CMPSC 580
Junior Seminar
Spring 2022

Assignment 6:
Prototype: A Foundational Deliverable

Objectives

To design and implement a prototype tool that can run experiments, collect data, and analyze results. To write clear documentation, which allows others to use the tool, and to demonstrate the implemented prototype.
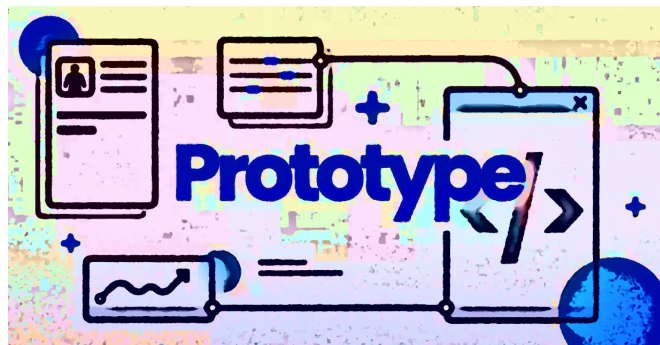

Figure 1: Build a part of your original idea to showcase some specific functionality. A prototype is an incomplete but functioning version of the software that is being developed to show something of your research.

Introduction

By this point, you should have a well-defined idea that fills a knowledge gap and research questions, the answer to which can fill the identified knowledge gap. You have also thought through the experiment to ensure your proposed idea is feasible. It is now time to put your idea to a small test.

This week is an implementation week! Specifically, you are to design and implement a prototype tool that can run experiments, collect data, and analyze results as described in Figure 1. Prototype typically refers to an incomplete but functioning version of the software that is being developed and as such prototype only involves a few main features of the final software product (reference: Wikipedia `https://en.wikipedia.org/wiki/Software_prototyping`).

By the end of next week, you should be able to demonstrate key features of your tool. Specifically, the completion of a prototype deliverable must fulfill the below requirements.

- **Code construction**: Build a *software-based creation* that implements some key features of your idea. This software creation must do *something* to facilitate some substantial part of the research method into the questions, established in a previous assignment. For instance,

this deliverable could be a tool, a piece of software written by you, that demonstrates that you have a good understanding of your research project's main deliverable. Please remember that this step also argues volumes that you will be able to complete the work.

- **Working code and error checking**: Implement automated tests (such as using `pytest` for Python code) to show something of the correctness of the code. In this task, you are demonstrating that your code is free of errors, and is able to produce results which are meaningful to your larger research problem.

- **Documentation**: Write clear documentation to accompany the software-based creation or deliverable. Your documentation is to facilitate the general comprehension, functionality and application of the deliverable. As you write your code, please be sure to include comments and information regarding the goals of each code block or major decision being made in your code. In addition, please write up software documentation in a Markdown README document to help the user to use your software-based creation.

Clone Your Assignment Repository

Today's assignment repository can be found at the below link to a GitHub Classroom repository. Here you will work on the implementation of your prototype and then you must push your work to the GitHub cloud where the instructor will be able to view your work for grading. **Please note that the instructor cannot view your submission and cannot grade your work unless you `push` your work to the GitHub cloud!**

<center>

https://classroom.github.com/a/Rl_lWB1A

</center>

To use this link, please follow the steps below.

- Click on the link and accept the assignment

- Once the importing task has completed, click on the created assignment link which will take you to your newly created GitHub repository for this lab,

- Clone this repository (bearing your name) and work locally

- As you are working on your lab, you are to commit and push regularly. The commands are the following.

  - `git add -A`
  - `git commit -m ''Your notes about commit here''`
  - `git push`

Part 1, Today (Tuesday): What To Do

Today you should complete the design of your tool. Specifically, in your design, you should:

- Identify the software requirements of your tool (all the things your tool must do).

- Create a GitHub Project Board in your prototype implementation repository by clicking on the "Projects" tab in your repository, select a non-Beta "Projects" on the left menu, then click "New project", fill out the relevant information and "Create project". Now convert the software requirements from the previous point to actionable items and add them to the TODO column by clicking "'+" icon.

- Create a technical diagram of the tool (workflow of how it works). Use a design tool of your choice.

As you have time, you should begin implementation of your prototype.

Part 2, Lab (Wednesday): What To Do

Continue to work on the implementation of your prototype. By the class on Thursday, you should have a clear design for the prototype and have some implementation to demonstrate its functionality in light of your project.

Part 3, Next Class (Thursday): What To Do

Continue to work on the implementation of your prototype. Be sure to begin working on your larger documentation project written in Markdown which is to educate the user about how to use the project's code. In addition, this documentation is to inform the user about necessary resources to secure before running the code. For instance, in your documentation, you are to discuss the software, libraries, drivers, data and other resources which would be necessary to run the software.

Sneak Peak to Next Week

We will continue this work next week. Each part listed below will indicate a class meeting or a lab session.

Part 4, Next Week (Tuesday): What To Do

Peer review of the current status of the implementation and continuation of the implementation. Partner with a colleague in class, have the partner try out your tool without your assistance, and then walk each other through your current progress in prototype implementation. Post the response to the following questions as an issue in the project's repository.

- Briefly describe their prototype tool.

- Were you able to follow documentation and use the tool? If not, outline the challenges that you faced. Post them as issues in their repository.

- What has been completed?

- What remains to be implemented for the prototype to be ready for a demo?

- What suggestions do you have?

Handed out: 5th April 2022

Part 5, Next Week Lab (Wednesday): What To Do

Continue to work on the implementation of your prototype. Consider the suggestions from your colleague to be used to further develop the project.

Part 6, Next Week (Thursday): What To Do

Lightening talk. Prepare a three min presentation / demonstration of the software creation or tool. Please be ready to discuss how this tool is relevant to the research ideas which are being studied in your project.

Summary of deliverables

- **THIS week's work**

    - **For class (Tuesday)**: Work on the design of your prototype.
    - **For lab (Wednesday**: Work on the implementation of your prototype.
    - **For class (Thursday)**: Continue to work on the implementation of your prototype. Now, begin to add working documentation to your project.

- **NEXT week's work**

    - **For class (Tuesday)**: Peer review work.
    - **For lab (Wednesday**: Continue to work on the implementation of your prototype. Add suggestions to improve project.
    - **For class (Thursday)**: Lightening talk.

- **The software resources that you create for this work are to be submitted in the `src/` directory**

- **Web notebook**: Please provide a cohesive blog post to introduce your work on your website (min 200 words). Please title your posting, *Prototype.*

Help?

Please let the instructor know of any questions that you may have. Please use email or make office-hour appointment slots if you would like to discuss an issue.