JavaScript Applications Website

A Project Report Submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

Ayush Gupta
(181500172)
Kshitij Gupta
(181500337)

Under the Guidance of

Mr. Pankaj Kapoor

Department of Computer Engineering and Applications

Institute of Engineering and Technology

GLA University
Mathura- 281406, India
Dec, 2020

GLA UNIVERSITY MATHURA

Department of computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

Declaration

We hereby declare that the work which is being presented in the FullStack Project “**JavaScript Applications Website**”, in fulfillment of the requirements for FullStack project in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of our own work carried under the supervision of **Mr. Pankaj Kapoor (Technical Trainer).**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Kshitij

Sign ________
Name of Candidate: Kshitij Gupta
University Roll No.: 181500337

Sign _____________
Name of Candidate: Ayush Gupta
University Roll No.:181500172

GLA UNIVERSITY MATHURA

Department of computer Engineering and Applications

GLA University, Mathura

17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

Certificate

This is to certify that the above statements made by the candidate are correct to the best of my/our knowledge and belief

Supervisor
Mr. Pankaj Kapoor
Technical Trainer

Date :7 Dec, 2020

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech FullStack Project undertaken during B. Tech. Third Year. This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.

Our heartiest thanks to Dr. (Prof). Anand Singh Jalal, Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.

We owe a special debt of gratitude to Mr. Pankaj Kapoor, Technical Trainer, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.

We also do not like to miss the opportunity to acknowledge the contribution of all instructors who are available on YouTube and Stackoverflow. I would like to thank all my friends who helped me in making this project.

Last but not the least, I would like to express our deep sense of gratitude and earnest thanks giving to our dear parents for their moral support and heartfelt cooperation during the project.

ABSTRACT

Our website is pertaining to javascript applications where users can learn while playing. Basically in this project we are promoting the idea of fun with learning. This website is especially for those users who face the same problem when they want to relax their mind but don't want to waste their time on any game which has no use. So, to overcome this problem we have built a website where users can improve their skills with fun and also manage their work.

We have built an interactive user interface with the help of HTML, CSS, JavaScript and Bootstrap which illustrates the things in an efficient way so that it is easy to use and understand. In this project users can find applications like ToDo List, Typing Speed Game and last but not least Simon game.

With the help of ToDo List, the user can write their daily tasks in the list and after completion of each task they can simply remove it. With the help of Typing Speed Game users will be able to relax their mind as well as improve their typing skill. Now with Simon Game which is a memory skill game invented by Ralph H. Baer and Howard J.Morrison. This game is a little bit challenging and fun also and we hope that users who love brain games will love that also.

Currently in this website we have developed a front-end part which is using HTML, CSS, Bootstrap, JavaScript and jQuery.

CONTENTS

1. Introduction
1.1 Overview and Motivation

We are going to build a website pertaining to some of the JavaScript Applications. Our website will be built using web languages and frameworks. It will provide an interface to the users to interact with some cool JavaScript applications through which they learn while playing games.

The applications which we are going to make on this website are a TODO List, Simon Game, and Typing Speed Game. These applications not only relax the users mind but also boost their skills. We choose these applications to show most of the functionality of JavaScript to the user.

With the help of ToDo List, the user can save their daily tasks in the list and after completing each task they can simply remove it. So, this type of application is very useful for those users who make a task list before doing something.

With the help of Typing Speed Game users will be able to relax their mind as well as improve their typing skill. This is a fun activity for those users who want to check their typing speed and also want to play games.

Our Last Application but not the least is Simon Game which is a memory skill game invented by Ralph H. Baer and Howard J.Morrison. This game is a little bit challenging and fun also and we hope that users who love brain games will love that also.

We are using HTML-5 to provide structure to our website, CSS-3 for styling our web pages, including colours, style, font, layout and many more., Bootstrap as a framework for the ease of designing and JavaScript/JQuery (which is a JavaScript Library) to provide functionality to our website.

Problem Statement:

There are many websites on the internet which provide only games, but our idea is to create a website through which users can learn while playing games. So, basically we are promoting the idea of learning with fun. Considering that, we are building all our applications as a game through which a user can learn while playing.

Objective:

The aim of our project is to target thousands of users to use and engage with our website. We have decided to work on this project because most of the people face the same problem when they want to relax their mind but don't want to waste their time on any game which has no use. So, to overcome this problem we will build a website where users can improve their skills with fun and also manage their work.

We will build an interactive user interface with the help of HTML, CSS, JavaScript and Bootstrap which illustrates the things in an efficient way so that it is easy to use and understand.

We are making applications like a Typing Speed game so that users can relax their mind with improving their typing skills. Our next application will be ToDo List; with the help of this application users can manage their work and life in a very easy and efficient manner. Our last application will be Simon Game, which will help users to increase their memorizing power.

2.Software and Requirement Analysis
2.1 Software Requirement

Visual Studio:

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug- ins that enhance the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new tool sets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++, C++/CLI, Visual Basic

.NET, C#, F#,JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python,Ruby, Node.js, and M among others is available via plug-ins. Java (and J#) were supported in the past.

Web Browser:

A **web browser** (commonly referred to as a **browser**) is a software application for accessing information on the World Wide Web. Each individual web page, image, and video is identified by a distinct Uniform Resource Locator (URL), enabling browsers to retrieve these resources from a web server and display them on the user's device.

A web browser is not the same thing as a search engine, though the two are often confused. For a user, a search engine is just a website, such as google.com, that stores searchable data about other websites. But to connect to a website's server and display its web pages, a user needs to have a web browser installed on their device.

The most popular browsers are Chrome, Firefox, Safari, Internet Explorer, and Edge.

2.2 Language and Framework Requirements

HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

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), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997. HTML code ensures the proper formatting of text and images so that your Internet browser may display them as they are intended to look. Without HTML, a browser would not know how to display text as elements or load images or other elements. HTML also provides a basic structure of the page, upon which Cascading Style Sheets are overlaid to change its appearance. One could think of HTML as the bones (structure) of a web page, and CSS as its skin (appearance).

CSS (Cascading Style Sheets)

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Bootstrap

Bootstrap is a free and open front-end framework for designing websites and web applications. It contains HTML - and CSS -based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many earlier web frameworks, it concerns itself with front end development only.

Bootstrap is the second most-starred project on GitHub, with more than 129,000 stars. Bootstrap comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields. In version 1.3, the following JavaScript plugins are supported: Modal, Dropdown, Scrollspy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel and Typeahead.

Java Script (JS)

JavaScript, often abbreviated as **JS**, is a high-level, interpreted programming language that conforms to the ECMAScript specification. It is a programming language that is characterized as dynamic, weakly typed, prototype-based and multi-paradigm.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

The terms *Vanilla JavaScript* and *Vanilla JS* refer to JavaScript not extended by any frameworks or additional libraries. Scripts written in Vanilla JS are plain JavaScript code.

Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design. JavaScript was influenced by programming languages such as Self and Scheme.

jQuery

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax.It is free, open-source software using the permissive MIT License. As of May 2019, jQuery is used by 73% of the 10 million most popular websites.[5] Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having at least 3 to 4 times more usage than any other JavaScript library.

jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, themeable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications.

The set of jQuery core features—DOM element selections, traversal and manipulation—enabled by its *selector engine* (named "Sizzle" from v1.3), created a new "programming style", fusing algorithms and DOM data structures. This style influenced the architecture of other JavaScript frameworks like YUI v3 and Dojo, later stimulating the creation of the standard *Selectors API.* Later, this style has been enhanced with a deeper algorithm-data fusion in an heir of jQuery, the D3.js framework.

Microsoft and Nokia bundle jQuery on their platforms. Microsoft includes it with Visual Studio for use within Microsoft's ASP.NET AJAX and ASP.NET MVC frameworks while Nokia has integrated it into the Web Run-Time widget development platform.

