OULU UNIVERSITY OF
APPLIED SCIENCES

Rakesh Phaiju

FULLSTACK WEB APPLICATION USING REACTJS AND FIREBASE

FULL STACK WEB APPLICATION USING REACTJS AND FIREBASE

Rakesh Phaiju
Bachelor's Thesis
Autumn 2020
Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Degree programme in Information Technology,

Author: Rakesh Phaiju
Title of the bachelor's thesis: Full stack Web Application Using ReactJS and Firebase
Supervisor: Veijo Väisänen
Term and year of completion: Autumn 2020 Number of pages: 27

The purpose of this bachelor's thesis was to implement a JavaScript based full stack web application for a professional photographer. This application was motivated by self-interest to learn and implement the full stack JavaScript based web application.

The application was developed using tools such as ReactJS, Google API, Google Firebase. ReactJS was used for implementing user interface and Firebase for storing the images and authentication of the admin panel.

As a result, there is an attractive, responsive, and user-friendly working web application for a photographer which can be used to show his skills to his potential customers.

Keywords: ReactJS, API, Web Applications, Firebase, User Interface, EmailJS

PREFACE

The basis for this thesis originally stemmed from my interest of learning about the development of a JavaScript based web application. The development of the web application and writing of this thesis took place in Oulu, Finland.

I would like to thank my friend Ramesh Sainju for trusting and providing me his beautiful photographs to use in this website along with guidance and suggestions throughout the development of the web application. Also, I would like to thank my thesis supervisor Veijo Väisänen and my language teacher Heidi Hedström for giving me opportunity to write this thesis paper along with their constant feedback and support.

Oulu, 20.11.2020
Rakesh Phaiju

CONTENTS

VOCABULARY

API – Application Programming Interface

Apps- Application

CSS- Cascading Style Sheets

HTML- Hypertext Markup Language

JS- JavaScript

NPM- Node Package Manager

SQL- Structured Query Language

UI- User Interface

VS code- Visual Studio Code

1 INTRODUCTION

This thesis is related to the implementation of a web application that could be used by a professional photographer to show his skills. This application was a self-learning project developed by the author to learn about the development of full stack web apps using ReactJS and Firebase.

Along with ReactJS, tools such as Google API was also used to retrieve the videos from the YouTube, CSS to make interactive, responsive, and attractive web application. ReactJS libraries such as react-reveal was used to animate the images in the application. Firebase was used as a backend tool. Google Cloud Storage was used to store the images, Firebase Authentication was used to authenticate the admin of the application. EmailJS was used to send the email from the form in the application. The secondary objective was to learn to implement the full stack web application using ReactJS and Firebase.

This thesis report provides detailed information about the technologies used during the development of the application along with the implementation process.

2 TECHNOLOGIES USED

Different technologies are used for creating web applications. The process of web development consists of two part i.e. Front-end and Back-end.

2.1 FRONT-END

Front end, also known as client-side, refers to the user interface of the web page. Basically, it is the part of the application where the user can view the content of the application. The following section describes the technology that were used while developing the application.

2.1.1 HTML

HTML stands for Hypertext Markup Language. HTML describes the structure of web page. HTML consists of various elements and represented with tags such as “head”, “body”, “div”. These tags are enclosed in <>. [1]

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

FIGURE 1. Simple HTML document. [1]

In FIGURE 1, the basic example of the HTML document is shown.

2.1.2 CSS

CSS stands for Cascading Style Sheets. CSS is used to style an HTML document. CSS is used to define styles for the web pages, including the design, layout and variations in display for different devices and screen sizes.

```
body {
  background-color: lightblue;
}

h1 {
  color: white;
  text-align: center;
}

p {
  font-family: verdana;
  font-size: 20px;
}
```

FIGURE 2. The basic CSS syntax. [2]

In FIGURE 2, basic CSS syntax is shown.

2.1.3 ReactJS

ReactJS, also known as React or React.js is a front-end library developed by Facebook. It is used for handling the view layer for web and mobile apps. ReactJS allows us to create reusable UI components. It is one of the most popular JavaScript libraries and has strong foundation and large community behind it [3]. React version of 16.14.0 was used while developing and implementing this full stack application.

