PRELIM SDD MAJOR PROJECT DOCUMENTATION

Oscar - Reinitz, Dominic Saleh

Allocation of Tasks

Gantt Chart: Oscar

Screen Designs: Oscar

Data Dictionary: Dominic

HTML and CSS setup: Oscar

Javascript Local storage: Dominic

Javascript Binary Search: Oscar

Logbook

2021-08-01 Oscar - Removed Submit.js File

2021-04-19 Oscar - Initial File Setup

2021-04-19 Oscar - Initial commit

2021-04-22 Dominic -Update index.html

2021-04-22 Dominic -Update index.html

2021-04-26 Oscar - Merge branch 'main' of https://github.com/Oscar -44/Major-Project-2021_Subject-Selection into main

2021-04-26 Oscar - Pure CSS Setup

2021-04-26 Oscar - Update readme with section from proposal

2021-04-26 Oscar - Licensing for Pure CSS Layouts

2021-04-27 Oscar - PureCSS Experimentation

2021-04-29 Oscar - HTML homepage setup

2021-04-30 Oscar - Search page setup

2021-04-30 Oscar - Added additional HTML pages

2021-05-13 Dominic -Submission page css

2021-05-14 Oscar - Formatted Search Page

2021-05-16 Dominic -Subject names and header

2021-05-17 Oscar - Search JS setup

2021-05-19 Oscar - Search Function array Setup

2021-05-20 Oscar - Binary Search function planning

2021-05-31 Oscar - adjustment to commit log batch file

2021-05-31 Oscar - Commit logging bat file test

2021-06-02 Oscar - Significant development to binary search. Implemented div hiding and search algorithm. Search term simplification WIP

2021-06-02 Oscar - Transferred commit logging process to out of repository

2021-06-03 Dominic -Subject Selection setup

2021-06-03 Oscar - Switch statement for term simplification function

2021-06-07 Oscar - Switch statement fixed, search function debugging begun

2021-06-08 Oscar - Binary Search function completed and working. CSS centering for search page fixed. Div hide/show function error corrected.

2021-06-10 Oscar - added cases to switch statement

2021-06-16 Oscar - Adjusted background image and css styling for home page

2021-06-16 Oscar - Added subject units to localstorage variables fixed typos, added validation for subject selections, compiled purecss locally

2021-06-17 Oscar - Built Submission page

2021-07-14 Oscar - Replaced searchbar with simpler version, Fixed an issue where arrays would be called from differing js scripts due to same names, css adjustments to search page, added button to reset search

2021-07-14 Oscar - Adjustment To CSS for larger screens

2021-07-15 Oscar - Local storage of subject object array

2021-07-26 Oscar - Update README.md

2021-07-26 Oscar - Documentation Merged with GitHub

2021-07-28 Oscar - Coded Validation for Submission Page

2021-07-31 Oscar - Moved script linking
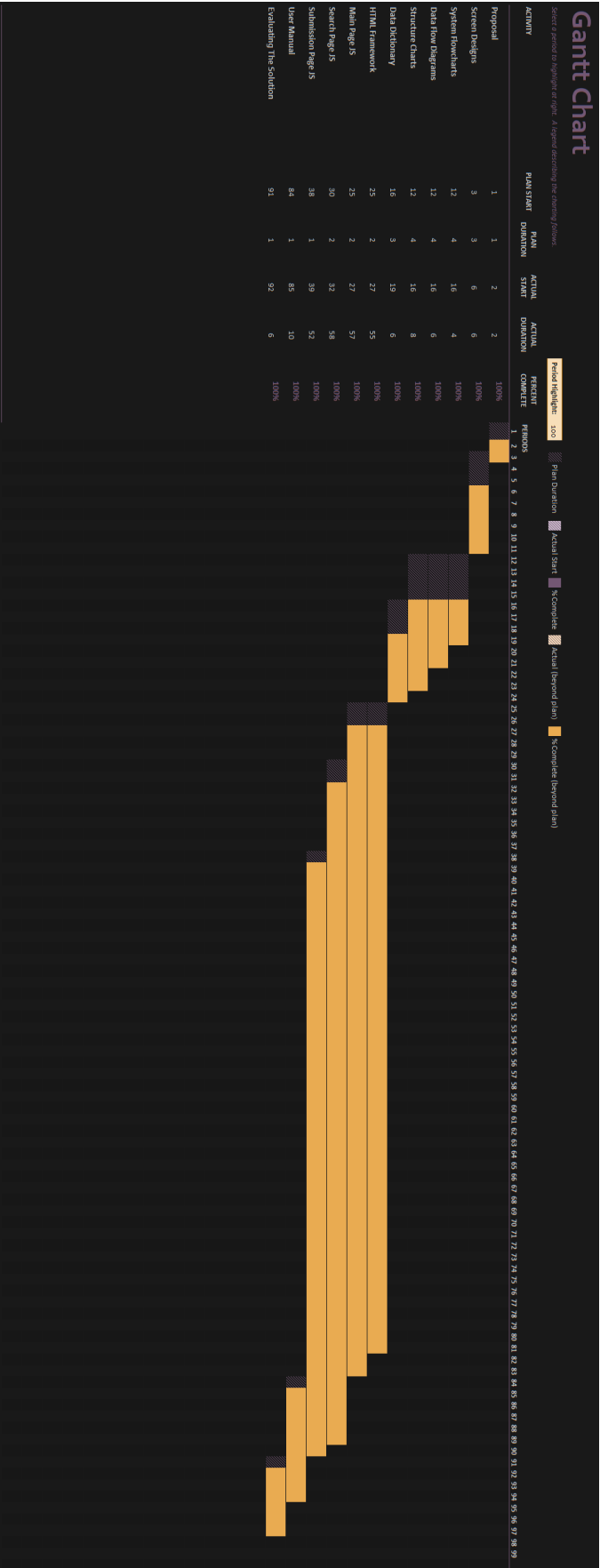
2021-07-31 Oscar - Submission Page Bugfixes