2.3 Software And Hardware requirement:

Following are the hardware and the software requirements for our project:

1. Hardware:

- Laptop/Desktop
- 1.8 GHz or faster processor. Quad-core or better recommended
- 4 GB of RAM and core i3 processor
- Hard disk space: Minimum of 500MB

2. Software:

- Windows 8.1 and above
- Visual Studio Code
- Web Browser
- Github Desktop

3. Language and Framework Requirements:

- HTML
- CSS
- Bootstrap
- JavaScript
- jQuery

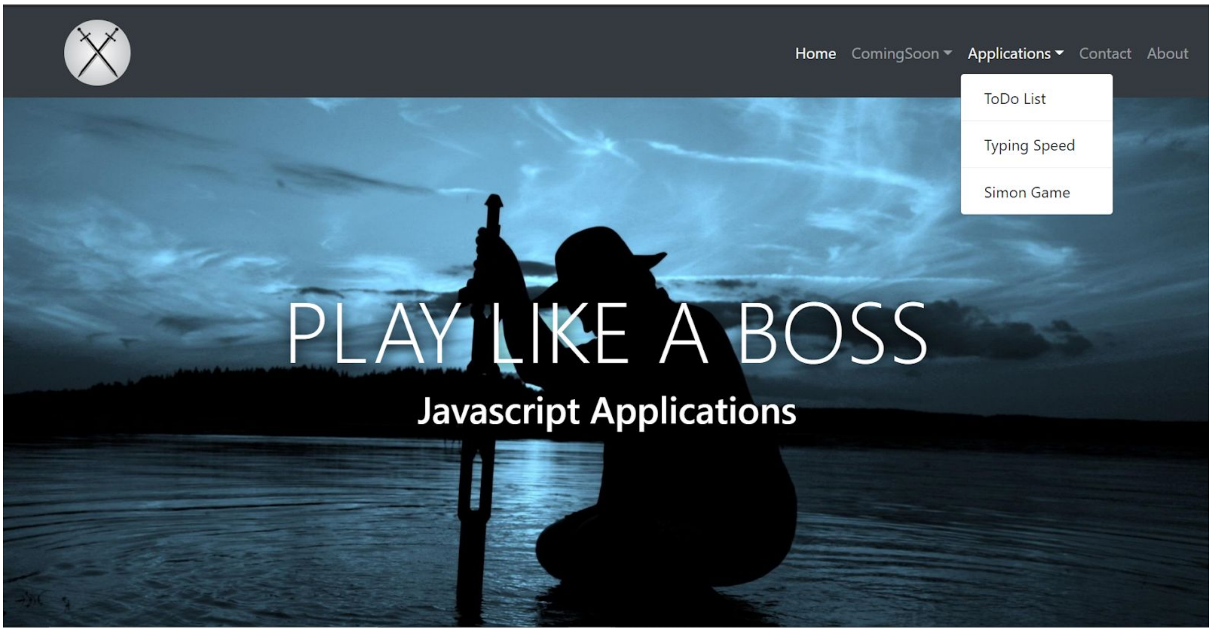
Implementation Details

We have started our project with the HTML to give structure to our website. We have used HTML-5 (Hypertext Markup Language) for the structure purpose. We have made a home page which consists - of a Navigation Bar in the navbar we have provided our website logo and some links and information through which users can navigate to each section of our website then we have a Background Section where we have provided an image that interacts users to our website then we have a section for Todo List through which user can get all information about todo list and navigate to-do list also then we have Fun Section where users can see interactive emojis then we have a section for Typing Speed Game through which user can navigate to the game then we have a section for Simon Game through which user can navigate to the game. After this, we have implemented a footer which consists of contact and about sections.

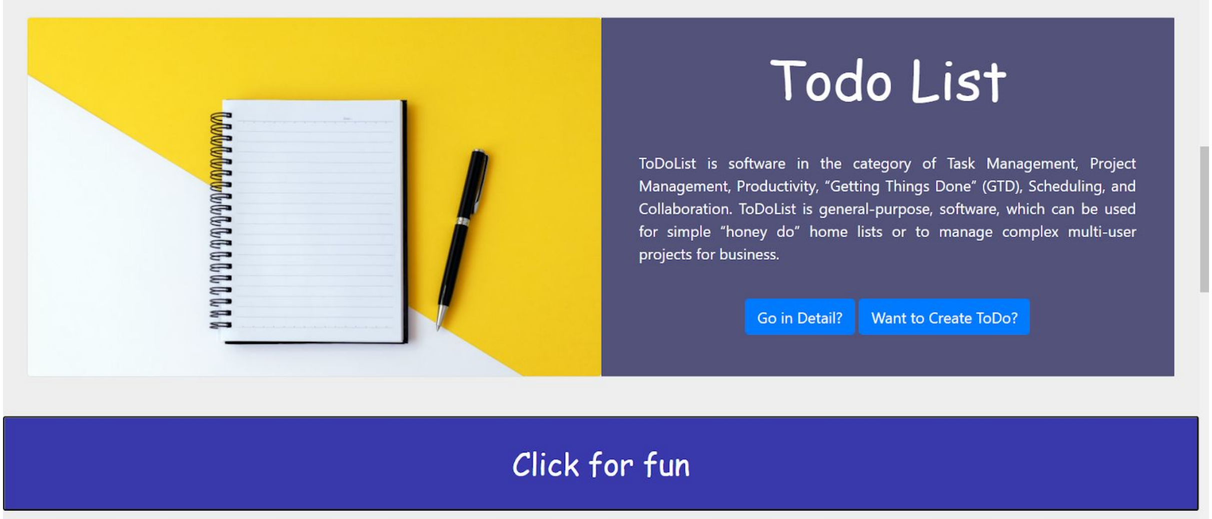
We have used CSS-3 (Cascading Style Sheets) for describing the presentation of Web pages, including colors, layout, and many more. We have also used bootstrap for designing purposes because Bootstrap is a potent front-end framework used to create modern websites and web apps. With the help of bootstrap we can easily make our website that adjusts on mobile devices, tablets and desktops. To provide functionalities to our applications we have used JavaScript and jQuery.

Some Screenshots

HomePage: Navigation Bar

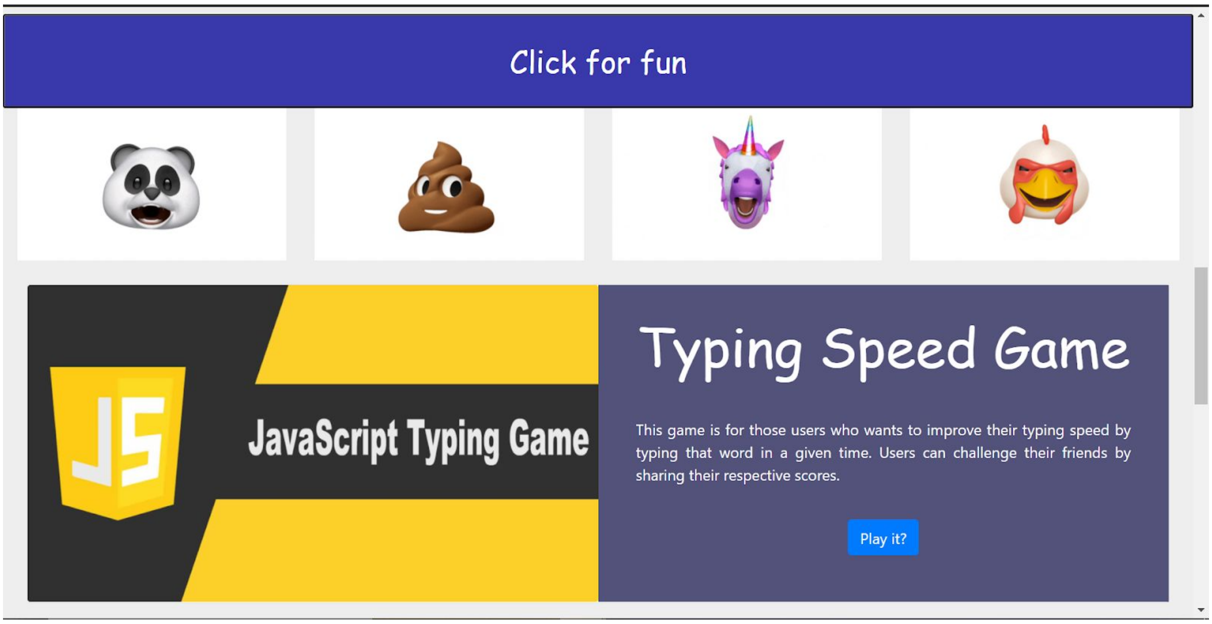


ToDo List Section:


Todo List

ToDoList is software in the category of Task Management, Project Management, Productivity, "Getting Things Done" (GTD), Scheduling, and Collaboration. ToDoList is general-purpose, software, which can be used for simple "honey do" home lists or to manage complex multi-user projects for business.

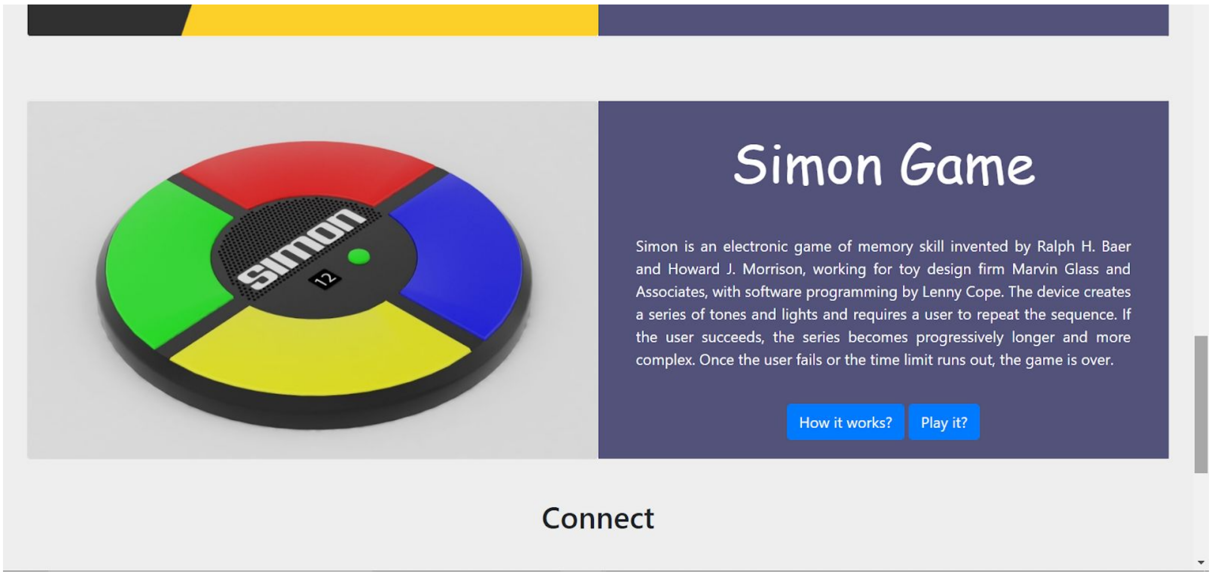
Typing Speed Game Section:



Typing Speed Game

This game is for those users who wants to improve their typing speed by typing that word in a given time. Users can challenge their friends by sharing their respective scores.

