INTEGRATED PROJECT REPORT

On

File Sharing App

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CS 203) of

COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2019

in

JUNE-2022

Submitted By: Mayank
Name(s): Sourav Kumar, Bisat, Mayank, Aayush kumar, Jay Goyal
University Roll No(s):1910990249,1910990232,1910990218,1910991071,1910990256
Semester: 6th
Session: 2019-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CHITKARA UNIVERSITY

PUNJAB

CERTIFICATE

This is to be certified that the project entitled “File Sharing App” has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2022-May-2022 is a bona fide piece of project work carried out by “Sourav Kumar, Bisat, Mayank, Aayush kumar, Jay Goyal of the students group” towards the partial fulfillment for the award of the course Integrated Project (CS 203).

Sign. of Project Guide :

Name of Project Guide

Gaganpreet Kaur (Mentor)

(Designation & Department)

CANDIDATE'S DECLARATION

We, Sourav Kumar, Bisat, Mayank, Aayush kumar, Jay Goyal OF THE STUDENTS GROUP, B.E.-2019 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled "File Sharing App" is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

Sign. of Student 1 Sign. of Student 2 Sign. of Student 3 Sign. of Student 4 Sign. of Student 5
Sourav Kumar, Bisat, Mayank, Aayush kumar, Jay Goyal

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as a part of the curriculum.

We also extend our sincere appreciation to "Vikash Kumar" who provided his valuable suggestions and precious time in accomplishing our Integrated project report.

Lastly, We would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to-day experience and received lots of suggestions that improve our quality of work.

1. Abstract

This chapter gives an introduction to the project by defining the technicality of working of an online file sharing application, the main objectives that the system expects to achieve and a brief introduction to existing solutions.

The project’s objective is to design the same application with the given analogy.

In today’s era most of the file sharing happens online. It is very difficult to manage physical hard drives.

We can share our files via various platforms such as Google Drive, Telegram, etc, but the problem is that the receiver is first required to wait for the sender's uploading time and then can start downloading.

In order to solve this problem we studied how one can share files in real time with the expectation of their security without any middle storage server so we found peer to peer file sharing that's the same technology on which torrents function.

So we have designed a real time file sharing web application, where you can share files and upload and download takes place in parallel.

2.Introduction to the project

2.1 Background :

In this chapter we’ll know about the platform used and the languages and their background. We will also see how the use of those languages help the project.

2.2 Problem Statement :

File sharing is the practice of sharing or offering access to digital information or resources, including documents, multimedia (audio/video), graphics, computer programs, images and e-books. It is the private or public distribution of data or resources in a network with different levels of sharing privileges.

File sharing can be done using several methods. The most common techniques for file storage, distribution and transmission include the following:

- Removable storage devices
- Centralized file hosting server installations on networks
- World Wide Web-oriented hyperlinked documents
- Distributed peer-to-peer networks

3. Software and Hardware Requirement Specification

3.1 Programming Environment :

HTML

HTML Markup HTML pages are created by tagging textual information with HTML markup. HTML markup consists of tags, which appear inside angled brackets < and > An example of an HTML tag is , which causes text to appear in bold. only notes where text should begin to appear in bold, while the tag marks the end of the emboldening. Most HTML tags have a corresponding end tag, which is specified by the name of the tag preceded by the / character. So, to create the text: Internet Commerce is great! The text is marked up as: Internet Commerce is great! Another example of an HTML tag is , which causes text to appear in italic. In HTML 4.01, the tag was used to render text in italics. However, this is not necessarily the case with HTML5. Style sheets can be used to format the text inside the element. This will be demonstrated later. Note that tags are not case-sensitive. In other words, or are the same tag, both specifying bold text

HTML stands for HyperText Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web

pages.

HTML is a markup language that is used by the browser to manipulate text, images, and other content to display it in the required format.

Characteristics of HTML:

Easy to understand: It is the easiest language you can say, very easy to grasp this language and easy to develop.
Flexibility: This language is so much flexible that you can create whatever you want, a flexible way to design web pages along with the text.
Linkable: You can make linkable text like users can connect from one page to another page or website through these characteristics.
Limitless features: You can add videos, gifs, pictures or sound anything you want that will make the website more attractive and understandable.
Support: You can use this language to display the documents on any platform like Windows, Linux or Mac.

HTML Structure: The structure of HTML document are given below:

```
<!DOCTYPE html>

<html>

 <head>

 <title> <!-- title bar --> </title>

 <!-- header for the website -->

 </head>

 <body>


 <!-- body section of the website -->


 </body>

</html
```

CSS

Cascading Style Sheets, fondly referred to as CSS, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables

you to do this independent of the HTML that makes up each web page.

There are three types of CSS which are given below:

Inline CSS
Internal or Embedded CSS
External CSS

Inline CSS: Inline CSS contains the CSS property in the body section attached with an element is known as inline CSS. This kind of style is specified within an HTML tag using the style attribute.

