NIIT

Final Project (Ecommerce Application) Documentation – Mr. David Adediji

# Project Details

|  |  |
| --- | --- |
| Project Name | Tech Course Ordering Ecommerce Website |
| Student Name | Mr. David Adediji |
| Name of Instructor | Mr. Fredrick Jones |
| Presentation Date | 27th of May, 2023 |
| Program Name | MMS |
|  |  |

# Acknowledgements

I want to deeply express gratitude to my Program Instructors, Mr. Belba, Mr. Imor, Mr. Efe Otega & Ms. Esther Iyege, Mr. Fredrick Jones, for taking the class diligently through this course. Their lessons were very concise and easy to understand. I also appreciate the opportunity given to defend a project pertaining to what I had been thought and had understood. More had been possible with the assistance of my helpful colleagues who helped make the lessons interactive, fun and had even contributed indirectly to the development of my JQuery Website Project.

# Project Overview

* This web project using jQuery and Django was created to enable users to buy tech courses.
* This project idea is inspired by Code with Mosh’s approach to ecommerce for tech courses.
* The web application was created using HTML, CSS, jQuery programming language and Django REST API framework alongside other resources.

# Project Process

* I designed a wireframe using pen and paper
* I downloaded font awesome
* Design of flowcharts for program logic
* Creation of Website markup with HTML
* Styling of website with CSS
* Programming of Website Logic with jQuery
* Built backend with Django
* Payment Processing with Stripe
* Testing

# jQuery lessons applied in project

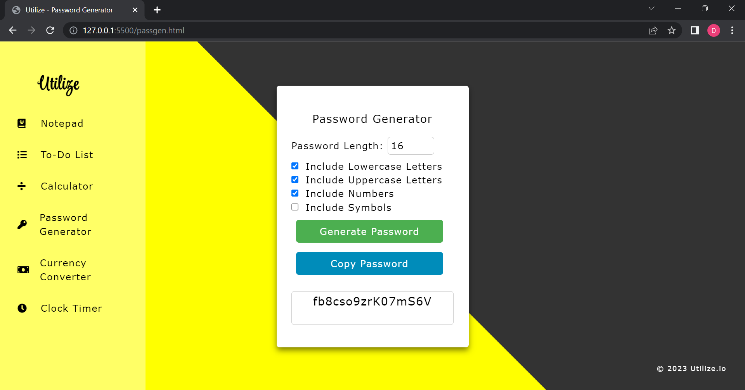
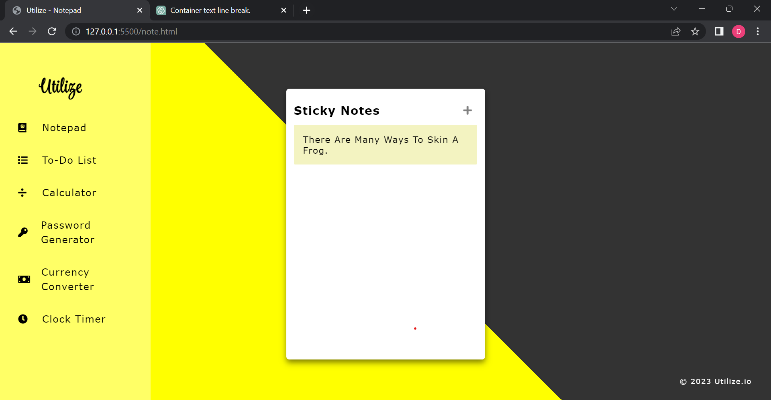
* Ajax requests
* Load function
* jQuery selectors
* Local Storage
* jQuery functions
* String Literals
* jQuery Event Listeners
* High order functions

