**STUDENT TOOLBOX**

**A PROJECT REPORT**

***Submitted By***

**JAI SARAN. S (Reg. No. 20203071401116)**

**MUTHU GANESH. R (Reg. No. 20203071401121)**

**RAM KUMAR. T (Reg. No. 20203071401129)**

**SUDHARSUN. M (Reg. No. 20203071401138)**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF COMPUTER APPLICATIONS**

****

**DEPARTMENT OF COMPUTER APPLICATIONS**

**GOVERNMENT ARTS AND SCIENCE COLLEGE**

PALKULAM, KANYAKUMARI – 629401

**APRIL 2023**

**GOVERNMENT ARTS AND SCIENCE COLLEGE**

PALKULAM, KANYAKUMARI – 629401

**BONAFIDE CERTIFICATE**

Certified that this Project Report entitled “STUDENT TOOLBOX” is the bonafide work of **JAI SARAN. S (Reg. No: 20203071401116), MUTHU GANESH. R (Reg. No: 20203071401121), RAM KUMAR. T (Reg. No: 20203071401129)** and **SUDHARSUN. M (Reg. No: 20203071401138)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Mrs. P. VASANTHI Dr. A. FERDINAND CHRISTOPHER

Assistant Professor & Head Assistant Professor & Supervisor

Department of Computer Applications Department of Computer Applications

Government Arts and Science College Government Arts and Science College Palkulam, Kanyakumari – 629401 Palkulam, Kanyakumari – 629401

Submitted for the viva –voce held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

External Examiners

**ACKNOWLEDGEMENT**

An endeavor carried about for a long period is complete only with the help and support of many well wishers. First and foremost I would like to thank God Almighty for helping us to successfully complete this project.

We thank Dr. G. SAROJA, the Principal of Government Arts and Science College, Palkulam, Kanyakumari, for allowing us to proceed with our project.

We specially acknowledge our thanks to Mrs. P. Vasanthi, Assistant Professor and Head, Department of Computer Applications, for providing timely advice and all facilities in making our venture a complete success.

We express our gratitude to our Supervisor Dr. A. Fredinand Christopher, Assistant Professor & Organizing Secretary of ASCA, Department of Computer Applications, for being a wonderful guidance to us in our efforts and whole heartedly helping us to do this project in a wonderful manner.

We express our thanks to all the staff members of the Department of Computer Applications. Last but not the least we extend our heartfelt thanks to our family and friends for their constant support.

**JAI SARAN. S (Reg. No: 20203071401116)**

**MUTHU GANESH. R (Reg. No: 20203071401121)**

**RAM KUMAR. T (Reg. No: 20203071401129)**

**SUDHARSUN. M (Reg. No: 20203071401138)**

**ABSTRACT**

Student Toolbox is a project aimed at providing a comprehensive set of tools for students to assist them in their academic and professional pursuits. The toolbox comprises four major sections: Resume Builder, Flappy Bird Game, Calculator using VB, and Document Converter. Each section serves a distinct purpose and is designed to be user-friendly and intuitive.

The Resume Builder section allows students to create a professional resume tailored to their specific needs.

The Flappy Bird Game section is a fun and engaging game that allows students to take a break from their academic work and have some leisure time.

The Calculator using VB section provides students with a powerful tool for solving mathematical problems. The calculator is user-friendly.

The Document Converter section allows students to convert their documents into different formats. It supports various file formats, including PDF, DOC, and JPG, making it easy for students to share their documents with others.

Overall, the Student Toolbox project is a valuable resource for students, providing them with tools to succeed academically and professionally while also offering a fun and engaging way to take a break from their studies.

**TABLE OF CONTENTS**

**CHAPTER TITLE PAGE NO**

**ABSTRACT** III

**INTRODUCTION**  1

* 1. GENERAL 1
  2. NEED FOR THE STUDY 1
  3. OBJECTIVE OF THE STUDY 3

1. **SYSTEM ANALYSIS** 4
   1. FEASIBILITY STUDY 4
      1. OPERATIONAL 4
      2. TECHNICAL 5
      3. ECONOMICAL 6
2. **REQUIREMENT ANALYSIS** 7
   1. HARDWARE REQUIREMENT 7
   2. SOFTWARE REQUIREMENT 7
   3. EXSISTING SYSTEM 7
   4. PROPOSED SYSTEM 8
3. **DESIGN AND PLANNING** 9
   1. DEVELOPMENT LIFE CYCLE MODEL 9
      1. WATERFALLS MODEL 9
   2. GENERAL OVERVIEW 11
   3. USE CASE DIAGRAM 12
   4. ER DIAGRAM 13
4. **IMPLEMENTATION** 14
   1. FRONT END 14
   2. BACK END 21
   3. SOURCE CODE
5. **OUTPUT** 24
6. **FUTURE ENHANCEMENT**
7. **REFERENCES**

**INTRODUCTION**

* 1. **GENERAL**

Student Tool Box is a comprehensive project designed to help students with their academic and personal needs. The toolbox features four major sections, each designed to address a specific area of concern.The first section is the Resume Builder, which provides an easy-to-use platform for students to create professional resumes. With customizable templates and helpful tips, students can create impressive resumes that showcase their skills and achievements.The second section is the Flappy Bird Game, a fun and challenging game that provides a break from studying and allows students to unwind.The third section is the Calculator using VB, which provides a powerful tool for students to solve complex mathematical equations. With a user-friendly interface and advanced features, this calculator is designed to help students with their academic work.Finally, the fourth section is the Document Converter, which allows students to convert files from one format to another with ease. This tool is particularly useful for students who need to submit assignments in specific formats or who need to convert files for compatibility with different devices.Overall, Student Tool Box is a valuable resource for students, offering a wide range of tools to help them succeed both academically and personally. Whether you need to create a resume, take a break with a fun game, solve a complex math problem, or convert a file, Student Tool Box has you covered.

Top of Form

* 1. **NEED FOR THE STUDY**

Student Tool Box is a valuable resource that can address the academic and personal needs of students. With the rising demands of academic work and the need to excel in different areas, students require a variety of tools to help them succeed. The following are some of the reasons why Student Tool Box is necessary for students:

Resume Builder: The ability to create a professional and impressive resume is essential for students who are preparing to enter the workforce. With the customizable templates and helpful tips provided by the Resume Builder section of Student Tool Box, students can easily create a resume that highlights their skills and achievements, making them stand out to potential employers.

Flappy Bird Game: While academic work is important, students also need to take breaks to unwind and relieve stress. The Flappy Bird Game section of Student Tool Box provides a fun and challenging game that allows students to take a break from their studies and engage in a fun activity.

Calculator using VB: Complex mathematical equations can be daunting for students, and having a powerful tool like the Calculator using VB section of Student Tool Box can help them solve these problems with ease. With a user-friendly interface and advanced features, this calculator can assist students with their academic work and help them understand complex mathematical concepts.

Document Converter: With different file formats and compatibility issues, students may struggle to convert files for different devices or submit assignments in specific formats. The Document Converter section of Student Tool Box can help students convert files from one format to another with ease, making it easier for them to complete their assignments and work with different devices.

In conclusion, Student Tool Box is a valuable resource for students, addressing their academic and personal needs. The various sections of the toolbox, including the Resume Builder, Flappy Bird Game, Calculator using VB, and Document Converter, provide students with the necessary tools to succeed academically and personally.

* 1. **OBJECTIVE OF THE STUDY**

The objective of the study on Student Tool Box is to evaluate its effectiveness in addressing the academic and personal needs of students. The study aims to assess the usefulness and effectiveness of each section, including the Resume Builder, Flappy Bird Game, Calculator using VB, and Document Converter, in helping students create professional resumes, take breaks, solve complex mathematical equations, convert files, and succeed academically and personally. Ultimately, the study seeks to determine the overall effectiveness of Student Tool Box as a resource for students.

**SYSTEM ANALYSIS**

System analysis is the process of examining an existing system or methodology with intent of improving it through better procedure and method. Before planning for improvement of the existing system, complete understanding about the system is essential. System analysis can collect the relative facts about the system. System is simply a set of components that interact to accomplish some purpose. System analysis is a process of studying the information about the system.

**2.1. FEASIBILITY STUDY**

A feasibility study is a high-level capsule version of the entire System analysis and Design Process. The study begins by classifying the problem definition. Feasibility is to determine if it’s worth doing. Once an acceptance problem definition has been generated, the analyst develops a logical model of the system. A search for alternatives is analyzed carefully. There are 3 parts in feasibility study.

1. Operational Feasibility
2. Technical Feasibility
3. Economical Feasibility

**2.1.1. OPERATIONAL**

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The operational feasibility assessment focuses on the degree to which the proposed development projects fits in with the existing business environment and objectives with regard to development schedule, delivery date, corporate culture and existing business processes. To ensure success, desired operational outcomes must be imparted during design and development. These include such design-dependent parameters as reliability, maintainability, supportability, usability, predictability, disposability, sustainability, affordability and others. These parameters are required to be considered at the early stages of design if desired operational behaviors are to be realized. A system design and development requires appropriate and timely application of engineering and management efforts to meet the previously mentioned parameters. A system may serve its intended purpose most effectively when its technical and operating characteristics are engineered into the design. Therefore, operational feasibility is a critical aspect of systems engineering that needs to be an integral part of the early design phase

**2.1.2. TECHNICAL**

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on outline design of system requirements in terms of input, processes, output, fields, programs and procedures. This can be qualified in terms of volume of data, trends, frequency of updating in order to give an introduction to the technical system. The application is the fact that it has been developed on windows XP platform and a high configuration of 1GB RAM on Intel Pentium Dual core processor. This is technically feasible .The technical feasibility assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system. It is an evaluation of the hardware and  
software and how it meets the need of the proposed system.

**2.1.3. ECONOMICAL**

Establishing the cost-effectiveness of the proposed system i.e. if the benefits do not outweigh the costs then it is not worth going ahead. In the fast paced world today there is a great need of online social networking facilities. Thus the benefits of this project in the current scenario make it economically feasible. The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. It includes quantification and identification of all the benefits expected. This assessment typically involves a cost/benefits analysis.

**REQUIREMENT ANALYSIS**

Even though there are lots of online quiz examination system software’s available in the market it offers wide range of facilities with high end technologies. So cost & technology wise it is very difficult to maintain for users like us. Our “Online Quiz System” paves the way to conduct examinations during this pandemic which is far better than buying commercial software available in the global market. In future we can customize the software according to our need.

* 1. **HARDWARE REQUIREMENTS**
* Processor : Pentium 1 and above processors
* Memory : 2 GB RAM
* Hard Disk Space : 250 GB or more

**3.2. SOFTWARE REQUIREMENTS**

* Operating system : Windows XP / Windows 7 / 8.1 / 10 / 11
* Front End : HTML, CSS, Bootstrap, JavaScript, Visual Basic, Typescript
* Back End : Node.JS, React.js
* IDE : Microsoft Visual Studio Code Editor
* Browser : Google Chrome, IE, Opera, Mozilla Firefox

**3.3. EXSISTING SYSTEM**

In existing system, the PDF converter are being prepared manually by separated option with their third-party ads. It took a long time to do that process. There was a lot of ads and add premium membership to download high-quality file. Existing system has many disadvantages in it.

**3.4. PROPOSED SYSTEM**

Separate assessment is prone to errors and is not time efficient as discussed previously. So why not automate the whole assessment process? Why would a student spend his/her precious time waste to watch an Ads or spend money with premium membership? So our main objective is an online service to work with PDF file completely free and easy to use.