Example:

```
<!DOCTYPE html>

<html>
  <head>

 <title>Inline CSS</title>

 </head>



 <body>

 <p style = "color:#009900; font-size:50px;

 font-style:italic; text-align:center;">

 GeeksForGeeks

 </p>

 </body>

</html>
```

here are many ways to include CSS file which are listed below:

External style sheet (Using HTML <link> Tag): External CSS contains separate CSS file which contains only style property with the help of tag attributes (For example class, id, heading, … etc). CSS property written in a separate file with .css extension and should be linked to the HTML document using link tag. This
means that for each element, style can be set only once and that will be applied across web pages. The link tag is used to link the external style sheet with the html webpage.
`<link rel="stylesheet" href="style.css">`

External style sheet (Using the @import At-Rule): At-rule method must

be included either within <style> tag or else inside style sheet.

```
<style>
@import url(style.css);
</style>
```

Internal style sheet (Using the <style> Element): This can be used when a single HTML document must be styled uniquely. The CSS rule set should be within the HTML file in the head section i.e the CSS is embedded within the HTML file.

```
<style>
element {
// CSS property
}
</style>
```

Inline Style Inline CSS contains the CSS property in the body section attached with element is known as inline CSS. This kind of style is specified within an HTML tag using style attribute. It is used to apply a unique style for a single element.

```
<h1 style="style property">Geeksforgeeks</h1>
```

Best Approach: The External Style Sheet (using HTML <link> Tag) is the best method which is used to link the element. Maintaining and re-using the CSS file across different pages is easy and efficient. The <link> tag is placed in the HTML <head> element. To specify a media type=”text/css” for a Cascading Style Sheet <type> attribute which is used to ignore style sheet types that are not supported in a browser.

JavaScript

JavaScript is a lightweight, cross-platform and interpreted scripting language. 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.

Example:

```
<!DOCTYPE html>
<html>
 <head>
 <title></title>
```

```
</head>

<body bgcolor="white">

<p>Paragraph 1</p>

<script type="text/javascript">

document.bgColor ="lightblue";

</script>

</body>

<html>
```

Run on IDE
Output:

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 needed.

React Js

React (also known as React.js or ReactJS) is an open-source, front end, JavaScript library[3] for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies.[4][5][6] React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality

```
import React from 'react';
import ReactDOM from 'react-dom';

class Car extends React.Component {
 render() {
 return <h2>Hi, I am a Car!</h2>;
 }
}

ReactDOM.render(<Car />, document.getElementById('root'));
```

3.3 Requirements to run the applications:

VS- Code (Software)

Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS.[7] Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

Microsoft has released Visual Studio Code's source code on the VSCode repository of GitHub.com, under the permissive MIT License, while the compiled binaries are freeware.

In the Stack Overflow 2019 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents reporting that they use it

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js and C++. It is based on the Electron framework, which is used to develop Node.js Web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language. It supports a number of programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface, but can be accessed via the command palette.

Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, and debuggers, perform static code analysis, and add code linters using the Language Server Protocol.

Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free

alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

Visual Studio Code allows users to set the code page in which the active document is saved, the newline character, and the programming language of the active document. This allows it to be used on any platform, in any locale, and for any given programming language.

3.4 Package Manager

npm is the package manager for the Node JavaScript platform. It puts modules in place so that node can find them, and manages dependency conflicts intelligently. It is extremely configurable to support a wide variety of use cases. Most commonly, it is used to publish, discover, install, and develop node programs.

4. Program's Structure Analyzing and GUI Constructing (Project Snapshots)

4.1 Structure Code

Detail.js

```
import { useEffect, useState } from "react";
import { useParams } from "react-router-dom";
import styled from "styled-components";
import db from "../firebase";

const Detail = (props) => {
 const { id } = useParams();
 const [detailData, setDetailData] = useState({});

 useEffect(() => {
 db.collection("movies")
 .doc(id)
 .get()
```

5. Conclusion

5.1 Chapter Overview

This chapters draws the project report to a close and reflects on the design decisions made throughout. It also discusses possible future development ideas.

5.2 Project Overview

The system achieved all of its proposed priority 1 and priority 2 functional requirements and even some priority 3 outlined in Section 3.5.

However, the initial project plan and gannt chart had to be modified as the project became about a month behind due to underestimations on the time to implement some desired features. This meant that some of the lower priority requirements had to be scrapped.

5.3 Further Development

This project was developed under time constraints of 120 hours. Therefore the proposed features specified in the requirements were what the developer thought to be realistic targets.

However, if more time became available the following could be implemented.

5.4 Graphical User Interface (GUI)

Currently the GUI was adequate to do the job but maybe the GUI could have had a more appealing look and feel. This all beckons on whether or not Java was the best programming language to use to generate the best looking GUI. Maybe a web developed GUI could have been a better alternative but with the developer having little experience in web development this would not have been ideal.

6. Future Scope

Bibliography

[1] Ian Alexander. Computing & control engineering. Volume 14(Issue 1):22–26, April 2003. [2] EzeeBurrp. Software Features. Accessed on 14 October, 2009. http://www.ezeeburrp.com/ features.html. [3] HSQL db. Database comparison. Accessed on 13 September, 2009. http://hsqldb.org/images/ imola_retrieve.jpg.

[4] Business Dictionary. Stakeholder. Accessed on 2 April, 2010. http://www.businessdictionary. com/definition/stakeholder.html.

[5] Geoffrey Elliot. Global business information technology: an integrated systems approach. Pearson Education, 2004.

[6] Youtube courses on React