Program Description

I will create a game similar to that of Bomberman. It will allow for same-keyboard play, where players share the same keyboard and play offline; online play, where players can host and join games of up to four players; and possibly single player mode with AI. When the game launches, the user is shown a main menu screen, where the player can choose a profile from a previously created name, choose different maps available, look at leaderboards, and options/key binds. The user can also choose different maps to play on that will have different layouts and mechanics. When not in a game, the user can navigate pages that allow for customization of key bindings as well as modifications to the players' sprites. The user can also access a leaderboard database which tracks every other user's score (kills, deaths/games played). The database can be sorted. While playing the game, the display shows the different objects in the game, which include: moveable players, unbreakable and breakable walls, bombs, timer, score, and power ups. All of these objects will be presented as sprites and the users will be able to edit/customize, save, and load their own character sprites. The last player standing wins the round. If the timer runs out and there is more than one player alive then it is considered a draw and nobody gets a point. The player can choose the time limit and number of rounds when the map is chosen.

Objects cannot move through another. Objects will block other objects' paths. The only exception is when a player drops a bomb, he will be allowed to move off it, instead of being trapped on it. The players will be able to move smoothly throughout the grid, like a map, and will be able to drop bombs that can be pushed/kicked until it hits another object (i.e. a wall). These bombs will explode after a certain amount of time or another bomb's explosion reaches the bomb. They will be able to destroy the breakable walls, kill players, and blow up other bombs. When the bomb explodes it will spread a fire in every direction possible (up, down, left, right). The size of the fire will depend on the power ups of the player who dropped it and if it touches a player, the player dies. The map will randomly distribute breakable walls throughout the grid-like map where unbreakable and breakable walls will not trap players. The breakable walls will drop the player power ups which gives the players perks to make them more efficient at blowing things up with their bombs. There are different types of power-ups that give different perks. The power ups only last until the round is over.

The destruction of walls, exploding of bombs, player movement, death, and dropping bombs will all be animated using multi-frame sprites to act as animations.

The game will also have an in-game soundtrack that is played from an audio file, that can loop to keep continuous music. The audio file will loop as long as the application is open. Each action in the game will also have its own sound. Such as when the character moves, kicks, and when a bomb explodes. These sounds will play based on user input. Clicking the UI will also play sounds to act has a feedback mechanism. The separate sound files will be part of the game assets.

