**Project Proposal**

**Description:**

Name: Magic Cube Solver

The project will take the colors from an unsolved Rubik’s Cube through a live webcam and give the user the moves required to solve the Rubik’s Cube from that permutation.

**Competitive Analysis:** Several Rubik’s Cube solvers can be seen in previous iterations of 15-112 like “Cubing Coach” (Fall 2019 Pittsburgh) and “Rubik’s Cube Solver” (Spring 2019 Pittsburgh). This project will use the same method as “Cubing Coach” (CFOP method) but a different one from “Rubik’s Cube Solver”. Furthermore, this project will not include a coach like the one seen in “Cubing Coach” and will focus more on giving the user the solution.

**Structural Plan:** The project will be divided into four files:

Rubik’s Cube class: simply makes a functioning Rubik’s Cube that can move, will be used later to find what the cube will look like after each step.

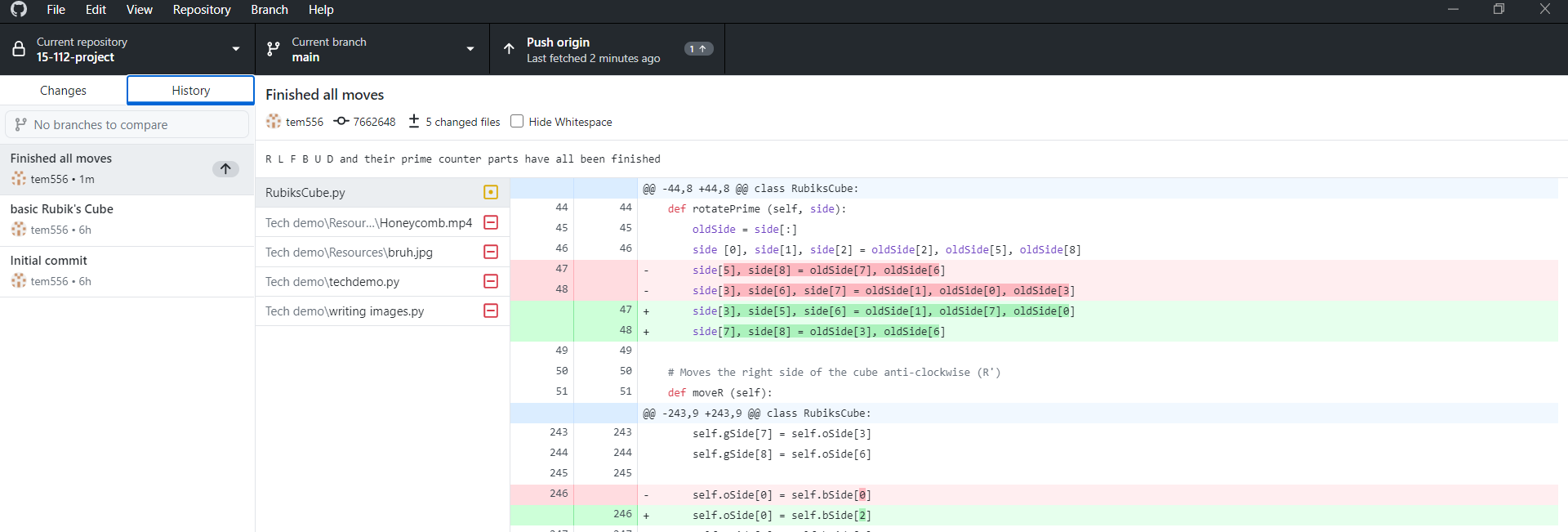
Solver algorithm: Finds the moves required to solve a Rubik’s Cube. Uses the CFOP (not a computer algorithm) method for solving the Rubik’s cube. I will achieve that by implementing it in python.

Image Processing: takes the unsolved Rubik’s Cube’s colors through a webcam and gives the colors to the solver algorithm.

User Interface: will have the code for the graphical user interface for this project and gives the user the moves required for solving.

**Algorithm Plan:** The trickiest part seems to be the Solver algorithm. This project will rely on the algorithmically heavy CFOP method (also called Fredrich method) for solving the Rubik’s Cube. This part will involve implementing this real-life algorithm into Python. Will have to recognize the colors at each step and then choose the correct algorithm for that position. Not only that but there are quite a few different permutations possible at each stage, which I will have to include the solution for.

**Timeline Plan:** I will finish the following by: Finish Rubik’s Cube class by TP1 (17/11). Finish the Solver algorithm and the image processing part by TP2 (22/11). And finish the GUI part by TP3(1/12).

**Version control plan:** Will be using GitHub to back up all files. 

**Module List:** OpenCV

**TP2 Update:**

No major design changes, there will still be four files for the project. Only change in plan is delaying the image processing part till next deadline (TP3).

**TP3 Update:**

No major design changes. Only added a 2D representation of the cube that walks the user through the solution. And now the program can recognize when the user enters an invalid cube.