

ENSF 607 - Web-Development Project

Deliverables	Demo Day
Pre-Project Exercise on REACT and Django	Friday Dec 11 During the lab
<p>Milestone 1: As described in the project description below, you will implement a web application that performs the functionalities of the attached excel template. In this milestone, you will implement a React application with the following functionalities of the excel file:</p> <ul style="list-style-type: none">• Course Information• Learning Outcomes• Final Grade Determination <p>Note that certain functionalities such as the grade breakdown, requires the user to choose many grade components. Therefore, your React application should support dynamic adding and removing of UI elements. You will create the database and back-end in the second milestone.</p>	Friday Dec 18
<p>Milestone 2: In this milestone, you will create a Django REST API that creates the models in the database based on the information it receives from the React app. In addition, you will create a new React page that displays the last information saved in the database.</p>	Friday Jan 15

Pre-Project Exercise on Django

Create a REST API with the following end point: <http://localhost:8000/api/echos/>.

GET/api/echos

Should list the link to all the echos

POST/api/echos

body {"message": "a string"}

Should create an echo with the above information

GET/api/echos/number (e.g GET/api/echos/1)

Should show the echo information in json. (e.g {"message": "a string"})

PUT/api/echos/number

body {"message": "another string"}

Should change the content of an existing echo

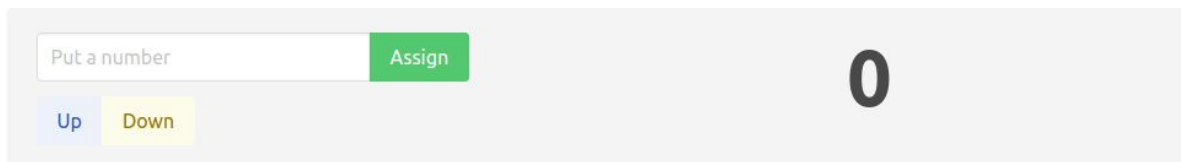
DELETE/api/echos/number

Should delete the content of an existing echo

Pre-Project Exercise on REACT

In this project, you will build a simple Counter application with React. The *gif* below shows how the app should work. There are two buttons: Up and Down, which increments and decrements the counter value, respectively. There is also a text input where you can put a number and assign it to the counter. To assign a new value, you can either press Enter when the input box is focused, or click on the Assign button.

You can design your own user interface, and it doesn't have to be like the design shown here. Although UI design is not graded for this assignment, there are bonus points for nicely done UIs.



Project Documentation:

Roles: Instructor

Introduction

You are asked to create a web application that assists professors with creating their course outlines by providing a convenient and user-friendly interface.

Currently, this task is completed using an excel template (see the excel template on D2L) which allows a professor to input their data and then create a pdf.

Here are the requirements for the entire system:

The system will have 2 roles

- Instructor
- Admin (The development of the admin is not part of this project)

The instructor dashboard would have the following features:

- A list of their past course outlines
 - When an instructor clicks a past course, they should see all their previous data in that course outline
- The functionality that allows them to create a new course outline

The webpage for creating a course outline must have several sections including, Course name and number, course calendar description, etc. Please consult the excel template for the complete list of sections and purpose of each section.

The instructors should be able to download their course outline as a pdf after they are done.

Please note the following about data in the course outline builder page:

- The grey fields in the excel template are not to be included in the final pdf. They are to be saved in the database only. This data will be useful for the admin dashboard (not included in this project scope) and will be used to analyze data for all courses (used for accreditation of the engineering program!)

Technology Specifications

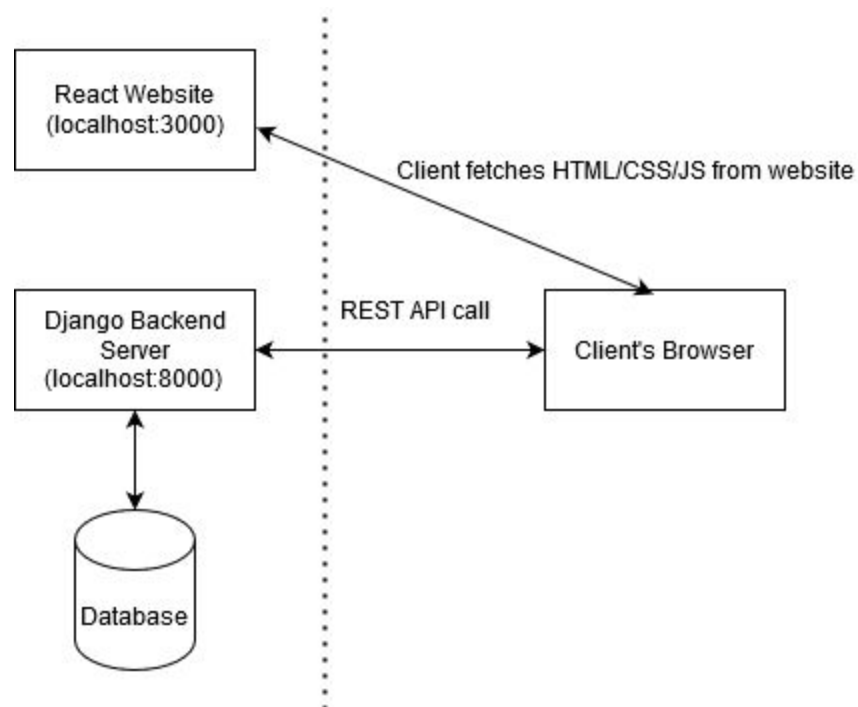
In this project you will implement the frontend using React and the backend using Django.

The frontend communicates with the backend using REST API calls. The backend uses SQLite database to store data.

For bonus points, you can deploy the application to Amazon AWS. If you deploy it to AWS, you will need to use MySQL instead of SQLite.

Interaction between Django and React:

When the user navigates to the website URL (localhost:3000), the client fetches HTML, CSS and JS from the React website, after that all the code runs on the client side. When the client sends a REST API call to the Django server, it performs some processing, queries the database and sends the response back to the client to show in the browser.



- Please explain how they will have to develop the frontend in Django, the backEnd in React, and deployment on amazon will be bonus marks

Backend Implementation Requirement

Use Django version 3.8 and Django Rest Framework version 3.11. Moreover, make sure to use virtual environment to manage your project and save all your dependencies in requirements.txt file.

The Django Rest API should have rest endpoints for various sections of the data models. For example

- <http://localhost:8000/api/calendarinfo/>
- <http://localhost:8000/api/finalgrades/>

For database, you can use SQLite which is Django's default database.

Frontend Implementation Requirements

Use create-react-app to initialize your React app. Moreover, use functional components and React hooks to create components and manage the state. Finally, use Material-ui framework (<https://material-ui.com/>) to design UI components.

- For more information about functional components visit <https://reactjs.org/docs/components-and-props.html>
- For more information about React hooks and state visit <https://reactjs.org/docs/hooks-state.html>

UI/UX requirements

These are the sample design of the application, however, you have the flexibility to design the application as your desire.

Examination

Examinations

Note: The timetable for Registrar Scheduled exams can be found at the University's Enrolment Services website, <http://www.ucalgary.ca/registrar/>.

Final Grades

Component	Learning Outcome(s)	Weight
Assignment title *	<input type="checkbox"/> Learning outcome description	25
Quiz title *	<input type="checkbox"/> Learning outcome description	25
Midterm title *	<input type="checkbox"/> Learning outcome description	25
Final title *	<input type="checkbox"/> Learning outcome description	25
Total		100