# HTTP5222 Project (APIs)

Create an Express web app which **consumes at least two APIs**. This should be a website with an interface (i.e. like a typical website and **NOT** building an API). Using an open data source in addition to these would be even better. This is your chance to use your creativity and really see what you can do with APIs. This project can be a good portfolio piece.

**First steps:** Take a look at available APIs and see if you can come up with cohesive concept which uses at least two. This means that your website should not be random API integrations. Instead, try to see if you can make a useful website and use the APIs to supplement your code for better functionality.

**SDKs:** For this project, if the API you want to use has an SDK (Software Development Kit)—in other words, a code library—you are allowed to use it.

Start thinking of your concept now. There will be a **checkpoint (5%) for you to submit a short description listing the APIs you plan to use and what you plan to create**. I'll give you a mark for your idea so that you can get an idea as to whether or not it needs tweaking.

For the final submission (20%), you will be marked:

* **API integration (5)**
* **Code quality (5)**
* **Design (5)**
* **Creativity/Cohesiveness (5)**

You can submit by either:

* zipping up files (minus the ***node\_modules*** folder) and submit via Blackboard
* **OR** push to GitHub and submit the link on Blackboard

**Items to include (to ensure I can run the code):**

* A README with any information I may need (e.g. login, etc.)
* **(OPTIONAL)** You can also develop and upload the code to your host so I can reliably test working code. It's also a good idea to start getting some projects online now if you don't have them up yet.

Due dates:

Checkpoint (5%) – **March 12, 2025 at 11:59pm**Final Submission (20%) – **April 2, 2025 at 11:59pm**

| **Criteria** | **Excellent**  **5** | **Good**  **4 – 5 pts** | **Satisfactory**  **3 – 4 pts** | **Somewhat Unsatisfactory**  **2 – 3 pts** | **Unsatisfactory**  **0 – 2 pts** |
| --- | --- | --- | --- | --- | --- |
| **Creativity/**  **Cohesiveness** | Interesting and original concept. APIs/data sources make sense together as a concept. | Concept not new but there's a twist; shows a lot of thought. APIs/data sources mostly make sense but a few minor issues. | Concept not new, but it's good. APIs/data sources mostly make sense together but at one major issue. | Concept not very new but an attempt. APIs/data sources somewhat make sense together but idea feels a little disjointed. | Concept/ideas are not new. APIs/data source are used randomly. There is no cohesive underlying reason for them to be tied together. |
| **API Integrations** | Shows excellent understanding of APIs. Integrations work without issue. | Shows good understanding of APIs. Integrations mostly work well but one or two minor issues. | Shows some knowledge of APIs. Integrations mostly work but one or two major issues. | Shows small knowledge of APIs. Integrations don't always work and there are a few major issues. | Shows little to no understanding of APIs. Integrations do not work or have many major issues. |
| **Code Quality** | Code follows best practices, is neat, organized, and efficient. | Code mostly follows best practices, is neat, organized, but not always efficient. | A good attempt at following best practices but some issues. | An attempt at following best practices but many issues. | Code is very problematic. No organization, formatting, and code is very inefficient. |
| **Design** | Professional design, good colours and whitespace. | Mostly professional design with a few minor issues. | Okay design but some things make it feel like student work. | An attempt but is clearly student work. | Design has major issues such as clashing colours, fonts and inconsistent layout. The design doesn't work. |