2021-08-01 Oscar - Merged All Diagrams to Folio. Resolved Missing JS file for search page
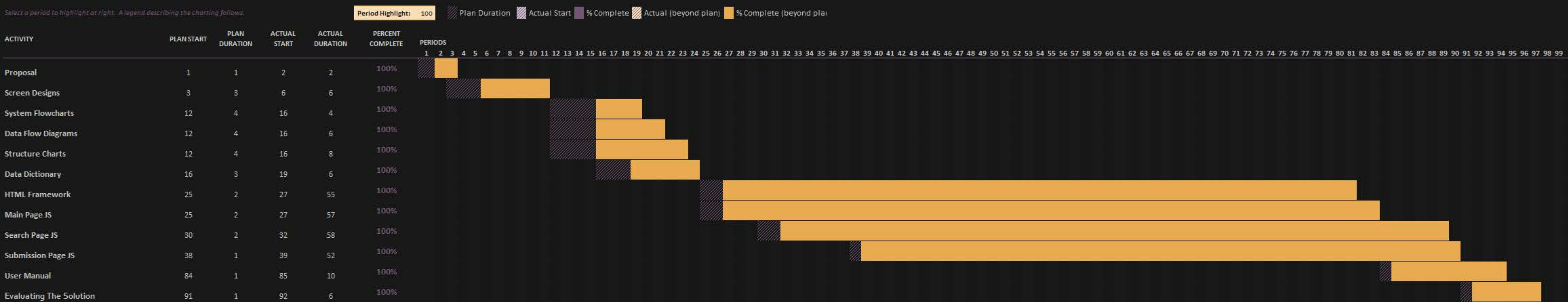
2021-08-01 Oscar – Finalised Folio

2021-08-02 Oscar – Additional Code Comments and User Manual Merge

2021-08-02 Oscar – PDF Compilation

Gantt Chart (Excel File Included)

Gantt Chart

Select a period to highlight at right. A legend describing the charting follows.

Period Highlight: 100

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Proposal	1	1	2	2	100%
Screen Designs	3	3	6	6	100%
System Flowcharts	12	4	16	4	100%
Data Flow Diagrams	12	4	16	6	100%
Structure Charts	12	4	16	8	100%
Data Dictionary	16	3	19	6	100%
HTML Framework	25	2	27	55	100%
Main Page JS	25	2	27	57	100%
Search Page JS	30	2	32	58	100%
Submission Page JS	38	1	39	52	100%
User Manual	84	1	85	10	100%
Evaluating The Solution	91	1	92	6	100%

PERIODS 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

Legend: Plan Duration | Actual Start | % Complete | Actual (beyond plan) | % Complete (beyond plan)

Gantt Chart

Period Highlight: 100

Legend: ▨ Plan Duration ▨ Actual Start ▇ % Complete ▨ Actual (beyond plan) ▇ % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Proposal	1	1	2	2	100%
Screen Designs	3	3	6	6	100%
System Flowcharts	12	4	16	4	100%
Data Flow Diagrams	12	4	16	6	100%
Structure Charts	12	4	16	8	100%
Data Dictionary	16	3	19	6	100%
HTML Framework	25	2	27	55	100%
Main Page JS	25	2	27	57	100%
Search Page JS	30	2	32	58	100%
Submission Page JS	38	1	39	52	100%
User Manual	84	1	85	10	100%
Evaluating The Solution	91	1	92	6	100%

PERIODS: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

Screen Designs

Homepage	Page Hyperlinks

Welcome Text

Clear and Refresh Button

List Of Selected Subjects

Search Page	Page Hyperlinks

Clear and Refresh Button

Search Bar

Search Results

Submit Page	Page Hyperlinks

```
┌─────────────────────────────────────────────────────┐
│ │Clear and Refresh Button│                           │
│ ┌───────────────────────────────────────────────┐   │
│ │                                               │   │
│ │                                               │   │
│ │                                               │   │
│ │            List Of Selected Subjects          │   │
│ │                                               │   │
│ │                                               │   │
│ │                                               │   │
│ └───────────────────────────────────────────────┘   │
│   │    Submit Button    │                            │
│ ┌───────────────────────────────────────────────┐   │
│ │                                               │   │
│ │               Validation Result               │   │
│ │                                               │   │
│ └───────────────────────────────────────────────┘   │
│                                                     │
└─────────────────────────────────────────────────────┘
```

Data Dictionary

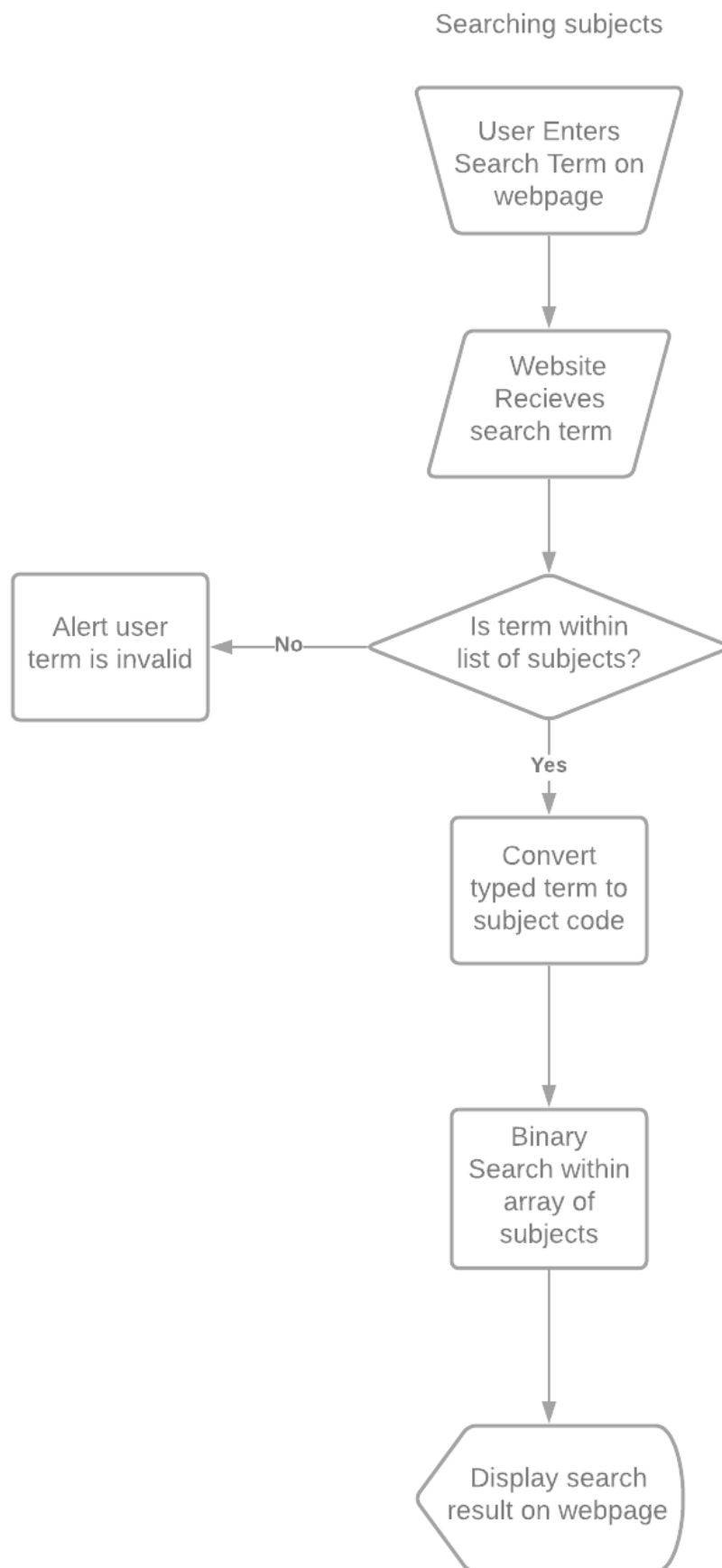| Variable Name	Data Type	Scope	Length	Description	Example
subjectsCode	Object Array	Global – search.js	40	Contains the name codes for the 40 available subjects	"enStand"
selectedSub	Array	Global – search.js	Varies 0-8	Holds up to 8 subject name codes when they are selected	"enStand"
subjectHeaders	Array	Global – search.js	13	Contains the subject header name code for usage in hiding divs	"headEn"
searchTerm	Integer from HTML element	Local – search.js	Varies	The string that the user enters into the search bar	"Advanced English"
tempTerm	String	Local – search.js	Varies	Holds a the resulting simplified version of the search term from the simpTerm function	"enAdv"
searchResult	String	Local – search.js	Varies	Holds the resulting name code from the binary Search function	"maAdv"
items	String	Local – search.js	Varies	Sorted array for the binary search algorithm	"enAdv maAdv"
value	String	Local – Search.js	Varies	Search term for the binary search algorithm	"enAdv"
middle	Integer	Local – Search.js	Varies	Used to calculate the middle during the binary search	"4"