**DESIGN AND PLANNING**

**4.1. DEVELOPMENT LIFE CYCLE MODEL**

**4.1.1. WATERFALLS MODEL**

The waterfall model was selected as the SDLC model due to the following reasons:

* Requirements were very well documented, clear and fixed.
* Technology was adequately understood.
* Simple and easy to understand and use.
* There were no ambiguous requirements.
* Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
* Clearly defined stages.
* Well understood milestones. Easy to arrange tasks.

Requirement

Analysis

System Design

Implementation

Maintenance

Deployment

Testing

Figure: 4.1 Waterfalls Model

**4.2. GENERAL OVERVIEW**

Figure: 4.2 General Overview Model

**4.3. USE CASE DIAGRAM**

Student

Figure: 4.3 Student Module

**4.4. ER DIAGRAM**

**­**

Student

Resume

Calc Park

Doc wizard

Flappy Bird

Play a Game

Calculator

Edit Resume

Build Resume

Convert PDF

Edit PDF

Simple Calc

Precentege Calc

Resume file

PDF file

Download

Figure: 4.6 ER – Diagram

**IMPLEMENTATION**

**5.1. FRONT END**

Let’s start at the beginning. So what is front-end web development, also called client-side development, refers to writing Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript code for a website or web application so that users can see and interact directly with them.

To delve further into this subject, it is important to know the basic differences between front-end development, back-end development and full-stack development. Simply put, a front end developer works on the parts / aspects of an app or website that users get to see and interact with. Whereas a back-end developer handles the behind-the-scenes aspects of the said app or website, say the infrastructure and databases. On the other hand, a full-stack developer does both, so it’s someone who deals with the whole design process from start to finish.

The main front-end development languages, as we have seen already, are HTML, CSS and JavaScript. JQuery has also been used but it is said to be going out of style whereas many legacy projects still use the JavaScript library, ensuring it continues to be taught at front-end boot camps. Front-end development also deals extensively with responsive design along with typography, layouts, grid system, and color theory. Now we are further going to discuss the languages that are used in front-end development of this project.

**5.1.1. HTML**

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

**5.1.2. CSS**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

CSS information can be provided from various sources. These sources can be the web browser, the user and the author. The information from the author can be further classified into inline, media type, importance, selector specificity, rule order, inheritance and property definition. CSS style information can be in a separate document or it can be embedded into an HTML document. Multiple style sheets can be imported. Different styles can be applied depending on the output device being used; for example, the screen version can be quite different from the printed version, so that authors can tailor the presentation appropriately for each medium. The style sheet with the highest priority controls the content display. Declarations not set in the highest priority source are passed on to a source of lower priority, such as the user agent style. The process is called cascading.

One of the goals of CSS is to allow users greater control over presentation. Someone who finds red italic headings difficult to read may apply a different style sheet. Depending on the browser and the web site, a user may choose from various style sheets provided by the designers, or may remove all added styles and view the site using the browser's default styling, or may override just the red italic heading style without altering other attributes.

**5.1.3. Bootstrap**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components. Bootstrap is the third-most starred project on GitHub, with more than 135,000 stars, behind only free Code Camp (almost 305,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap getbootstrap.com is in the top-2000 in US while vuejs.org is in top-7000 in US.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

**5.1.4. JavaScript**

JavaScript is a high-level, dynamic programming language that is widely used for creating interactive web pages and web applications. It is one of the three core technologies of the World Wide Web, alongside HTML and CSS. JavaScript code can be embedded directly into HTML documents, or it can be included in separate JavaScript files that are referenced by the HTML document.

JavaScript was first introduced in 1995 by Brendan Eich while he was working at Netscape Communications Corporation. It was initially called Mocha, then renamed to LiveScript, and finally renamed to JavaScript to capitalize on the popularity of the then-new Java programming language. Since then, JavaScript has evolved into a powerful and versatile language that is used by millions of developers around the world.

One of the key features of JavaScript is its ability to manipulate the Document Object Model (DOM) of a web page. The DOM is a hierarchical tree-like structure that represents the content and structure of an HTML document. With JavaScript, you can add, remove, or modify elements of the DOM in real time, allowing you to create dynamic and interactive web pages that respond to user input.

JavaScript also supports a wide range of programming paradigms, including object-oriented, functional, and procedural programming. It has a large and active community of developers who have created numerous libraries and frameworks that make it easier to develop complex web applications. Some of the most popular JavaScript frameworks include jQuery, React, Angular, and Vue.js.

**5.1.5. Visual Basic**

Visual Basic 6.0 is a programming language and integrated development environment (IDE) used to develop software applications for Windows operating systems. The language is an object-oriented programming (OOP) language that is designed to be easy to learn and use. Visual Basic 6.0 allows developers to create user interfaces and interact with databases and other software systems.

The Visual Basic 6.0 IDE consists of several components, including a code editor, a form designer, and a debugger. The code editor allows developers to write and edit code, while the form designer allows them to create user interfaces visually. The debugger is used to find and fix errors in code.

Visual Basic 6.0 uses a variety of data types, including integers, strings, and Boolean values, to store data. It also supports arrays, which allow developers to store and manipulate large sets of data. Visual Basic 6.0 also includes a set of built-in functions and objects, such as the MsgBox function for displaying messages to users and the FileSystemObject object for working with files and folders.

Visual Basic 6.0 can be used to create a wide variety of applications, including database applications, games, and utility programs. The language is particularly well-suited for creating applications with graphical user interfaces (GUIs) and for working with databases. In addition to the built-in functionality provided by the language, Visual Basic 6.0 can also be extended with third-party libraries and add-ons.

**5.1.6. TYPESCRIPT**

TypeScript is a free and open-source programming language developed by Microsoft. It is a strict syntactical superset of JavaScript, which means that any valid JavaScript code is also valid TypeScript code. TypeScript adds optional static typing and other features to JavaScript, making it more powerful and robust.

One of the main advantages of TypeScript is its strong type system. TypeScript allows developers to define types for variables, parameters, and functions, making it easier to catch errors early on in the development process. This can lead to more reliable and maintainable code.

TypeScript also supports object-oriented programming (OOP) principles, including classes, interfaces, and inheritance. This makes it easier to organize and structure code for larger projects, and it can help improve code reusability.Another benefit of TypeScript is its tooling support. TypeScript is supported by many code editors and integrated development environments (IDEs), which provide features such as code completion, error checking, and refactoring. This can help developers write code more efficiently and reduce the risk of bugs.

TypeScript can be used for both client-side and server-side programming, and it can be integrated with other popular web frameworks and libraries, such as React and Angular. TypeScript can also be compiled into JavaScript, which means it can run on any browser or server that supports JavaScript.

Overall, TypeScript is a powerful and flexible programming language that can improve code quality and reduce the risk of errors. Its strong type system, OOP support, and tooling make it an excellent choice for building large, complex web applications.

**5.2. BACK END**

Back-end development, also called server-side development, alludes to the behind-the-scenes activities that take place when an action is performed on a website. This action could be logging in to one’s account or purchasing a watch from an online store. So a backend developer trains his/her sights on databases, scripting and the architecture of websites. The code written by such a back-end developer helps in passing on the database information to the browser.

**5.2.1. NODE.JS**

Node.js is a popular open-source runtime environment that allows developers to run JavaScript code on the server-side, making it an excellent choice for backend development. Node.js is built on the V8 JavaScript engine used in Google Chrome, which makes it fast and efficient.

One of the main advantages of Node.js is its non-blocking I/O model, which allows it to handle large amounts of data and requests without slowing down. This makes it a great choice for building scalable and high-performance web applications.

Node.js also has a vast ecosystem of packages and modules available through its package manager, npm. These packages can be easily integrated into Node.js applications, allowing developers to save time and effort when building complex applications.

Additionally, Node.js is cross-platform, which means it can run on various operating systems, including Windows, macOS, and Linux, making it a versatile choice for developers. Its popularity has led to the creation of many online communities, forums, and resources that provide support and information for developers using Node.js.

Overall, Node.js is a powerful tool for building fast, scalable, and efficient server-side applications, making it an excellent choice for backend development.

**5.2.2. REACT.JS**

React.js, often referred to as simply React, is an open-source JavaScript library that is widely used for building user interfaces (UIs) for web applications. React was developed by Facebook, and it has become one of the most popular front-end frameworks in the world.

One of the key features of React is its ability to handle complex UIs with ease. React uses a component-based architecture, which allows developers to break down complex UIs into smaller, reusable components. This makes it easier to maintain and update the codebase over time.

Another advantage of React is its virtual DOM, which is a lightweight copy of the real DOM. When a user interacts with a React-based web application, the virtual DOM is updated instead of the real DOM. This makes React faster and more efficient, as it only updates the parts of the UI that have changed, rather than re-rendering the entire page.

React also integrates well with other libraries and frameworks, making it easy to build complex web applications. React Native, for example, is a framework that allows developers to use React to build native mobile applications for iOS and Android devices.

Overall, React is a powerful and flexible tool for building user interfaces for web applications. Its popularity, large community, and vast ecosystem of libraries and tools make it an excellent choice for front-end development.

**5.3. SOURCE CODE**

**Index.html**

<!DOCTYPE html>

<html>

<head>

<title>Student Toolbox</title>

<!-- Include Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="css/style.css">

<!-- Include jQuery -->

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>

<!-- Include Bootstrap JS -->

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>

<style>

.tool-box {

background-color: #fff;

border-radius: 10px;

box-shadow: 0px 2px 10px rgba(0, 0, 0, 0.3);

padding: 20px;

margin-bottom: 20px;

height: 450px;

text-align: center;

}

.tool-box:hover {

box-shadow: 0px 2px 20px rgba(0, 0, 0, 0.5);

}

.tool-box img {

max-width: 100%;

max-height: 180px;

margin-bottom: 15px;

}

.tool-box h3 {

font-size: 24px;

margin-bottom: 15px;

color: #5a5a5a;

font-weight: bold;

}

.tool-box p {

font-size: 18px;

line-height: 24px;

color: #999;

margin-bottom: 20px;

}

.tool-box a.btn {

background-color: #007bff;

color: #fff;

border-radius: 5px;

padding: 10px 30px;

font-size: 18px;

font-weight: bold;

transition: all .3s ease;

margin-top: auto;

}

.tool-box a.btn:hover {

background-color: #0056b3;

color: #fff;

text-decoration: none;

box-shadow: 0px 2px 20px rgba(0, 0, 0, 0.5);

}

</style>

</head>

<body>

<div class="title">

<a href="docwizard.html" id="navbartitle">Student Toolbox</a>

</div>

<nav class="sticky-top">

<div class="nav">

<ul>

<a href="#download">

<li>Calc Park</li>

</a>

<a href="../Flappy Bird/flappybird.html" target="\_blank">

<li>Flappy Bird</li>

</a>

<a href="pdf to excel.html">

<li>Docwizard</li>

</a>

<a href="http://localhost:3000/" target="\_blank">

<li>Resume Builder</li>

</a>

</ul>

</div>

</nav>

<br><br>

<div class="container">

<div class="row">

<div class="col-md-6">

<div class="tool-box" >

<img src="images/calcpark.jpg" alt="Calculator Park">

<h3>Calc Park</h3>

<p>Calc Park is a Visual Basic application that provides a normal and percentage calculator. It is a useful tool for performing basic and percentage calculations quickly and easily.</p>

<a href="../Calc Park/CALC PARK.rar" id="download" class="btn btn-primary">Download Calc Park Application</a>

</div>

</div>

<div class="col-md-6">

<div class="tool-box">

<img src="images/docwizard.jpg" alt="Docwizard">

<h3>Doc Wizard</h3>

<p>Doc Wizard is a tool for converting documents from one format to another. It supports a variety of file types, including PDF, DOC, and JPG.</p>