ReactJS came into existence in 2011, when Jordan Walke, a software engineer at Facebook created the library [4]. React was influenced by the concept of Component, such as XHP in PHP as an HTML component. The first use of ReactJS was in Facebook newsfeed in 2011. Later, it was picked by Instagram

to use it in their system. React was made as an open source by Facebook in May 2013.

```
class Cinematography extends Component {
  render() {
    return (
      <div className="about">
        <h1>Cinematography</h1>
        <Youtube />
      </div>
    );
  }
}

export default Cinematography;
```

FIGURE 3, A simple React component

2.1.3.1 React Reveal

React Reveal is an animation framework for React [5]. It can be used to create various animations such as Fade, Flip, Rotate, Zoom. For the animations of the images in this application, animations such as Flip and Rotate from react reveal was used.

2.2 BACK-END

Generally, back-end refers to the server-side which also mean internal working part of the application. This part of the website does not interact directly with the user, but it is responsible for manipulating and storing the data of an application. This section includes the explanation of the technologies that were used during the development of this application.

2.2.1 Firebase

Firebase is a cloud database that stores data as JSON and synchronizes in real time to subscribed devices. It is a unique approach, super-fast and simple to

use [6]. Firebase is a powerful platform for both mobile and web applications. Firebase acts as the back end of the application, including data storage, user authentication, static hosting and many more [7].

Before Firebase was known as Firebase, it was a start-up called Envolve. As Envolve, it provided developers with an API that enabled the integration of online chat functionality into their website. But developers used to pass application data that was more than just a message using Envolve. This leads the founders of Envolve, James Tamplin and Andrew Lee to create Firebase as a separate company that provided Backend-as-a-Service with real- time functionality [7]. Firebase was acquired by Google in 2014 and it has been growing as a powerful platform for mobile and web application since then. The following section describes some of the services provided by firebase which was used during the implementation of the application.

2.2.1.1 Cloud Firestore

Cloud Firestore is a flexible, scalable database for mobile, web and server development from Firebase and Google Cloud Platform. It is NoSQL document database that lets the user to store, sync and query the data for the mobile and web apps. [7]

2.2.1.2 Authentication

Authentication provides the backend services, easy to use SDKs and ready-made UI libraries to authenticate users to the application. It supports authentication using passwords, phone numbers, emails and other identity providers like Google, Facebook, Twitter and many more.[8]

2.2.1.3 Cloud Storage

Cloud Storage is developed for the developer who need to store and serve user- generated content, such as photos, videos, or audios. Cloud Storage for Firebase is simple, powerful, and cost-effective object storage service built for Google scale.

2.2.2 EmailJS

EmailJS helps sending emails using client-side technologies. While using the EmailJS, a server is not required. The developers just need to connect EmailJS to one of the supported email services. The developer should create an email template and use it to trigger an email. Some other alternatives are also there to EmailJS. Below you can see some of them.

- Nodemailer
- PHPMailer
- MailKit
- Context.IO
- Mailparser.io

3 IMPLEMENTATION OF WEB APPLICATION

3.1 Design

Before the development of the website, the design was made using Figma tool. Figma is a free, online UI tool to create websites, apps or other UI components that can be integrated into other projects. The design for this web application can be found here. This design was approved by the photographer himself.

3.2 Code Editor

Along with the development process, it is important to select a good code editor. It really has a great impact on productivity of implementation process. Some of the popular and free code editors available are VS code (Visual Studio Code), Atom, Sublime Text, Notepad ++. [10]

For the development of this web application, VS code was selected as a code editor. VS code is a free source code editor made by Microsoft for Windows, Linux and macOS. It has built-in terminal, and it supports Git commit.

