

Create a planning document that describes (a) your testing strategy and (b) your development strategy. Remember with Clue we had high-level goals such as testing the deal, test making an accusation, test disproving a suggestion, etc. For the testing strategy, your document should list the program functionality that you will test, with a brief explanation. For the development strategy, you should list the major functional aspects of your program and indicate which you plan to code in Part I and which in Part II. Note that your plans may change... the point of this is simply to encourage communication. Also, you may merge the testing/development strategies, as long as it's clear.

## Testing Strategy

### Part I

- **Test board configuration load:** Ensure that the board configuration loads properly
- **Test level configuration load:** Ensure that the levels are loaded correctly
- **Test that the level increments when the player beats a level:** Ensure that, when a level is beaten, the game goes to the next level.

### Part II

- **Test minion advance:** Ensure that the minions advance at the correct speeds and that they stay on the paths.
- **Test tower fire angle:** Place a minion in the firing range of a tower and ensure that when the tower fires the minion dies.
- **Test :**

## Development Strategy

### Part I

- Create the classes from the UML with unimplemented functions
- Load board configuration
- Load level configuration
- Game logic (level increment, lives tracking)
- Draw board and squares

### Part II

- Place towers
- Minions attack on path
- Towers attack minions, implement the various types of attacks
- HUD GUI

## Functionality if Time

- Multiplayer over LAN