**CSE 310 – Applied Programming**

**Module Plan**

|  |  |
| --- | --- |
| **Name:** | Alex Nielsen |
| **Date:** | 10/4/23 |
| **Teacher:** | Ken Walters |
| **Module # (1-6):** | 2 |

1. Identify which module you have selected to work on. Place an “X” under the “Selected Module” column.

|  |  |
| --- | --- |
| **Modules** | **Selected Module** |
| Cloud Databases |  |
| Data Analysis |  |
| Game Framework |  |
| GIS Mapping |  |
| Mobile App |  |
| Networking |  |
| SQL Relational Databases |  |
| Web Apps |  |
| Language – C++ | X |
| Language – Java |  |
| Language – Kotlin |  |
| Language – R |  |
| Language – Erlang |  |
| Language – JavaScript |  |
| Language – C# |  |
| Language - TypeScript |  |
| Language – Rust |  |
| Choose Your Own Adventure |  |

1. At a high level, describe the software you plan to create that will fulfill the requirements of this module. This may change as you learn more about the technology or language you are learning.

I will make a command line-based calculator that can add, subtract, divide, multiply, and fulfill other basic arithmetic functions. I will also focus on having a fully implemented error handling system so that the calculator is immune to user error.

1. Create a detailed schedule using the table below to complete your selected module during this Sprint. Include details such as what (task), when (time), where (location), and duration. You are expected to spend 24 hours every Sprint working on this individual module and other activities in the course. Time spent on this individual module should be at least 12 hours.

|  |  |  |
| --- | --- | --- |
|  | **First Week of Sprint** | **Second Week of Sprint** |
| **Monday** |  | Begin writing code, practicing syntax, and experimenting 2hr |
| **Tuesday** |  |  |
| **Wednesday** | Planning and beginning research 2hr | Coding in classes, functions, and majority of backend dealing with the operations and error handling 3hr |
| **Thursday** |  |  |
| **Friday** | Beginning layout and program design, determining any classes needed, interaction, etc. 3hr | Finishing error handling and creating user input 2hr |
| **Saturday** |  | Flex time to finish and complete extra debugging 2hr |

1. Identify at least two risks that you feel will make it difficult to succeed in this module. Identify an action plan to overcome each of these risks.
2. Identifying all of the potential user-errors which I will need to account for in my program. This will take a little bit of brute force trial and error, attempting to break the code in anyway in order to fix it.
3. Learning how to use functions to take a variable number of inputs, and how to handle several operations in the command line. I will use ample research time and experimentation in a sandbox environment in order to learn this skill properly.