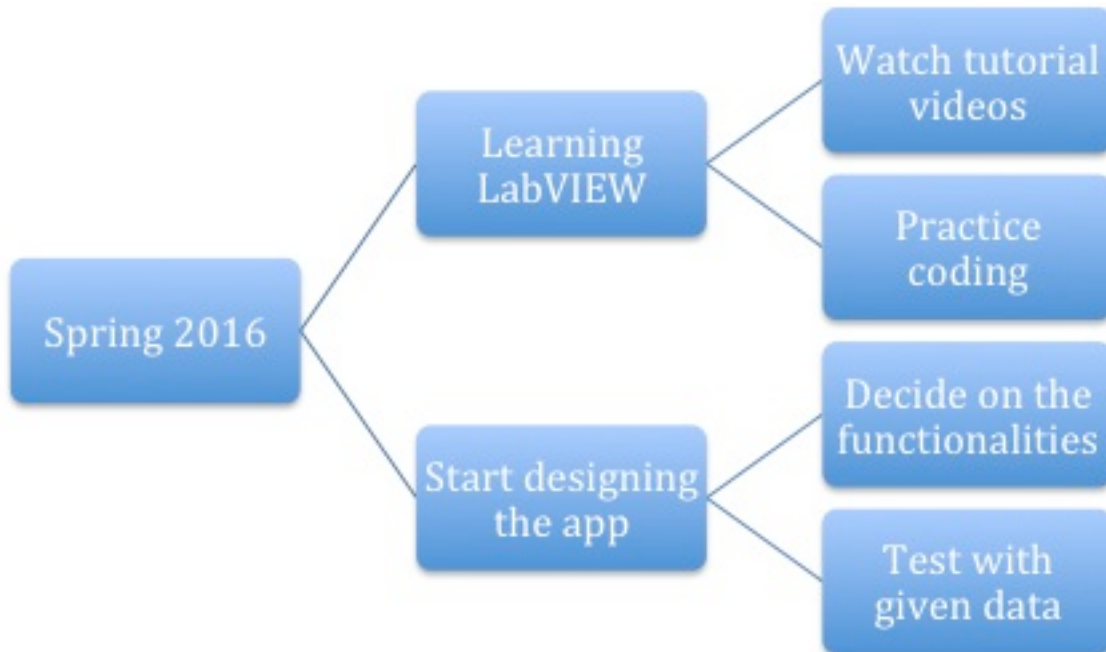# Semester Schedule

Task Map

Figure 1: Spring 2016 Task Map

Task List

1. Watching tutorial videos to learn LabVIEW - 02/24 - all members 2. Practice Coding - 03/23 - all members

- Loops: Learned about how for and while loops are used in LabVIEW. All group members had previous experiencing with loops, but the unique aspects of LabVIEW made it some- what confusing. However, tutorial videos as well as practice with LabVIEW helped promote understanding. Finished learning about loops 2/15.

- Shift Registers: Shift registers were studied because converting output values into input values in loops became a necessity as more complex practice was carried out. Finished learning about and practicing with shift registers 2/15

- State Machines: State machines allowed the members of the group to essentially create a set of if statements. Certain actions were carried out if certain specifications were met. A mock vending machine was created using a state

machine. Finished learning about and practicing with state machines 2/22

- Sub-VIs: Sub-VIs are essentially functions, and can be utilized in future programs to reduce the amount of code that must be written. This makes the code more user friendly, which is the main goal of this application. Finished learning about and practicing with Sub-VIs 3/3

- Executables: Youtube videos about executables were watched; however, the topics were quite complex, so actual testing with executables was not conducted. Further research into exe- cutables will be done in the upcoming week.

- Configuration Files: Although the team has looked into Configuration Files, more time needs to be spent on this subject.

- Type Definition: Further research into the uses of type definitions must be conducted in the upcoming week.

- Graphical Analysis: Using the graphical representation tools provided by LabVIEW will be further studied this week, because it will be very important as the group begins working on the actual application. 3. Start editing the code given by Casey Garland- 03/15

- Take specific parts of the code and put them into subfunctions. Also, get rid of unnecessary parts of the original code.

- Create a new VI with these subfunctions. This new VI is more organized and no longer has the unnecessary parts of the original code. 4. Decide on functionality of the final program - 03/29 - all members 8

5. Make program be able to process multiple photos continuously- 04/20- all members• Use event structures and loops to allow multiple photos to be processed continuously. • A VI will be created that completes this action.

6. Create a working version of the post processing image analysis part of the program - 05/10 - all members

7. Order hardware needed to upload photos to program and test the code with continuously generated photos- 05/20- all members