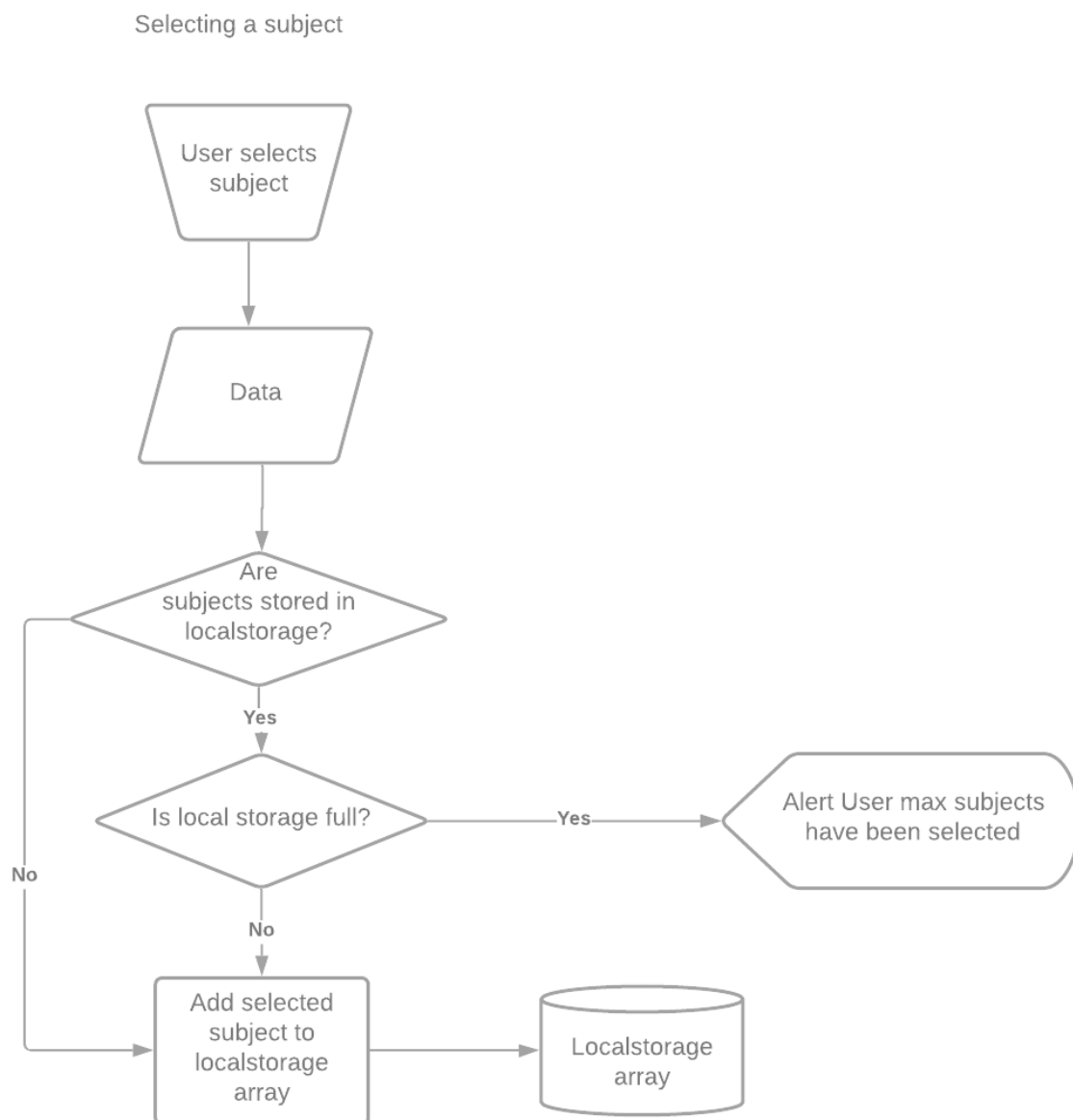| term	String	Local – Search.js	Varies	Hold the search term when used in the simpTerm function	"english studies"
startIndex	Integer	Local – Search.js	Varies	Value used when binary searching to indicate which value of the array to begin each search	"0"
stopIndex	Integer	Local – Search.js	Varies	Value used when binary searching to indicate which value of the array to be at the end of each search	"6"
thingA	String	Local – Search.js	Varies	Value used in sort(compare) function	N/A
thingB	String	Local – Search.js	Varies	Value used in sort(compare) function	N/A
comparison	Integer	Local – Search.js	1	Value used as a result from comparing thingA and thingB	"0"
subCount	Integer	Global – main.js	Varies	Value held in localstorage that indicates how many subjects have been selected so far	"4"
subjects	Array	Global – main.js	Varies	Array containing all subjects, their unit value and if they are currently selected	"English Studies"
subName1	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits1	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName2	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits2	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName3	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits3	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName4	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"

