### Team Name: Five Guys

### Names of Those Present:

### Zachary Chenausky, Jigme Rinji Sherpa, Clay Lewis, Haris Javed, Saad Javed

## **Accomplishments**

During the course of Sprint 1, our team has been prioritizing getting speech functionality into VSCode extensions. Through our investigation of the properties available to VSCode extension, we found that using an external server program to communicate with the extension was the most functional path. We have updated the previous team’s testing document to include testing processes for updates that our team is making to the current extension. In addition, the development environment and extension have been repaired for further use, as there had been issues with functionalities not being available after changes in the system details since the previous team’s work.

## **What went well**

We were able to get the environment updated to smoothly transition in the future as functionalities and dependencies change. Part of the issue previously in the way was a lack of information about necessary parts and how they function. Now we have alleviated that by pulling relevant information from the user’s current installation to set dependencies and expectations up instead of hard coded entries.

## **What could be improved**

We spent a large amount of time attempting to get voice recognition fully within the scope of the extension, but we were not successful in adding this functionality. VSCode does not handle the appropriate modules or communicate with system input resources in a way that could be used to facilitate this function. We need to work on the accuracy of the speech recognition when the user is being quiet. Accuracy is currently strongly correlated with the volume of the user, as too low of a user volume doesn’t ensure accurate results from the speech recognition software.

## **Plans for Sprint 1**

As we expand on the functionality of the speech recognition functions, we will be finding and handling common misinterpretations from the speech recognition. This would include homonyms and other misinterpretations that can arise in a speech recognition system. Once connected into the extension, we will go about designing the functions based off of the prior written prompt script. This will prompt the user and take voice input as necessary to build out functions with parameters and clean function design. We hope to expand on the Autocomplete functionality, allowing users to receive a relevant selection of options through sound cues, allowing ease of use without vision.

The autocomplete functionality will be implemented using the Language Server Protocol. The language server protocol will be used to maintain a connection between the client i.e, vscode and the Language Server. The language server is responsible for parsing the python file and supplying autocomplete values back to vscode and the voice control system. We will implement functionalities where users can enable or disable autocomplete suggestions. These suggestions will be passed to the voice control system and can be outputted via voice/sound. For the convenience of the users, they will have the option to filter the type of suggestions. For example, they can ask to suggest local variables only or an object’s functions. We hope to achieve these functionalities implemented or at least the basic functionalities for the next sprint.