can be extracted from the system. The local version control system was used a long time ago, as shown in Figure 17, where the files with the different versions were saved in the local computer.[20]
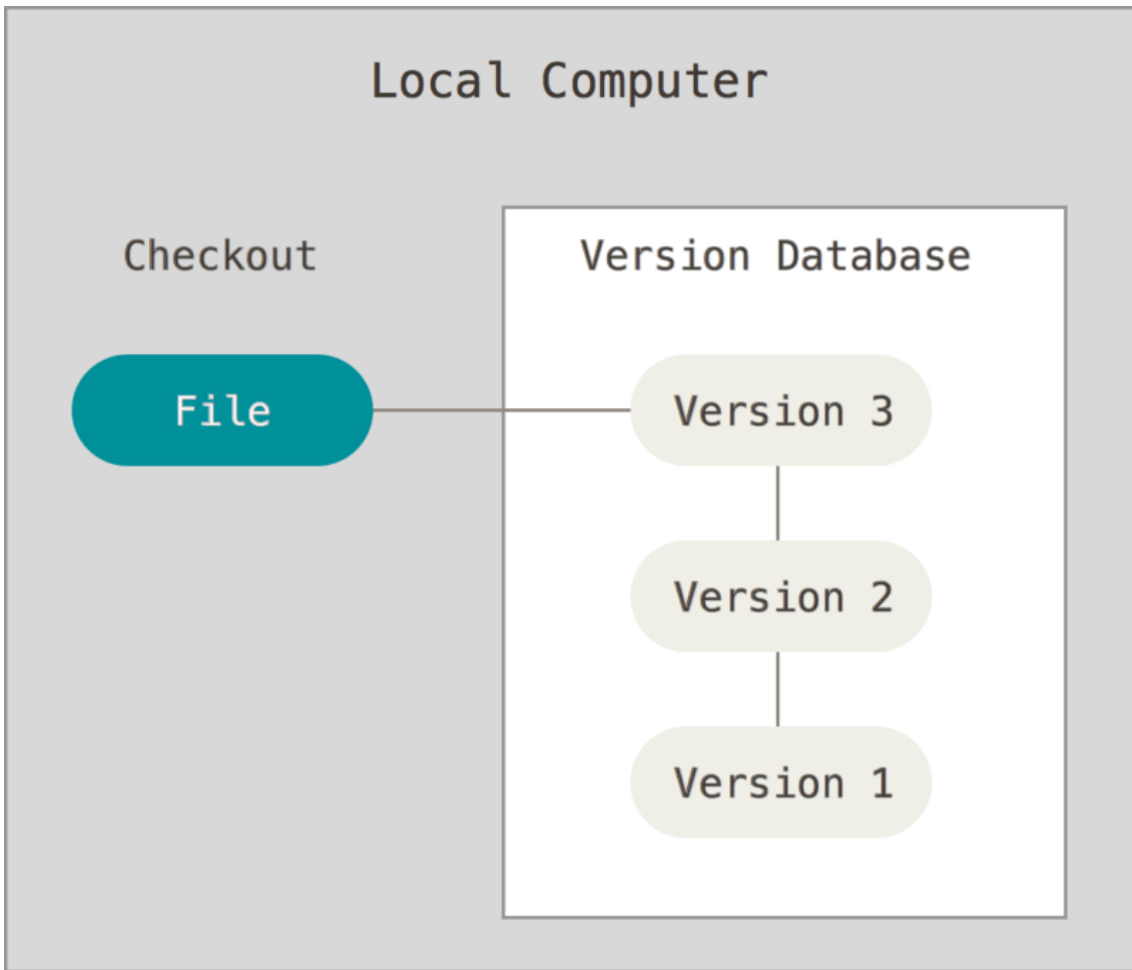
18