3.3 Project File Structure

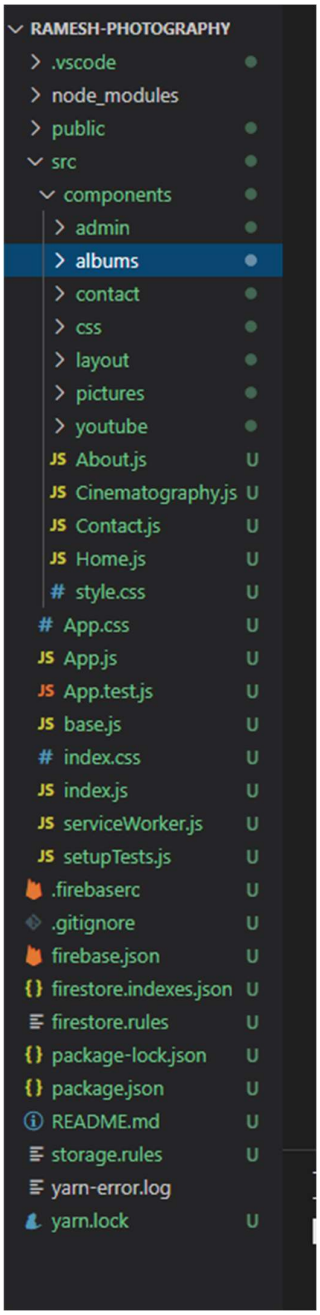

FIGURE 4. File structure of the project

FIGURE 4 shown above is an image of the file structure of the project.

The file structure looks like a basic react project file along with the firebase rules where admin folder contains components related to admin authentications, albums folder contains components related to images, contact folder contains corresponding components and so on. The layout folder consists of the components like header and footer. The package.json file consists of information for packages used and its corresponding versions.

3.4 UI of the Website

3.4.1 Navigation Bar and Footer

The navigation bar allows the user of the application to navigate through different pages of the website. In this application, the navigation bar provides the access to the Gallery, Cinematography, About and Contact pages of the website.

FIGURE 5. The navigation bar for the desktop view

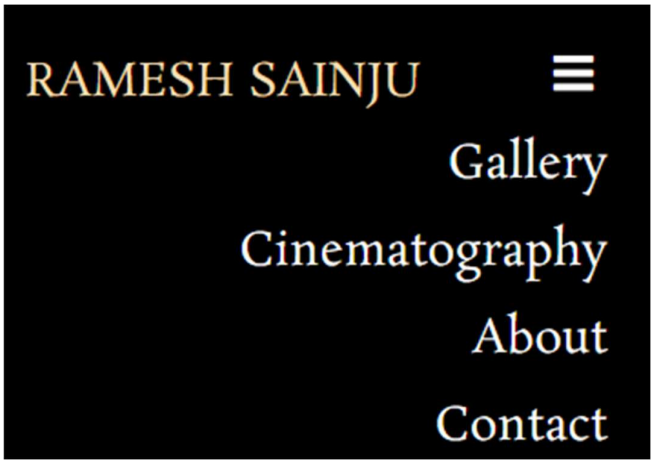

FIGURE 6. The navigation bar for the mobile view

FIGURE 5 shown above is the navigation bar for the desktop view whereas in FIGURE 6, it changes to a hamburger menu in mobile view as the whole application is also responsive to any devices.

FIGURE 7. Footer of the application

FIGURE 7 shown above consists of the icons which redirects the social media handle of the photographer.

3.4.2 Landing Page

The Landing page is the page where the user lands as soon as they visit the website. In the landing page of this application, it includes the various albums of the photos as shown in FIGURE 7. These lists of albums are rendered from the Firebase. The thumbnail images are the first uploaded image in the corresponding album.