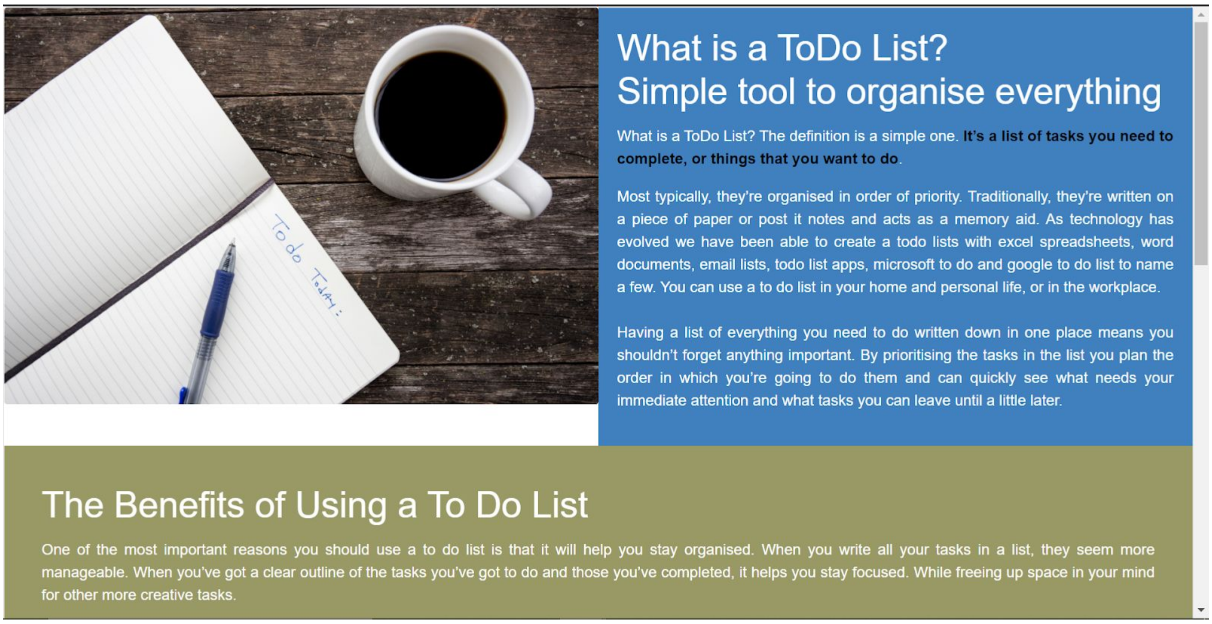
Simon Game Section:



Simon Game

Simon is an electronic game of memory skill invented by Ralph H. Baer and Howard J. Morrison, working for toy design firm Marvin Glass and Associates, with software programming by Lenny Cope. The device creates a series of tones and lights and requires a user to repeat the sequence. If the user succeeds, the series becomes progressively longer and more complex. Once the user fails or the time limit runs out, the game is over.

ToDo Detailed Section:



What is a ToDo List?
Simple tool to organise everything

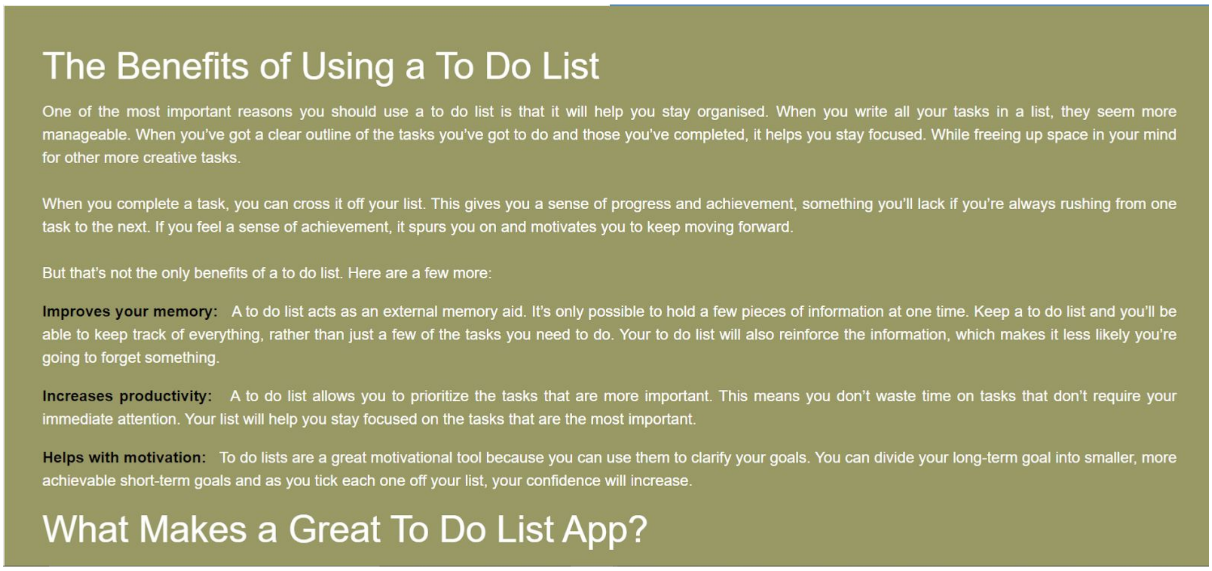
What is a ToDo List? The definition is a simple one. **It's a list of tasks you need to complete, or things that you want to do**.

Most typically, they're organised in order of priority. Traditionally, they're written on a piece of paper or post it notes and acts as a memory aid. As technology has evolved we have been able to create a todo lists with excel spreadsheets, word documents, email lists, todo list apps, microsoft to do and google to do list to name a few. You can use a to do list in your home and personal life, or in the workplace.

Having a list of everything you need to do written down in one place means you shouldn't forget anything important. By prioritising the tasks in the list you plan the order in which you're going to do them and can quickly see what needs your immediate attention and what tasks you can leave until a little later.

The Benefits of Using a To Do List

One of the most important reasons you should use a to do list is that it will help you stay organised. When you write all your tasks in a list, they seem more manageable. When you've got a clear outline of the tasks you've got to do and those you've completed, it helps you stay focused. While freeing up space in your mind for other more creative tasks.



The Benefits of Using a To Do List

One of the most important reasons you should use a to do list is that it will help you stay organised. When you write all your tasks in a list, they seem more manageable. When you've got a clear outline of the tasks you've got to do and those you've completed, it helps you stay focused. While freeing up space in your mind for other more creative tasks.

When you complete a task, you can cross it off your list. This gives you a sense of progress and achievement, something you'll lack if you're always rushing from one task to the next. If you feel a sense of achievement, it spurs you on and motivates you to keep moving forward.

But that's not the only benefits of a to do list. Here are a few more:

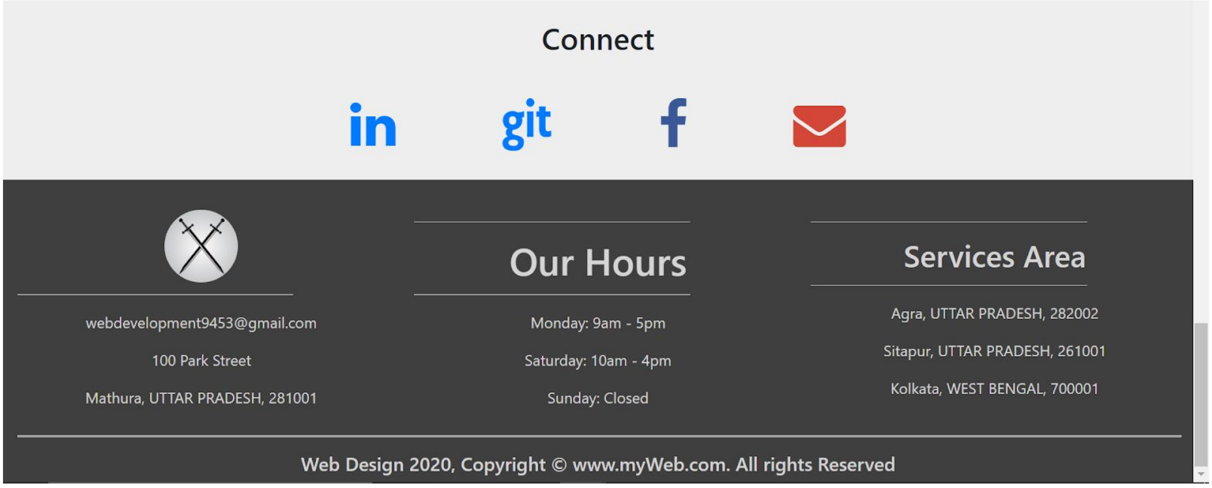
Improves your memory: A to do list acts as an external memory aid. It's only possible to hold a few pieces of information at one time. Keep a to do list and you'll be able to keep track of everything, rather than just a few of the tasks you need to do. Your to do list will also reinforce the information, which makes it less likely you're going to forget something.