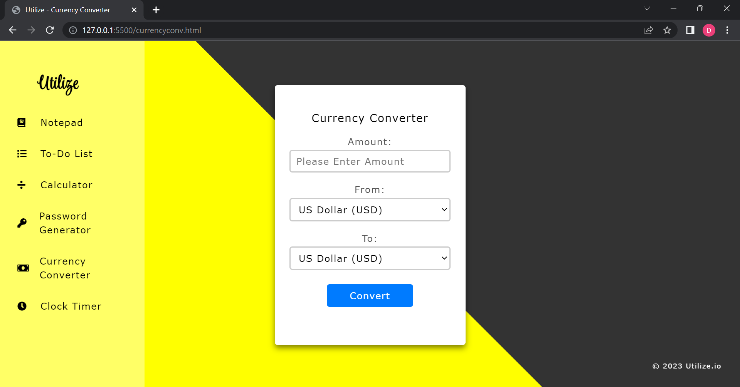
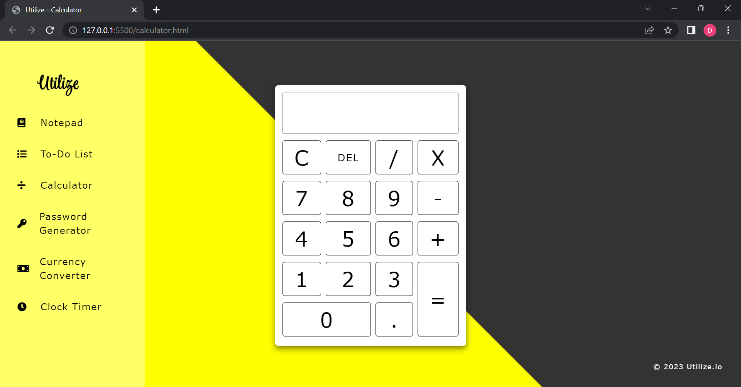
# Project Images

# 

# 

# 

# Source Code

Notepad script

 <script>

*// get notes from local storage*

        let notes = JSON.parse(localStorage.getItem("notes")) || [];

*// display notes on page load*

        displayNotes();

*// add event listener for "Add Note" button*

        $("#add-note-btn").on("click", function () {

*// create note-writing section and append it to the notepad div*

          let noteWritingSection = `

            <div id="note-writing" class="sticky\_\_new-note">

              <div class="sticky\_\_header">

                <h3>New Note</h3>

                <div>

                  <button id="back-btn" class="sticky\_\_add-note"><i class="fas fa-save"></i></button>

                  <button id="cancel-note-btn" class="sticky\_\_add-note"><i class="fas fa-times"></i></button>

                </div>

              </div>

              <textarea id="note-text" class="sticky\_\_note-text"></textarea>

            </div>

          `;

          $("#notepad").append(noteWritingSection);

*// hide note-list and show note-writing section*

          $("#note-list").hide();

*// $("#note-writing").show();*

        });

*// add event listener for "Back" button*

        $("#notepad").on("click", "#back-btn", function () {

          let noteText = $("#note-text").val();

          if (noteText !== "") {

            notes.push(noteText);

            localStorage.setItem("notes", JSON.stringify(notes));

            $("#note-text").val("");

            displayNotes();

          }

          $("#note-list").show();

          $("#note-writing").remove();

        });

*// add event listener for "Cancel" button*

        $("#notepad").on("click", "#cancel-note-btn", function () {

          $("#note-text").val("");

          $("#note-list").show();

          $("#note-writing").hide();

*// remove note-writing section from notepad div*

          $("#note-writing").remove();

        });

*// add event listener for "Save Note" button*

        $("#notepad").on("click", "#save-note-btn", function () {

          let noteText = $("#note-text").val();

          if (noteText !== "") {

            let noteIndex = $(*this*).data("note-index");

            if (noteIndex !== undefined) {

*// if noteIndex is defined, this is an edited note*

              notes[noteIndex] = noteText;

            } else {

*// otherwise, this is a new note*

              notes.push(noteText);

            }

            localStorage.setItem("notes", JSON.stringify(notes));

            $("#note-text").val("");

            displayNotes();

            $("#note-list").show();

            $("#note-writing").hide();

*// remove note-writing section from notepad div*

            $("#note-writing").remove();

          }

        });

*// add event listener for note list item*

        $("#note-list ul").on("click", "li", function () {

          let noteIndex = $(*this*).index();

          $("#note-text").val(notes[noteIndex]);

*// create note-writing section and append it to the notepad div*

          let noteWritingSection = `

            <div id="note-writing" class="sticky\_\_edit-note">

              <div class="sticky\_\_header">

                <h3>Edit Note</h3>

                <div class="sticky\_\_header-btns">

                  <button id="save-note-btn" class="sticky\_\_add-note" data-note-index="${noteIndex}"><i class="fas fa-save"></i></button>