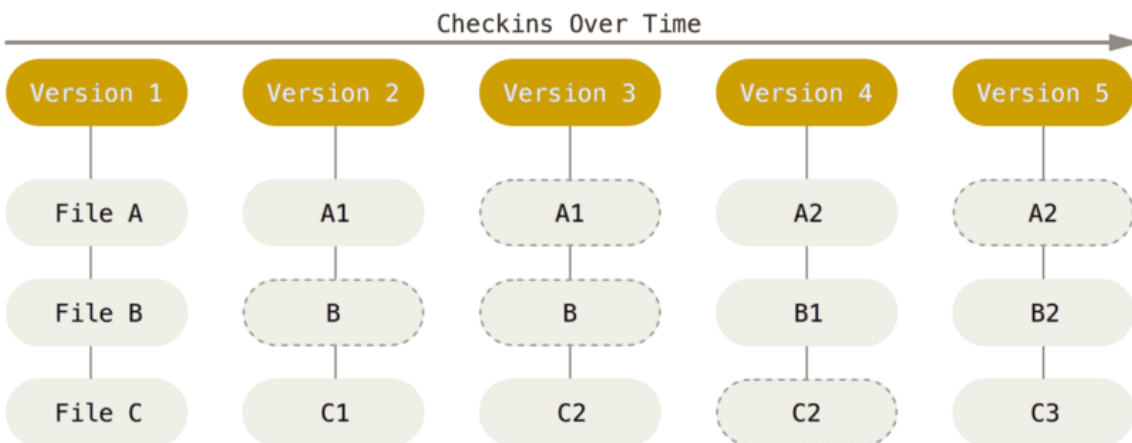*FIGURE 17. Local version controller workflow.[20]*

Figure 18. Storing the data as snapshots of the project.[21]

With the introduction of Git, the way of saving the version is changed. As shown in Figure 18 Git behaves a file as a series of snapshots of the miniature project. Git takes the snapshot of the file and saves a reference of it. If the file is not changed, Git does not save the file again.[21]

4 DESIGN AND DEVELOPMENT

Design and Development elaborates how the project was planned, designed and developed in details.

4.1 Requirements

A User story was made to foresee the requirements of the project. After creating the user story from the user's point of view and owner's prospects, a roadmap was designed on how to take things forward.

User stories for the web-application are below:

- As an owner, I would like to display the company info/profile and its motto.
- As an owner, I would like to display my product catalogue and its range.
- As an owner, I would like to point out the services provided.
- As an owner I would like to let the customer to know about the location, and contact info.
- As an owner, I would like to receive feedback from user.
- As an owner I would like to be in communication with the customer.
- As an owner, I would like to send the newsletter to the subscribed user.
- As a user, I would like to know about the company and its motto.
- As a user, I would like to find the location, contact information of company from the website.
- As a user, I would like to receive the catalogue of the range of products
- As a user, I would like to receive the list of service provided by the company.
- As a user, I would like to get in communication with company.
- As a user, I would like to subscribe the newsletter about campaigns and products.
- As a user, I would like to give feedback to the company.

4.2 Front-End

Front-end represents the interactive unit of the web application. It is also known as client-side development. HTML, CSS and JS are generally used to make an interactive front-end part of the web application[22]. Many approaches could be taken to enhance the feature and design of the application. To reduce time and make the programming journey more efficient, many frameworks and tools are used, some of which are discussed below.

4.2.1 Designing

Designing is done in the first phase of the project. It allows the developer and client to visualise the project. It helps the developer to predict the time for the development. Design can be a sketch in the paper, an elaborated drawing in the whiteboard or it can be done by using the modern-day online editors.

Balsamiq was used as a design tool to make the wireframe of the project application. Balsamiq is a quick and easy approach for designing. It allows to reuse the components and to create an interlink between the component itself.[23]