Increases productivity: A to do list allows you to prioritize the tasks that are more important. This means you don't waste time on tasks that don't require your immediate attention. Your list will help you stay focused on the tasks that are the most important.

Helps with motivation: To do lists are a great motivational tool because you can use them to clarify your goals. You can divide your long-term goal into smaller, more achievable short-term goals and as you tick each one off your list, your confidence will increase.

What Makes a Great To Do List App?

Connect and Contact Section:



Connect

webdevelopment9453@gmail.com

100 Park Street

Mathura, UTTAR PRADESH, 281001

Our Hours

Monday: 9am - 5pm

Saturday: 10am - 4pm

Sunday: Closed

Services Area

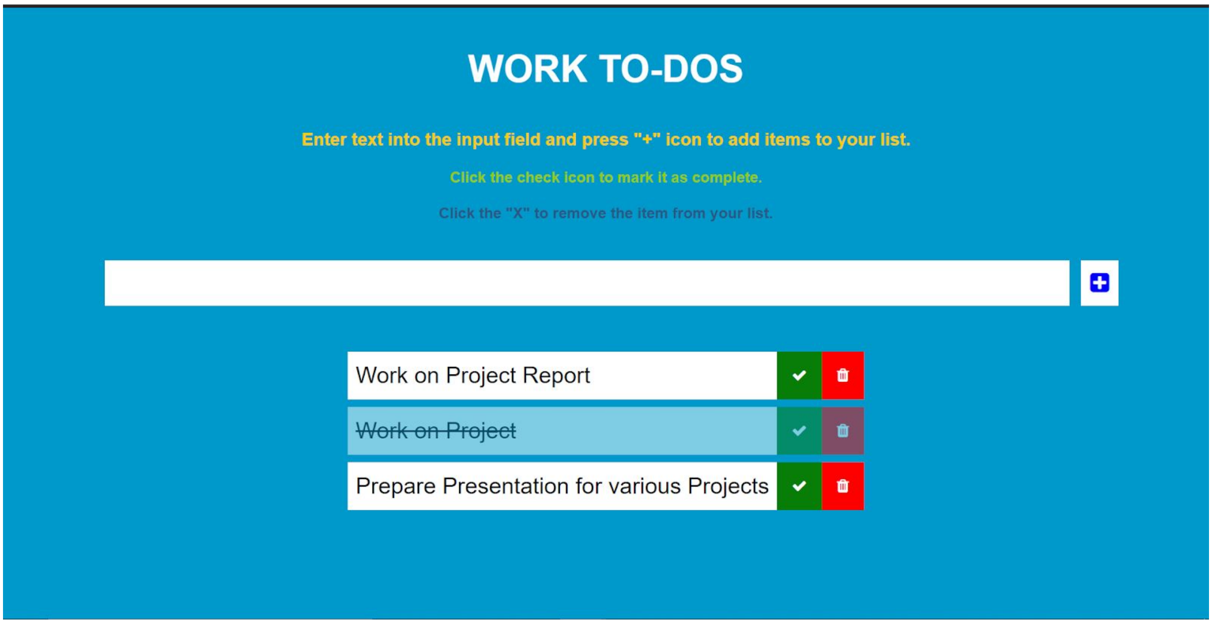
Agra, UTTAR PRADESH, 282002

Sitapur, UTTAR PRADESH, 261001

Kolkata, WEST BENGAL, 700001

Web Design 2020, Copyright © www.myWeb.com. All rights Reserved

ToDo List Page:



WORK TO-DOS

Enter text into the input field and press "+" icon to add items to your list.

Click the check icon to mark it as complete.

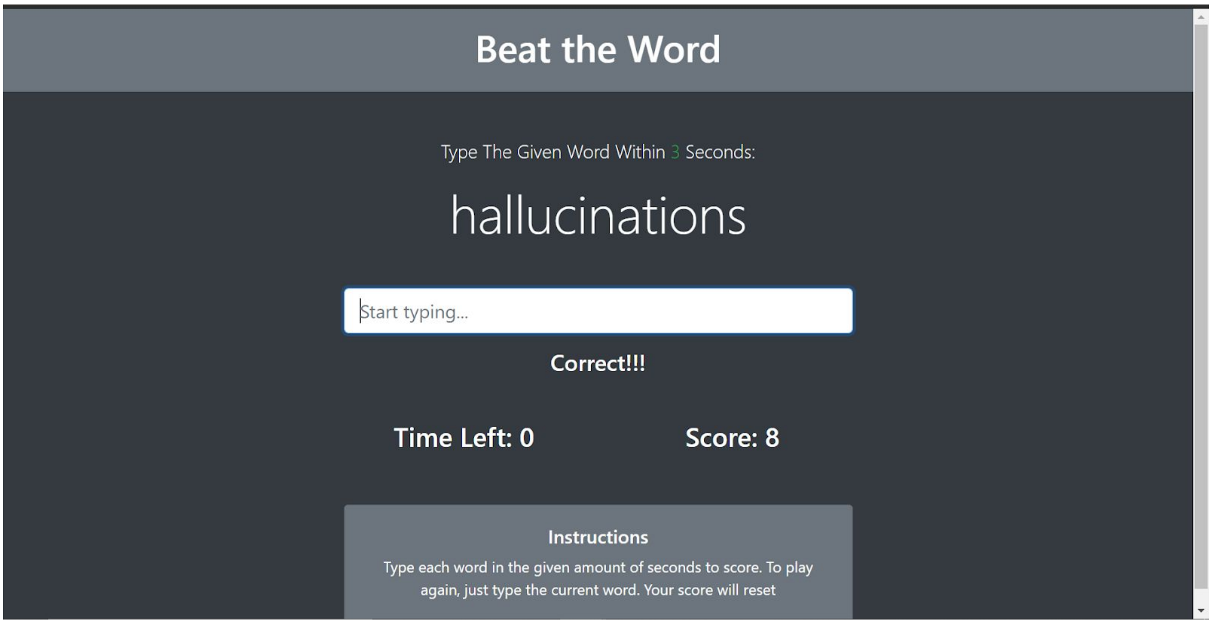
Click the "X" to remove the item from your list.

Work on Project Report

~~Work on Project~~

Prepare Presentation for various Projects

Typing Speed Game:



Beat the Word

Type The Given Word Within 3 Seconds:

hallucinations

Start typing...

Correct!!!

