**Putt Putt Golf Tournament**

**Project Overview**

The project we intend to build is a web application that will run a Putt Putt Golf tournament. The users for our application are managers, players, sponsors, and drink meisters. There is only one manager at any given time, and he receives all the money from the drink orders and entry fees. The manager also verifies the sponsors and manipulates the drink menu. The sponsors donate money or prizes for the Tournaments and can select which Tournament they would like to sponsor, i.e., Monday’s Tournament or Thursday’s Tournament. Each player tracks which hole they are on and their current score, which is the number of strokes they have taken. The maximum number of strokes per hole is 5 so a lower score is better. The winner of the tournament will be displayed after all players have finished and the players in 1st, 2nd, and 3rd place will receive prizes. Drinkmeisters will be able to make drinks and deliver them once they are ready.

**Team Organization**

Project Manager: Evelyn Teeples (may change over the course of the project)

Designers and Developers: Carter Parks, Eathan Hodgkinson, Josh Williams, Lane Barnes

**Software Development Process**

(Taken from the Project Plan template on canvas)

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

**Communication Information**

Our team will be meeting through discord daily from X to Y, beginning each meeting with a stand-up in order to assess progress and hold each other accountable. New tasks will be assigned to developers as needed.

Discord – Main channel for communication. Used for group calls, file sharing, and other collaborative activities.

Google Drive – Storage for files needing collaborative efforts.

GitHub – Formal repository used for submissions, version control, data tracking, and communication with Professor Dan Watson.

**Risk Analysis:**

* Database
  + Likelihood - Low
  + Severity - Very high
  + Consequences - The whole app would break down since everything from inputting golf scores to buying drinks requires fetching information from the database
  + Work-Around - None
* Account Creation / Login
  + Likelihood - Low
  + Severity - Very high
  + Consequences - You will need to login in order to use the app, so issues with the account creation/ login system will prevent users from using the app
  + Work-Around - None
* Bank Account / Payment System
  + Likelihood - Low
  + Severity - Very high
  + Consequences - Users won’t be able to buy drinks, sponsors won’t be able to give money to their favorite players, managers won’t earn money since purchases won’t be possible
  + Work-Around - None
* Player Specific Pages (Current for golf tournament)
  + Likelihood - Low
  + Severity - Very high
  + Consequences - Players won’t be able to log and track scores
  + Work-Around - None
* Manager Specific Pages (Verifying a Sponsor/Drinkmeister, scheduling tournaments, modifying the drink menu)
  + Likelihood - Low
  + Severity - Very high
  + Consequences - Manager won’t be able to verify sponsors, schedule tournaments or modify the drink menu
  + Work-Around - None
* Sponsor Specific Pages (Amount of money they are providing, selecting which tournament they want to sponsor i.e. Monday’s tournament or thursday’s Tournament)
  + Likelihood - Low
  + Severity - Very high
  + Consequences - Sponsors won’t be able to donate money to players
  + Work-Around - Manager can collect money from sponsors and donate it to the desired players
* Drinkmeister Specific Pages (Current orders, finished orders, drink recipes)
  + Likelihood - low
  + Severity - High
  + Consequences - Users won’t be able to order drinks, reducing manager’s revenue
  + Work-Around - Manager can handle drink orders directly if necessary

**Configuration Management Plan**

See the README.md file in the project’s Git repository.