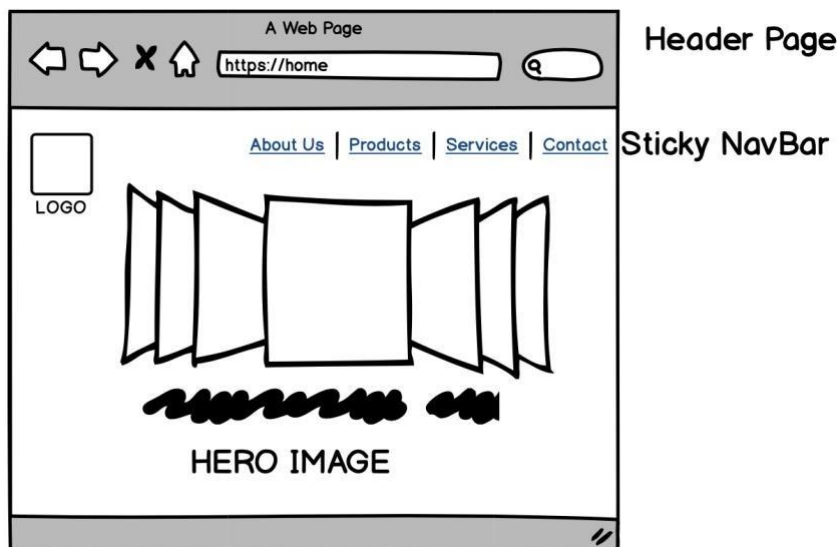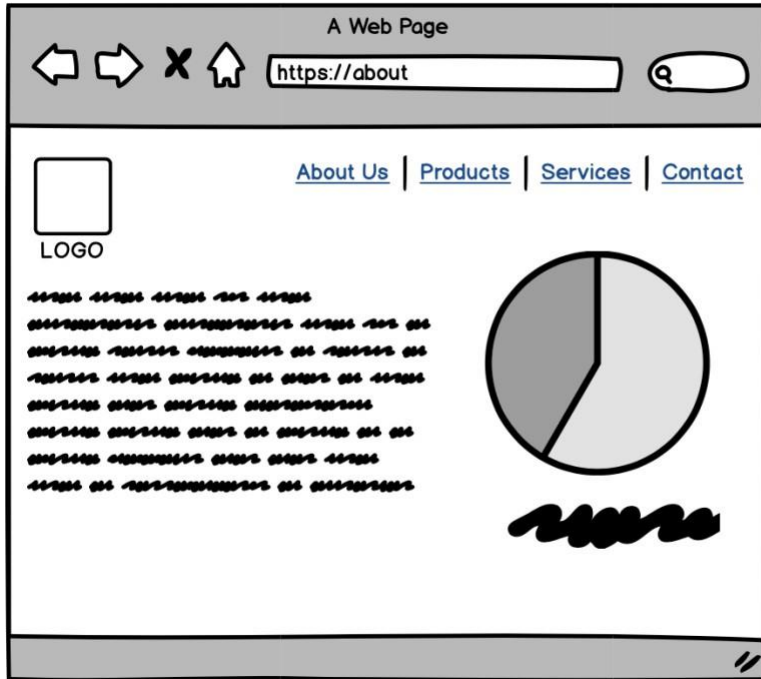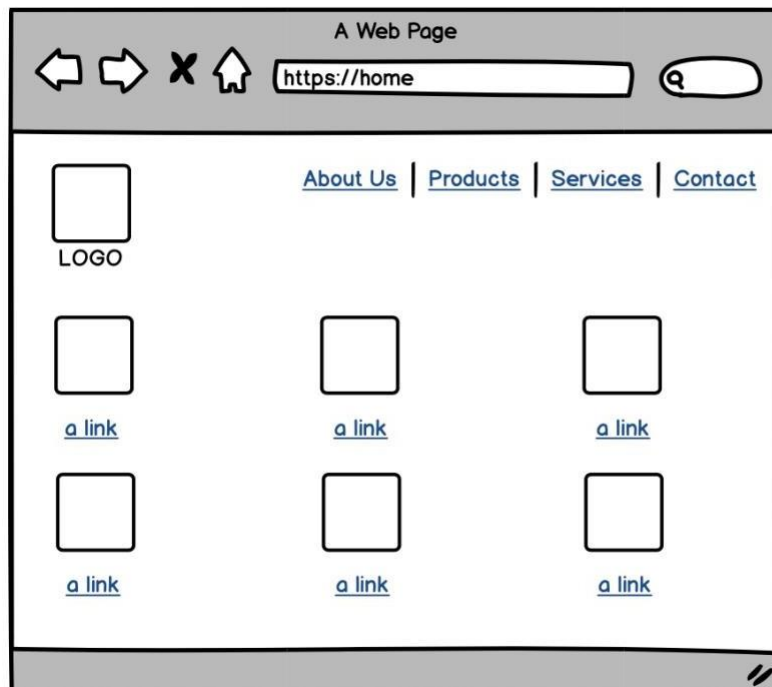Some of the design using the Balsamiq is shown in Figures 19 - 23.
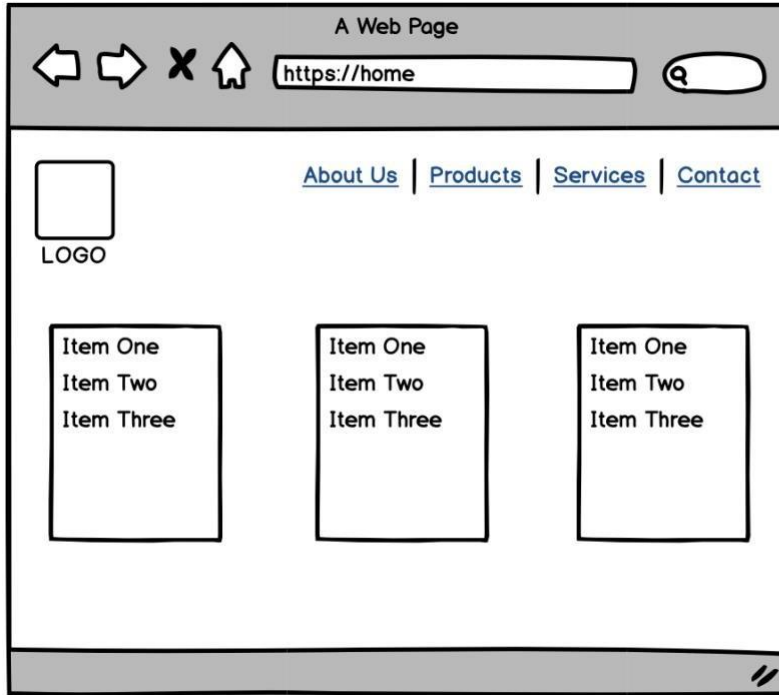