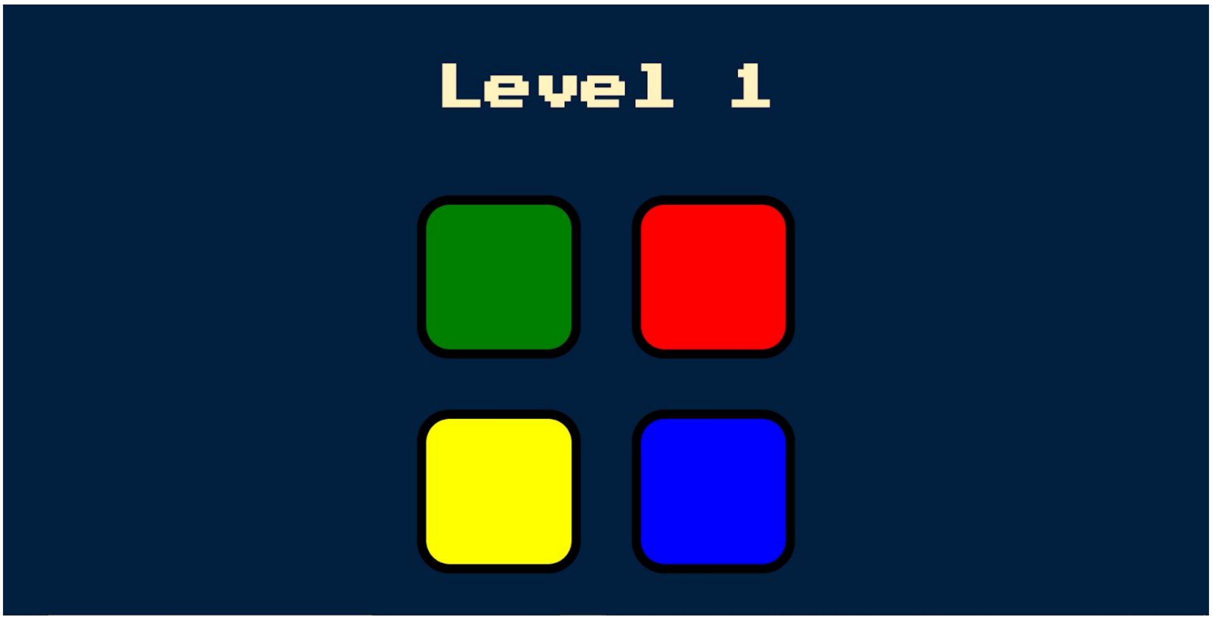
Time Left: 0

Score: 8

Instructions

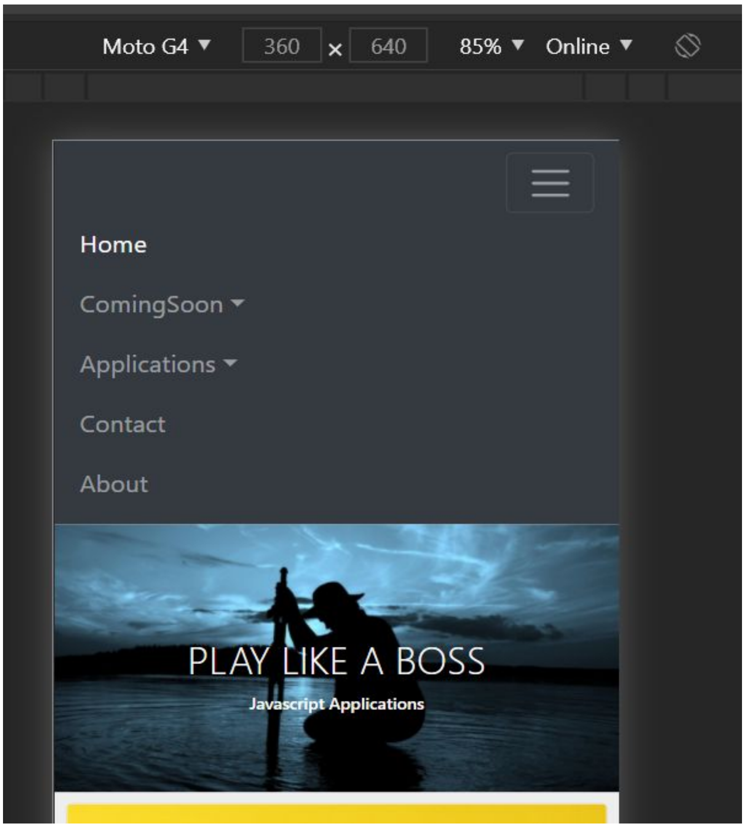
Type each word in the given amount of seconds to score. To play again, just type the current word. Your score will reset

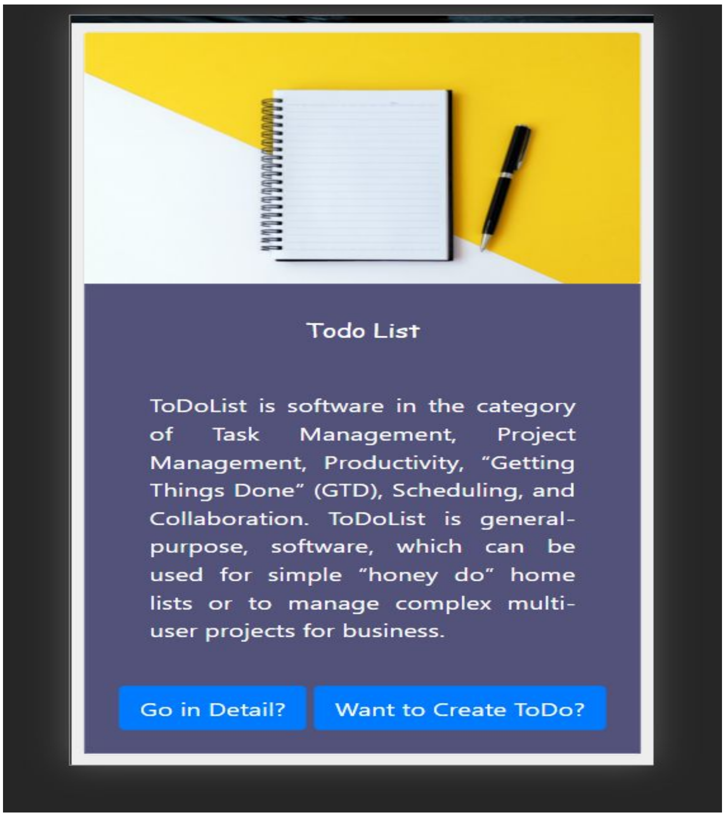
Simon Game:



Level 1

Responsive?

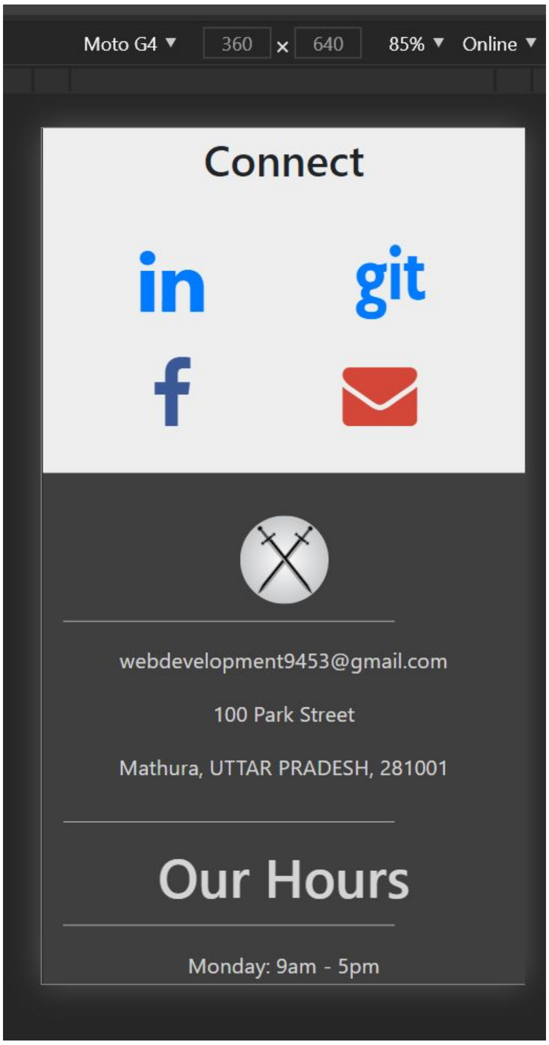




Todo List

ToDoList is software in the category of Task Management, Project Management, Productivity, “Getting Things Done” (GTD), Scheduling, and Collaboration. ToDoList is general-purpose, software, which can be used for simple “honey do” home lists or to manage complex multi-user projects for business.

Connect and Contact Section in mobile screen:



Moto G4 ▼ 360 × 640 85% ▼ Online ▼

Connect

webdevelopment9453@gmail.com

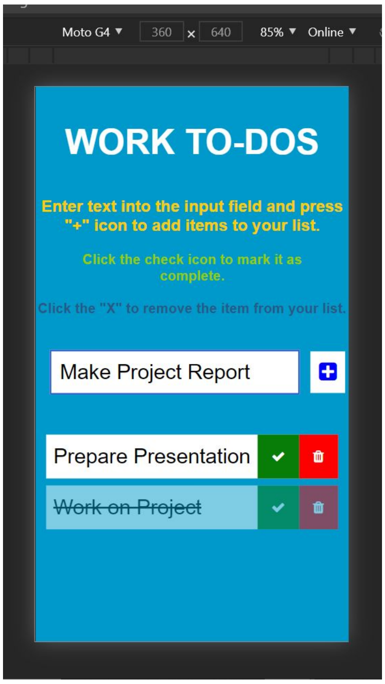
100 Park Street

Mathura, UTTAR PRADESH, 281001

Our Hours

Monday: 9am - 5pm

ToDo List in mobile screen:



Moto G4 ▼ 360 × 640 85% ▼ Online ▼

WORK TO-DOS

Enter text into the input field and press "+" icon to add items to your list.

Click the check icon to mark it as complete.

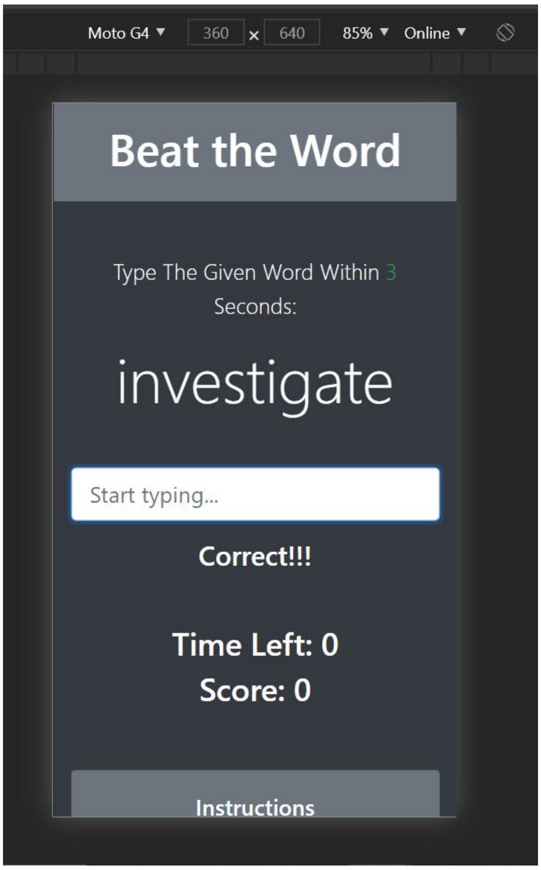
Click the "X" to remove the item from your list.