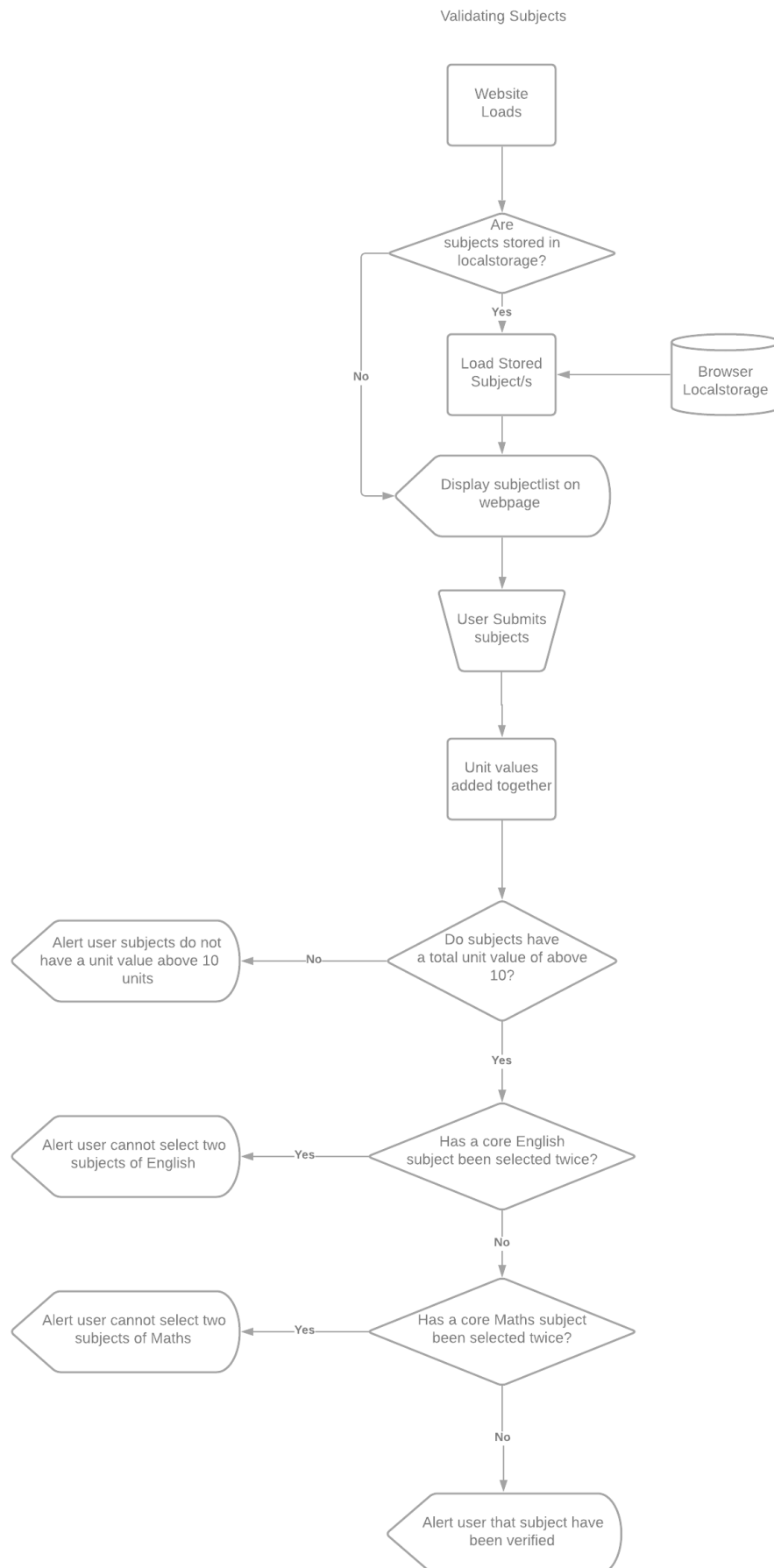|					
subUnits4	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName5	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits5	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName6	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits6	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName7	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits7	Integer	Local – main.js	Varies	Number that holds the units of a currently selected subject	"2"
subName8	String	Local – main.js	Varies	String that holds the name of a currently selected subject	"English Studies"
subUnits8	Integer	Local – main.js	1	Number that holds the units of a currently selected subject	"2"
sub1	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub2	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub3	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub4	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub5	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub6	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub7	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"
sub8	Integer	Local – main.js	1	Holds number of units in a given selected subject	"1"

subs	Array	Local - main.js	1-2	Holds sum of units in subject selection	"12"
sublist	Array	Local – main.js	0-8	Array that holds the names of the currently selected subjects	"English Studies"
subN1	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN2	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN3	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN4	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN5	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN6	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN7	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subN8	String	Local – Main.js	Varies	Holds name of selected subject	"English Studies"
subNs1 - subNs8	Array	Local – Main.js	Varies	Used to hold the first word in a subject name	"English"
subNp1 - subNp8	Array	Local – Main.js	Varies	Splits a subject name up into an array of each word	"English" "Studies"
subNP	Array	Local – main.js	Varies	All subNp variables concatenated together	"English" "Studies"
subNS	Array	Local – main.js	Varies	Array of all subNs variables	"English"

System Flowcharts:

Clearing Selected Subjects

Searching subjects

User Enters
Search Term on
webpage

Website
Recieves
search term

Alert user
term is invalid   ←No—   Is term within
list of subjects?

Yes

Convert
typed term to
subject code

Binary
Search within
array of
subjects

Display search
result on webpage

Selecting a subject

User selects subject

Data

Are subjects stored in localstorage?