FIGURE 19. Wireframe of the Display page.

About Page

FIGURE 20. Wireframe of the About page.

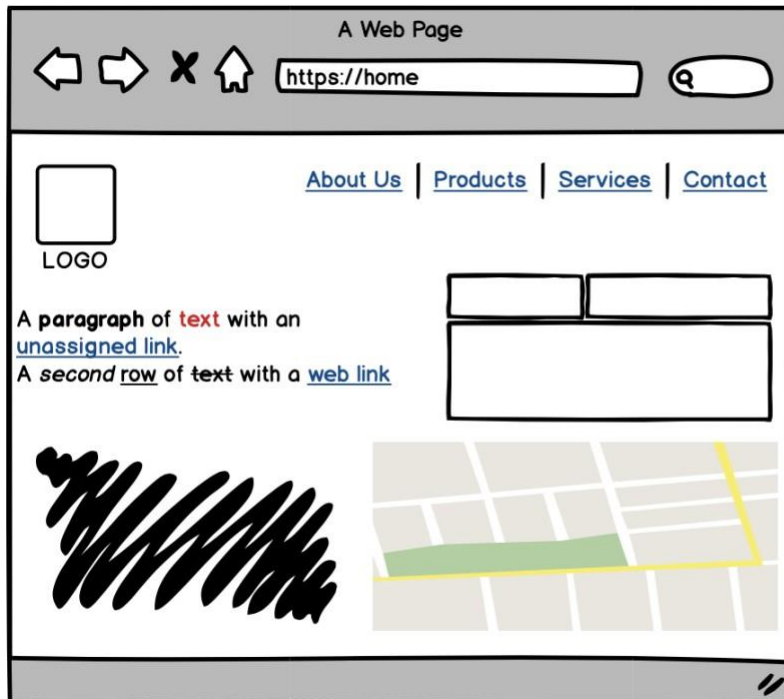**Product Page**

FIGURE 21. Wireframe of the Product page.

FIGURE 22. Wireframe of the Service page.

FIGURE 23. Wireframe of the Contact page.

4.2.2 React

React is a popular JavaScript library from Facebook used for building the user interface. React is component based, and components are reused as much as possible[13]. This approach cuts the time of making each component every time and minimizes the need of writing the same series of code multiple time.

Developing the web application in React automatically transmits into a single-page application mode. React uses the Virtual DOM where snapshot of the DOM is captured and if some part of the page needs the refresh, only the respective component is reloaded. This approach of the React makes the UI very smooth and uninterrupted.[24]

React uses JavaScript XML or shortly JSX. It is an extension to the JavaScript syntax. The syntax looks like HTML tags and is very easy to read and write. Under the hood, the compiler or babel translates the code into machine readable code or JavaScript. The use of JSX in React has made the programming a very simplified experience. React component uses the render() method, which takes input data and returns the output, as in Figure 24. It also shows the JSX in use, the same code is written in pure JavaScript which is shown in Figure 25.

```
1 class HelloMessage extends React.Component {
2   render() {
3     return (
4       <div>
5         Hello {this.props.name}
6       </div>
7     );
8   }
9 }
10
11 ReactDOM.render(
12   <HelloMessage name="Rachit" />,
13   document.getElementById('hello-example')
14 );
```