<a href="http://127.0.0.1:3000/" target="\_blank" class="btn btn-primary">Use Doc Wizard</a>

</div>

</div>

</div>

<div class="row">

<div class="col-sm-6">

<div class="tool-box">

<img src="images/resume.jpg" alt="Resume Builder">

<h3>Resume Builder</h3>

<p>Resume Builder is a web-based tool that allows users to create professional-looking resumes quickly and easily. It provides a variety of templates and customization options to help users create a resume that suits their needs.</p>

<a href="http://localhost:3000/" target="\_blank" class="btn btn-primary">Build a Resume</a>

</div>

</div>

<div class="col-sm-6">

<div class="tool-box">

<img src="images/flappybird.jpg" alt="Flappy Bird Game">

<h3>Flappy Bird Game</h3>

<p>Flappy Bird is a popular game that challenges players to navigate a bird through a series of obstacles. This version of the game is built using HTML, CSS, and JavaScript.</p>

<a href="../Flappy Bird/flappybird.html" target="\_blank" class="btn btn-primary">Play Flappy Bird</a>

</div>

</div>

</div>

</div>

</body>

</html>

**flappybird.html**

<html>

<head>

<style>

/\* Load the flappybird font from an external source \*/

@font-face {

font-family: flappybird;

src: url('http://www.mediafont.com/storage/contents/3184/font.eot');

/\* IE \*/

src: url('http://www.mediafont.com/storage/contents/3184/04B\_19\_\_.TTF');

/\* non-IE \*/

}

body {

margin: 0;

/\* Make sure the body has 0 margin initially \*/

width: 100%;

/\* Define we're using full width of the browser \*/

height: 100%;

/\* Define we're using full height of the browser \*/

overflow: hidden;

/\* Hide scrollbars \*/

}

#Canvas {

position: relative;

/\* Position relative, so all absolute items will be "contained"" through here \*/

width: 100%;

/\* Define that we're using the full width of the body \*/

height: 100%;

/\* Define that we're using the full width of the body \*/

background: url(sky.jpg);

background-size: cover;

/\* Temporary color background \*/

}

#Birdy {

/\* Ratio is about 1.449 (Width/Height) \*/

width: 2.9%;

padding-bottom: 2%;

background-image: url('http://flappybird.io/img/bird.png');

background-size: 300%;

/\* Make the width 300% of the current width, so that only 1 frame fits and the other 2 frames are hidden \*/

position: absolute;

/\* Absolutely positioned, so it's out of flow of regular elements, position wherever we'd like \*/

top: 50%;

/\* We're going to be around middle of the screen (a bit lower due to height)\*/

left: 20%;

/\* We're going to be at around 20% left of the screen \*/

z-index: 150;

/\* Display in front of pipes even after getting hit by them \*/

}

.FallenBirdy {

-webkit-transform: rotate(90deg);

-moz-transform: rotate(90deg);

-ms-transform: rotate(90deg);

-o-transform: rotate(90deg);

transform: rotate(90deg);

}

#PauseButton {

position: absolute;

/\* Position button so it's not hugging the screen edge \*/

top: 2%;

right: 1%;

width: 3%;

/\* Make the width scale with the canvas size \*/

height: 0;

/\* Set an initial 0 height \*/

/\* Use padding of same width% to create a square

Padding uses percentages based on width, so we're going to use padding that'll add up to 3%\*/

padding-top: 1.25%;

padding-bottom: 1.75%;

background: orange;

border-radius: 5px;

/\*Rounded corners\*/

font-size: 2.6vw;

/\* Scale Font to 3% of ViewPort\*/

text-align: center;

/\* Horiziontally center pause text \*/

z-index: 50;

/\* Make it lay on top of the pipes \*/

cursor: pointer;

/\* Make it appear like it's a button so the cursor turns into a hand when hovered \*/

}

#PauseButton span {

line-height: 3%;

/\*Vertically center the pause text \*/

margin-left: 3%;

/\* Adjust pause text position \*/

}

.Pipe {

position: absolute;

/\* Position it absolutely, not in document flow \*/

top: 0;

/\* Hug the top of the screen \*/

left: 100%;

/\* Start off the screen \*/

width: 5%;

/\* Arbitrary pipe percentage \*/

background: lime;

border: double yellow 1px;

animation: PipeMovement 15s linear;

/\* Run the movement animation in 15 seconds linearly \*/

-webkit-animation: PipeMovement 15s linear;

/\* webkit prefix \*/

}

@keyframes PipeMovement {

from {

left: 100%

}

/\* Define that we'll start at position 100% left (off screen to the right) \*/

to {

left: -25%

}

/\* Define that we'll end at -25% left (off screen to the left) \*/

}

@-webkit-keyframes PipeMovement {

/\* webkit prefix \*/

from {

left: 100%

}

to {

left: -25%

}

}

/\* Pause the animation when the user hits the pause button \*/

.paused {

-ms-animation-play-state: paused;

-o-animation-play-state: paused;

-moz-animation-play-state: paused;

-webkit-animation-play-state: paused;

animation-play-state: paused;

}

/\* Prevents the user from highlighting elements we don't want highlighted' \*/

.noSelect {

-webkit-touch-callout: none;

-webkit-user-select: none;

-khtml-user-select: none;

-moz-user-select: none;

-ms-user-select: none;

user-select: none;

}

#LostScoreScreen {

/\* Position the score box so it's perfectly centered on screen \*/

position: relative;

/\* Position relative inside the absolute container \*/

left: -50%;

/\* Move it so half of it is left of the center \*/

margin-top: -70%;

/\* Raise it so that it's about center \*/

padding: 5px 0px 5px 12px;

/\* Center the text properly and give it a bit of space \*/

background: #ded895;

/\* Random color, kinda matches flappy bird \*/

border-radius: 4%;

/\* Give it rounded edges \*/

border: 2px solid black;

/\* 2px thick black border for the score screen \*/

text-align: center;

/\* Horiziontally center the text \*/

display: none;

/\* Hide initially \*/

font-family: flappybird;

/\* Use the flappybird font for the score \*/

font-size: 2vw;

/\* Scale the font to 2% of viewport \*/

color: white;

/\* White font \*/

text-shadow: /\* Give it a 2px of "text shadow" to act as an outline to the score text \*/

-2px -2px 0 #000, 2px -2px 0 #000, -2px 2px 0 #000, 2px 2px 0 #000;

z-index: 150;

/\* Pop the box above the pipes \*/

}

#CurrentScoreCard {

/\* Position the score box so it's perfectly centered on screen \*/

position: relative;

/\* Position relative inside the absolute container \*/

left: -50%;

/\* Move it so half of it is left of the center \*/

margin-top: -700%;

/\* Raise it so that it's above center \*/

text-align: center;

/\* Horiziontally center the text \*/

font-family: flappybird;

/\* Use the flappybird font for the score \*/

font-size: 4vw;

/\* Scale the font to 2% of viewport \*/

color: white;

/\* White font \*/

text-shadow: /\* Give it a 2px of "text shadow" to act as an outline to the score text \*/

-2px -2px 0 #000, 2px -2px 0 #000, -2px 2px 0 #000, 2px 2px 0 #000;

z-index: 50;

/\* Raise above the pipes \*/

}

#DebugInfo {

position: absolute;

top: 0;

left: 0;

width: 150px;

height: 250px;

z-index: 25;

background: gray;

opacity: 0.7;

color: white;

display: none;

}

</style>

<!-- Bootstrap CSS from MaxCDN -->

<link rel="stylesheet" href="http://netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">

</head>

<body>

<div id="Canvas">

<!-- Inline CSS -->

<div id="InstructionBox" style="position: absolute; left: 50%; display:none;">

<div style="position: relative; left: -50%; z-index: 150; font-family:flappybird; color:white; font-size: 2vw;" class="noSelect">

Press W to Fly <br> Enter to Reset <br> "P" to Pause

</div>

</div>

<div style="position: absolute; left: 50%; top: 50%;">

<div id="CurrentScoreCard" class="noSelect">

<span id="CurrentScore">0</span>

</div>

</div>

<div style="position: absolute; left: 50%; top: 80%;width:400px;">

<div id="LostScoreScreen" class="noSelect">

<span>Score</span>

<br>

<span id="FinalScore">0</span>

<br>

<span>Best</span>

<br>

<span id="BestScore">0</span>

<br>

<button id="ResetButton" type="button" class="btn btn-warning">Reset</button>

</div>

</div>

<div id="DebugInfo" class="noSelect"></div>

<div id="Birdy" style=" background-position-x: 400%;">

</div>

<div id="PauseButton" class="noSelect">

<span class="glyphicon glyphicon-pause"></span>

</div>

</div>

<!-- jQuery from Google CDN -->

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>

<script>

$(function() {

$('#InstructionBox').slideDown(); //Slide down the instructions box

setTimeout(function() {

$('#InstructionBox').slideUp();

}, 5000); //Make it slide up after 5 seconds

var canvasObject = $('#Canvas');

var gameLoopIntervalID = 0;

var Paused = true;

var Lost = false;

function pauseGame() {

clearInterval(gameLoopIntervalID); //Stop the timer

$('.Pipe').addClass('paused'); //Pause the CSS3 animation so the pipes don't move.

$('#PauseButton span').removeClass('glyphicon-pause').addClass('glyphicon-play'); //Swap out the pause text with the play text

Paused = true; //Set paused flag to true

}

function startGame() {

if (Lost) {

return; //Don't unpause if we're already dead

}

//Run the gameLoop every 100ms and save the ID for stopping

gameLoopIntervalID = setInterval(function() {

gameLoop();

}, 35);

$('.Pipe').removeClass('paused'); //Unpause the CSS3 animations

$('#PauseButton span').removeClass('glyphicon-play').addClass('glyphicon-pause'); //Swap out the play text with the pause text

Paused = false; //Set pause flag to false

}

function endGame() {

Lost = true; //Set the lost flag to true, so they can't unpause;

pauseGame(); //"Pause" the game

var cookieScore = getCookie('HighScore');

console.log(Math.max(CurrentScore, cookieScore));

console.log(cookieScore);

setCookie('HighScore', Math.max(CurrentScore, cookieScore), 30000); //Store the highest score between current score and saved score

Birdy.BirdyObject.animate({

top: '90%'

}, 1500, 'linear'); //Drop our birdy

$('#FinalScore').html(CurrentScore); //Update the current score

$('#BestScore').html(Math.max(CurrentScore, cookieScore)); //Update the current score

$('#LostScoreScreen').slideDown(); //Have a nice slideDown for the score screen

}

function resetGame() {

pauseGame(); //Pause the game first, so we'll kill the last game loop

$('.Pipe').remove(); //Remove all pipes

Lost = false; //Reset the lost flag

CurrentScore = 0; //Reset the score

Birdy.Reset(); //Reset the bird object

startGame(); //Restart the game loop

$('#LostScoreScreen').slideUp(); //Have a nice slideUp for the score screen

}

//Toggle the pause function

function togglePause() {

if (!Paused) {

pauseGame();

} else {

startGame();

}

}

var CurrentScore = 0;

$('#PauseButton').mousedown(function(event) {

event.stopPropagation(); //Don't allow the click to register as a "jump" command, by stopping it's propagation through the DOM tree.

togglePause(); //Toggle the pause.

});

$('#ResetButton').click(function() {

resetGame();

});

$('body').keydown(function(event) {

if (event.which == 87) {

Birdy.jump(); //Jump when the user clicks

}

});

$('body').keydown(function(event) {

if (event.which == 13) {

resetGame();

}

if (event.which == 80) {

togglePause(); //Toggle the pause.

