

Project 1 - Vaccine Distribution System

Overall Project Description

A.Hornof - April 5, 2021

Due Dates

Initial PP/SRS/SDS - Monday, April 12, 9PM

Final Project Submitted on Canvas - Monday, April 26, 9PM

Develop a Project Idea

The initial project proposals have been published, with instructor comments, at

<https://canvas.uoregon.edu/courses/176767/files/folder/Project%201%20Proposals%20with%20Comments>

Develop and use any of the following four proposed project ideas:

Group 1 Proposal 1

Group 3 Proposal 1 or 2

Group 4 Proposal 1

Cite any proposals that you derive ideas from, and are not yours.

Requirements analysis is under-emphasized in this project.

This project focuses on design, implementation, testing, project management, and not on requirements analysis. Please provide a reasonable justification for the system in your SRS, but do not start making up stories to support made-up claims as to why the system is needed. Perhaps focus more on obviously-useful technical requirements, and not on made-up human requirements.

Project materials

Read all of the materials under "Project Handouts" on

<https://classes.cs.uoregon.edu/21S/cis422/>

For example...

1. See https://classes.cs.uoregon.edu/21S/cis422/Project_Evaluation.html for a description of the Initial Project Plan / SRS / SDS.
2. Use the templates at <https://classes.cs.uoregon.edu/21S/cis422/Templates.html> for your SRS and SDS.

Technical Requirements

You will need to develop your system requirements, but at least include the following:

1. The delivered system should be complete. For example, if the system will use a server to collect data, the system must include full instructions on how to set up such a server. The steps involved, and the instructions must be at least as simple as this:

<https://systems.cs.uoregon.edu/wiki/index.php?n=Help.ToolsMysql>

2. Systems should use standard libraries to the extent possible. For example, if written in Python, the systems should primarily use the Python Standard Library at

<https://docs.python.org/3/library/index.html>

Written permission must be obtained to use any libraries or packages beyond the standard libraries for a language. Permission is granted to use (a) Flask and (b) the Google Maps API.

3. Installing and running the system should require little or no software to be installed. To this end, no virtual environments, and no gaming engines such as Unity, may be used.

4. Instructions must be provided to compile, run, and install all of the code necessary to use the system.

5. Roughly half of your design and development should focus on *a novice user* installing the system and seeing how it works. You should specifically design the steps and processes that a novice would use to see every aspect of the system working. You should provide substantial realistic data, not dummy data like "XXX 123", for the novice to load into the system, so the novice can see the system working in real time.

6. Additional Programming Constraints

- The system may be built in C/C++, the C++ standard library, Cocoa, and no other components. (Note that an XCode command line tool could fulfill many of the requirements.)
- The system may be built using Python 3 along with The Python Standard Library <https://docs.python.org/3/library/index.html>, but no other imports.
- The system may be built using Java along with Java Standard Edition modules <https://docs.oracle.com/en/java/javase/12/docs/api/index.html>, but no other imports.
- C++ code must comply with C++11.
- Python code must run in Python 3.8 and 3.9.
- Java code must run in Java 7 or 8.
- Instructions must be provided for how to compile the code.
- No virtual environments may be used.
- No gaming engines such as Unity may be used.

7. Installation Constraints

- There can be at most 10 steps to compiling the code and running the program.
- An experienced computer programmer should not require more than 30 minutes working alone with the submitted materials to compile and run the code.

Exceptions to the above requirements will be considered. The requests will carry the most weight if they follow the guidance under "A group asks about using Flask" at

<https://classes.cs.uoregon.edu/20S/cis422/P1/addendums.html>