Yes

Is local storage full? — Yes → Alert User max subjects have been selected

No

No

Add selected subject to localstorage array → Localstorage array

Validating Subjects

```
                        ┌─────────────┐
                        │   Website   │
                        │    Loads    │
                        └──────┬──────┘
                               │
                               ▼
                          ╱─────────╲
                         ╱    Are    ╲
              ┌─────────  subjects stored in
              │          ╲ localstorage? ╱
              │           ╲─────────╱
              │               │ Yes
              │               ▼
              │          ┌─────────────┐      ┌──────────────┐
              │          │ Load Stored │◄─────│   Browser    │
            No│          │  Subject/s  │      │ Localstorage │
              │          └──────┬──────┘      └──────────────┘
              │                 │
              │                 ▼
              │          ╱─────────────╲
              └────────► │ Display subjectlist on │
                         ╲    webpage    ╱
                          ╲─────────────╱
                               │
                               ▼
                          ╱─────────╲
                          │  User Submits │
                          │   subjects    │
                          ╲─────────╱
                               │
                               ▼
                         ┌─────────────┐
                         │ Unit values │
                         │ added together │
                         └──────┬──────┘
                                │
                                ▼
                           ╱─────────╲
   ╱──────────────╲       ╱ Do subjects have ╲
  │ Alert user subjects do not │◄──No── ╲ a total unit value of above ╱
  │ have a unit value above 10 │        ╲      10?     ╱
  │       units       │         ╲─────────╱
   ╲──────────────╱               │ Yes
                                  ▼
   ╱──────────────╲          ╱─────────╲
  │ Alert user cannot select two │◄──Yes── ╲ Has a core English ╲
  │    subjects of English │         ╲ subject been selected twice? ╱
   ╲──────────────╱           ╲─────────╱
                                  │ No
                                  ▼
   ╱──────────────╲          ╱─────────╲
  │ Alert user cannot select two │◄──Yes── ╲ Has a core Maths subject ╲
  │    subjects of Maths  │         ╲ been selected twice? ╱
   ╲──────────────╱           ╲─────────╱
                                  │ No
                                  ▼
                           ╱─────────────╲
                          │ Alert user that subject have │
                          │      been verified      │
                           ╲─────────────╱
```
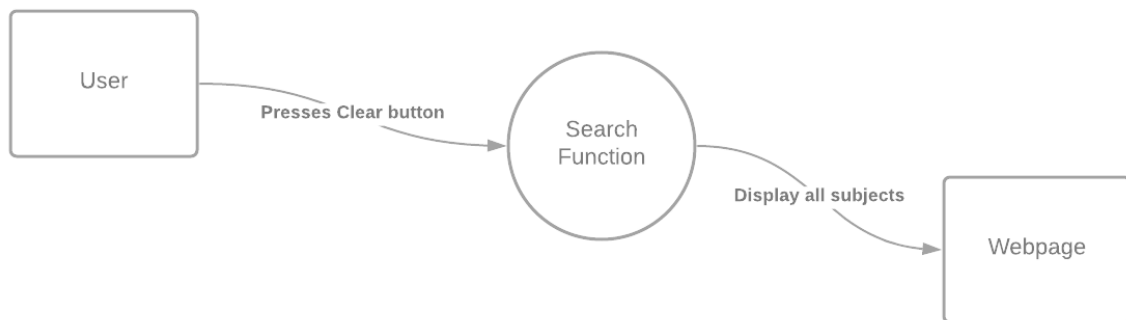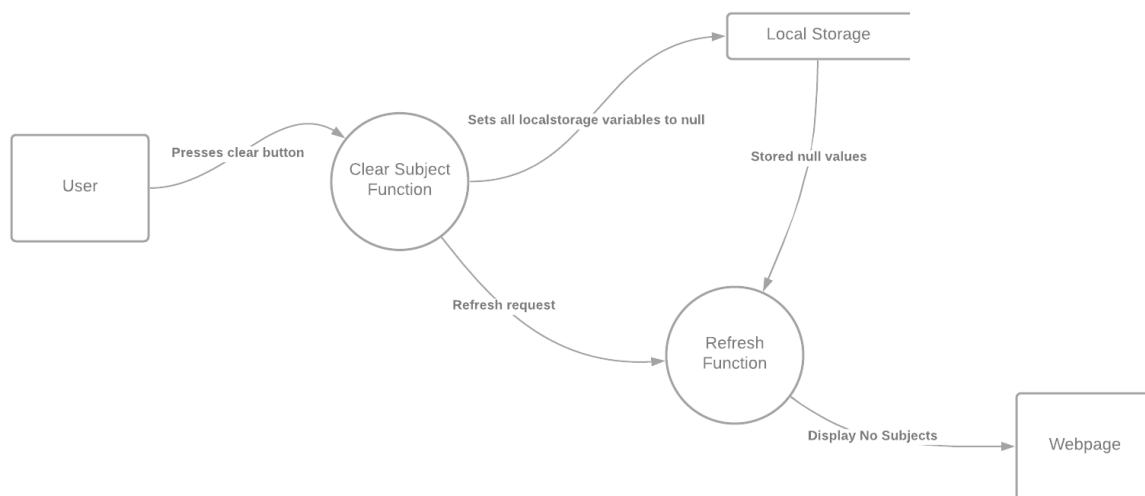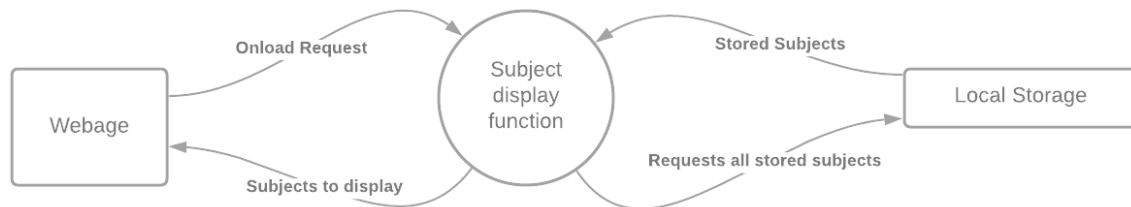
Data Flow Diagrams
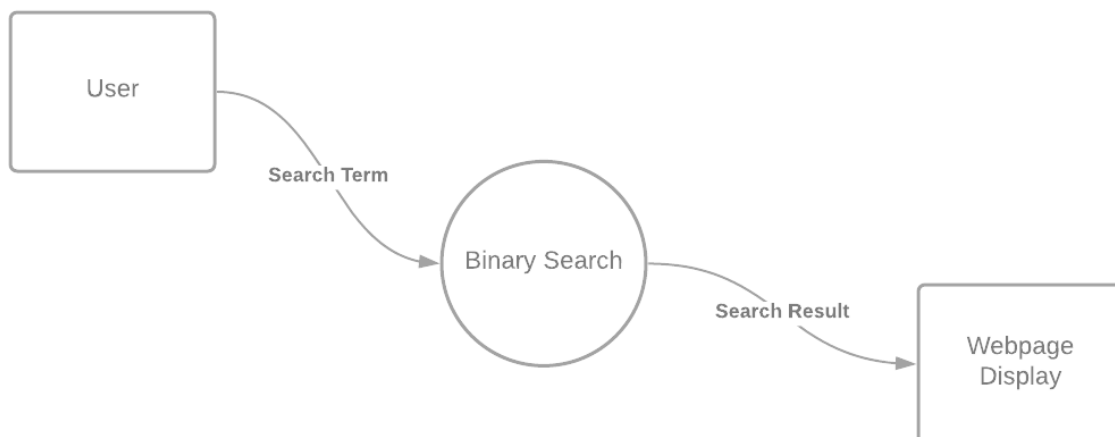
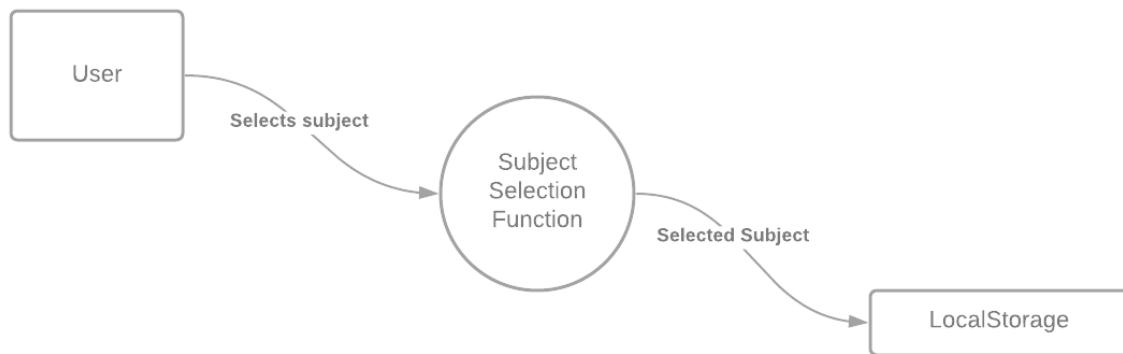### Clearing Search Results