FIGURE 24. Use of JSX inside React.

```
 1 class HelloMessage extends React.Component {
 2   render() {
 3     return React.createElement(
 4       "div",
 5       null,
 6       "Hello ",
 7       this.props.name
 8     );
 9   }
10 }
11
12 ReactDOM.render(React.createElement(HelloMessage, { name: "Rachit" }),
   document.getElementById('hello-example'));
```

FIGURE 25. Raw JavaScript code.

4.2.3 Reactstrap

Reactstrap is an open source tool and is based on Bootstrap. It is a component based and React compatible prebuilt Bootstrap tool. It consists of readily available Bootstrap components, which are sure to work with React components. The code is written in JSX unlike the raw Bootstrap code. Figure 26 shows the use of className instead of class attribute in the JSX. It can also be classified as a Front-End Frameworks tool.

It is installed by using npm.

npm install --save reactstrap react react-dom

or by including CDN to the application bundle.

https://cdnjs.cloudflare.com/ajax/libs/reactstrap/4.8.0/reactstrap.min.js

```
1 import React from 'react';
2 import { Alert } from 'reactstrap';
3
4 const Example = (props) => {
5   return (
6     <div>
7       <Alert color="danger">
8         <h4 className="alert-heading">Well done!</h4>
9         <p>
10          Lorem Ipsum is simply dummy text of the printing and typesetting industry.
11        </p>
12        <hr />
13        <p className="mb-0">
14          Lorem Ipsum is simply dummy text of the printing and typesetting industry.
15        </p>
16      </Alert>
17    </div>
18  );
19 };
20
21 export default Example;
```

FIGURE 26. ReactStrap in use

5 AKALADEVI TRADING CONCERN COMPANY PROJECT

Making a large application takes a lot of time and sometimes requires the developer to learn new skills on the way. This part discusses the programming journey of the author, shortcoming, learnings and in general how the project was carried out.

5.1 Initial build

The author started with simple and one-page HTML and CSS code. The basic design was achieved by creating the HTML layout and styling using CSS. A Bootstraps component was used with the HTML attributes, which enhanced the visual factor and helped author achieve responsiveness.

Separate index.html and style.css files were created and linked. A media query was used to achieve responsiveness alongside with Bootstraps. The breakage of the web-responsiveness was recorded around the width of 750px and 480px. A mobile view and compression of the navigation bar were created with the following definitions.

```css
@media screen and (max-width: 750px) {

  .col-sm-4 {

    text-align: center;

    margin: 25px 25px 0px 0px;

  }

}

@media screen and (max-width: 450px) {
```

```
.logo {

  font-size: 150px 150px 0px 0px;

 }


}
```

All the code was divided into a component structure, which made transforming the code into React much easier.

5.2 Transforming into React application

The author had no previous experience with React or any other React-based frameworks. The approach of creating the application by using the known tools, such as HTML, CSS and JS was practiced. When the basic functionality and structure were achieved, the code was divided into multiple components and later used in React.

Reusing the component was the main priority and learning. All the components were isolated and used where it was necessary. CSS and style responsible for the respective components were added inside the style folder with a corresponding name as shown in Figure 27. Style files were imported in the component file when necessary as shown in Figure 28.
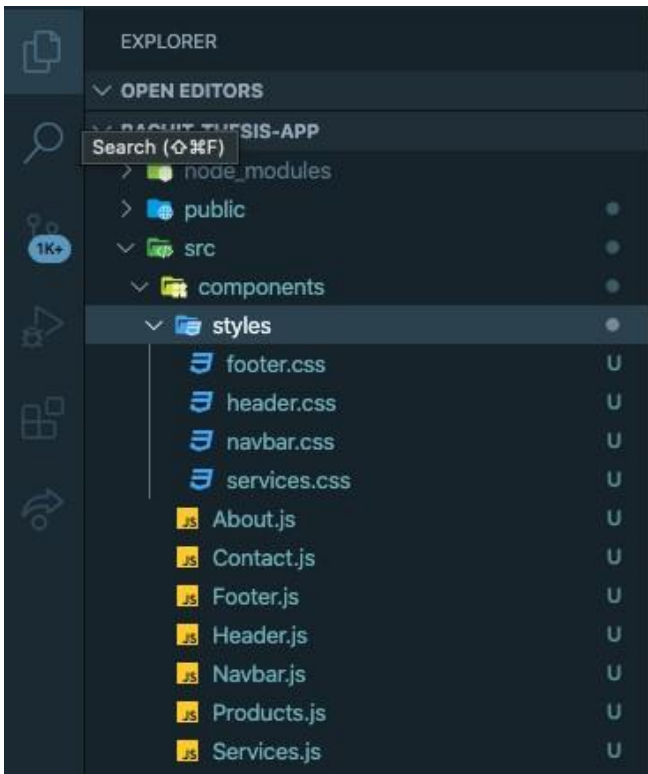
FIGURE 27. Folder Structure and list of components.