                  <button id="cancel-note-btn" class="sticky\_\_add-note"><i class="fas fa-times"></i></button>

                </div>

              </div>

              <textarea class="sticky\_\_note-text" id="note-text">${notes[noteIndex]}</textarea>

            </div>

          `;

          $("#notepad").append(noteWritingSection);

*// hide note-list and show note-writing section*

          $("#note-list").hide();

          $("#note-writing").show();

        });

        function displayNotes() {

          let noteList = $("#note-list ul");

          noteList.empty();

          for (let i = 0; i < notes.length; i++) {

            let noteText = notes[i];

            let noteItem = $('<li class="sticky\_\_note-item">').text(noteText);

            let deleteButton = $(

              '<button class="sticky\_\_delete-note-btn"><i class="fas fa-trash"></i></button>'

            );

            deleteButton.click(function () {

              let noteIndex = $(*this*).parent().index();

*// to remove the note*

              notes.splice(noteIndex, 1);

              localStorage.setItem("notes", JSON.stringify(notes));

              displayNotes();

            });

            noteItem.append(deleteButton);

            noteList.append(noteItem);

          }

        }

      </script>

Calculator Script

    <script type="text/javascript">

*// to show number on the calc display*

      function insert(num) {

        $(".calc-display").val($(".calc-display").val() + num);

      }

*// to evaluate the expression*

      function eql() {

        $(".calc-display").val(eval($(".calc-display").val()));

      }

*// to clear the calc display*

      function c() {

        $(".calc-display").val("");

      }

*// to delete the last number from the calc display*

      function del() {

        value = $(".calc-display").val();

        $(".calc-display").val(value.substring(0, value.length - 1));

      }

      $(document).ready(function () {

        $("#nav-placeholder").load("nav.html");

        $("#copyright-div").load("statement.html");

      });

    </script>

Navigation Script

<script type="text/javascript">

  function showNavUtil() {

    $(".side-nav").toggleClass("responsive-nav");

    $(".header .btn-show").toggleClass("invisible");

  }

  $(document).ready(function () {

*// get the initial width of the viewport*

    var width = $(window).width();

*// check the width on window resize*

    $(window).resize(function () {

*// get the new width of the viewport*

      var newWidth = $(window).width();

*// if the width has increased beyond the mobile breakpoint, remove the responsive-nav class*

      if (newWidth >= 768 && width < 768) {

        $(".responsive-nav").removeClass("responsive-nav");

      }

*// update the width variable*

      width = newWidth;

    });

  });

</script>

Timer Script

<script>

      var initialHours, initialMinutes, initialSeconds;

      var hours, minutes, seconds;

      var timer;

      var running = false;

      $("#hours").on("input", function () {

        let hours = parseInt($(*this*).val());

        if (isNaN(hours)) {

          hours = 0;

        }

        console.log(hours);

        $("#hours").val(hours > 23 ? 00 : hours);

        $("#hours-display").text(

          hours < 10 ? "0" + hours : hours > 23 ? 00 : hours

        );

      });

      $("#minutes").on("input", function () {

        let minutes = parseInt($(*this*).val());

        if (isNaN(minutes)) {

          minutes = 0;

        }

        $("#minutes").val((minutes = 0 || minutes > 59 ? 00 : minutes));

        $("#minutes-display").text(

          minutes < 10 ? "0" + minutes : minutes > 59 ? 00 : minutes

        );

      });

      $("#seconds").on("input", function () {

        let seconds = parseInt($(*this*).val());

        if (isNaN(seconds)) {

          seconds = 0;

        }

        $("#seconds").val((seconds = 0 || seconds > 59 ? 00 : seconds));

        $("#seconds-display").text(

          seconds < 10 ? "0" + seconds : seconds > 59 ? 00 : seconds

        );

      });

      $(document).ready(function () {

        var hoursInput = $("#hours");

        var minutesInput = $("#minutes");

        var secondsInput = $("#seconds");

        var hoursDisplay = $("#hours-display");

        var minutesDisplay = $("#minutes-display");

        var secondsDisplay = $("#seconds-display");

        var startButton = $("#start");

        var pauseButton = $("#pause");

        var resumeButton = $("#resume");

        var paused = false;

        startButton.click(function () {

          if (!running && !paused) {

            initialHours = parseInt(hoursInput.val());

            initialMinutes = parseInt(minutesInput.val());

            initialSeconds = parseInt(secondsInput.val());

            if (initialHours + initialMinutes + initialSeconds == 0) {

              return;