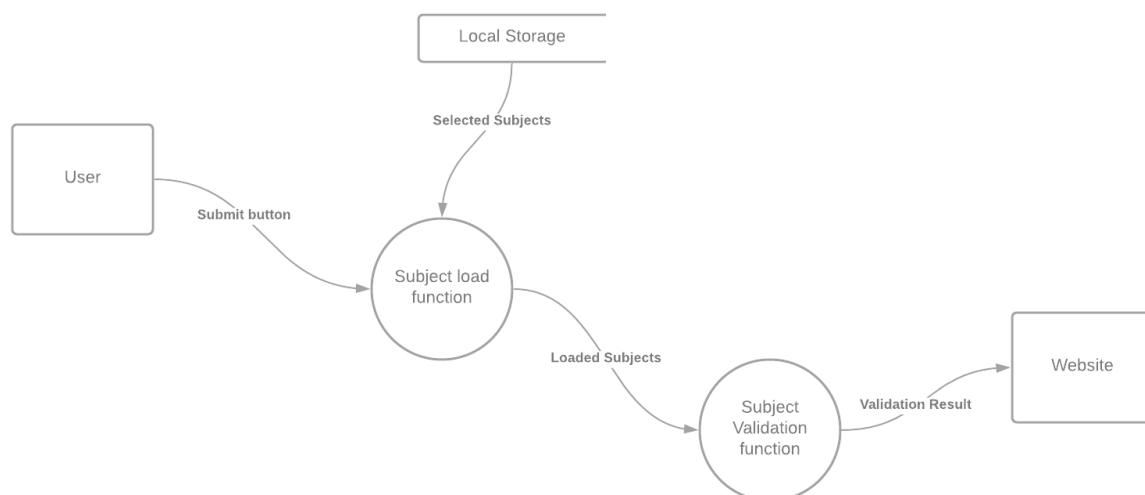### Clearing Subject Selection

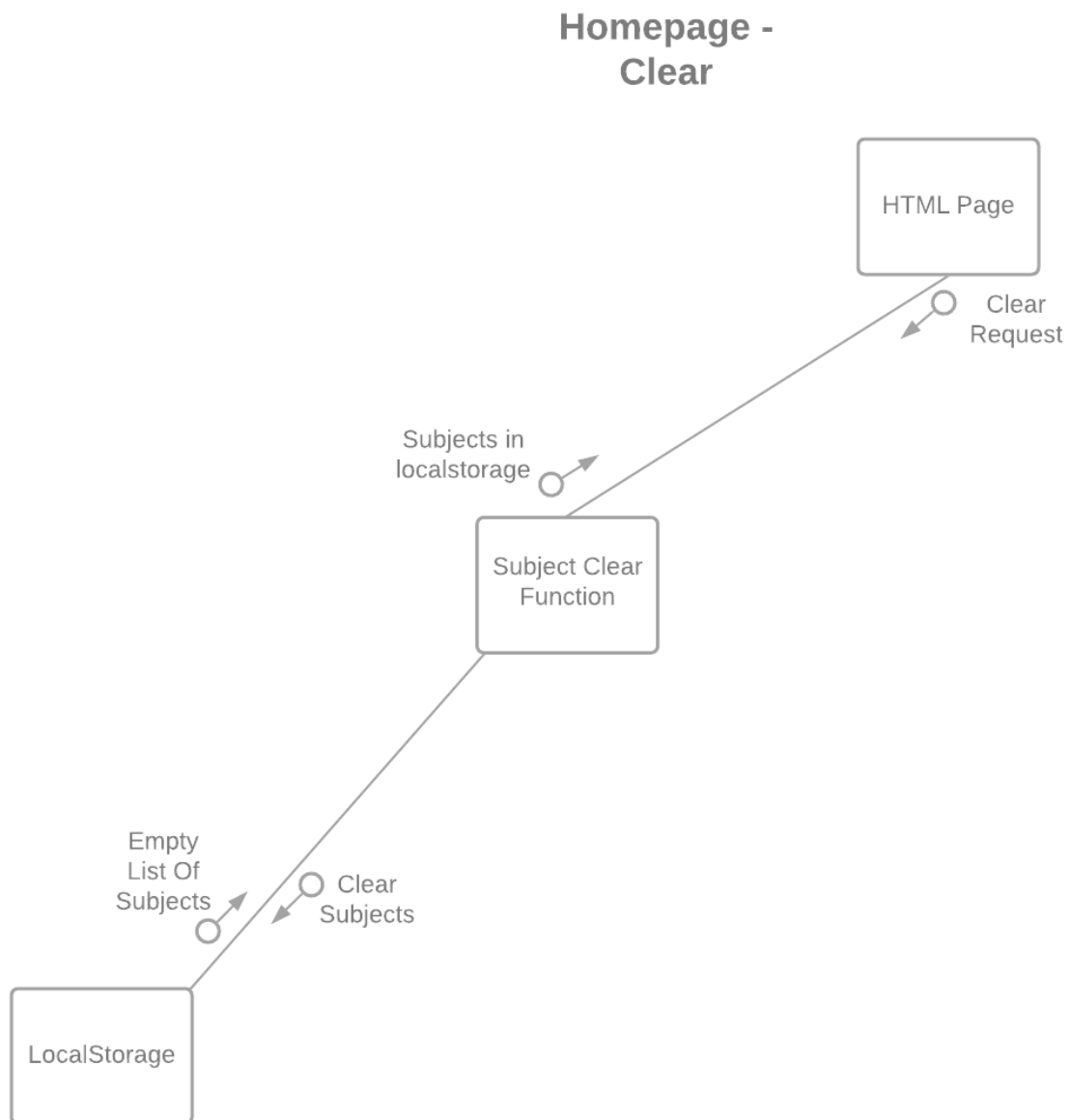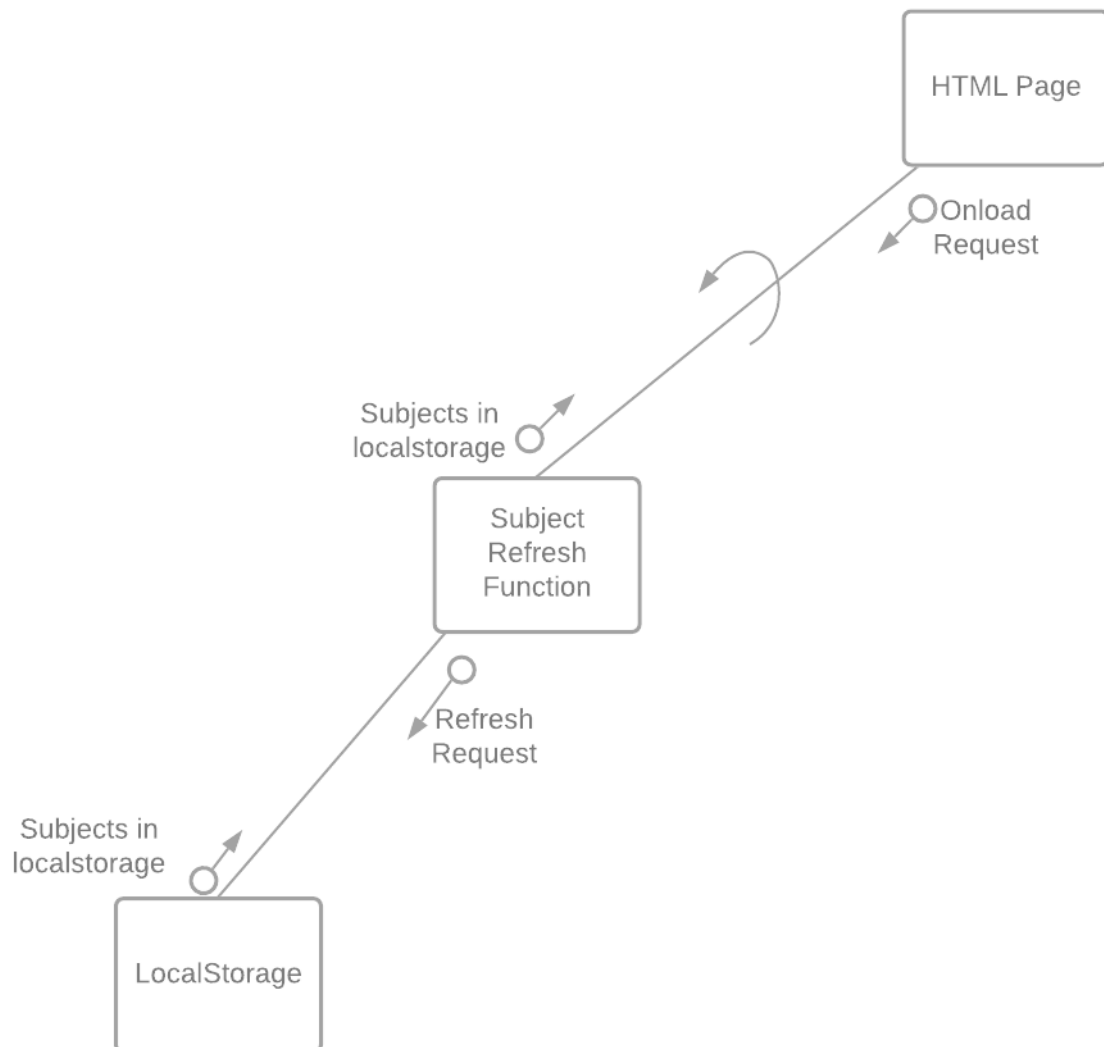**Refreshing Subjects**

Searching subjects

Selecting Subjects

User → **Selects subject** → Subject Selection Function → **Selected Subject** → LocalStorage

Validating Subjects

Local Storage → **Selected Subjects** → Subject load function

User → **Submit button** → Subject load function → **Loaded Subjects** → Subject Validation function → **Validation Result** → Website

