

Week 2: Individual Report

- Basic information
 - (___/1pt) The project title.
 - Water Carrier
 - (___/1pt) The URL to the GitHub repository
 - If your GitHub account is protected, add the extra information for checking out the GitHub Repo.
 - <https://github.com/dathaotrinh/WaterCarrier>
 - (___/1pt) Milestones with deadlines
 - Refresh about Python programming and learn about Flask by 2/4/2024
 - Research on how to draw bar chart using Flask 2/4/2024.
 - Design and setup login and signup page by 2/9/2024.
 - Design the UML by 2/11/2024
 - Flask is the Frontend and backend technology tool to be used
 - (___/1pt) The algorithms/AI schemes used in the core engine
 - Use basic math to solve the problem.
 - Container with Most Water problem.
 - (___/1pt) Describe the market space the application is related to and the selling points of your product.
 - Imagine you have a bunch of walls of different heights. Each wall can hold water on top of it, and you want to find the pair of walls that, when treated like the sides of a container, can hold the maximum amount of water. The score will be determined by how fast the user solves the problem.
- Progress
 - a list of action items that were
 - (___/1pt) committed to since the previous meetup
 - Did research on math and computer science concepts.
 - (___/1pt) completed
 - Decided to build a web application that asks the users to solve the “Container with Most Water” problem.
 - Setup GitHub repository with the directories as described below.
 - Updated README file.
 - Understood the concept of “Container with Most Water” problem.
- (___/1pt) Your GitHub account

- (Create file README.md with the above basic information about your project)
- Make the following directories under the root directory (do not misspell the directory names.)
 - "presentation_slide"
 - If there is a presentation, upload presentation slides there as weekn.pdf where n is the week number.
 - "weekly_report"
 - upload weekly reports there as week2.pdf, week3.pdf ...
 - "dev" that contains all project codes
 - "docs" that contains project documentation with following subdirectories
 - expert_users
 - **instructions for building your project for expert users like your colleagues, TA and the instructor.**
 - **0 points if the project cannot be built successfully based on your instructions.**
 - app_users (To be done later)
 - developers (To be done later)
 - Summarize the bug/problems found in the past week
 - - /docs/developers/to_be_fixed.pdf (To be done later)
 - List the bug/problems to be fixed with the expected dates for fixes.
 - /docs/developers/not_to_be_fixed.pdf (To be done later)
 - For the enhancement

problems that cannot or will not be implemented, add the problems along with the reasons for not fixing them. (To be done later)

- /pub (To be done later)
 - product_design_spec.pdf
 - test.plan.pdf