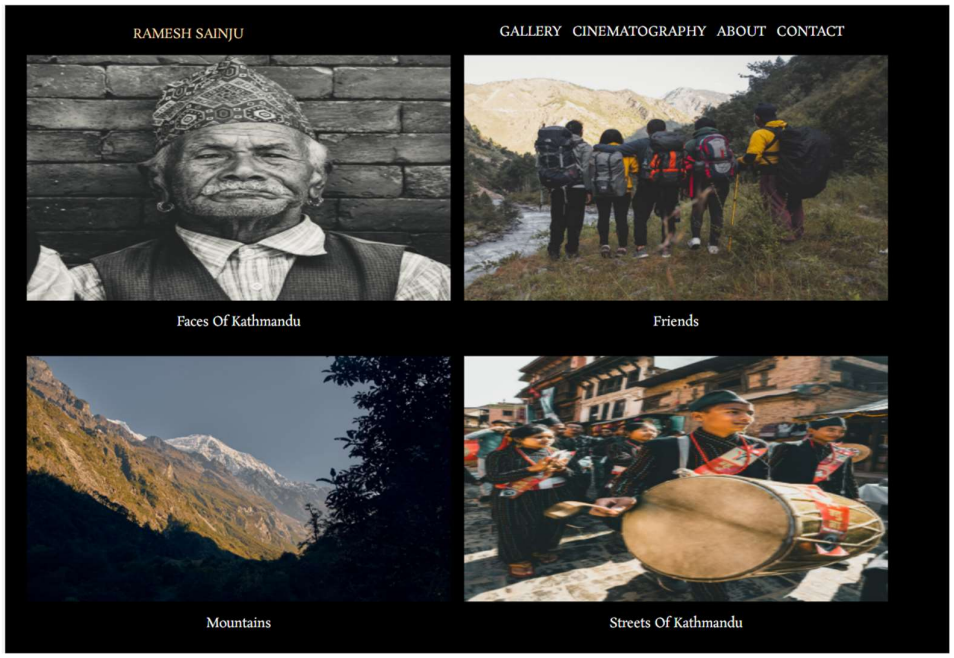

FIGURE 8. Landing page of the application

3.4.3 Album Page

When the user clicks a certain album which he/she wants to see, the images inside that corresponding album will be loaded in the route /album_name. The images from these albums are rendered from the Firebase and stored in the Google Cloud Storage. FIGURE 9 below is shown as an example view of the Album page.

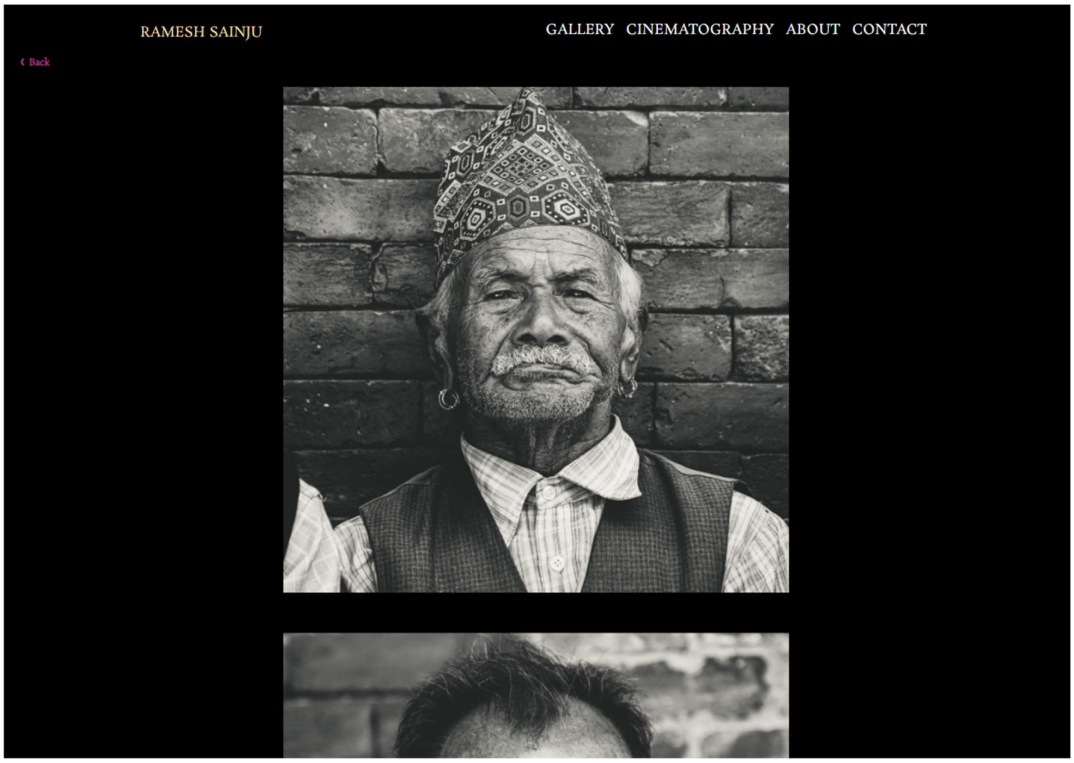

FIGURE 9. Album page

3.4.4 Cinematography page

In Cinematography page, videos from YouTube channel are fetched using Google API and channel id. The users can see the cinematography skill of the owner of the website. This page can be visited in /cinematography route.

