**Product Documentation: Technical Features of “VClass”**

**Compatibility:**

Devices: Mac, PC

Browsers: Chrome, Internet Explorer, Microsoft Edge, Safari

Additional Hardware: Trackpad, Mouse, Keyboard

**Currently not compatible/Untested on:**

Devices: iPhone, Android, other handheld devices, Linux

Browsers: Firefox, Opera, Safari

Additional Hardware: Keyboard, Stylus/Drawing Pad

**Files:**

There are 3 files that are used, in the languages HTML, CSS and JavaScript respectively. The central file is “index.html” (the webpage), the linked styling sheet is “majorProject.css”. The JavaScript file is “majorProject.js”, which uses the JavaScript library JQuery and contains all the variables followed by the functions and then the control structures. This file also interacts with the database.

The images folder contains all images and audio files referenced to within my HTML and JavaScript files.

**Functions:**

The ‘global’ functions at the top of my JavaScript folder are to reduce complexity within click handlers and to allow functions to be reusable through parameters.

The majority of the functions are called for in the event of a button click, or keyboard press in combination with hiding and showing HTML elements using JQuery.

The functions used to access the databases within resdb.io:

* The functions are named according to the database name to reduce errors within the code.
* There are 3 functions to access the user database
  + “getVClassUsers”, which accepts a url and an API key. This function is used to retrieve all the registered users and put it into a global array called “arrVClassUsers”. It is run at the when the webpage first loads because the data is required for the login and register functions.
  + “addVClassUser”, which accepts an item, url and an API key. This function adds a new user to the database when they register.
  + “editUser”, which accepts an item, a url containing the object ID and the API key. This function is used to add another class code to a user’s current array of class codes.
* 2 functions to access the app classes
  + “getVClassClassrooms”, which accepts a url and API key. This function is used to retrieve all the classroom names and their corresponding class codes. It is run when the webpage first loads, to then display the appropriate classrooms when the user goes onto their homepage.
  + “addVClassClassroom”, which accepts an item, url and API key. This function is used when a new class is created and adds a new object to the VClassClassrooms database
* 2 classes to access the app ‘pinboards’
  + “getVClassPinboards”, which accepts a url, API key and relevantClassroom. This function is called as soon as the user selects a particular classroom, and linear searches to append the relevant pinboard information onto the virtual class pinboard.
  + “addVClassPinboards”, which accepts an item, url and API key. This function is used when a new pinboard post is created and adds a new object to the VClassPinboards database.

Homepage functions:

The “homePage” function is used when the home icon image is selected.

* It displays the homepage HTML elements, including the user’s classrooms and inserts the user’s name to welcome them.
* If it is the first time the function is run there is a linear search through the user’s class codes and the array of classrooms to find all relevant classroom information. This is put into the array “arrClassesToSort” which is passed through the parameter and sorted by the “selectionSortClasses” function. This function places classroom divs onto the homepage and sorts them anti-alphabetically.

The “newClass” function is used when the “imgNewClass” image is selected. This function checks whether the user is a teacher or student then displays the HTML to create a new class or add a new class respectively.

The “generateString” function is used when a teacher is creating a new class, and the class code becomes a unique identifier for the when students are adding their new class. A length is passed into its parameter, and it returns a random string selected from the characters A-Z, a-z, 0-9.

Randomising aesthetic functions:

The “randomBackgroundColour” function is used to randomise the background in the virtual classroom environment, to make it seem a bit more unique each time. This returns a random hex code taken from an array called “arrBackgroundColours” of random colours.

Likewise, the “randBorderColour” function randomises the colours of each class icon on the homepage display. This function returns a random hex code taken from an array called “arrImgBorders” of random colours.

Video livestream function:

the function “init” is run when the user is on the joining classroom page, asking for the users camera permission. The feed called “myVideo” is then then transferred from the “outOfClassUser” div on the joining classroom page to the “inClassUser” inside the virtual space, which is then moved around by the user.

Interactive virtual element functions:

The “fishProgress” function is run from when the webpage is loaded, this shortens the progress bar by 10px every 10 minutes.

The “checkSound” function is run every time the user presses their keyboard arrows. If statements are used to check the user’s placement on the screen, if they are close enough to the stereo image then music will start playing, the closer they get, the louder the music will be.

Likewise, the “checkFish” and “checkPinboard” functions are used to check whether the user div is near the visuals and will display a prompt to let the user know they can be interacted with.

**Variables:**

All variables and elements use camel case.

Arrays:

All arrays start with the prefix “arr”

* “arrVClassUsers” contains all user information retrieved from the database VClassUsers. This is used for the register and login functions as well, once the user is created or identified their information is then placed into a “currentUser” variable for easy identification and to identify their unique classrooms.
* “arrVClassClassrooms” contains all information retrieved from the database VClassClassrooms. This is searched through and compared with the current user’s class codes to find the relevant classroom data. The current class name is also stored in a variable called “currentClassName” for easy access.
* “arrVClassPinboards” contains all information retrieved from the database VClassPinbaords. This is searched through and compared to the “currentClassName” to find and display the relevant pinboard information to the class that the user has selected.
* “arrClassesToSort” contains the relevant classroom data to the user. This array is then intended to be passed into the SelectionSortClasses function to be sorted alphabetically and then display classroom divs on the hompage.

Objects:

The majority of the objects created are with the purpose of putting them into the database. They are generally called “tempItem”. For example, “tempItemUser” is created by taking input from the input boxes and displaying them within required fields, this is added to the “currentUser” universal variable to then be passed into the function “addVClassUser” and added to the database.

Boolean:

The boolean variable “stopMovement” is used to ensure that the virtual classroom background and user don’t move around in the background when the user is interacting with the pinboard display. “stopMovement” is initially set to false but when the pinboard is selected then it is set to true and no further events happen on keyboard arrow press.

The Boolean variable “initialHomeSetup” is used so that the classroom divs on the homepage are only created once. “initialHomeSetup” is initially set to false, the first time the “homePage” function is run and the divs are printed then the Boolean is set to true.

**Databases:**

The game uses one database with three collections.

The user collection is where all the user details are stored.

* The fields “Email” and “Password” are necessary during the login and register processes
* The field “UserType” contains only teacher or student and will indicate whether a user can add or create a class.
* The field “UserClasses” is an array that contains strings of class codes. This is used to connect this collection with the user classrooms collection.

The classrooms collection is where the classroom names and codes are stored.

* This contains the fields “ClassCode”, which connects this collection with the user collection and “ClassName” which collects this collection with the pinboard collection.

The pinboard collection is where post title, content and creator are stored, but also class name to link it to the other collections.

* The fields “PostTitle”, “PostText” and “UsersName” are all used when displaying the pinboard post to the user. The “ClassName” field is primarily used to connect the collection to the classrooms collection.