Project Plan

**Introduction:**

**We are going to create a restaurant menu application. It will be a cutting-edge application that will elevate the convenience of dining. It’s designed to allow customers to choose their favorite dishes from a breakfast, lunch, or dinner menu and go to checkout and submit their order without any further steps. The project’s intention is to create a user-friendly application that will enable the user to make selections and seamlessly proceed to the checkout.**

* Project Organization

|  |  |
| --- | --- |
| **Team Members** | **Roles of the members** |
| **Drashaun Morrow** | **Leader** |
| **Ryan Jarret** | **Coding** |
| **Kacie Jordan** | **Analyst** |
| **George Xiao** | **Testing** |

* **Roles: Leader, Analyst, Testing, Coding**
* **Risk Analysis:**

|  |  |  |  |
| --- | --- | --- | --- |
| Risk Name | Risk Level Explanation | Risk Effect | Risk Planning |
| Project Exceeds Budget | Low due to the amount of free resources that we can utilize | This could cause the later parts of the project to be of much lower quality than the earlier parts | By utilizing free resources, we should be able to complete the project without spending anything. |
| Scheduling | High due to the possibility of members’ schedules not lining up and possibly missing meetings | This could slow down the project due to attending members needing to help missing members catch up on what they missed | By keeping meetings infrequent and focusing more on messaging services, we can allow members to work on their own time. |
| Requirements Shortfall | High due to the limited time available to complete the project | Could make the final version of the project fall short and possibly even make it unusable | We have split up the work to allow members to focus on their part of the project. Should one fall behind, others can pick up some slack. |
| System and Equipment Maintenance | Low due to services that help us back up data, even if a machine fails other members can temporarily pick up the slack | A machine failing could potentially lead to lost data and force other members to take on more work | Should someone's machine fail, we have contingency plans to split up their work between other members until the problems are fixed. |
| Data Breach | High due to no system being perfect, if there's even one crack bad-faith actors will jump on it | Could expose people's sensitive data, as well as make the customers and public lose trust in us | By implementing plenty of checks and testing them frequently, we can hopefully reduce the chances of our security being broken. |
| Miscommunication | High due to communication being done over long distances | This could cause members to do unnecessary work and slow down the project considerably | By keeping our important points in writing, members will be able to ask others to clarify anything they are uncertain about. |
| Too Many Features | Low due to time constraints | This could cause members to focus on features that we don’t have time to implement, taking away the time to work on important features | By focusing on what needs to be done first and adding additional features only if we have the time, we should be able to reduce or eliminate any scope creep |
| Failure to Gather Resources | Low due to the amount of free resources that we are already starting to utilize | This could cause the project to take longer, or finish in an incomplete state | By several free resources provided or learned from previous projects, we should be able to find everything we need for the project. |

* Hardware and Software requirements:
  + **For our restaurant menu application, it will be developed in Python and several potential libraries such as Tkinter, SQLite, and maybe numpy to run smoothly on a wide range of windows versions and computers. We also might use Visual Studio to create this application.**
  + **Hardware**
    - **4 GB of ram for larger amounts of data and customers**
    - **Possibly 5 GB of storage for data, images, and other media or resources.**
    - **Keyboard & mouse**
    - **Monitor**
  + **Software** 
    - **Python**
    - **Windows Operating system**

