# ­­SINGAPORE POLYTECHNIC

**SCHOOL OF COMPUTING**

Specialist Diploma in Full Stack Web Development (NSFSW)

|  |  |  |
| --- | --- | --- |
| Module Code | : | IT8906 |
|  |  |  |
| Module Name | : | WEB AND MOBILE DEVELOPMENT USING FRAMEWORKS |
|  |  |  |
| Acad Year/Semester | : | 2022/2023 Semester 2 |
|  |  |  |
| Set by | : | Jeremiah Ang |
|  |  |  |
| Assignment Title | : | CA1 – React Core Concepts |
|  |  |  |
| Assignment Type | : | Individual |
|  |  |  |
| Weighting | : | 30% |
|  |  |  |
| Deadline | : | End Week 11 (1st Jan 2023, 2359) |
| Deliverables | : | Application files (html, js, css, json etc) |
| Mode of Submission | : | Pushed to GitHub main branch |

WARNINGS:

Plagiarism means passing off as one's own the ideas, works, writings, etc., which belong to another person. In accordance with this definition, you are committing plagiarism if you copy the work of another person and turning it in as your own, even if you would have the permission of that person. Plagiarism is a serious offence and disciplinary action will be taken against you. If you are guilty of plagiarism, you may fail all modules in the semester, or even be liable for expulsion.

50% of marks to be deducted for submission of assignment within ONE calendar day after the deadline. No marks shall be awarded for assignments submitted more than one day after the deadline.

**CA1 – React Core Concepts**

Students will create a simple website to display their competency in understanding React Core Concepts.

**Assignment Objectives**

1. Injecting React Elements into a Webpage
2. Creating reusable components using Props
3. Create and maintain component states and lifecycle events
4. Adding interaction using Events, conditional rendering, and list rendering.
5. Handling form states
6. Styling React elements

**Assignment Instructions**

* You are to work on this project individually.
* The website you build will be a Movie Management System where admin will go in to manage the movies in their system
  1. Assume only admin have access to the system.
* Your website can be a single page application or multi-page application, but should contain the following features:
  1. Create Movie
     1. Admin can fill up a form to add a new movie to the list
  2. Retrieve Movie
     1. Admin can see a list of movies
     2. Admin can toggle hide/show past movies
     3. Admin can filter movies based on genre
  3. Delete Movie
     1. Admin can remove individual movies from the list
     2. Admin can select multiple movies to be deleted together
* Connecting to backend server/database not required
* Persistent storage is not required, the website should always start with a fix set of movies.

**Marking Scheme**

|  |  |
| --- | --- |
| JSX   * Defining elements using JSX * Embedding JS expressions in JSX | 5% |
| Component   * Creating reusable components (e.g., as rows in table) * Breaking down website into smaller components | 10% |
| States   * Using states in components (e.g., remembering hide/show option) * Lifting states up (e.g., selected items) | 10% |
| Events   * Detecting events to update states (e.g., selected movies, form submission) | 10% |
| Conditional Rendering   * Using of Conditional AND or Ternary Operator to control what is displayed (e.g., hide/show) | 10% |
| List Rendering   * Using Loops or Array functions to create list of React Elements/Components (e.g., rows in table) | 10% |
| Forms   * Tracking form’s state (e.g., Adding new row) | 10% |
| Styling | 5% |
| Conformance to required features | 20% |
| Code quality   * Well and neatly indented * Sensible variable and component names * Sensible filenames | 10% |
| **Total** | **100%** |