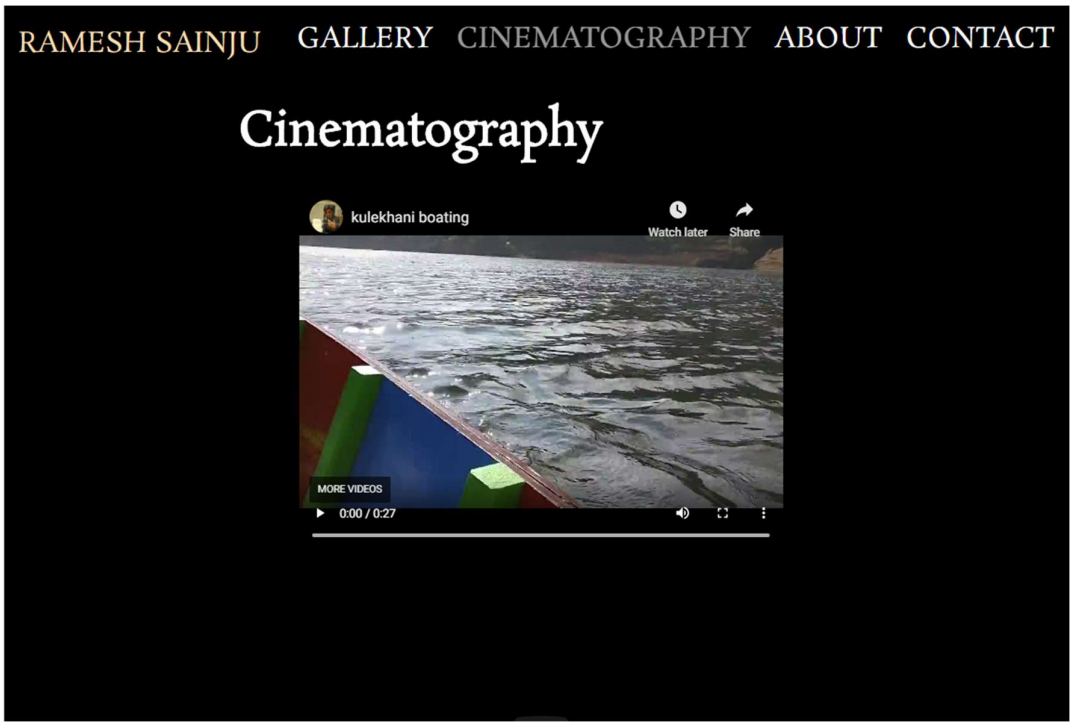

FIGURE 10. Cinematography page of the application

3.4.5 About Page

In this page, the general information about the owner of the website i.e., the photographer is shown. As shown in FIGURE 9, About page contains image of the photographer himself and general information about him. This page can be visited using /about route.

FIGURE 11. About page of the application

3.4.6 Contact Page

In Contact page, it consists of a small form to send the message to the owner of the application. EmailJS was used to send the message via the form available on this page. This allows the customer to send the message to the photographer's email.

