

# Themes for Hack The Web

## Basic Track Themes -

### 1. Create a Quiz game website:

- Students will be judged on the basis of their design
- Code Factoring
- Creativity in taking responses from user
- User involvement
- Number of useful and intuitive features
- Responsive design

### 2. Create a Personal Website:

- Students will be judged on the basis of their design
- Code Factoring
- User readability and clarity for sharing information
- Intuitive and creative way of information sharing
- Responsive design

### 3. Club website Landing page:

- You can design a landing page for your favourite club.
- The landing page may contain the About Us, Featured Events, Contact Us, etc. sections.
- Students will be judged on the basis of their design
- Code Factoring
- User readability and clarity for sharing information
- Intuitive and creative way of information sharing
- Features available on website(all feature don't need to be working like login form should be present but it will be fine if it doesn't work)
- Responsive design

### 4. Arcade Game Website:

- Students will be judged on the basis of their design
- Code Factoring
- Smooth working of site and Game

- User involvement
- Number of useful and intuitive features
- Responsive design

## 5. Startup or Product Landing Page:

- Students will be judged on the basis of their design
- Code Factoring
- Focus on presenting The product.
- Use creative ways of presenting the product features.
- Responsive design

## 6. Showcasing a story through a static page:

- Students will be judged on the basis of their design
- Code Factoring
- Creativity and focus on story.
- User interaction if possible.

# Intermediate Track Themes -

## 1. Weather Forecasting using a Weather API.

- Use any available weather forecasting API.
- Students will be judged on the basis of their design
- Code Factoring
- Responsive design
- Use creative ways of displaying information.

## 2. Music player using spotify's API

- Search and Use spotify's music API
- Students will be judged on the basis of their design
- Code Factoring
- Various features like filters based on artists etc.
- Try to add search functionality.

## 3. Codeforce rank predictor or analytics app using codeforces API

- Students will be judged on the basis of their design
- Code Factoring
- Responsive design
- Use Codeforces official API to create a rank predictor or analytics app
- Add features for using certain time frames for analytics.
- Use various graphs to display various information.
- Try to add comparison features etc.

## 4. Bulk SMS sender APP

- Use [Bulk SMS sender](#) API
- Students will be judged on the basis of their design
- Code Factoring
- Responsive Design

## 5. Calendar and To-do list app integrated with google calendar

- Use [Google Calendar API](#) to fetch resources and integrate it into your website.
- To-do list should display information about events in a day as fetched from google's API.
- Students will be judged on the basis of their design.
- Code Factoring.
- Creative ways of presenting info.
- Responsive design.

## 6. Online Messaging app using 3rd party API.

- Use [Telegram's messaging API](#) to create a messaging app
- Add various features like notification alert
- Try to add emoji feature in your app
- Students will be judged on the basis of their design
- Code Factoring

# Advanced Track Themes -

## 1. Use WebGL to create a landing page or any creative website:

- Students will be judged on the basis of their design
- Creativity is the main judging criteria for this project
- Use of proper designing techniques.
- You can use this [youtube video](#) for reference.

## 2. Create a client side app for event management:

- Displaying past and upcoming events.
- Displaying event details on clicking over an event.
- Add a new event.
- Register for an event.

## 3. Quizzing app

- Should have multiple topics of quizzing.
- Adding new topics and questions to that topic.
- Taking a quiz and getting the scores.
- Leaderboard

## 4. Stock management app

- List all the stocks available with a club.
- Add a new stock item (take input of stock item, quantity, cost per item, date and who added it)
- Use a stock item (stock item, quantity, date, who took it for use)
- Borrow a stock item (stock item, who borrowed, quantity, date and due date to return)

## 5. Code reader and editor

- Take images as input from users and use node packages to convert these images into text files.
- Users should be able to edit these text files in your web application.
- Try to implement various helpful features.
- Users should be able to download or save these files in application.