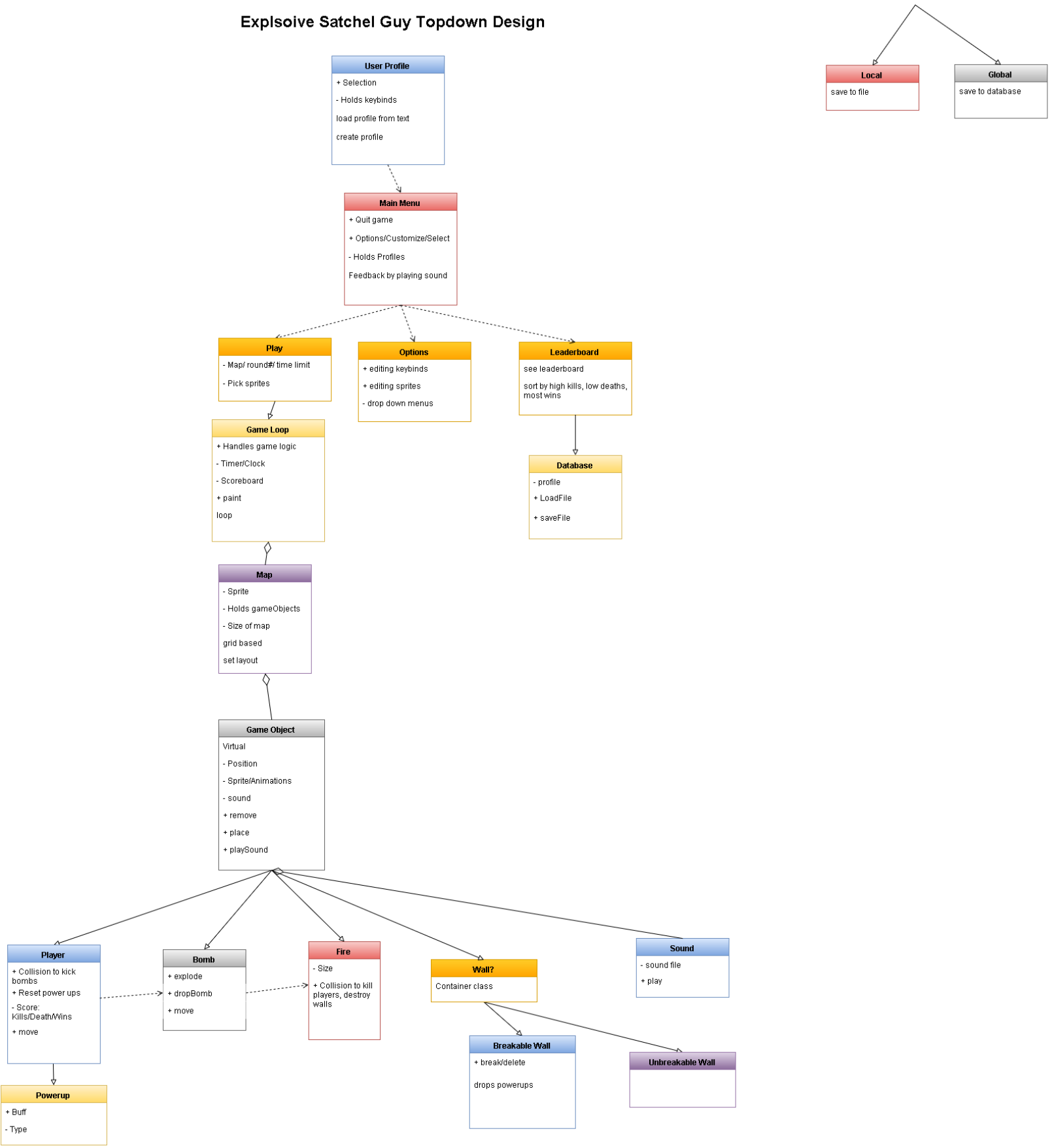
The database will hold all the users' information, including username, favorite sprite, number of kills, number of games played, number of deaths, and possibly more. Multithreading will be necessary in order to make sure the game can run as smoothly as possible, considering

there are going to be many objects on the field at any one time with their own movement, and action functions. Internet networking will be necessary in order to give this game the ability to be played online. It will also allow users to access the database.

