Alt
School

Semester-1 Holiday Challenge

What

Solidify what you've already learnt by building *simple* things you can showcase

Why

Practice makes perfect. We're getting you ready for all the fun projects in the coming semester

How

1. Build any one of the below projects as a group / team, using Repl.it
2. Vanilla HTML/CSS and Javascript is sufficient to build these projects as fully functional apps, so don't use any fancy libraries or frameworks. Doing so can cost you some points!
3. It's a healthy competition amongst teams, so try your best to make your application functional, clean and presentable

Getting Setup for The Challenge

1. Setup is to be completed by the team Rep or anyone nominated by the team. Other team members are to collaborate and help make sure the steps are correctly carried out. We recommend doing the setup as a team (e.g on a Zoom or Google Meet call). Team members need to have a GitHub and Repl.it account
2. Follow this visual guide.
3. Edit and update the app.properties file in the project. Make sure to correctly add the needed details in there. They include:
    a. Name of the team
    b. name/email of the team rep
    c. Names/email of the team members
2. As a team, decide on which of the listed projects to work on. They are all simple and cool, so be ready to have some fun.
3. If for any reason, the team rep, project or team members change make sure to UPDATE the details in the app.properties file
4. The portion of each project tagged as **Advanced features** are totally optional. Feel free to attempt it if you have completed the main requirements and want to go over and beyond. Also, if there are team members who are more experienced and feel the

project/features are too simple, pls encourage them to tackle the advanced features while the rest of the team focus on the main features.

5. Go make magic!!

Projects

As a team, decide which of the below projects you are excited about, then go ahead and build it.

Make sure to *always* update the `app.properties` file within the project codebase to reflect the choice of project as well as details of the team Rep and all the members working to build it

01 - Phonie: Identify the telecoms carrier from a phone number

Scenario

A user is filling a form in your web application and they are required to enter their phone number into a form field.

Outcome

After typing the phone number into the form field, the UI indicates that it is an MTN or GLO or Airtel or Etisalat number e.t.c (e.g if its an MTN line, display the MTN icon/logo somewhere beside the form field)

Advanced features (OPTIONAL- build any, some or all if you feel up to it)

- Add validation - Use the `pattern` attribute of the HTML form field to restrict phone numbers to a certain carrier, e.g restrict to only Airtel such that entering an MTN/GLO number would be invalid
- Support +XYZ country codes (e.g +234 for Nigeria) - Still detect the carrier even if the user prefixed the number with their +XYZ country code
- Show auto-completion - Show suggestions as the user starts typing the phone number. E.g once they type 080, show them that 080*3* and 080*6* and other possible *matching* variants of the carrier

02- Redactr - Scramble some words from a body of text content

Scenario

Your app is to help people safely share content on social media by scrambling portions of the text they don't want the world to read/see. A user will write/paste the some content into a form field and then specify the words they need to be scrambled (using another field). Note the user is allowed to specify more than one word (separated by space) to be scrambled.

Outcome

After clicking a “Redact Now” button, the app scans through the original text and replaces the specified words to be scrambled with asterisks (i.e ***)

Advanced features (OPTIONAL- build any, some or all if you feel up to it)

1. Allow the user specify what to use in replacing the words, so that the app can do the scrambling with what the user provides, be it asterisks, questions marks, dashes, underscores, or even other words
2. Display some stats like how many words were scanned, how many matched the words to be scrambled, how many characters were scrambled in total, how long it took (in seconds) to complete the scrambling task e.t.c. Feel free to be creative here

03 - SchoolStat - A simple dashboard for a tech school

Scenario

You have data about which instructors are taking the courses in a school, as well as data about which students are in the frontend, backend or cloud tracks. Your users need to see some basic stats about the school

Outcome:

When the user opens your dashboard app, they see HTML tables showing the following statistics/summaries

- How many courses a given instructor is taking
- How many students are in each of the tracks (frontend, backend and cloud)
- How many students are male, female or other genders/orientation
- Minimum, maximum, and average age of students in the school

Advanced features (OPTIONAL- build any, some or all if you feel up to it)

- Make the tables sortable by (at least) the column holding the numbers. Indicate what columns are sortable (e.g with an icon or by adding the <> characters in the column header e.t.c). When the user clicks on a sortable column header, sort the records in the table by that column (e.g switch the current sort order such that if the column was already sorted ASC, sort in DESC)