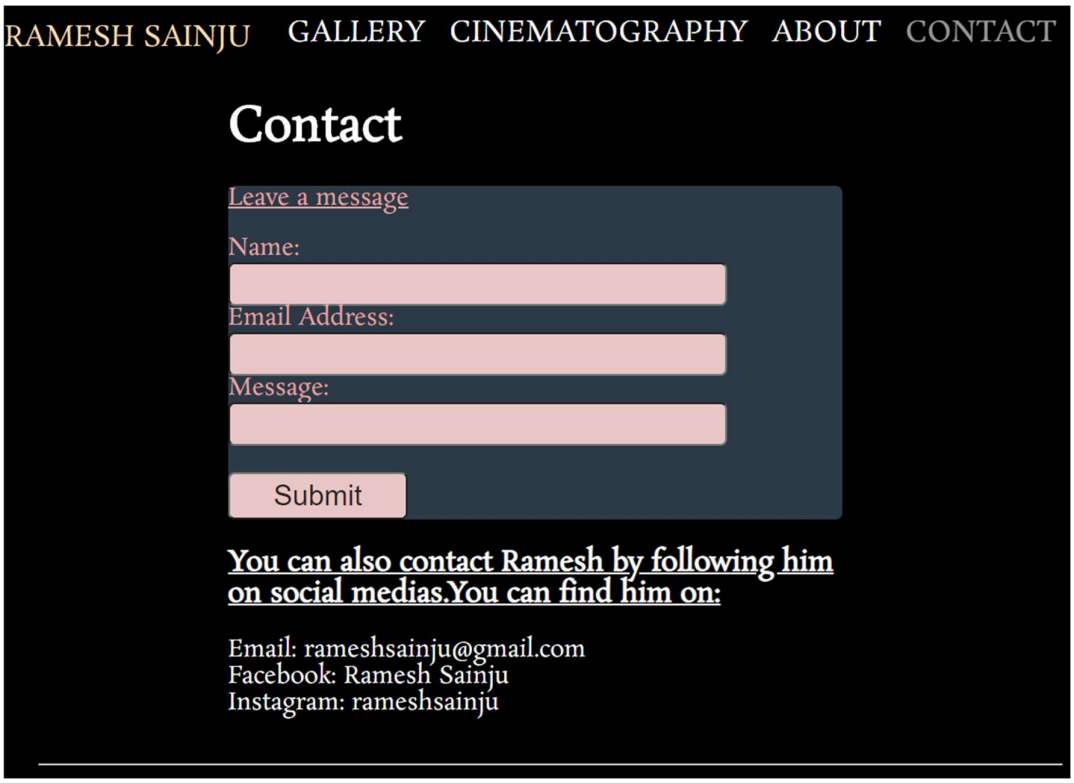

FIGURE 12. Contact page of the application

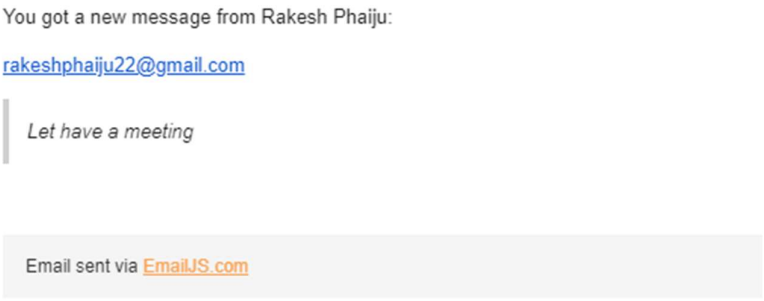

FIGURE 13. Message in email via EmailJS

In FIGURE 13, we can see the message or content of the form in email which is sent via EmailJS.

3.4.7 Admin Panel of the website

In this application, admin panel is accessible by authorized personnel only. The admin panel is accessed in a route /admin and must be logged in using admin credentials. Once the admin is authorized, the user can add new albums for the images, add new images to the corresponding albums and also delete the albums.

FIGURE 14. Login page for the admin

The above FIGURE 14 shown is the page for logging in the authorized admin with the credentials (email and password). Firebase authentication was used to authenticate the admin user.

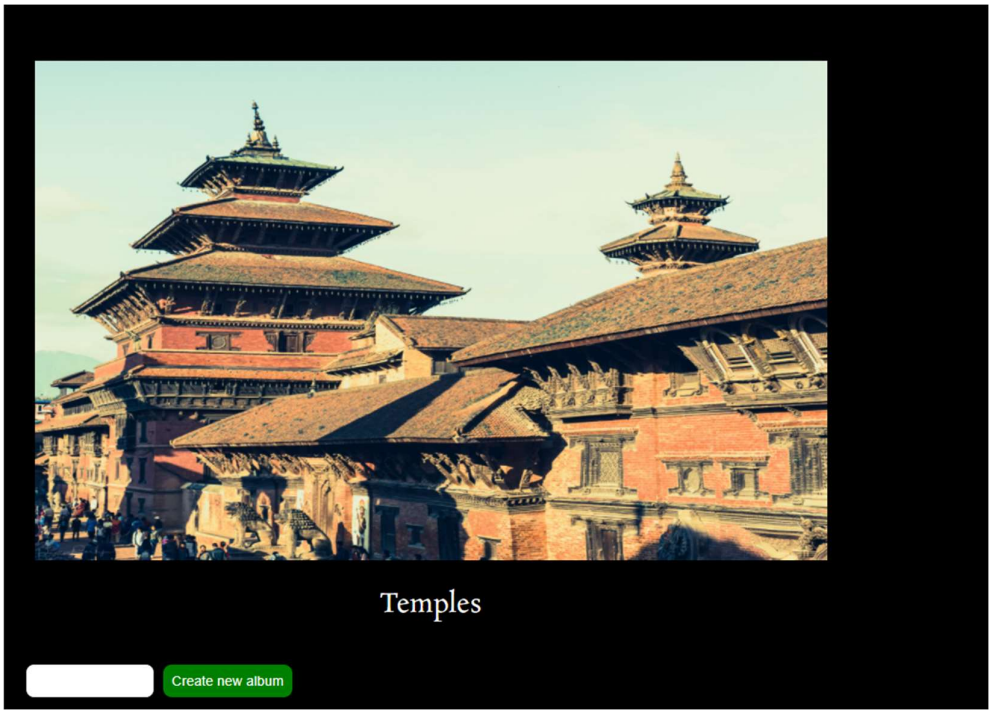

FIGURE 15. Album addition component

After authenticating with the admin credentials the component for adding new album will be visible at the bottom of Landing page where admin can add a new album which is shown in FIGURE 15.