}

});

//Count how many gameLoops have passed

var gameLoopCounter = 0;

function gameLoop() {

if (gameLoopCounter % 3 === 0) { //Reduce to every 2 loops to improve performance

incrementScore();

checkCollisions();

}

isInBound(Birdy.BirdyObject, canvasObject);

Birdy.fall();

//TODO: Add decrease time between pipes as score increases

if (gameLoopCounter % 90 === 0) {

addPipe(); //Add a pipe every 90 loops ~2.7 seconds

cleanPipes(); //Remove pipes as well

}

if (gameLoopCounter % 7 === 0) { //Flap the wings every 7 loops

Birdy.flapWings();

}

gameLoopCounter++;

}

var Birdy = new(function() {

var selectorObject = $('#Birdy');

var jumping = false;

var gravVeloc = 0;

var gravAccel = 0.3;

var terminalVelocity = 5;

//Current angle of the bird

var Angle = 0;

//Wing flap counter

var WingPosition = 0;

//Sequence of the wing flapping

var WingPositions = [0, 1, 2, 1];

this.Reset = function() {

jumping = false;

gravVeloc = 0;

Angle = 0;

WingPosition = 0;

selectorObject.stop().rotate(0).css('top', '50%');

}

this.fall = function() {

if (!jumping) { //Check if we're jumping

//If we're currently not jumping

selectorObject.stop().animate({

top: '+=' + gravVeloc + '%'

}, 30, 'linear'); //Stop any current animations and then drop the bird by Velocity%

gravVeloc += gravAccel; //Add the acceleration scalar to the velocity scalar.

//Limit the maximum velocity of the bird

if (gravVeloc > terminalVelocity) {

gravVeloc = terminalVelocity;

}

var AdjustedAngle = Angle + 15 \* (gravVeloc / terminalVelocity) ^ 2;

adjustAngle(Math.min(AdjustedAngle, 90));

$('#DebugInfo').html('Gravity: ' + gravVeloc);

} else {

gravVeloc = 0; //Reset the falling velocity.

//console.log('Grav Disabled');

}

};

this.jump = function() {

if (Paused) { //If we're paused, don't let our bird jump

return;

}

jumping = true; //Set that we're jumping right now

adjustAngle(-45);

selectorObject.stop().animate({

top: '-=10%'

}, 100, 'linear', function() { //stop any current animations and "Jump" up 9% in 100ms linearly.

jumping = false; //Set our jumping flag to false right after our jump

Birdy.fall(); //Run the fall immediately

});

};

this.flapWings = function() {

WingPosition++; //Increment wing flapping counter

if (Angle > 45) { //If the bird is falling, put the wings back to center

WingPosition = 1;

}

selectorObject.css("background-position-x", WingPositions[WingPosition % 4] \* 50 + "%"); //Move the backgroud position of the bird to animate the flapping

}

function adjustAngle(angle) {

selectorObject.rotate(angle); //Rotate the bid

Angle = angle; //Set the current angle of the bird so we can read it

}

this.BirdyObject = selectorObject;

});

//Generate a new pipe

function addPipe() {

var PipeGap = 30, //Gap between pipes in %

MinPipeHeight = 5; //Minimum pipe height, so the pipes are still okay sized.

var MaxTopPipeHeight = 100 - PipeGap - 2 \* MinPipeHeight; //Max height the top pipe can be

var TopPipeHeight = Math.random() \* MaxTopPipeHeight + MinPipeHeight; //Actual top pipe height, which is random.

var BottomPipeTop = TopPipeHeight + PipeGap; //This calculates where the bottom pipe would be in position

var BottomPipeHeight = 100 - BottomPipeTop; //This calculates how tall the bottom pipe should be

//Let's create the top pipe, give it the correct height, and add it to the canvas. Mark them as non-scored.

$('<div/>').addClass('Pipe').css('height', TopPipeHeight + '%').data('scored', false).appendTo(canvasObject);

//And now the bottom pipe, but this time we need to tell how far from the top it is, and how tall it is as well. Mark them as non-scored.

$('<div/>').addClass('Pipe BottomPipe').css({

height: BottomPipeHeight + '%',

top: BottomPipeTop + '%'

}).data('scored', false).appendTo(canvasObject);

}

//Delete all the pipes that are already off screen

function cleanPipes() {

$('.Pipe').each(function() {

//If the position percentage is less than -20%, it's off the screen

if ($(this).offset().left / $(this).parent().width() < -0.2) {

$(this).remove();

}

});

}

function checkCollisions() {

$('.Pipe').each(function() {

if (isIntersecting(Birdy.BirdyObject, $(this))) {

console.log('Hit!');

endGame();

}

});

}

function isIntersecting(obj1, obj2) {

//Get the coordinates of the left, top, right, and bottom of the bird and pipe

var obj1Dimensions = [obj1.offset().left, obj1.offset().top, obj1.offset().left + obj1.width(), obj1.offset().top + obj1.height()];

var obj2Dimensions = [obj2.offset().left, obj2.offset().top, obj2.offset().left + obj2.width(), obj2.offset().top + obj2.height()];

/\*

\* It's easier to prove that we aren't intersecting, so we'll

\* prove that our bird isn't touching any pipes, and then

\* we'll negate the result to find out if it touching the pipe.

\*/

return !(obj1Dimensions[3] < obj2Dimensions[1] //If our bird's bottom edge is above the pipe's top edge OR

||

obj1Dimensions[1] > obj2Dimensions[3] //If our bird's top edge is below the pipe's bottom edge OR

||

obj1Dimensions[0] > obj2Dimensions[2] //If our bird's left edge is right of the pipe's right edge OR

||

obj1Dimensions[2] < obj2Dimensions[0]); //If our bird's right edge is left of the pipe's left edge OR

}

function isInBound(birdy, canvas) {

//We're out of bounds if the bottom of the bird is below the bottom of the canvas

//OR if the bird's top is above the canvas top

if (birdy.offset().top + birdy.height() > canvas.offset().top + canvas.height() || birdy.offset().top < canvas.offset().top) {

console.log('Out of Bounds!'); //Print out that we're out of bounds

endGame();

}

}

//Increase the score as the birdy passes a pipe

function incrementScore() {

$('.BottomPipe').each(function() {

var BirdyBeakXPos = Birdy.BirdyObject.offset().left + Birdy.BirdyObject.width(); //Calculate the X coordinate of the bird's beak.

var PipeRightXPos = $(this).offset().left + $(this).width(); //Calculate the X coordinate of the pipe's right edge.

if (!$(this).data('scored') && BirdyBeakXPos > PipeRightXPos) { //If we're ahead of the pipe and we haven't scored on it yet, add to the score.

CurrentScore++; //Increment score

console.log(CurrentScore); //Write to console the score we currently have for debugging purposes

$(this).data('scored', true); //Mark the pipe as scored, so we won't look at it again

}

});

$('#CurrentScore').html(CurrentScore); //Update current score display

}

//Run the start game which starts the gameLoop function

startGame();

//jQuery function to set the rotation of an element using the CSS3 transform property.

jQuery.fn.rotate = function(degrees) {

return $(this).css({

'-webkit-transform': 'rotate(' + degrees + 'deg)',

'-moz-transform': 'rotate(' + degrees + 'deg)',

'-ms-transform': 'rotate(' + degrees + 'deg)',

'transform': 'rotate(' + degrees + 'deg)'

});

};

function setCookie(cname, cvalue, exdays) {

var d = new Date();

d.setTime(d.getTime() + (exdays \* 24 \* 60 \* 60 \* 1000));

var expires = "expires=" + d.toGMTString();

document.cookie = cname + "=" + cvalue + "; " + expires;

}

function getCookie(cname) {

var name = cname + "=";

var ca = document.cookie.split(';');

for (var i = 0; i < ca.length; i++) {

var c = ca[i].trim();

if (c.indexOf(name) == 0) return c.substring(name.length, c.length);

}

return "";

}

});

</script>

</body>

</html>

**Calc park.vbp**

'SIMPLE CAL

Dim x, y, res, a, b, c As Double

Dim o As Boolean

Dim opr As String

Private Sub num\_Click(Index As Integer)

If o = False Then

Text1.Text = Text1.Text + num(Index).Caption

x = Val(Text1.Text)

Else

Text1.Text = Text1.Text + num(Index).Caption

y = Val(Text1.Text)

End If

End Sub

Private Sub cmdadd\_Click(Index As Integer)

If x Or y Then

o = True

opr = "+"

Text1.Text = " "

Else

o = False

End If

End Sub

Private Sub cmdmin\_Click(Index As Integer)

If x Then

o = True

opr = "-"

Text1.Text = " "

Else

o = False

End If

End Sub

Private Sub cmdmul\_Click(Index As Integer)

If x Then

o = True

opr = "\*"

Text1.Text = " "

Else

o = False

End If

End Sub

Private Sub cmddiv\_Click(Index As Integer)

If x Then

o = True

opr = "/"

Text1.Text = " "

ElseIf Text1.Text = 0 Then

Text1.Text = "Zero Division Error"

o = False

Else

o = False

Text1.Text = " "

End If

End Sub

Private Sub cmdeq\_Click(Index As Integer)

If opr = "+" Then

res = x + y

Text1.Text = res

x = res

o = False

ElseIf opr = "-" Then

res = x - y

Text1.Text = res

x = res

ElseIf opr = "\*" Then

res = x \* y

Text1.Text = res

x = res

ElseIf opr = "/" And y <> 0 Then

res = x / y

Text1.Text = res

x = res

ElseIf opr = "/" And y = 0 Then

Text1.Text = "Zero Division Error"

End If

End Sub

Private Sub Clr\_Click()

Text1.Text = ""

x = 0

y = 0

End Sub

Private Sub Ext\_Click(Index As Integer)

End

End Sub

Private Sub switch\_Click()

If scalc.Visible = True Then

scalc.Visible = False

pcalc.Visible = True

switch.Caption = "SIMPLE CALCULATOR"

ElseIf pcalc.Visible = True Then

scalc.Visible = True

pcalc.Visible = False

switch.Caption = "PERCENTAGE CALCULATOR"

End If

End Sub

'Percentage Calculator

Private Sub Command1\_Click()

a = 0

b = 0

c = 0

If Val(t1.Text) = 0 Then

t10.Caption = "Zero Divide Error or Provide Valid Numbers."

Else

b = Val(t2.Text)

a = Val(t1.Text)

c = a \* b / 100

t3.Caption = c

t10.Caption = t1.Text & " x " & t2.Text & " / 100 = " & c

End If

End Sub

Private Sub Command2\_Click()

a = 0

b = 0

c = 0

b = Val(t5.Text)

If Val(t4.Text) = 0 Then

t10.Caption = "Zero Divide Error or Provide Valid Numbers."

Else

t10.Caption = ""

a = Val(t4.Text) / b

c = a \* 100

t6.Caption = c & "%"

t10.Caption = t4.Text & " / " & t5.Text & " x 100 = " & c

End If

End Sub

Private Sub Command3\_Click()

a = 0

b = 0

c = 0

b = Val(t8.Text)

If Val(t7.Text) = 0 Then

t10.Caption = "Zero Divide Error or Provide Valid Numbers."

