**Approach:** Our approach to implementing Phase 2 is to implement the agile-unified process to develop the classes in a top-down structure, where we prioritized finishing the character classes before moving on to the map.

**Changes:** We changed CharacterCollision changed to a Boolean because it can only have two values, either a collision or not. Also, axe/sword/speed were changed from Hostages to be implemented inside an arraylist because it made more sense while implementing it. We made simplifications to the game due to time constraints.

**Management:** We had an initial meeting where we decided on the Java library we would use for the game and a plan for the implementation. We arranged meetings and decided Myron would focus on the user interface, Josh would work on implementing the character classes, and Stephanie would work on implementing the maze.

**External Libraries:** To implement our program we decided to use the JavaFX library. We chose JavaFx because we were able to find documentation and support for JavaFx with Intellij IDEA, which is the IDE we used to develop our game. In addition, JavaFx provides a Scene Builder tool which allowed us to create GUI components simply by dragging and dropping them.

**Measure taken to enhance quality:** We added pause button and resume buttons to improve flexibility when playing the game. In addition, we added comments to explain the methods used and their functionality. We used abstract classes (Person, GamerTime, Wall) to organize our program and this enhanced the quality of the code

because it simplifies it. Additionally, we used inheritance by implementing subclasses that extend superclasses and that made our code more reusable and readable.

**Challenges:** Learning the JavaFx library with little knowledge of Java was a far greater challenge than we imagined.  We spent more time reading documentation and watching tutorials than the actual implementation.