FIGURE 28. Importing respective CSS file in the Component file.

All the components were imported in the main root file (figure 29) or in the App.js and laid in the right order of the page layout.

```
src > JS App.js > ...
  1  import React from "react";
  2  import "./App.css";
  3  import Header from "./components/Header";
  4  import About from "./components/About";
  5  import Products from "./components/Products";
  6  import Services from "./components/Services";
  7  import Contact from "./components/Contact";
  8  import Footer from "./components/Footer";
  9
 10  function App() {
 11    return (
 12      <div
 13        className="App"
 14        id="myPage"
 15        data-spy="scroll"
 16        data-target=".navbar"
 17        data-offset="60"
 18      >
 19        <Header />
 20        <About />
 21        <Products />
 22        <Services />
 23        <Contact />
 24        <Footer />
 25      </div>
 26    );
 27  }
 28
 29  export default App;
 30
```

FIGURE 29. Importing the components in App.js

6 CONCLUSION

This thesis project was a real project developed for the company. Making the informative web-site with the responsive design concept by using modern day web technologies was achieved. Screenshots of the website are shown in the appendices 1 - 3. The screenshots feature dummy texts and placeholder images as the company denied the full exposure of the content in the thesis project. Real company text and related images will replace the dummy text and placeholder images at the time of launch.

The author has developed skills for developing the responsive website by using modern day tool. Learning and understanding React was the main learning out of the project. Using readily available code from the frameworks, such as Bootstraps transforms the website in minutes. Reusing the code whenever possible saves a lot of time and resources. This practice helps to shape good coding practices and embraces the dynamic coding practices.

Apart from the programming aspects, the author developed designing skills, as designing was the initial part of the project. It formed the layout and common understanding between both parties. Skills such as communication, working under pressure, working on completely new technologies, researching, surfing the Internet for solution, analysing and presentational skills were in effect rigorously.

Hosting the website, domain registration and security prospects of the project is the next part. The author has no experience on hosting the application but believes from the previous experience that anything can be done by learning on the way.

REFERENCES

1. Anonymous. 2020. How the Web works. Date of Retrieval Apr 1, 2020
   https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/How_the_Web_works

2. P,Christensson. 2015. Internet. Date of Retrieval Apr 1, 2020
   https://techterms.com/definition/Internet

3. P,Christensson. 2006. TCP/IP. Date of Retrieval Apr 1, 2020
   https://techterms.com/definition/tcpip

4. P,Christensson. 2015. HTTP. Date of Retrieval Apr 1, 2020
   https://techterms.com/definition/http

5. P,Christensson. 2014. DNS. Date of Retrieval Apr 1, 2020
   https://techterms.com/definition/dns

6. MDN contributors. 2019. HTML basics. Date of Retrieval Apr 4, 2020
   https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics

7. Anonymous. 2020. CSS Syntax. Date of Retrieval Apr, 4, 2020
   https://www.w3schools.com/css/css_syntax.asp

8. O,Mark & J,Thornton & Bootstrap. Introduction. Date of Retrieval Apr 5, 2020
   https://getbootstrap.com/docs/4.4/getting-started/introduction/

9. Anonymous. Introduction-JavaScript. Date of Retrieval Apr 5, 2020
   https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Introduction

10. A, Zlatkov. 2018. How JavaScript works: inside the V8. Date of Retrieval Apr 15, 2020
    https://blog.sessionstack.com/how-javascript-works-inside-the-v8-engine-5-tips-on-how-to-write-optimized-code-ac089e62b12e

