Dan’s Frapps

# Project Overview

This project aims to sell Frappuccino’s to people through a website application. This will be for an exclusive club of people who all have an account with Dan’s Frapps.

The ordering system will allow for both online orders to be placed as well as a Point-of-Sale contact order to be placed. It will make ordering Frappuccino’s easy for people and make keeping track of those orders easy. The system will also keep track of the inventory for the ingredients as well as their prices. It will also allow for the manager to order new inventory to keep inventory full.

Additionally, this system will act as a time tracker for the hours worked for each employee. They will be able to log the hours worked in their account and the manager will be able to pay them with the store funds. The app allows a manager to add or remove employees and set their individual wages.

# Team Organization

Team members: Jason Crandall, Joshua Frerichs, Reagan Baxter Spencer Lingwall

Supervisor: Dan Watson

We will primarily use an egoless programming structure that will have rotating leadership according to our strengths. The leadership responsibilities will be determined at the beginning of each scrum cycle.

# Software Development Process

The development will be broken up into five phases. Each phase will be a little like a Sprint in an Agile method and a little like an iteration in a Spiral process. Specifically, each phase will be like a Sprint, in that work to be done will be organized into small tasks, placed into a “backlog”, and prioritized. Then, using on time-box scheduling, the team will decide which tasks the phase (Sprint) will address. The team will use a Scrum Board to keep track of tasks in the backlog, those that will be part of the current Sprint, those in progress, and those that are done.

Each phase will also be a little like an iteration in a Spiral process, in that each phase will include some risk analysis and that any development activity (requirements capture, analysis, design, implementation, etc.) can be done during any phase. Early phases will focus on understanding (requirements capture and analysis) and subsequent phases will focus on design and implementation. Each phase will include a retrospective.

|  |  |
| --- | --- |
| **Phase** | **Iteration** |
| 1. | Phase 1 - Requirements Capture |
| 2. | Phase 2 - Analysis, Architectural, UI, and DB Design |
| 3 | Phase 3 - Implementation, and Unit Testing |
| 4 | Phase 4 - More Implementation and Testing |

We will use Unified Modeling Language (UML) to document user goals, structural concepts, component interactions, and behaviors.

# Communication policies, procedures, and tools

The primary communication method for this project will be a discord server organized with various channels for different topics pertaining to the project. Team meetings will be held primarily through discord or in person. Another option for meeting will be through zoom.

We will use Git as our version control with each person creating their own branch from the master branch. When changes are made, push to your branch, and then create a pull request to pull your changes into the master branch. This will allow most changes that are made to avoid merge conflicts.

For personal conflicts and issues within the team we will follow this system of conflict resolution. First, bring the issue up with the perpetrator and see if the problem can be resolved with them. If the conflict is not resolved through that method, then escalate it and bring it up with the team. Again, if the team cannot reach a resolution with the condemned individual, then we will escalate the issue to the Manager- Dan Watson. Through this whole process we will try to cultivate a culture of compassion and understanding.

# Configuration Management

See the README.md in the Git repository.