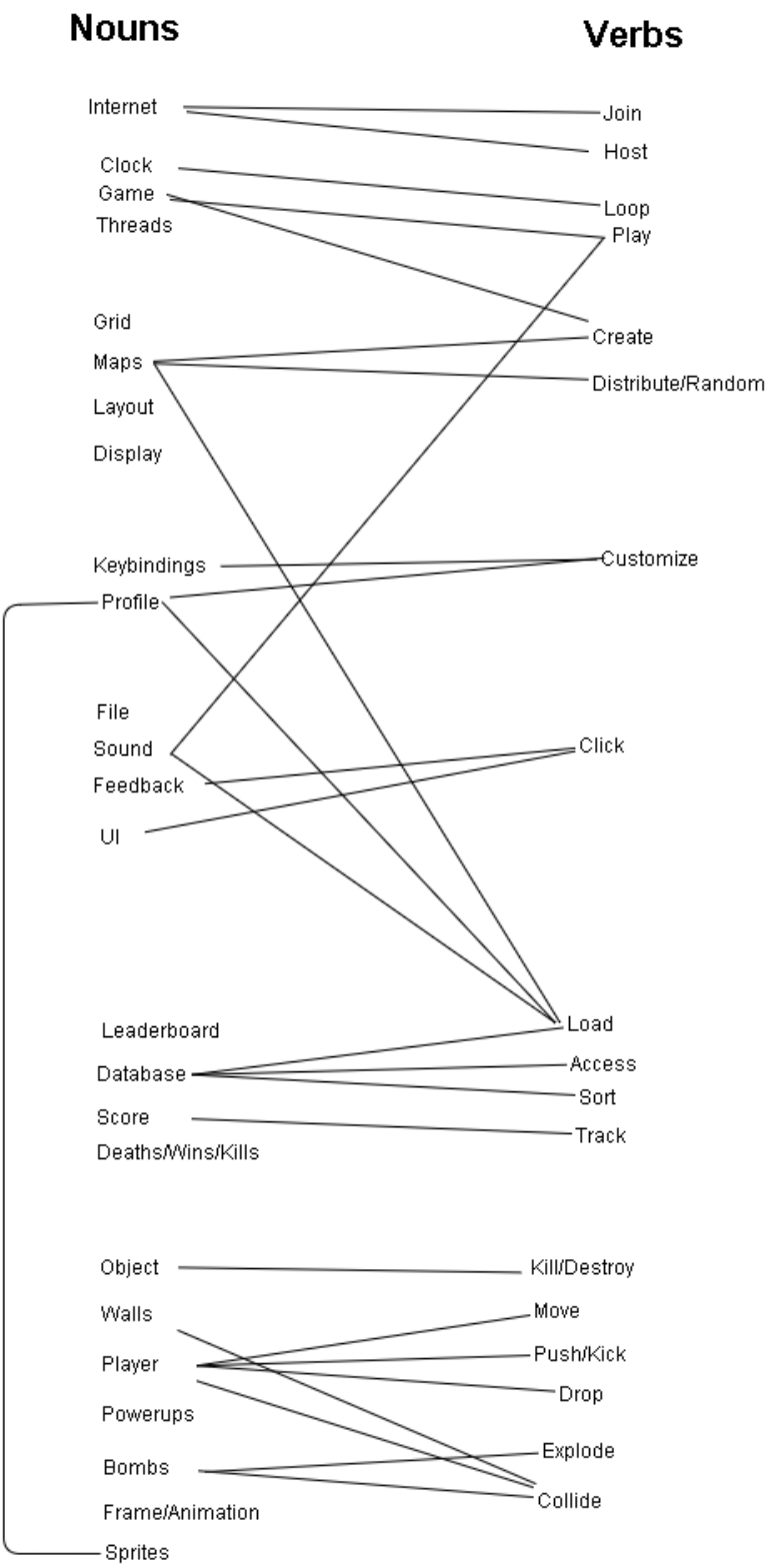
Technical Description

- **Database**
  - requires a server/ cloud/ host (Microsoft SQL Server)
  - hold profiles - username, wins, losses, kills, deaths, games played
- **Game objects/Map**
  - Player, Bombs, Fire, Power-ups, Walls
  - need their own sprites and/or custom sounds
- **Main menu/Play Screen/Options Screen/Leaderboard Screen**
  - needs images for background of each screen
  - buttons to navigate page will need their own sprites
- **Internet networking**
  - requires a server to connect to
  - needs a port to go through

Explsoive Satchel Guy Topdown Design



Explsoive Satchel Guy Bottomup Design



Installation Guide – Explosive Satchel Guy

- Run the Explosive Satchel Guy.jar file by double clicking it and enjoy the game. Use arrow keys to navigate menus, enter to select and option.
- Player one controls: WASD for movement, space to drop a bomb
- Player two controls: Arrow keys for movement, 0 numpad to drop bombs
- Escape to pause and quit or return to menu

Project Analysis – Explosive Satchel Guy

1. What areas of requirements are completed/covered?

   The base game is fully functional, and works as designed. I have players on a map that are able to drop bombs, which destroy some walls and other players. I also have power ups that the players can pick up to make themselves stronger. However, there are some extras that I was unable to complete.

   In order to code our project, I had to implement multithreading. As a result, our game runs quite smoothly.

2. What sections are missing?

   I was not able to implement databases or LAN hosting/joining. Also I did not implement player profiles or the options menu. However, I do believe that the player profiles and options menu would have been easily implemented if I had more time to do so.

3. What are the suggestions for future enhancements for this project?

   Some suggestions for the future of this project would be to implement the missing sections of our document as well as have online play. After that I could move on to make this game a mobile app and advertise it on both pc and mobile.

4. What are the suggestions for starting of the similar project?

   Keep everything broken down into generic classes that can be inherited from. This makes the classes be much easier to manage from one main class.