            }

            hours = initialHours;

            minutes = initialMinutes;

            seconds = initialSeconds;

            timer = setInterval(function () {

              if (seconds == 0) {

                if (minutes == 0) {

                  if (hours == 0) {

                    clearInterval(timer);

                    running = false;

                    startButton.text("Start");

                    return;

                  } else {

                    hours--;

                    minutes = 59;

                    seconds = 59;

                  }

                } else {

                  minutes--;

                  seconds = 59;

                }

              } else {

                seconds--;

              }

              hoursDisplay.text(hours < 10 ? "0" + hours : hours);

              minutesDisplay.text(minutes < 10 ? "0" + minutes : minutes);

              secondsDisplay.text(seconds < 10 ? "0" + seconds : seconds);

              $("#hours-display").addClass("pulse");

              $("#minutes-display").addClass("pulse");

              $("#seconds-display").addClass("pulse");

            }, 1000);

            running = true;

            startButton.text("Stop");

            pauseButton.show();

            resumeButton.hide();

          } else if (paused || running) {

            clearInterval(timer);

            running = false;

            paused = false;

            startButton.text("Start");

            $("#hours-display").removeClass("pulse");

            $("#minutes-display").removeClass("pulse");

            $("#seconds-display").removeClass("pulse");

            $("#hours-display").text("00");

            $("#minutes-display").text("00");

            $("#seconds-display").text("00");

            $("#pause").hide();

            $("#resume").hide();

          }

        });

        pauseButton.click(function () {

          clearInterval(timer);

          running = false;

          paused = true;

          pauseButton.hide();

          resumeButton.show();

        });

        resumeButton.click(function () {

          if (!running) {

            timer = setInterval(function () {

              if (seconds == 0) {

                if (minutes == 0) {

                  if (hours == 0) {

                    clearInterval(timer);

                    running = false;

                    paused = false;

                    startButton.text("Start");

                    return;

                  } else {

                    hours--;

                    minutes = 59;

                    seconds = 59;

                  }

                } else {

                  minutes--;

                  seconds = 59;

                }

              } else {

                seconds--;

              }

              hoursDisplay.text(hours < 10 ? "0" + hours : hours);

              minutesDisplay.text(minutes < 10 ? "0" + minutes : minutes);

              secondsDisplay.text(seconds < 10 ? "0" + seconds : seconds);

              $("#hours-display").addClass("pulse");

              $("#minutes-display").addClass("pulse");

              $("#seconds-display").addClass("pulse");

            }, 1000);

            running = true;

            pauseButton.show();

            resumeButton.hide();

          }

        });

      });

    </script>

To-do List Script

    <script>

      $(document).ready(function () {

        $("#nav-placeholder").load("nav.html");

        $("#copyright-div").load("statement.html");

*// localStorage.clear();*

*// Create a new div element to hold the toast notification*

        var toast = $("<div>");

*// Add classes to the toast element*

        toast.addClass("toast");

        toast.addClass("auto-save");

*// Set the text of the toast element*

        toast.text("Your tasks are auto-saved");

*// Add the toast element to the page*

        $("body").append(toast);

*// Fade in the toast element*

        toast.fadeIn(200);

*// Wait 3 seconds and then fade out the toast element*

        setTimeout(function () {

          toast.fadeOut(500, function () {

            $(*this*).remove(); *// Remove the toast element from the page*

          });

        }, 3000);

*// retrieve tasks from local storage*

        var tasks = JSON.parse(localStorage.getItem("tasks")) || [];

*// render tasks from local storage*

        $.each(tasks, function (index, value) {

          var checked = value.checked ? "checked" : "";

          $(".task-list").append(

            "<li class='" +

              checked +

              "' data-id='" +

              index +

              "'>" +

              value.text +

              '<i class="fas fa-check"> </i> <i class="fas fa-trash"></i> </li>'

          );

        });

        $("#input").change(function () {

          var input = $(*this*).val();