Else

t10.Caption = ""

a = Val(t7.Text)

c = (a / b) \* 100

t9.Caption = c

t10.Caption = t7.Text & " / " & t8.Text & " x 100 = " & c

End If

End Sub

**Resume Builder**

**index.tsx**

import type { NextPage } from 'next';

import Head from 'next/head';

import HomeLayout from 'src/modules/home/HomeLayout';

const HomePage: NextPage = () => {

return (

<div>

<Head>

<title>E-Resume: Home</title>

<meta name="description" content="Single Page Resume Builder" />

<link rel="icon" type="image/png" href="/icons/resume-icon.png" />

</Head>

<HomeLayout />

</div>

);

};

export default HomePage;

**HomeLayout.tsx**

import { NavBarActions, StyledButton } from '../builder/nav-bar/atoms';

import { motion, useAnimation } from 'framer-motion';

import { BsGithub } from 'react-icons/bs';

import { Button } from '@mui/material';

import FeatureSection from './components/Feature';

import Image from 'next/image';

import Link from 'next/link';

import Person from './components/Person';

const HomeLayout = () => {

const controls = useAnimation();

const animationEffectsHoverEnter = { scale: 1.05 };

const animationEffectsHoverLeave = { scale: 1 };

const animationEffectsFirstLoad = {

scale: [0.9, 1],

opacity: [0, 1],

};

const transtionEffects = {

type: 'spring',

stiffness: 400,

damping: 17,

};

return (

<motion.div initial={{ opacity: 0 }} animate={{ opacity: [0, 1] }} className="scroll-smooth">

<nav className="sticky top-0 z-20 h-14 w-full bg-resume-800 flex py-2.5 px-4 xl:px-60 items-center shadow-level-8dp">

<Link href="/">

<Image src={'/icons/resume-icon.png'} alt="logo" height="36" width="36" />

</Link>

<div className="flex-auto flex justify-between items-center ml-5">

<NavBarActions>

<Link href="/builder" passHref={true}>

<StyledButton variant="text">Editor</StyledButton>

</Link>

</NavBarActions>

<NavBarActions>

<Link href="#contribute" passHref={true}>

<StyledButton variant="text" className="max-md:hidden">

Contribute

</StyledButton>

</Link>

<Link href="#about-us" passHref={true}>

<StyledButton variant="text">About us</StyledButton>

</Link>

<a

href={'https://github.com/sadanandpai/resume-builder'}

target="\_blank"

rel="noopener noreferrer"

>

<BsGithub className="h-6 w-6" fill="white" />

</a>

</NavBarActions>

</div>

</nav>

<div

style={{

background: 'linear-gradient(180deg, #E7EEFA 50%, #FFFFFF 100%)',

fontFamily: "'Roboto Slab', serif",

}}

>

<div className="mx-6 md:mx-40 xl:mx-60 mb-6">

<motion.div

className="grid grid-cols-12 pt-12 md:pt-24"

initial={{ opacity: 0 }}

animate={animationEffectsFirstLoad}

transition={transtionEffects}

>

<div className="col-span-12 sm:col-span-4">

<motion.img

id="resume-3d"

src="/resume.webp"

alt="resume-3d"

className="w-6/12 sm:w-9/12"

onMouseEnter={() => {

controls.start(animationEffectsHoverEnter, transtionEffects);

}}

onMouseLeave={() => {

controls.start(animationEffectsHoverLeave, transtionEffects);

}}

animate={controls}

/>

</div>

<div className="col-span-12 sm:col-span-8">

<h3 className="text-xl md:text-2xl mb-2 text-resume-400">SIMPLEST WAY TO BUILD A</h3>

<h1 className="text-5xl mb-12 text-resume-800">Professional Resume</h1>

<div className="flex mb-10">

<div className="bg-resume-800 w-1 rounded-lg"></div>

<p className="text-lg ml-3 text-resume-800">

&ldquo;The secret to getting ahead is getting started&rdquo;

<br />

—Mark Twain

</p>

</div>

<Link href="/builder" passHref={true}>

<Button variant="contained" className="bg-resume-800 mb-2">

BUILD YOUR RESUME

</Button>

</Link>

<p

className="xl:invisible text-resume-800"

style={{ fontFamily: "'Roboto Slab', serif" }}

>

Desktop screen recommended

</p>

</div>

</motion.div>

</div>

</div>

<motion.div

className="mx-6 md:mx-40 xl:mx-60 my-32 w-75"

style={{ fontFamily: "'Roboto Slab', serif" }}

initial={{ opacity: 0 }}

animate={animationEffectsFirstLoad}

transition={transtionEffects}

>

<div className="grid grid-cols-1 lg:grid-cols-2 gap-6">

<FeatureSection />

</div>

</motion.div>

<div className="bg-resume-50 my-32">

<div

id="contribute"

className="mx-6 md:mx-40 xl:mx-60 py-12"

style={{ fontFamily: "'Roboto Slab', serif" }}

>

<div className="grid grid-cols-12 items-center text-center">

<div className="col-span-12 lg:col-span-4 mb-4 lg:mb-0 flex flex-col items-center gap-2">

<Image src={'/icons/palette.svg'} alt="logo" height="48" width="48" />

<p className="text-resume-800 text-xl mt-2">

Do you want to make your own <strong>template?</strong>

</p>

</div>

<div className="col-span-12 lg:col-span-1 mb-4 lg:mb-0 text-resume-800 text-4xl">

<p>+</p>

</div>

<div className="col-span-12 lg:col-span-2 flex flex-col items-center gap-2">

<Image src={'/icons/terminal.svg'} alt="logo" height="48" width="48" />

<p className="text-resume-800 text-xl mt-2">

Do you write <strong>React</strong> code?

</p>

</div>

<div className="invisible lg:visible lg:col-span-2 text-resume-800 text-4xl mx-6">

<p>=</p>

</div>

<div className="col-span-12 lg:col-span-3 mx-auto flex flex-col items-center gap-2">

<div className="mb-6">

<Image src={'/icons/wave.svg'} alt="logo" height="48" width="48" />

</div>

<div>

<a

href="https://github.com/sadanandpai/resume-builder"

target="\_blank"

rel="noreferrer"

>

<Button variant="contained" className="bg-resume-800 mt-2 lg:mt-5 mb-3">

CONTRIBUTE

</Button>

</a>

</div>

</div>

</div>

</div>

</div>

<div

id="about-us"

className="mx-6 md:mx-40 xl:mx-60 my-32"

style={{ fontFamily: "'Roboto Slab', serif" }}

>

<h2 className="text-resume-800 text-3xl mb-2 text-center lg:text-left">About us</h2>

<p className="text-resume-400 mb-8 text-center lg:text-left">

A bunch of developers and designers — who love to build open source projects and learn

along!

</p>

<div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-8">

<Person />

</div>

<p className="text-resume-400 my-8 text-center lg:text-left">

Read our design story on&nbsp;

<a

href="https://medium.com/@yakshag/e-resume-build-a-professional-resume-design-case-study-3dc02a6359ea"

target="\_blank"

rel="noreferrer"

className="underline"

>

Medium

</a>

↗

</p>

</div>

</motion.div>

);

};

export default HomeLayout;

**App.js**

const express = require('express');

const pdfToImage = require('pdftoimage');

const fs = require('fs');

const app = express();

app.use(express.static('public'));

const bodyParser = require('body-parser');

app.use(bodyParser.urlencoded({ extended: true }));

app.use(bodyParser.json());

app.get('/', (req, res) => {

fs.readFile('index.html', (err, data) => {

if (err) {

console.error(err);

return res.status(500).send('Error reading file.');

}

res.setHeader('Content-Type', 'text/html');

res.send(data);

});

});

app.get('/pdftojpg', (req, res) => {

fs.readFile('pdf to jpg.html', (err, data) => {

if (err) {

console.error(err);

return res.status(500).send('Error reading file.');

}

res.setHeader('Content-Type', 'text/html');

res.send(data);

});

});

app.post('/convert', (req, res) => {

console.log(req); // log the req object

const file = req.files.pdf;

if (!req.files || !req.files.pdf) {

return res.status(400).send('PDF file not found.');

}

pdfToImage.convert(file.data, {

density: 150,

quality: 90,

outputFormat: 'jpg',

outputType: 'base64'

}).then(images => {

res.send(`

<html>

<head>

<title>PDF to JPG Converter</title>

</head>

<body>

<h1>PDF to JPG Converter</h1>

<p>Here is your converted image:</p>

<img src="data:image/jpeg;base64,${images[0].data}" />

</body>

</html>

`);

}).catch(error => {

console.error(error);

res.send(`

<html>

<head>

<title>PDF to JPG Converter</title>

</head>

<body>

<h1>PDF to JPG Converter</h1>

<p>An error occurred: ${error.message}</p>

</body>

</html>

`);

});

});

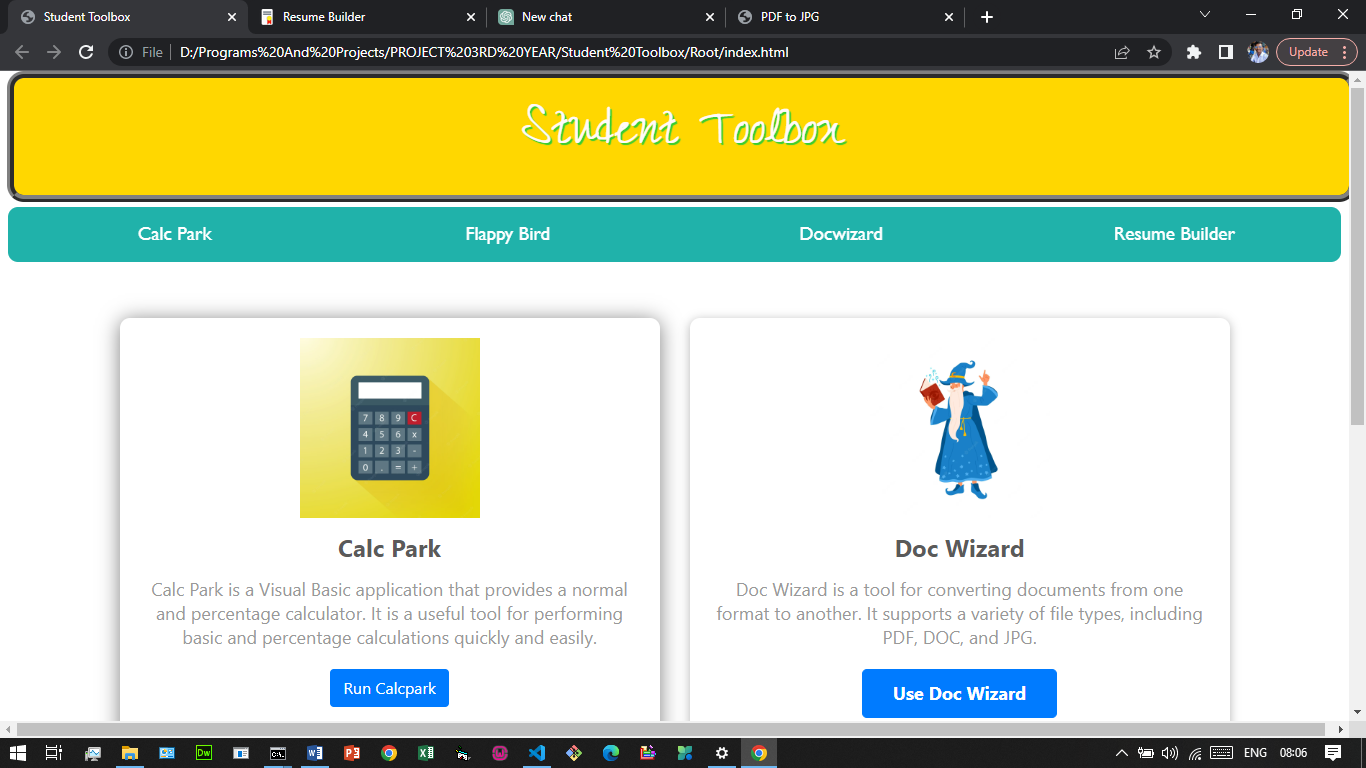
const port = 4000;