11. Google Developer. 2020. Angular. Date of Retrieval Apr 15, 2020
    https://angular.io/

12. Anonymous. Ember.js - A framework for ambitious web developers.
    Date of Retrieval Apr 15, 2020
    https://emberjs.com/

13. Anonymous. React – A JavaScript library for building user interfaces.
    Date of Retrieval Apr 15, 2020
    https://reactjs.org/

14. Anonymous. Vue.js. Date of Retrieval Apr 15, 2020
    https://vuejs.org/

15. Anonymous. What is Single Page Application - Benefits, Disadvantages,
    Examples | SCAND Blog. Date of Retrieval Apr 15, 2020
    https://scand.com/company/blog/single-page-application-vs-multi-page-application/

16. Anonymous. 2018. Single-page application vs. multiple-page application.
    Date of Retrieval Apr 15, 2020
    https://medium.com/@NeotericEU/single-page-application-vs-multiple-page-application-2591588efe58

17. Anonymous. Visual Studio Code - Code Editing. Redefined. Date of
    Retrieval Apr 15, 2020
    https://code.visualstudio.com/

18. Anonymous. Stack Overflow Developer Survey 2019. Date of Retrieval
    Apr 19, 2020
    https://insights.stackoverflow.com/survey/2019/?utm_source=social-share&utm_medium=social&utm_campaign=dev-survey-2019

19. Anonymous. Chrome DevTools | Tools for Web Developers. Date of
    Retrieval Apr 19, 2020
    https://developers.google.com/web/tools/chrome-devtools

20. Anonymous. Getting Started - About Version Control. Date of Retrieval Apr 19, 2020

    https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control

21. Anonymous. Getting Started – What is Git?. Date of Retrieval Apr 19, 2020

    https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F

22. Anonymous What Is a Front-End Developer? Date of Retrieval Apr 20, 2020

    https://frontendmasters.com/books/front-end-handbook/2018/what-is-a-FD.html

23. Anonymous Balsamiq Wireframes | Balsamiq. Date of Retrieval Apr 20, 2020

    https://balsamiq.com/wireframes/

24. Anonymous Visual Studio Code - Code Editing. Redefined. Date of Retrieval Apr 19, 2020

    https://code.visualstudio.com/

25. A, Ravichandran. 2018. React Virtual DOM Explained in Simple English.

    https://programmingwithmosh.com/react/react-virtual-dom-explained/

APPENDICES

List of Appendices:

Appendix 1: Layout of the web-site in tablet view.

Appendix 2: Layout of the website in different variant of tablets.

Appendix 3: Layout of the web-site in galaxy-tablet view

Logo   About Us   Products   Services   Contact

Akaladevi Trading Concern

Company Moto or Slogan!!!