Structure Charts

Homepage - Clear

HTML Page

Clear Request

Subjects in localstorage

Subject Clear Function

Empty List Of Subjects

Clear Subjects

LocalStorage

Homepage -
Refresh

Search and Selecting Page
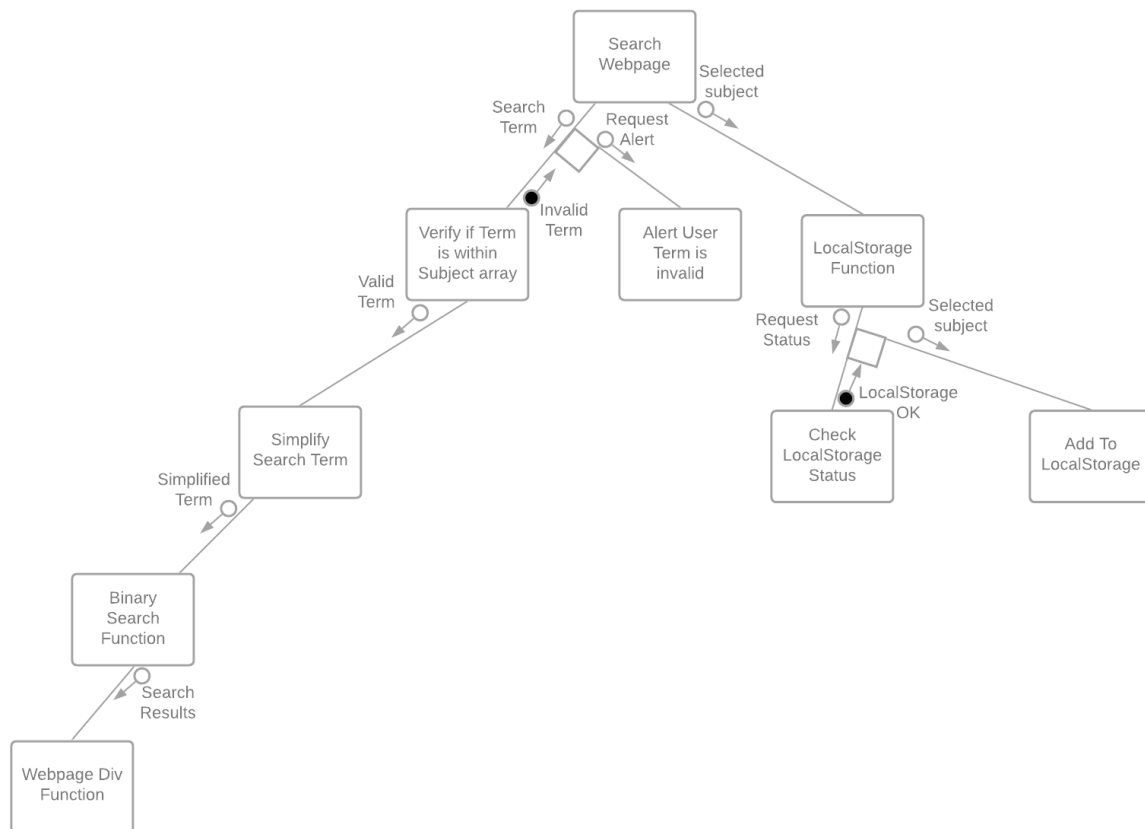
Validation Page
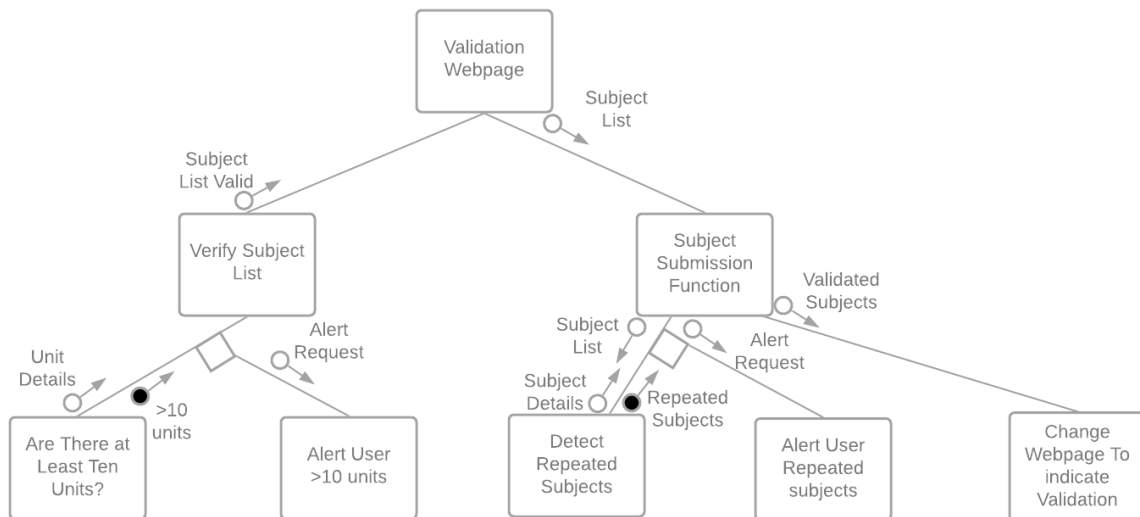
Evaluating The Solution

The solution was able to display Local Storage of the selected subjects in the browser and recall them throughout the three pages on the site. The binary search function was successfully implemented for a given term. Each subject could be added to an array which was accessible across the three pages via the usage of buttons. The validation page effectively evaluated the selected list of subjects dependant on the total unit value and overlapping subjects like English and Mathematics. However, the function that allowed sorting of subjects on the search webpage proved to be extremely difficult to develop due to the method that allowed the hiding of subjects in the binary search, this ended up in the feature to be scrapped after the binary search had been completed. The system to hide other units of subjects like English and Maths once one was selected also proved difficult to develop due to this same reason, being instead satisfied by the validation page's functions, alerting the user when they submit that the error has occurred.

Overall, the solution met the majority of the proposed goals, allowing for the main features of subject selection and validation to occur, with effective use of local storage and binary search, where only few features were unable to be implemented due to constraints of the solution's design.

User Manual

Features

Home Page

- Displays Currently selected subjects at base of page
- Features buttons to load/refresh and clear your subject selection from browser memory

Search Page

- Search Bar for searching the list of subjects
- Buttons for resetting the search and clearing selected subjects from memory
- Buttons on each subject that will select them when pressed

Validation Page

- Button to refresh subject selection from browser memory
- Button for submitting your subject list
  - o Will warrant a response from the website as to if your selection is valid or not

How to use the site

1. When first loading the site navigate to the search page to select your subjects, you may enter the name of a subject to filter out all other subjects
   a. Key terms will not work due to the nature of the binary search, the name of a single subject must be entered
2. Once up to 8 subject have been selected navigate back to the home page to review your subjects
3. If you are confident with the selection navigate to the validation page, you may now submit your subjects to receive a verdict on the selection.