Software Development Plan (SDP)

KyuSeungSim

Nyah Hughes

Cody Rogers

Eden Florianne Merigold

1. Product Description
   1. Non-work for hire

* Potential audience is User(player).
* The user goes to battle against various enemies using the deck they’ve built with their gathered cards
  + Each card has different properties (e.g., attack/deal damage, boost attributes, protect the player, etc.)
  + This will be programmed by our group
* The user controls a player via HUD to manage resources (e.g., manage cards, purchase from shop) and make decisions (e.g., battle an enemy)
  + This will be programmed by our group
* Processing
  + Input: player can click and drag & drop interaction object in GUI
  + Process: the game’s system will catch clicks and “drag and drop” positions in GUI
  + Output: the program’s GUI will show the result of the player’s input as images or text

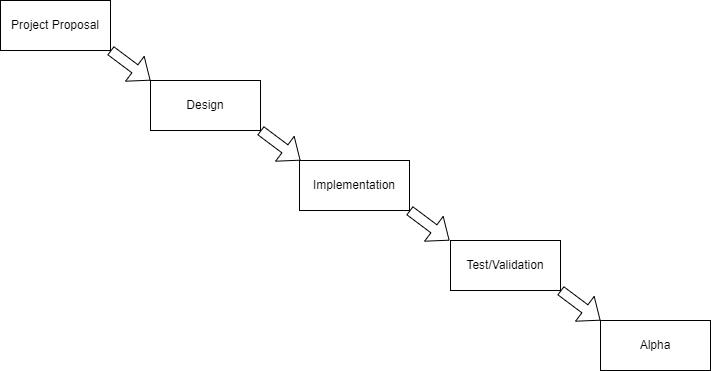
1. Team Description

* Experience in Java and Scene builder is required by all members
* Knowledge of some game design and the fundamentals of the game genre is necessary
* Some talent in graphic design will be for GUI images and game objects
* Experts will not be needed

1. Software Process Model Description

Waterfall model process:

* A level of simplicity appropriate for the group size
* Means of communication and projected scope size mitigate code inconsistencies and design integration flaws
* Project’s life cycle is relatively short, and maintenance and significant testing periods are unnecessary



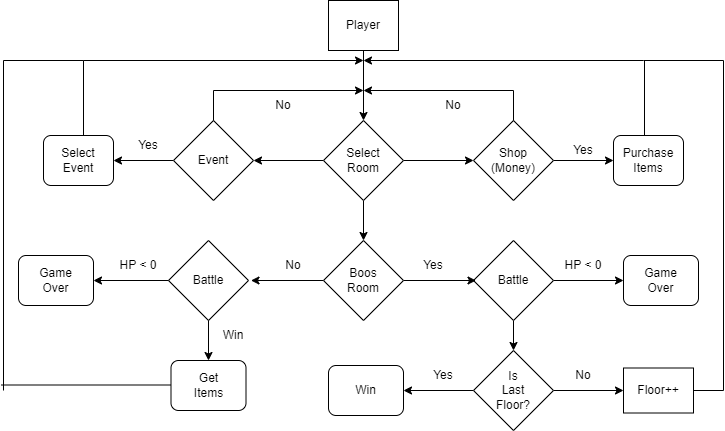
1. Project Definition

Novice:

* I hope Novice’s download Java to run JVM
* Play and enjoy the game, with the end goal to reach and defeat the boss
* Introduction to the roguelike and deckbuilding game genres

Expert:

* Already prepared Java and environment
* Set additional goals or restrictions for a more challenging, enjoyable experience, such as collecting every card, limited-resources challenge mode, or a speed run
* Already enjoy roguelike genre game.



1. Project Organization

* Plan

|  |  |
| --- | --- |
| Item | Date |
| Project Proposal | 11 February, 3 hours |
| Proposal Presentation | 15 February, 2 hours |
| Software Development Plan | 28 February, 3 hours |
| Design | 11 March, 10 hours |
| Implementation | 3 April, 21 |
| Progress Report | 24 March, 4 hours |
| Test | 1 May, 23 hours |
| Project Presentation | 5 May, 2 hours |

테이블이(가) 표시된 사진

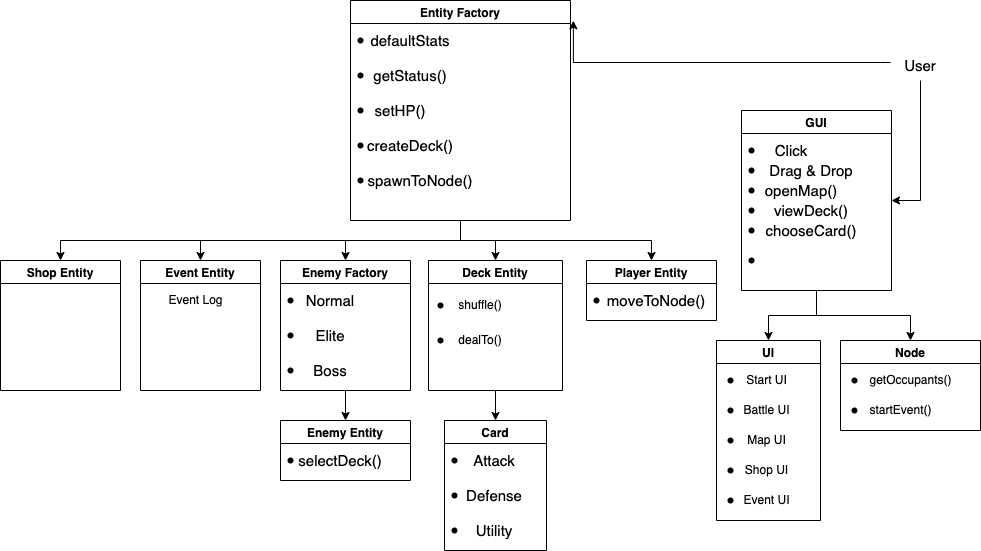
자동 생성된 설명

1. Validation Plan

Example entering a new room sequence:

1. Input: Player mouse clicks a new node on the GUI to move their character to
2. Output: The character’s position is changed on the map screen, and the battle screen GUI is opened
3. Output: Player and enemy’s decks are automatically shuffled, and player is given a hand at random. The individual card graphics for the current hand are displayed prominently for the player
4. Input: player clicks on the card graphic
5. Output: information panel updates with the card’s statistics and effects
6. Input: player can click button ‘a’ to play the card, or click button ‘b’ to unselect that card and return to hand selection
7. Configuration/Version Control

* Configuration



* Version Control
  + Design update
    - * + 1.0 Version: CLI interface
        + 2.0 Version: GUI interface
  + Method update
    - * + Player System
        + Battle System
        + Map System
        + Shop System
        + Event System
  + Bug Fix
    - * + Find every bug when run the program.

1. Tools

* Major System
  + Windows
  + Mac OS
* Subsystem
  + Java
  + Javafx
* Tool
  + Eclipse
  + Visual Studio
  + Screen Builder