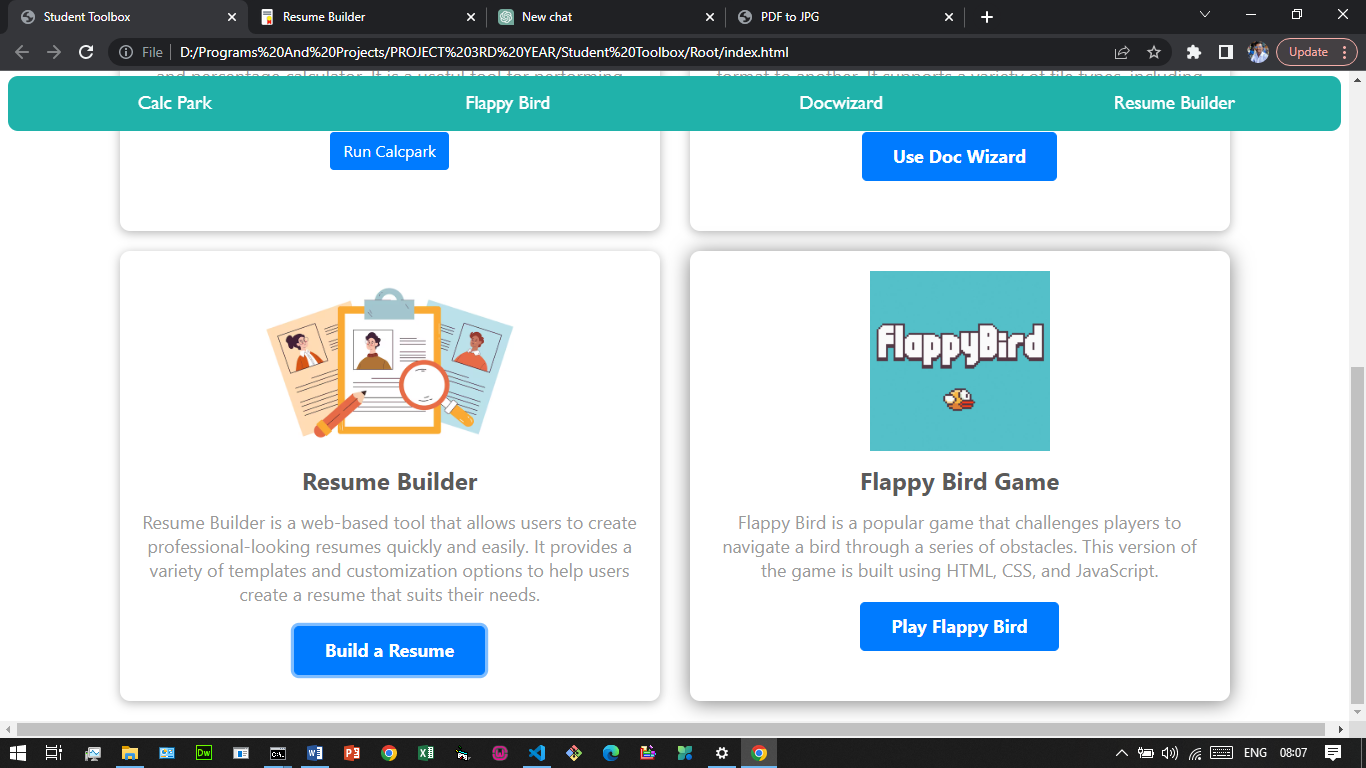
app.listen(port, () => {

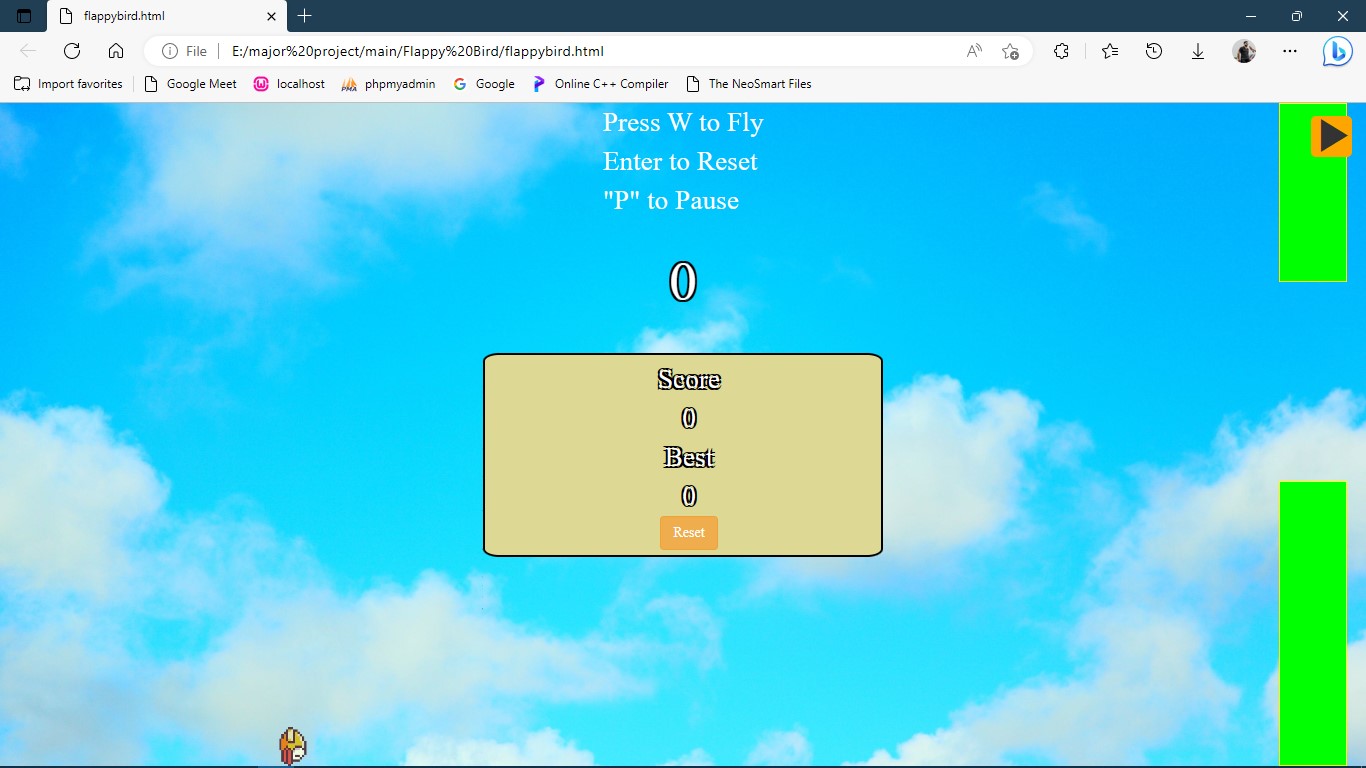
console.log(`Server listening on http://localhost:${port}`);

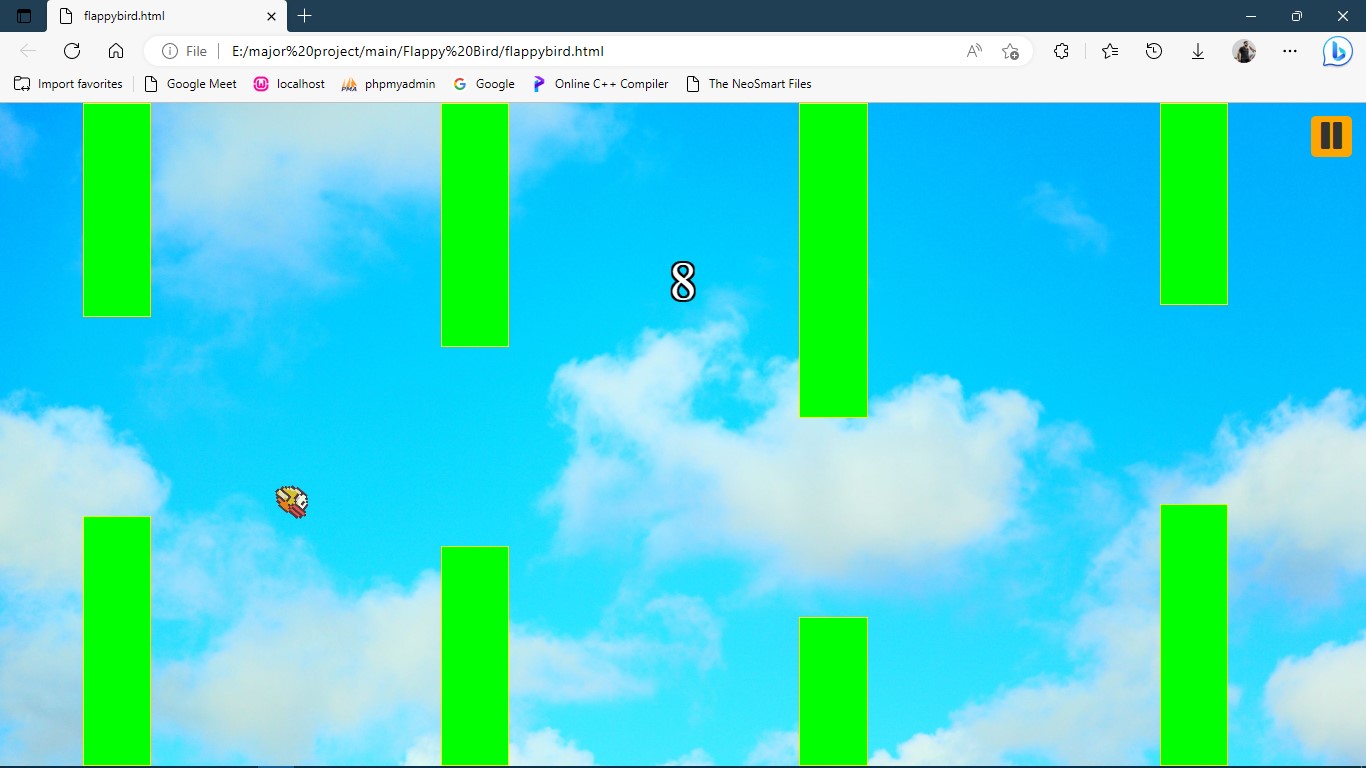
});