ABOUT COMPANY PAGE

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Get in Touch
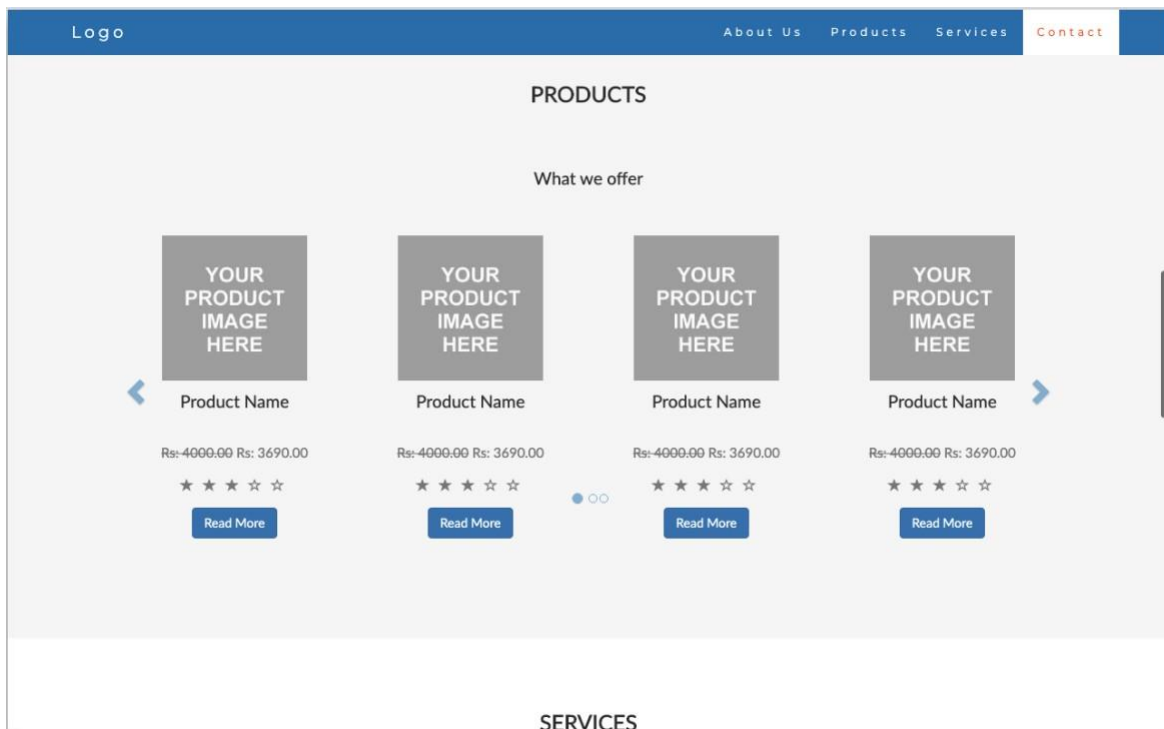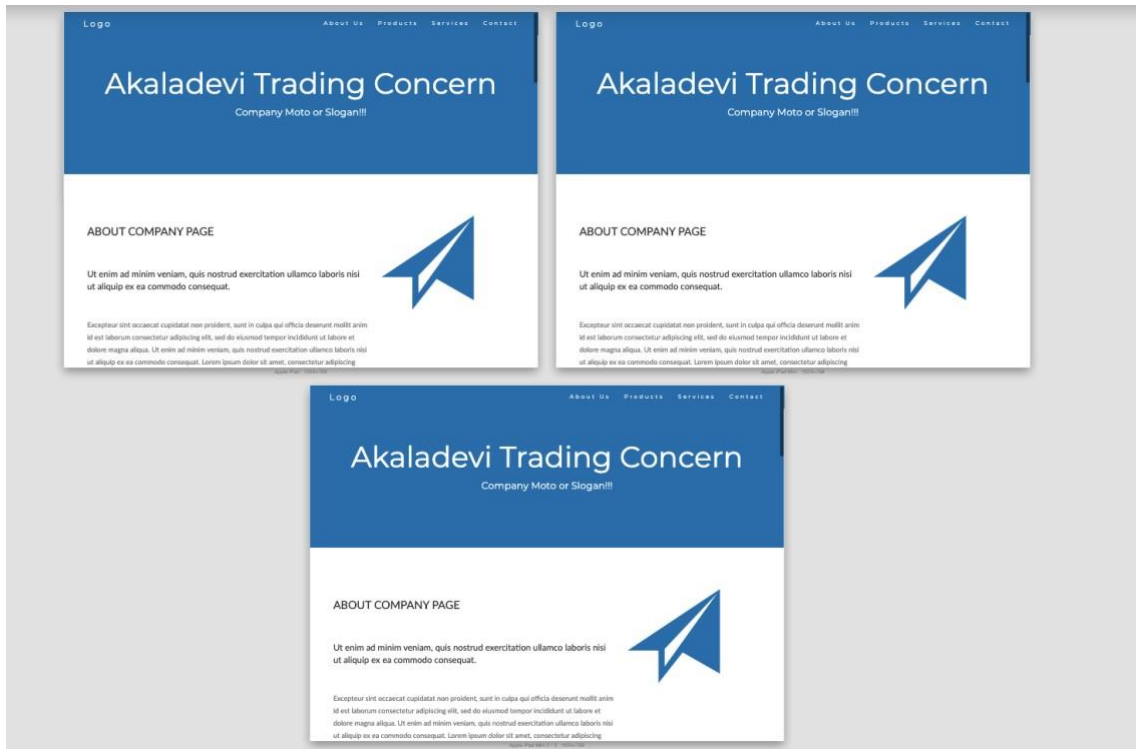
WHY US

Ut enim ad minim veniam

Enviroment Friendly	Home services	Easy on the cost
Lorem ipsum dolor sit amet..	Lorem ipsum dolor sit amet..	Lorem ipsum dolor sit amet..

PRODUCTS