**Work Breakdown:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Schedule** | | | | | | | | |
| **Month** | **November** | | | | **December** | | | **Dead line** |
| **Day** | **11/3** | **11/12** | **11/19** | **11/26** | **12/3** | **12/10** | **12/13** | **12/15** |
| **Planning (ALL)** |  |  |  |  |  |  |  |  |
| **Initial Design/Analysis (ALL)** |  |  |  |  |  |  |  |  |
| **Final Design (ALL)** |  |  |  |  |  |  |  |  |
| **Coding Drafts (ALL)** |  |  |  |  |  |  |  |  |
| **Code Review (ALL)** |  |  |  |  |  |  |  |  |
| **GUI Creation (Drashaun)** |  |  |  |  |  |  |  |  |
| **Coding: Welcome Page (Kacie)** |  |  |  |  |  |  |  |  |
| **Coding: Main Menu Page (Kacie)** |  |  |  |  |  |  |  |  |
| **Coding: Order Summary Page (Ryan)** |  |  |  |  |  |  |  |  |
| **Coding: Payment Page (George)** |  |  |  |  |  |  |  |  |
| **Code Review/Test (George & Kacie)** |  |  |  |  |  |  |  |  |
| **Fixes & Polish (All)** |  |  |  |  |  |  |  |  |
| **Documentation review (Ryan)** |  |  |  |  |  |  |  |  |
| **Implementation (All)** |  |  |  |  |  |  |  |  |
| **Presentation (All)** |  |  |  |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Project Schedule | | | |
| **Task** | **Week Start** | **Deadline Week** | **Identifier** |
| **Planning** | **1** | **1** | **M1** |
| **Initial Design & Analysis** | **2** | **2** | **M2** |
| **Final Design** | **3** | **3** | **M3** |
| **Coding Drafts** | **3** | **4** | **M4** |
| **Code Review** | **4** | **4** | **M5** |
| **GUI Creation** | **4** | **4** | **M6** |
| **Coding: Welcome Page** | **4** | **5** | **M6.1** |
| **Coding: Main Menu Page** | **5** | **5** | **M6.2** |
| **Coding: Order Summary Page** | **5** | **6** | **M6.3** |
| **Coding: Payment Page** | **6** | **7** | **M6.4** |
| **Code Review and Test** | **6** | **7** | **M7** |
| **Fixes & Polish** | **7** | **7** | **M8** |
| **Documentation Review** | **7** | **7** | **M8.1** |
| **Implementation** | **8** | **8** | **M9** |
| **Presentation** | **8** | **8** | **M9.1** |

Monitoring and Reporting

**For Monitoring and Reporting procedures we will be using a trello board to keep track of assigned tasks, and task completions. Our primary forms of communication will be held through a discord server for informal meetings, and a recorded zoom call for formal meetings. Files will be passed through the discord server as well, any files too large for discord will be shared through email, or through a repository.**

* **Collaboration**

**Collaboration has been decided by splitting work relatively equally, *simpler* assignments are assigned to one person, while larger tasks are assigned to multiple. For direct collaboration we will be using google docs for written portions to allow for live feedback, comments, and editing. As well as discord communication while working on other portions that are impossible to do through docs.**

* **Scheduling**

**Scheduling, due to our difficult schedules we set a drop-dead time as before sunday. Sunday will be spent reviewing each other's tasks to ensure mutual understanding and satisfaction. Any errors or problems discovered during the day will be fixed by turn in time Sunday night. Meetings will be scheduled before the start of the new week.**

* Appendix

|  |  |  |
| --- | --- | --- |
| * **Task Number** | * **Description** | * **Duration (weeks)** |
| * **M1** | * **Initial planning phase; present ideas and decide on solution** | * **1** |
| * **M2** | * **Initial design phase; create diagrams of program and discuss operation** | * **1** |
| * **M3** | * **Final design phase; Finalize and decide on operation, structure, and SOPs #See note 1** | * **2** |
| * **M4** | * **Coding Drafts; Begin programming sections of the software solution** | * **2** |
| * **M5** | * **Code Review; Collectively review progress, assist if needed, share insights, discoveries** | * **1** |
| * **M6** | * **GUI Creation; Begin creating windows and implementing images into app** | * **1** |
| * **M6.1** | * **Coding: Welcome Page; Create text and buttons to welcome user** | * **2** |
| * **M6.2** | * **Coding: Main Menu Page; Create a set of menus to choose from based on time of day** | * **2** |
| * **M6.3** | * **Coding: Order Summary Page; Start making a window that shows all the selections that the user has made.** | * **3** |
| * **M6.4** | * **Coding: Payment Page; Create page for user to enter personal and payment information** | * **2** |
| * **M7** | * **CR & Test; Review completed sections, and assemble and test solution.** | * **2** |
| * **M8** | * **Fix & Polish; Any errors, or bugs encountered during testing are fixed, variables consistent, all sections adhere to SOP.** | * **2** |
| * **M8.1** | * **Doc Review; Create, Combine, & Review user documentation for solution.** | * **1** |
| * **M9** | * **Implementation; Decide on implementation plan for software solution** | * **1** |
| * **M9.1** | * **Presentation; Present software solution** | * **1** |

* Note 1: Decide on standard programming practices, ex. pascalCase for variables, and other practices.

**Note 1: Decide on standard programming practices, ex. pascalCase for variables, and other practices.**