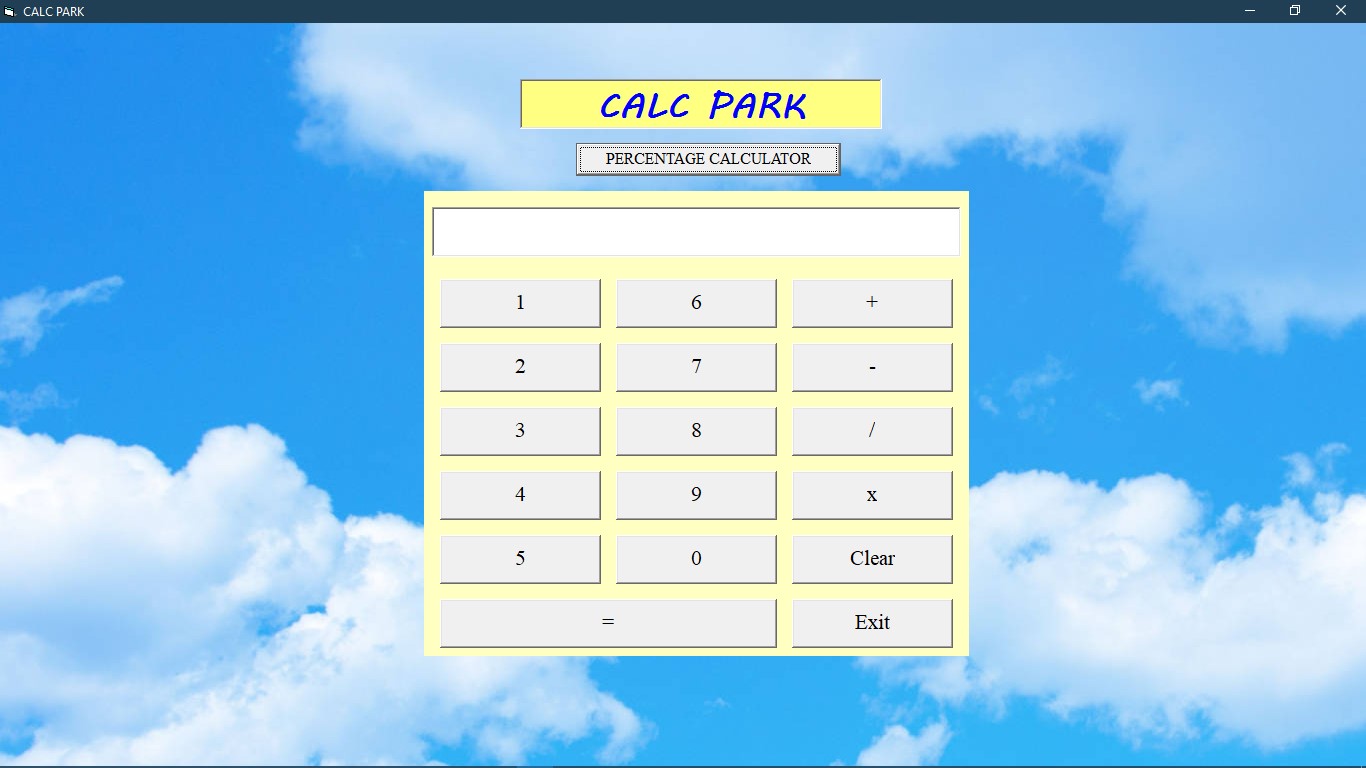
**6.OUTPUTS**



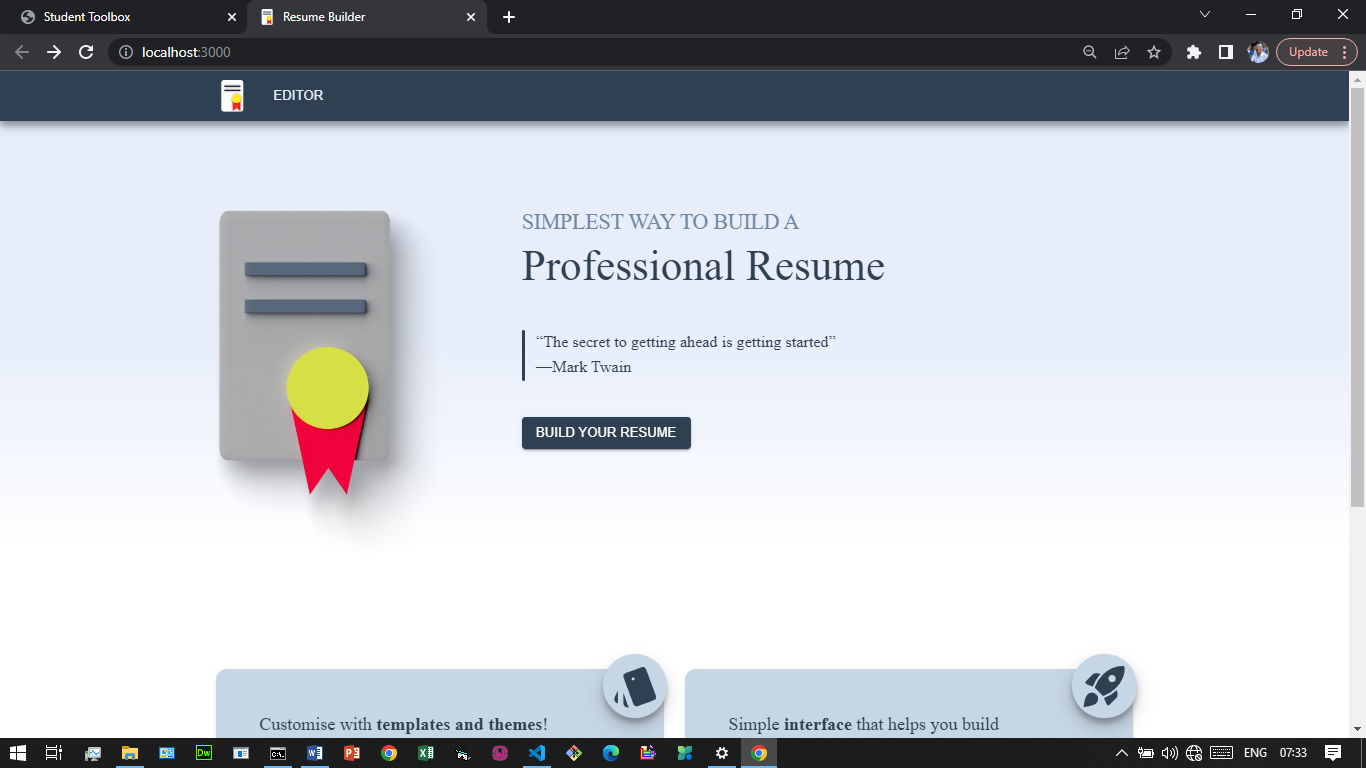


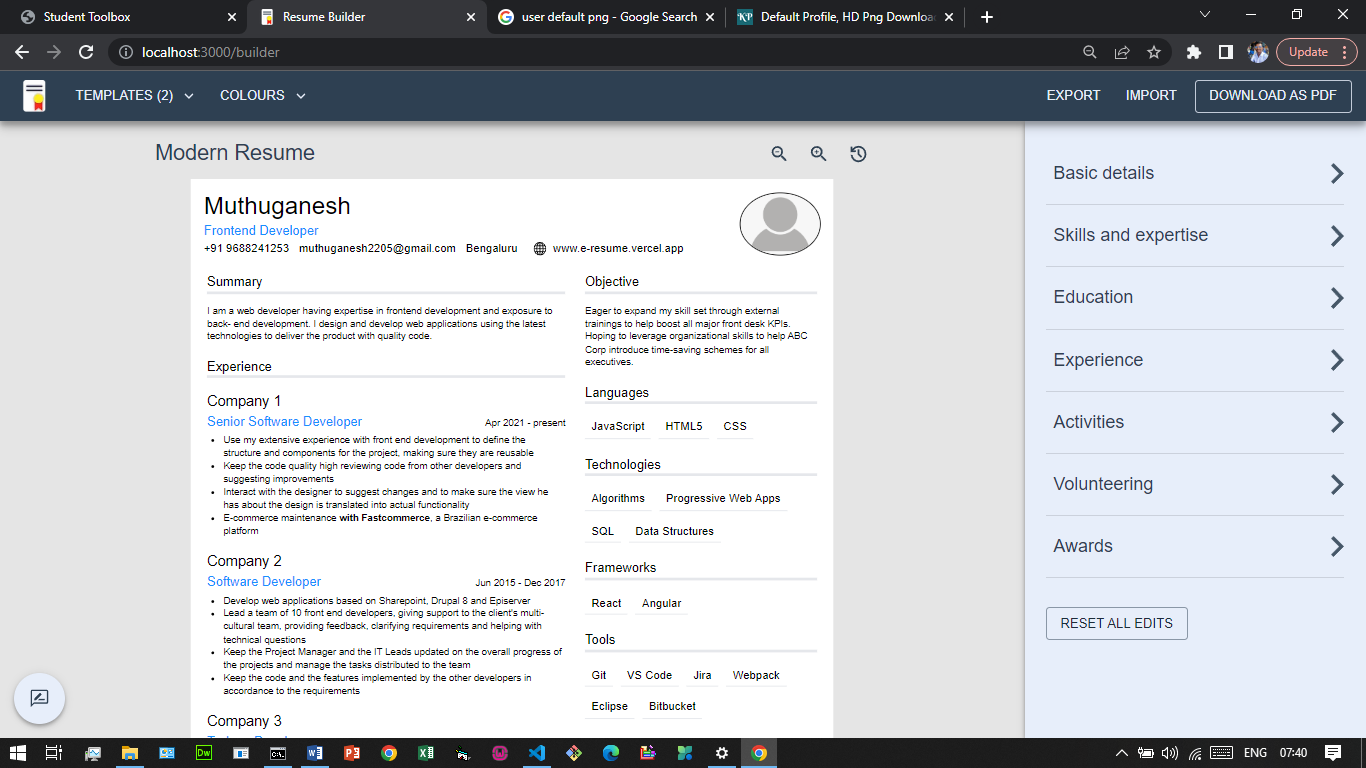
****

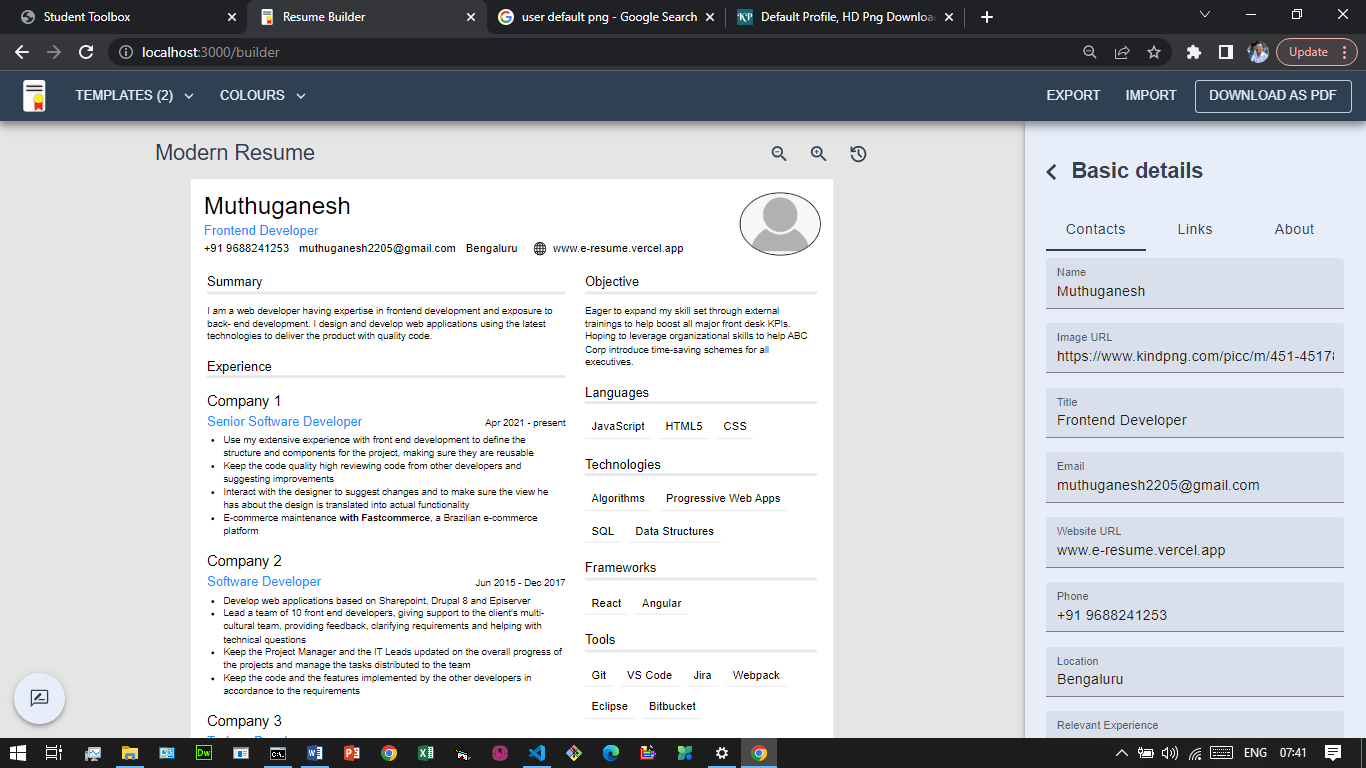
****

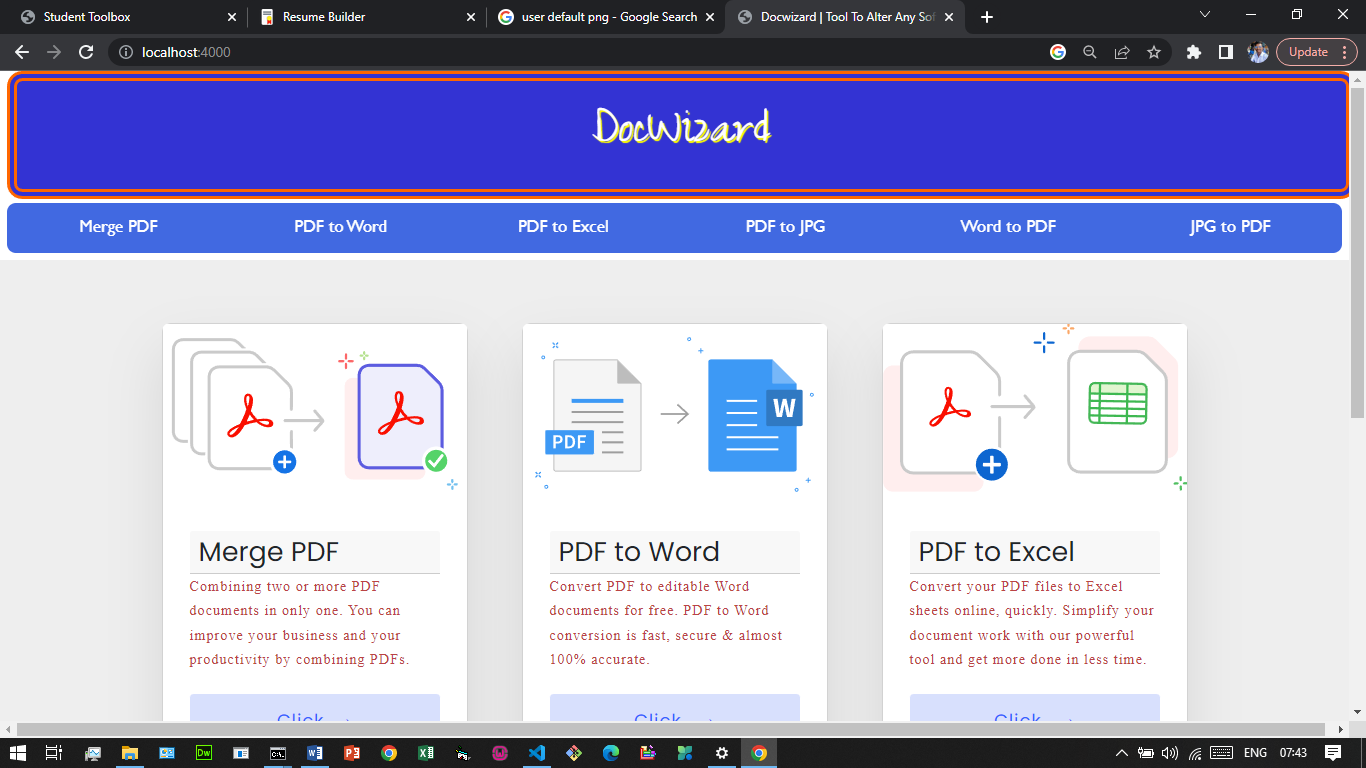
****

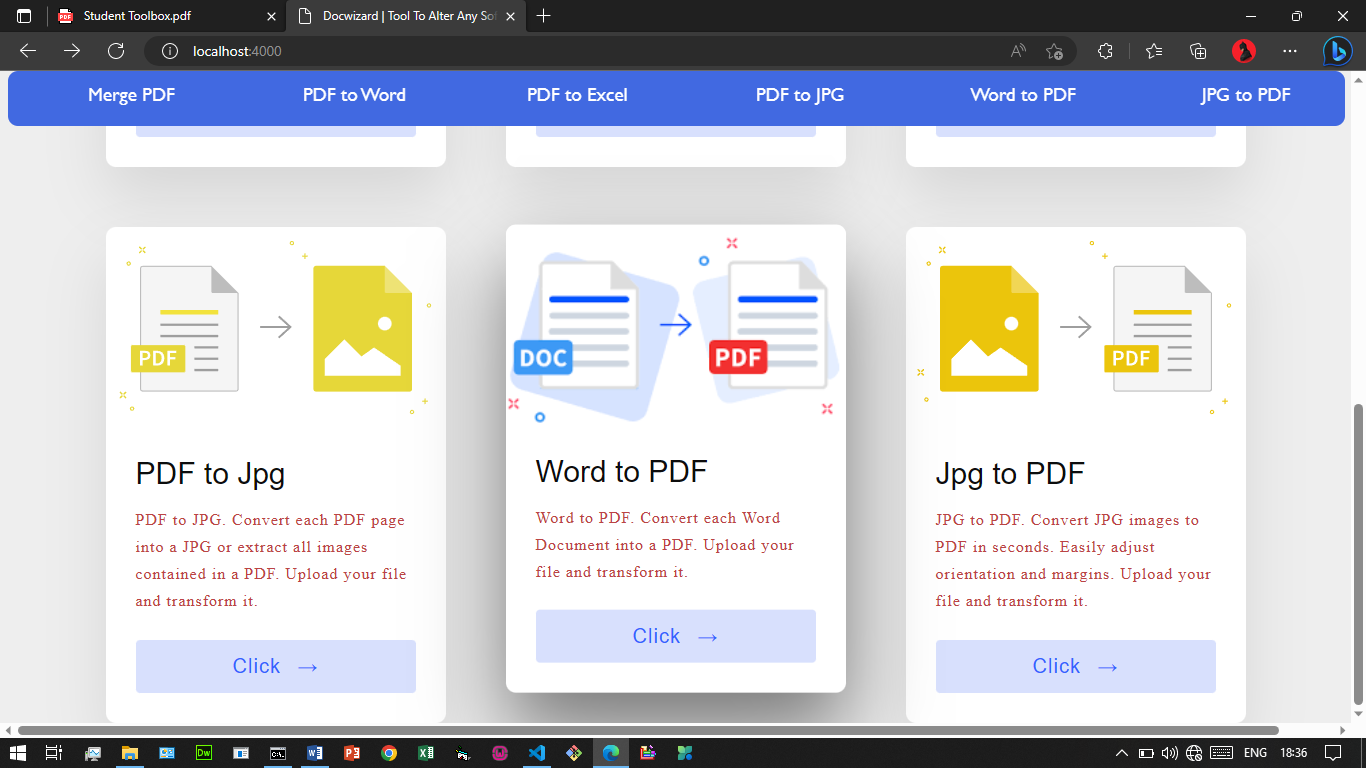
****

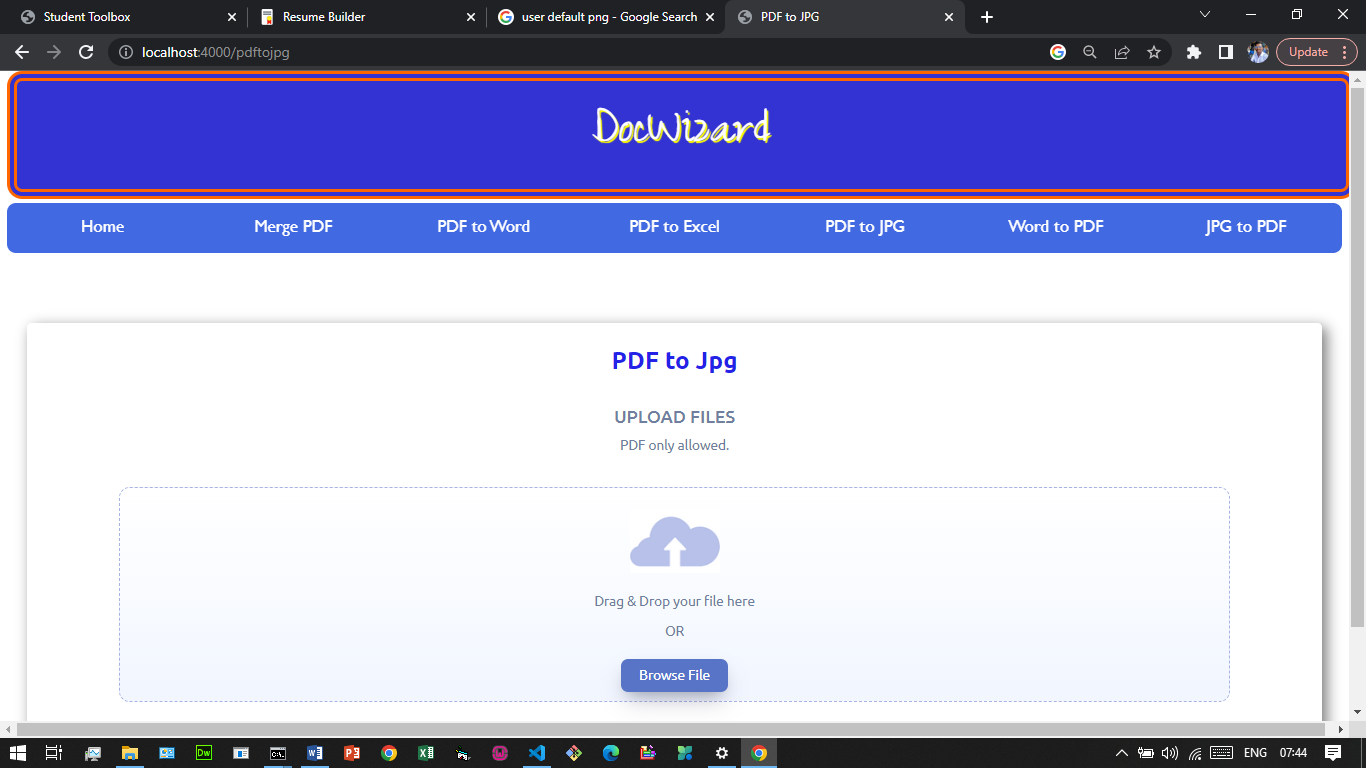












**7. FUTURE ENHANCEMENT**

* Allow users to save their resumes, documents, and game progress to cloud storage services like Dropbox or Google Drive. This will enable users to

access their files from any device with an internet connection.

* Consider adding support for different languages to make the tool accessible

to users who speak languages other than English.

* Allow users to share their resumes or high scores on social media platforms

like LinkedIn or Twitter. This can help students to showcase their skills and

achievements to potential employers.

* Add more games and calculators to the toolbox, such as a scientific

calculator, a graphing calculator, or a puzzle game.

* Continuously improve the user interface of the toolbox to make it more

intuitive and user-friendly.

* Consider creating a mobile app version of the toolbox that can be installed

on smartphones and tablets.

* Implement a feature that uses machine learning algorithms to suggest

improvements for users' resumes based on industry-specific keywords and

job requirements

**8. REFERENCES**

* https://www.tutorialspoint.com/index.htm
* https://www.javatpoint.com
* https://www.w3schools.com
* https://html.com
* https://nodejs.org/
* https://react.dev/
* https://github.com