Make Project Report

Prepare Presentation

~~Work on Project~~

Typing Speed Game in mobile screen:



Beat the Word

Type The Given Word Within 3 Seconds:

investigate

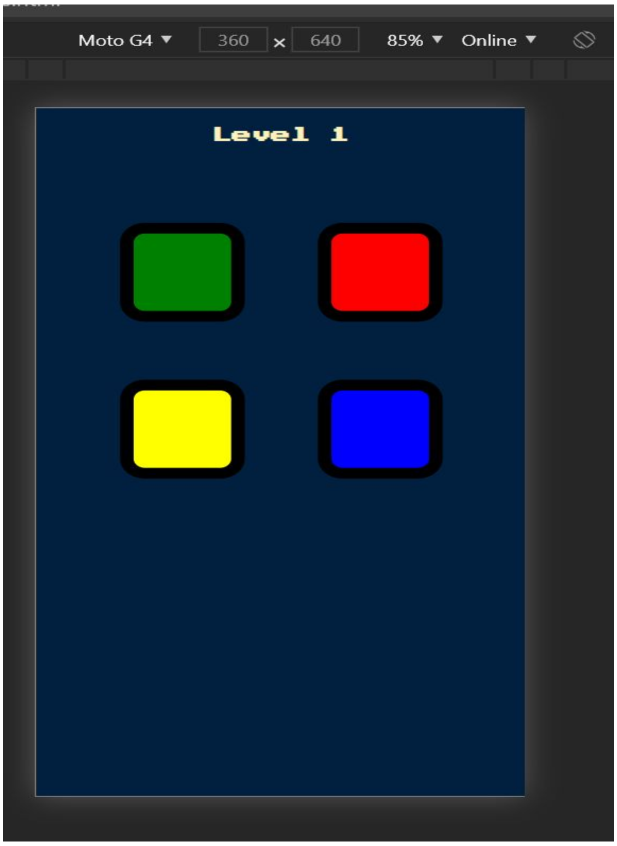
Start typing...

Correct!!!

Time Left: 0

Score: 0

Simon Game in mobile Screen:



Conclusion

It will be a wonderful and learning experience for us while working on this project. We decided to work on this project because we want to promote fun with learning and our all applications are based on this concept.

Fun with learning helps people to release their stress while not only playing the games, but also allows them to learn a new skill.

We have made a good interactive User Interface (UI) so that the user will not face any difficulties while using it. We have built different types of applications through which we can target different types of users according to their needs.

The joy of work and thrill involved while tackling the various problems and challenges will give us the feel of the developer industry.

Online Git Repository

https://github.com/ayushgupta6387/JavaScript_Applications_Website

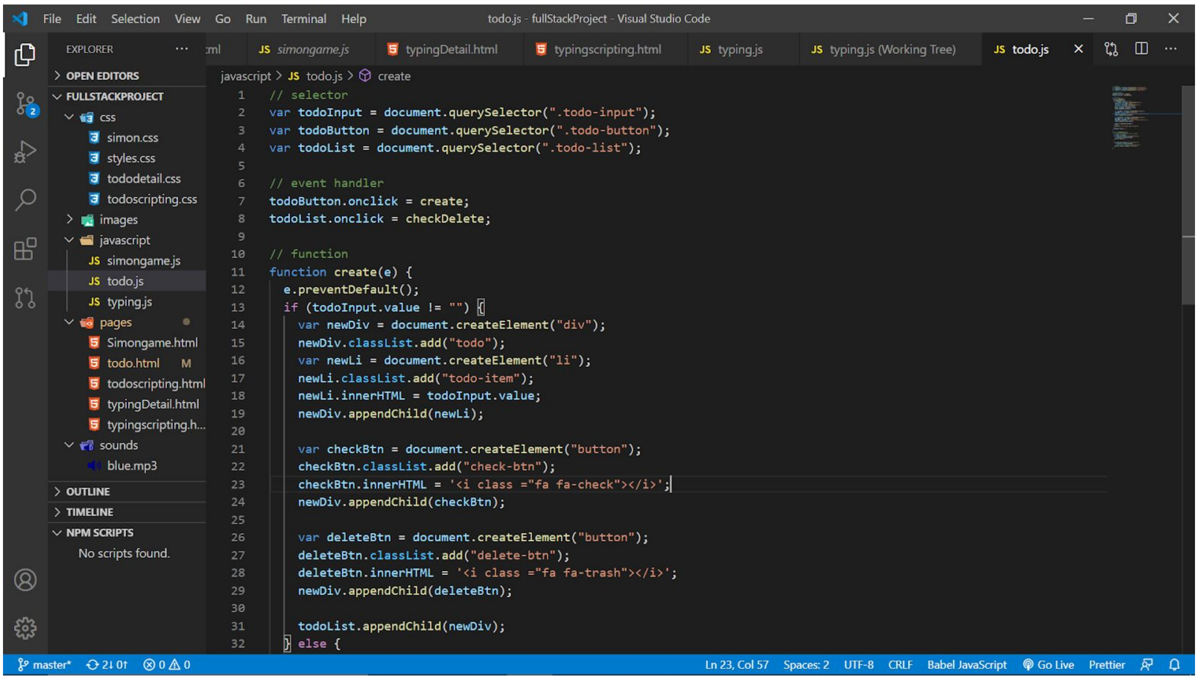
https://github.com/kshitijgupta468035/JavaScript_Applications_Website

Live Project Link

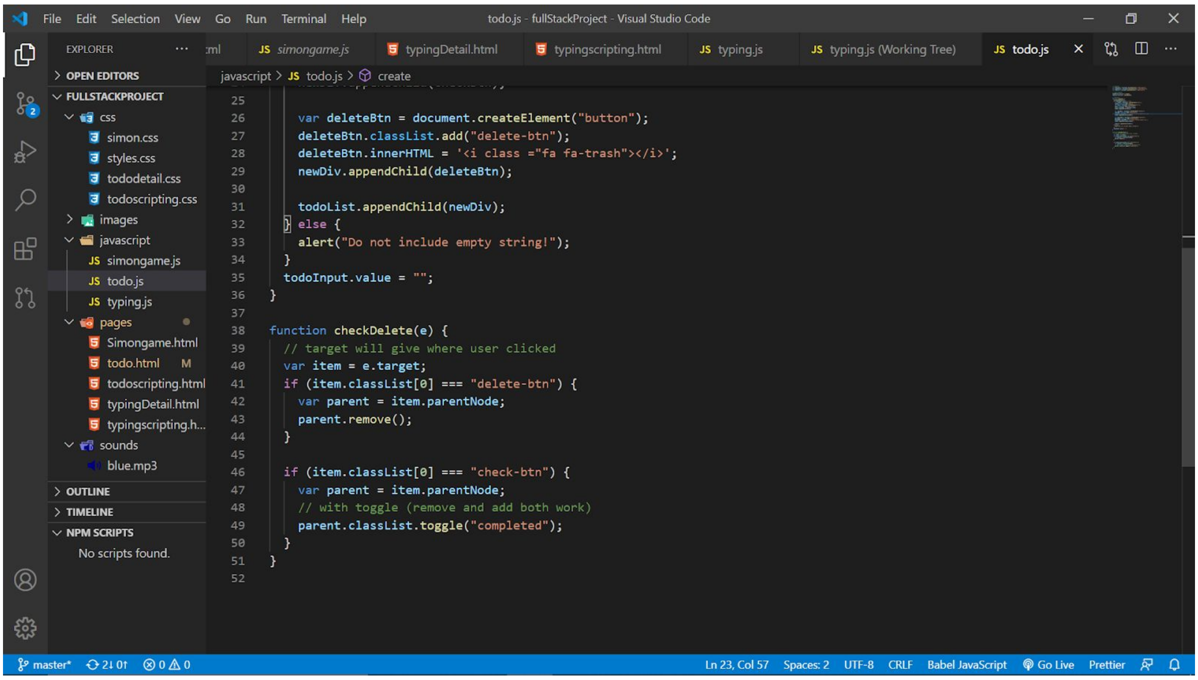
https://ayushgupta6387.github.io/JavaScript_Applications_Website/