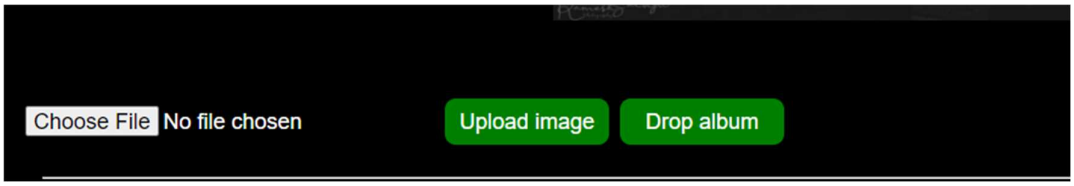

FIGURE 16. Image addition and album delete component

Similar to adding a new album, if the admin user is authenticated, the components for adding a new image and deleting a whole album as shown in FIGURE 16. These components will be visible inside all the existing albums.

FIGURE 17. Logout page

And to log out from the admin, the user has to visit the route /admin again where a logout button is visible as shown in FIGURE 17 above.

4 CONCLUSION

The main objective of this thesis study was to implement a full stack web application using ReactJS and Firebase. The idea of the project was to build a working website for a professional photographer which can be used by both a photographer and his potential customers. Mainly, this project was implemented to show the past work of a photographer to grow the business and get the work. Along with the difficulties while implementing the project, the main objective which was planned earlier in the beginning of the project was achieved by the author.

During the implementation, various React components and libraries, Google API were used which were new to the author. The author learnt to develop a full stack web application using ReactJS and Firebase and to use API in a web application.

This project allowed the author to learn new skills related to the technologies used. Using the various libraries related to React were new to the author. Also, Google Firebase and Google API were new technologies to the author. The author learnt to connect the ReactJS and Firebase in an application to build full stack serverless application. While developing the application building the various reusable react components were also learnt by the author.

The final website was tested with various devices to test the responsiveness of the website. Also, the final result was tested among various users and the owner to test the efficiency, interactivity and smoothness of the website. According to the users, the web application was simple yet attractive, user-friendly, responsive, and informative. The website also met the expectation of the photographer to showcase his pictures and videos.

In conclusion, the whole development and implementation process was a great learning experience for the author.

5 REFERENCES

1. HTML Introduction, Date of retrieval 07 November 2020
   https://www.w3schools.com/html/html_intro.asp

2. CSS Introduction, Date of retrieval 07 November 2020
   https://www.w3schools.com/css/css_intro.asp

3. Learn ReactJS JavaScript library, Date of retrieval 08 November 2020
   https://www.tutorialspoint.com/reactjs/index.htm

4. History of ReactJS, Date of retrieval 02 December 2020
   https://www.education-ecosystem.com/guides/programming/react-js/history

5. Introduction to React Reveal, Date of retrieval 02 December 2020
   https://www.react-reveal.com/docs/

6. Bruce Mcpherson, Going GAS, Date of retrieval 03 December 2020
   https://learning.oreilly.com/library/view/going-gas/9781491940457/ch16.html#:-:text=Fire-base%20is%20a%20cloud%20datab,-fast%2C%20and%20sim-ple%20to%20use.

7. Vrijraj Singh, Introduction to Firebase, Date of retrieval 03 December 2020
   https://medium.com/codingurukul/introduction-to-firebase-f9f6ccc8a785

8. Firebase Authentication, Date of retrieval 04 December 2020
   https://firebase.google.com/docs/auth

9. Firebase Cloud Storage, Date of retrieval 04 December 2020
https://firebase.google.com/docs/storage
10. Best Code Editors for 2019, Date of retrieval 08 November 2020
https://www.elegantthemes.com/blog/resources/best-code-editors