

CSCI 2720 – Building Web Applications

Course Project

Released: 7th November 2023

Project outline deadline: 23:59, 15th November 2023

Full submission deadline: 23:59, 15th December 2023

Synopsis:

In a group of 5 students, you are going to set up a web app to check information on some locations. Users and Admin will be able to log in and perform certain actions. Your project app will retrieve location details from an open dataset.

Data source:

A suggested dataset for this project:

Cultural Programmes <https://data.gov.hk/en-data/dataset/hk-lcsd-event-event-cultural>

You can choose the database that interests you. The only restriction is that it must be publicly available real-world location data. If you identify a comparable dataset from another source, you may ask Dr Colin Tsang (colintsang@cuhk.edu.hk) for permission.

System requirements:

Your project will be graded with the following:

- Google Chrome (almost latest version)
- Node v20.9.0 + npm 8.19.2
- MongoDB server 7.0.2

The backend use should be Node. The fronted platform is not restricted for this project. Yet, your project app must be a **Single Page Application**, without refreshing the page for any internal links. However, visits to all different views should be reserved in the browser history, with a proper URL.

You are allowed to utilize additional methods beyond what has been covered in our lectures and labs. However, please note that you cannot expect our TA to use any external software or tools to run your files. During the grading process, we will only use the above environment.

The data:

You need to perform pre-processing to the dataset for the local storage and display:

Dataset: Cultural Programmes

- Mainly consider the “Programme information” dataset
- Data is available as XML
- Pick only 10 venues to be shown in your app (where each should host at least 3 events)
- Handle the following data: title, venue, date/time, description, presenter, price.

You need to *get the real time information from API to database only once when the user logs in and loads your page*. Show the last updated time clearly. No auto-update is required after that.

You need to design the data schemas and models storing (caching) items. For the locations, you are required to maintain at least:

- Location name
- Latitude and longitude

Only English data is required for the project app. For the schema and models for users and other data, you may design freely to suit your needs.

You may need to consult data dictionaries and related data location details, and you can feel free to use extra APIs for your app. *Never use anything more than Free Tier.*

Access modes:

Your app will provide two modes of access:

1. **Users** – only authenticated users have access to the app’s contents. A user is recognized using a username and password pair. The user will be able to perform the “user actions”, which are specified on the next page.
2. **Admins** – admins will be able to perform arbitrary CRUD actions to the location data and the user data on your database.

Application requirements:

User actions:

1. List all locations in a table as links to single locations and allow sorting of the table with the number of events at venue.
2. Show all locations in a map, with links to each single location (suggested APIs: Google Maps or MapBox).
3. Search for locations which contain keywords in the name which will result in a table of location results.
4. A separate view for one single location, containing:
 - a. A map showing the location.
 - b. The location details.
 - c. User comments, where users can add new comments seen by all other users.
5. Add location into a list of user's favourite locations and see the list in another view.
6. Show events whose price under a specific number. (e.g., ≤ 100)
7. See the username in the top-right of screen and be able to log out.
8. Try to create at least one more feature in your app.

Admin actions:

1. CRUD stored event details in the local database.
 - a. We will not test other features (e.g., map, comments) if deleting an existing location.
2. CRUD user data (username and password only) in the local database.
 - a. We will not test other features (e.g., comments) if deleting an existing user.
3. Log out as admin.

Non-user actions:

1. Log in as user with username and password.
2. Log in as admin using username and password.

Project report:

You need to submit a report to describe your project, with reference to the **CSCI2720 course materials**, and other online references. Anything you consider important should be included in this report. Here are suggested components for your report:

1. *Abstract:*

- A summary of your work in no more than 100 words, with one screenshot of a representative screen of your site.

2. *Methodologies:*

- List the files submitted to Blackboard with short descriptions.
- Discussion on the pre-processing of the dataset.
- Discussion on all actions of your app, including the new features designed by you. E.g., explain if you have applied any restrictions on username and password.
- Discussion on the programming languages and important algorithms you have used.
- Design of data schemas and models of your database. Figures are suggested.
- Description of all libraries/frameworks used, other than HTML/CSS/JS. (natural language is better, but some core codes are acceptable)
- A comparison table of at least two advantages and two disadvantages (specific to your project app) of your chosen platform and technologies comparing to others.

3. *References:*

- Citation of all materials which are not originally written by you, including teaching materials in and out of our course.
- You must use the IEE style properly: https://www.ieee.org/content/dam/ieee-org/ieee/web/org/conferences/style_references_manual.pdf

4. *Appendix*

- A workload distribution. We allow more than one person to do a job together, but each member should have a unique workload.
- Anything you consider supplementary.

You are recommended to include more figures for this report. Please **use 4-7 pages for the report, with 11pt 1.5 line spacing**. Penalties will be applied for anything out of the allowable range.

Assessment:

Your project will be graded by:

1. Technical requirements – fulfilment and complexity (50%)
2. Usability – look and feel (20%)
This includes whether a smooth SPA experience is provided, with responsiveness in layout and clarity of text/colour presentation.
3. Project report (30%)

The project is designed to resemble an examination. Therefore, we will only clarify the specifications if they are unclear. To maintain fairness, we will not offer additional assistance or guidance. If you encounter difficulties, we encourage you to review the lectures or discuss them with your groupmates.

You will decide the complexity and aesthetics of your work. Make sure it is clear and useable by any users (e.g., your TAs who will grade this project) without much guesswork. You can decide on anything not specified in this document and extend beyond the basic requirements. You may freely decide the choice of technologies and frameworks to be used in this project.

The grading will be done on a normal desktop/laptop computer using Google Chrome and an environment specified on the first page of this document.

Submission:

Project outline (23:59, 15th November 2023):

You should submit a one-page document including the following:

1. Group members list (max 5 students), with full name and student number.
2. Data source (provide details if you propose a new dataset).
3. Anything you consider relevant.

Your group should select one student as a representative to submit the project outline to Blackboard.

If you couldn't find enough groupmates, or you don't have any groupmates at all, you should still submit a project outline. We will randomly form the ungrouped students into groups of 5. Please welcome new groupmates assigned by us. We will announce the finalized groups on Blackboard as soon as possible.

Full submission (23:59, 15th December 2023):

Include full names and student IDs of all members in all code files using comments. Zip all your files into a single zip file:

GROUP_[your group number].zip

Once again, select one student as a representative to submit it to Blackboard.

You do not need to submit the *node_modules* folder, but please keep the files *package.json* and *package-lock.json*.

Submit also a *readme.txt* (inside the zip file) to state the project server start commands, as well as your site URL. Inside the file, indicate clearly whether you have read this article carefully: <https://www.cuhk.edu.hk/policy/academichonesty/> and include the required declaration.