https://kshitijgupta468035.github.io/JavaScript_Applications_Website/

Appendices



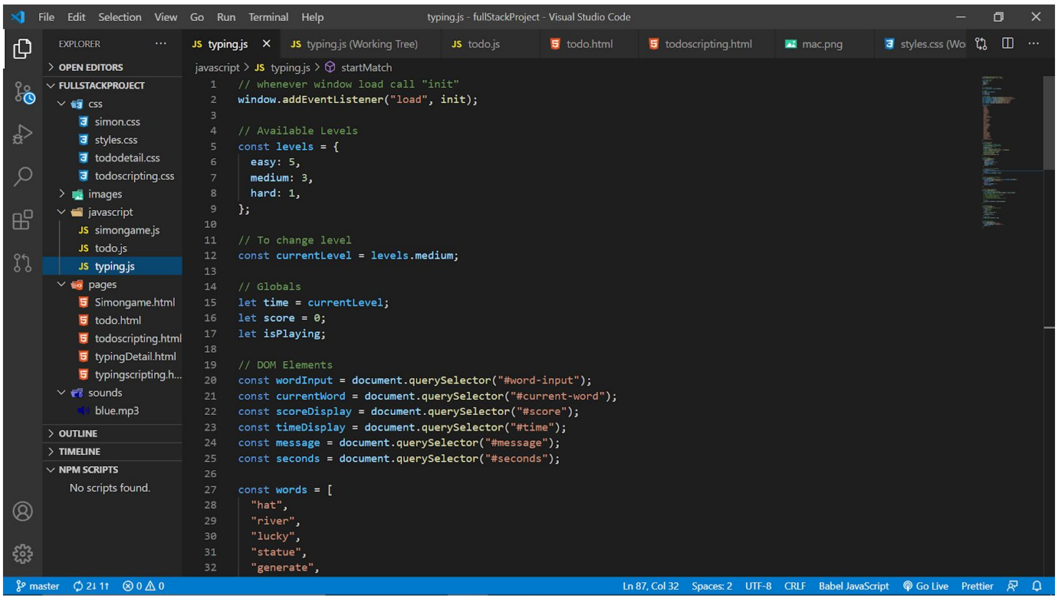
```javascript
// selector
var todoInput = document.querySelector(".todo-input");
var todoButton = document.querySelector(".todo-button");
var todoList = document.querySelector(".todo-list");

// event handler
todoButton.onclick = create;
todoList.onclick = checkDelete;

// function
function create(e) {
  e.preventDefault();
  if (todoInput.value != "") {
    var newDiv = document.createElement("div");
    newDiv.classList.add("todo");
    var newLi = document.createElement("li");
    newLi.classList.add("todo-item");
    newLi.innerHTML = todoInput.value;
    newDiv.appendChild(newLi);

    var checkBtn = document.createElement("button");
    checkBtn.classList.add("check-btn");
    checkBtn.innerHTML = '<i class ="fa fa-check"></i>';
    newDiv.appendChild(checkBtn);

    var deleteBtn = document.createElement("button");
    deleteBtn.classList.add("delete-btn");
    deleteBtn.innerHTML = '<i class ="fa fa-trash"></i>';
    newDiv.appendChild(deleteBtn);

    todoList.appendChild(newDiv);
  } else {
```



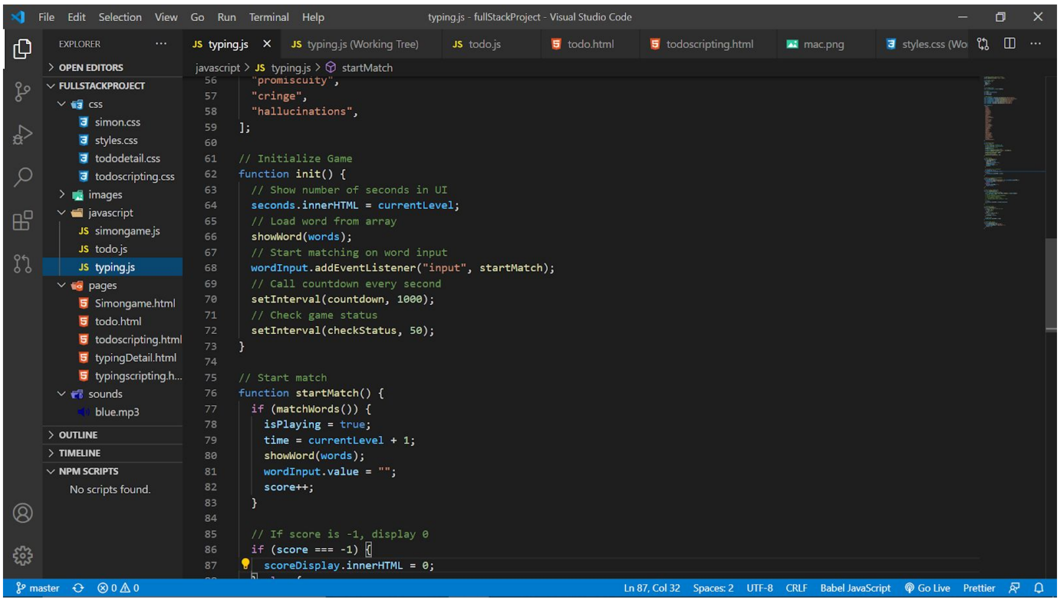
File Edit Selection View Go Run Terminal Help
todo.js - fullStackProject - Visual Studio Code
javascript > todo.js > create
master* 2↓ 0↑ 0 0
Ln 23, Col 57 Spaces: 2 UTF-8 CRLF Babel JavaScript Go Live Prettier

```javascript

    var deleteBtn = document.createElement("button");
    deleteBtn.classList.add("delete-btn");
    deleteBtn.innerHTML = '<i class ="fa fa-trash"></i>';
    newDiv.appendChild(deleteBtn);

    todoList.appendChild(newDiv);
  } else {
```

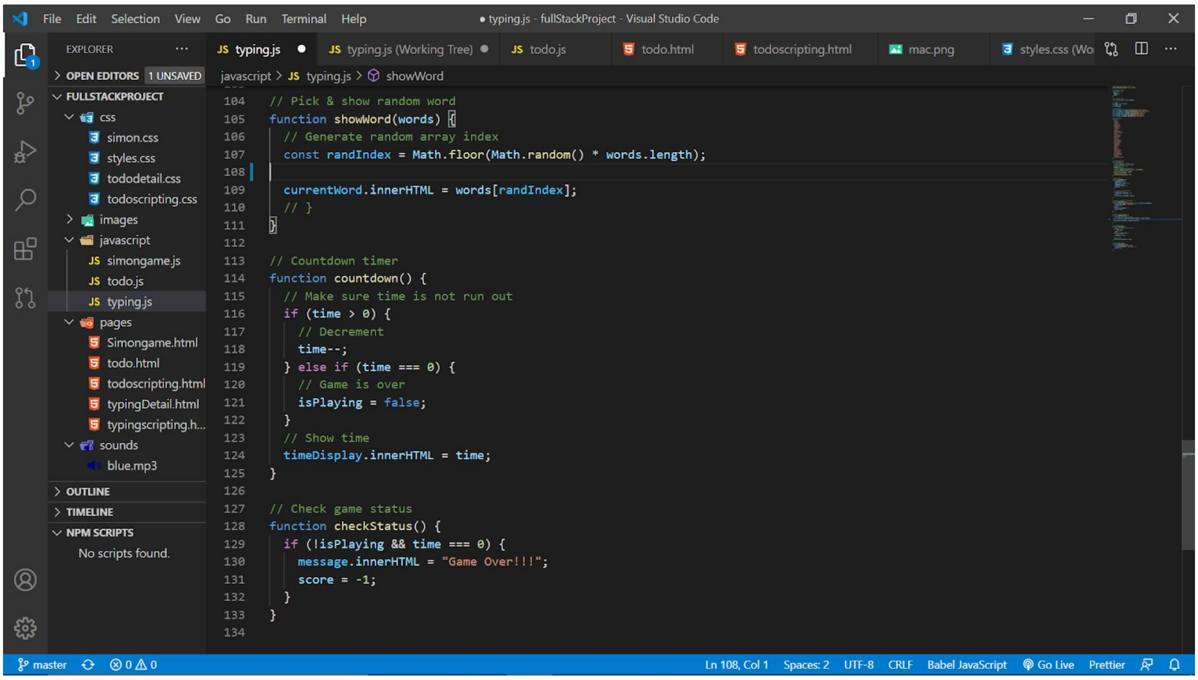
```javascript
    alert("Do not include empty string!");
  }
  todoInput.value = "";
}

function checkDelete(e) {
  // target will give where user clicked
  var item = e.target;
  if (item.classList[0] === "delete-btn") {
    var parent = item.parentNode;
    parent.remove();
  }

  if (item.classList[0] === "check-btn") {
    var parent = item.parentNode;
    // with toggle (remove and add both work)
    parent.classList.toggle("completed");
  }
}
```



