Deven Biehler
Aidan Smith
Olive Hehemann
Brian Chan
Mohammed Al Ahbabi
Subham Behera

Milestone 1

Wednesday, February 2, 2022

Team name: We Serve U (WSU)

For this milestone, we will provide the key tools, technology, and process model we plan to use for our restaurant automation project.

Key Tools:

GITHUB

- By using Github, we can effectively edit the same project in a remote repository without overwriting each other's additions. Because we are a larger team and with the current pandemic, being able to work remotely on our project is absolutely essential since meeting up in person anytime we want to work on it is simply impossible. Github also enables version control, thereby being a helping hand in the spiral method. This will help in allowing us to all work on the projects at the same time and create some sort of control over the revisions that we create, in addition to allowing people to work on their own features independently of others. This helps to add ownership of certain sections by specific people. We were also able to add the TAs to our Github repository so they can help us if needed.

VISUAL STUDIO

- Has a built-in syntax checker. It's free. Many keyboard shortcuts to help productivity. It's also well known so finding resources online will be easy, and everyone in the group has experience with some form of visual studio so we can help one another if we are having technical difficulties. Visual studio also has a software called Incredibuilder bundled inside that will accelerate C++ builds, this will benefit us since we are using C++

programming language. On top of that, visual studio supports C++ GUI which is one of the great options that visual studio supports.

Technologies:

C++

- We are using C++ for restaurant automation. A simple fact that C++ is what the majority of us have the most knowledge in. We think that it is most important to stick to a language that we are familiar with so that we can create the best project possible. C++ is a compiler-based programming language. It makes the code run smooth and fast. C++ also has a great GUI for application. The user-friendly interface will let the user feel more comfortable.

.NET FRAMEWORK

- For the GUI, we decided to go with a basic windows form application. Visual Studio makes windows form applications easy to set up and the .net framework works well with c++ which is our chosen language. This will allow us to build a GUI easily which will be necessary for what we have planned for the project.

Deven Biehler
Aidan Smith
Olive Hehemann
Brian Chan
Mohammed Al Ahbabi
Subham Behera

Milestone 1

Wednesday, February 2, 2022

DISCORD

- We will be using discord as a means of communication. Discord has a helpful server feature that will allow us to save conversations, transfer files, post announcements, and voice chat for meetings. Discord also has a feature that allows users to post easily readable code segments that we can use to ask each other for help. Discord also works well on both mobile and desktop devices, making it very easy for us to contact each other.

GOOGLE DOCS

- While we will use Discord for most communication, for filling out milestones (like this one) or planning, we will be using google docs to help keep track of meeting notes and discussion topics we want to discuss. This is because google docs is slightly more organized than discord is, and the formatting of google documents are easier to export for turning in or putting into our Github repository. Also, with google docs, we can all work at the same time and see other members' changes as they make them.

Process model:

The process model used here will be the XP model.

- We believe that the XP model is appropriate for this project. This is because the XP model fits our scope and our schedule. The XP model will allow us the flexibility to adapt

to new features or changes that we realize would be important as we go along. While the XP model can be hard to scale, that should not be too big of an issue.

- The XP model will be helpful when updating the project as it seamlessly can move to the next version with minimal reconstruction. Additionally, it is important that we use the XP model so that we retain flexibility throughout the project while still being able to maintain our timeline.
- The XP model takes the idea of code reviewing and amplifies it, allowing for continuous reviewing. This idea of continuous reviewing will be important so we will limit the risk of spreading bugs throughout versions.
- With a restaurant application, there is bound to be complaints from the many customers. With XP, we can take these complaints and build new requirements, and plan implementation of these new requirements right away.
- The risks of XP is that it lacks some of the other safety features of the scrum/sprint model. While it leaves us with more freedom, it also leaves more room for mistakes. Because of the tight timeline of our project, we think that the XP model is a better fit.