

# CS Capstone Design

## Technical Demo Grading Sheet (100 pts)

### TEAM: LOST EXPRES

**Overview:** The main purpose of the “Technical Demos” is to very clearly communicate the extent to which the team has identified key challenges in the project, and has proven solutions to those challenges. Grading is based on how complete/accurate the list of challenges is, , and how convincingly and completely the given demos cover the given challenges.

This template is fleshed out by the team, approved by CS mentor, and brought to demo as a grading sheet.

### Risky technical challenges

Based on our requirements acquisition work and current understanding of the problem and envisioned solution, the following are the key technical challenges that we will need to overcome in implementing our solution:

**C1: Docker Environment** Our team has no familiarity with the Docker Environment, and this is a requirement set upon us by our client. We also need to run python script to load a website in this environment. To prove capability we need to demonstrate loading a website from Python script in a Docker environment.

**C2: Query from MySQL** A main functionality of our product is to display the graphs. To prove capability we need to demonstrate the ability to request data from MySQL and receive the intended value back.

**C3: Populate MySQL** A requirement of launching the website is to have access to a working database with recorded data points. Through this site, our Client is able to populate the MySQL database with previously documented exposures. To prove capability we need to demonstrate loading a text document which will be used to populate the database.

**C4: Graph with Plotly** A main functionality of the finished website will be to plot the data that has been requested by the user. To demonstrate this, we will demonstrate graphing a sample set of data.

**C5: Sample UI with Vue** Our team has very limited experience with Vue, which we have chosen as our main framework for the product. To prove our design capabilities in Vue, a simple UI should be designed.

**C6: Login Differences** The project requires proof of different user levels. To demonstrate this we will provide a simple login screen showing the differences between a logged in user vs. a public user.

### Challenges covered by demos:

In this section, we outline the demonstrations we have prepared, and exactly which of the challenge(s) each one of them proves a solution to.

---

#### Demonstration 1: Show Use of Docker Environment

Challenges addressed: C1 (Docker Environment)

Flight Plan: Step by step overview of demo

1. First, a docker development environment must be set up locally by the programmer.
2. Inside the docker container, there will be a sample HTML file and a simple Python script to execute.
3. A HTML file will be loaded from the docker and accessed via a web browser.
4. After the HTML page is loaded, the user will interact with a button. This button will be tied to some Python code and will be used to test if the docker container is functioning properly.
5. If the script fails to execute, then the user will be notified with an error message.
6. If the Python script runs, then this demonstrates that a docker container can run any necessary Python scripts.

Evaluation:

- ✓ Convincingly demo'd each of listed challenges?
- ✓ Other evaluative comments:

---

**Demonstration 2: Graphing from Database**

Challenges addressed: C2 (Query from MySQL), C3 (Populate MySQL), C4 (Graph with Plotly), C5 (Sample UI with Vue)

Flight Plan: Step by step overview of demo

1. Using a local server, we will open a web application created with a Vue front-end framework.
2. The application will contain a display of an empty database, a button to display graph, and a prompt for a text file upload with a button.
3. By clicking on the upload button, it will allow the user to select a text file from their directory.
4. After navigating to the directory, a pre created text file will be selected. This text file will contain multiple points to be graphed.
5. Once the file is uploaded the displayed database will be populated with the data from the text file.
6. Then by clicking on the graph button, a graph will be displayed plotting the points from the database.

Evaluation:

- ✓ Convincingly demo'd each of listed challenges?
- ✓ Other evaluative comments:

---

### **Demonstration 3: Showing Differences in Log in**

Challenges addressed: C5 (Sample UI with Vue), C6 (Login Differences)

Flight Plan: Step by step overview of demo

1. Using a local server, we will open a web application created with a Vue front-end framework
2. The web application will require a username and password from the user before displaying any content
3. The initial user interface will show two input boxes. One will prompt the user for a username and the other for a password.
4. We will enter in a valid username and password.
5. The website will notify the user that their credentials were accepted and they are now logged in.
6. We will show the process again, this time entering in an invalid username and password.
7. The website will notify the user that their credentials were incorrect, and prompt the user to try again.

Evaluation:

- ✓ Convincingly demo'd each of listed challenges?
- ✓ Other evaluative comments:

### **Other challenges recognized by not addressed by demo:**

If there were challenges you listed earlier that were *not* covered by a demo, list here. This will hopefully be a short list...but better to be clear about where you are. If you have items here, you could list (if applicable) any pending plans to reduce these risks.