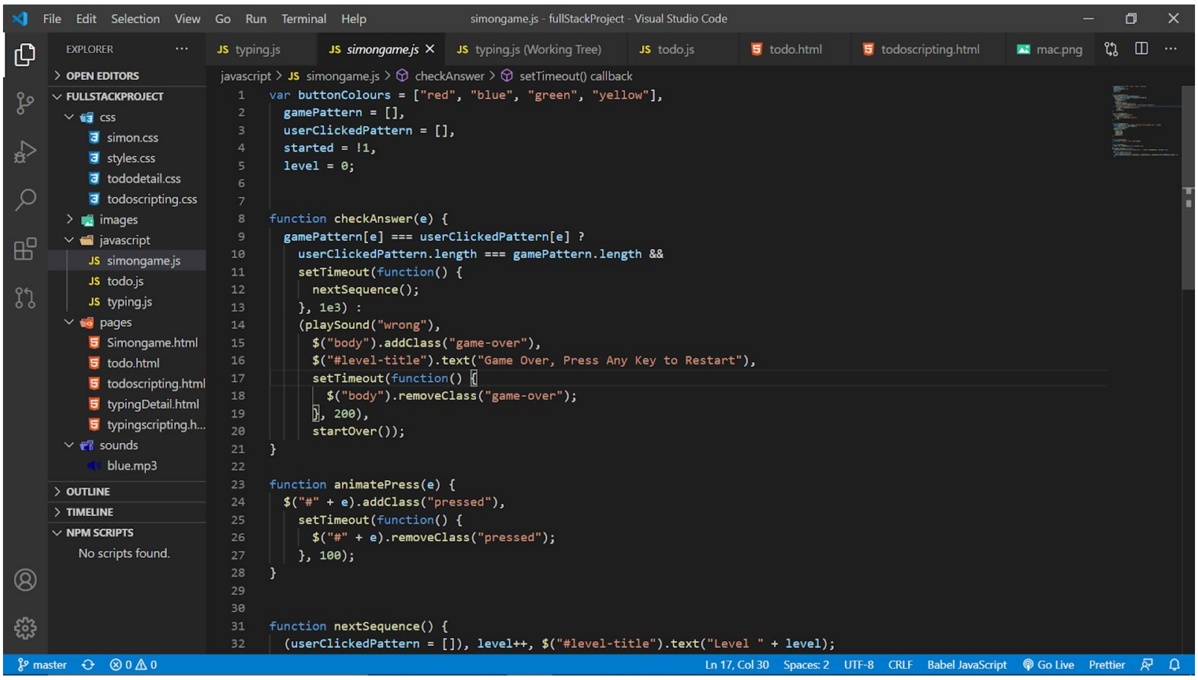
```
// whenever window load call "init"
window.addEventListener("load", init);

// Available Levels
const levels = {
  easy: 5,
  medium: 3,
  hard: 1,
};

// To change level
const currentLevel = levels.medium;

// Globals
let time = currentLevel;
let score = 0;
let isPlaying;

// DOM Elements
const wordInput = document.querySelector("#word-input");
const currentWord = document.querySelector("#current-word");
const scoreDisplay = document.querySelector("#score");
const timeDisplay = document.querySelector("#time");
const message = document.querySelector("#message");
const seconds = document.querySelector("#seconds");

const words = [
  "hat",
  "river",
  "lucky",
  "statue",
  "generate",
```



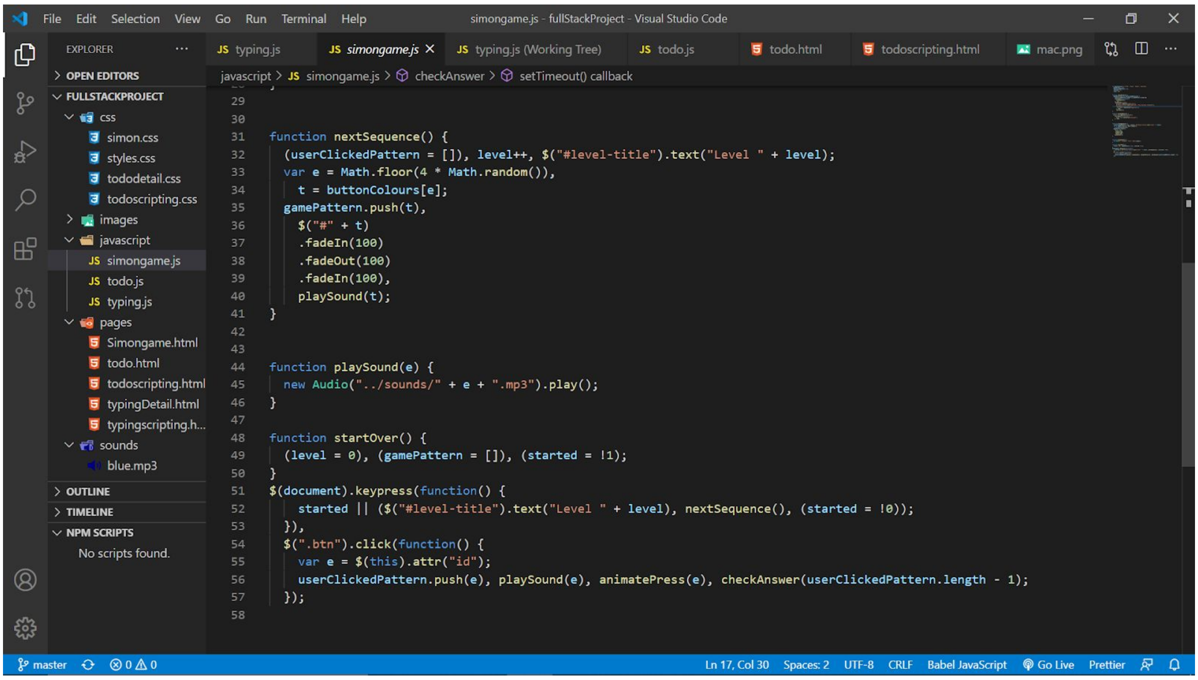
```
  "promiscuity",
  "cringe",
  "hallucinations",
];

// Initialize Game
function init() {
  // Show number of seconds in UI
  seconds.innerHTML = currentLevel;
  // Load word from array
  showWord(words);
  // Start matching on word input
  wordInput.addEventListener("input", startMatch);
  // Call countdown every second
  setInterval(countdown, 1000);
  // Check game status
  setInterval(checkStatus, 50);
}

// Start match
function startMatch() {
  if (matchWords()) {
    isPlaying = true;
    time = currentLevel + 1;
    showWord(words);
    wordInput.value = "";
    score++;
  }

  // If score is -1, display 0
  if (score === -1) {
    scoreDisplay.innerHTML = 0;
```



```
// Pick & show random word
function showWord(words) {
  // Generate random array index
  const randIndex = Math.floor(Math.random() * words.length);

  currentWord.innerHTML = words[randIndex];
  // }
}

// Countdown timer
function countdown() {
  // Make sure time is not run out
  if (time > 0) {
    // Decrement
    time--;
  } else if (time === 0) {
    // Game is over
    isPlaying = false;
  }
  // Show time
  timeDisplay.innerHTML = time;
}

// Check game status
function checkStatus() {
  if (!isPlaying && time === 0) {
    message.innerHTML = "Game Over!!!";
    score = -1;
  }
}
```



```javascript
var buttonColours = ["red", "blue", "green", "yellow"],
  gamePattern = [],
  userClickedPattern = [],
  started = !1,
  level = 0;


function checkAnswer(e) {
  gamePattern[e] === userClickedPattern[e] ?
    userClickedPattern.length === gamePattern.length &&
    setTimeout(function() {
      nextSequence();
    }, 1e3) :
    (playSound("wrong"),
      $("body").addClass("game-over"),
      $("#level-title").text("Game Over, Press Any Key to Restart"),
      setTimeout(function() {
        $("body").removeClass("game-over");
      }, 200),
      startOver());
}

function animatePress(e) {
  $("#" + e).addClass("pressed"),
    setTimeout(function() {
      $("#" + e).removeClass("pressed");
    }, 100);
}


function nextSequence() {
  (userClickedPattern = []), level++, $("#level-title").text("Level " + level);
```



```javascript


function nextSequence() {
  (userClickedPattern = []), level++, $("#level-title").text("Level " + level);
  var e = Math.floor(4 * Math.random()),
    t = buttonColours[e];
  gamePattern.push(t),
    $("#" + t)
    .fadeIn(100)
    .fadeOut(100)
    .fadeIn(100),
    playSound(t);
}


function playSound(e) {
  new Audio("../sounds/" + e + ".mp3").play();
}

function startOver() {
  (level = 0), (gamePattern = []), (started = !1);
}
$(document).keypress(function() {
    started || ($("#level-title").text("Level " + level), nextSequence(), (started = !0));
  }),
  $(".btn").click(function() {
    var e = $(this).attr("id");
    userClickedPattern.push(e), playSound(e), animatePress(e), checkAnswer(userClickedPattern.length - 1);
  });

```

References

Beta Labs

https://www.youtube.com/

DevDocs API Documentation

W3Schools Online Web Tutorials

Stack Overflow - Where Developers Learn, Share, & Build Careers

We have used these resources as a reference to build our project. These all resources are good e-learning platforms and give us a lot of information about the applications we are going to make on our website.

Signature of Project Guide: ______________________