          $(".task-list").append(

            "<li data-id='" +

              tasks.length +

              "'>" +

              input +

              '<i class="fas fa-check"> </i> <i class="fas fa-trash"></i> </li>'

          );

          $(*this*).val("");

*// add task to local storage*

          tasks.push({

            text: input,

            checked: false,

          });

          localStorage.setItem("tasks", JSON.stringify(tasks));

        });

        $(".task-list").on("click", ".fa-trash", function () {

          var taskId = $(*this*).parent("li").data("id");

          $(*this*)

            .parent("li")

            .fadeOut(200, function () {

*// remove task from local storage*

              tasks.splice(taskId, 1);

              localStorage.setItem("tasks", JSON.stringify(tasks));

            });

        });

        $(".task-list").on("click", ".fa-check", function () {

          var taskId = $(*this*).parent("li").data("id");

          $(*this*).parent("li").toggleClass("checked");

*// update task in local storage*

          var checked = $(*this*).parent("li").hasClass("checked");

          tasks[taskId].checked = checked;

          localStorage.setItem("tasks", JSON.stringify(tasks));

        });

      });

    </script>

Password Generator script

    <script type="text/javascript">

      function generatePassword() {

*// get/check parameters to generate password*

        const length = document.getElementById("password-length").value;

        const includeLowercase = $("#include-lowercase").prop("checked");

        const includeUppercase = $("#include-uppercase").prop("checked");

        const includeNumbers = $("#include-numbers").prop("checked");

        const includeSymbols = $("#include-symbols").prop("checked");

*// picking character set to use based on parameters*

        let charset = "";

        if (includeLowercase) charset += "abcdefghijklmnopqrstuvwxyz";

        if (includeUppercase) charset += "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

        if (includeNumbers) charset += "0123456789";

        if (includeSymbols) charset += "!@#$%^&\*()\_+-=[]{}|;:,.<>?";

        let password = "";

        for (let i = 0; i < length; i++) {

*// select a character from the set by index after picking index at random and add it to the password*

          password += charset.charAt(

            Math.floor(Math.random() \* charset.*length*)

          );

        }

*// Set the generated password in localStorage*

        localStorage.setItem("password", password);

        return password;

      }

      function closeNavUtil() {

        $(".side-nav").removeClass("responsive-nav");

      }

      $(document).ready(function () {

        $("#nav-placeholder").load("nav.html");

        $("#copyright-div").load("statement.html");

*// display result*

        $("#generate-password").click(function () {

          var password = generatePassword();

          $("#password").val(password);

        });

*// copy to generated password to clipboard*

        $("#copy-password").click(function () {

          var password = $("#password").val();

          if (password) {

            $("#password").select();

            document.execCommand("copy");

            alert("Password copied to clipboard!");

          }

        });

*// Retrieve the password from localStorage when the page loads - persistence*

        const password = localStorage.getItem("password");

        if (password) {

          document.getElementById("password").value = password;

        }

*// prevent the password length input field from exceeding 20*

        $(".pass-length").on("input", function () {

          if ($(*this*).val() > 20) {

            value = $(*this*).val();

            $(*this*).val(value.substring(0, value.length - 1));

          }

        });

      });

    </script>

Currency Converter Script

    <script>

      $(document).ready(function () {

*// Set API endpoint and key*

        var endpoint = "https://api.exchangerate-api.com/v4/latest/";

        var apiKey = "YOUR\_API\_KEY\_HERE";

*// Attach click event handler to the convert button*

        $("#convert").click(function () {

*// Get the input values*

          var amount = $("#amount").val();

          var from = $("#from").val();

          var to = $("#to").val();

*// Send GET request to API endpoint*

          $.ajax({

            url: endpoint + from,

            type: "GET",

            success: function (response) {

*// Calculate the conversion*

              var rate = response.rates[to];

              var result = amount \* rate;

*// Display the result*

              $("#result").text(

                amount + " " + from + " = " + result.toFixed(2) + " " + to

              );

            },

            error: function (xhr, status, error) {

*// Display an error message if the API request fails*

              $("#result").text("Error: " + error);

            },

          });

        });

        $("#nav-placeholder").load("nav.html");

        $("#copyright-div").load("statement.html");

      });

    </script>