For the final version, I decided to structure the project similarly to how we structured our mobile game project in the IMAT 2608 Mobile Games module from second year. This separates the project into Activities, Classes and Views. Activities contains the different ‘screens’ of the game, such as the start-up screen, the options screen, etc. Classes contains the java classes we use to define objects, for example the player, certain objects in the world, etc. Lastly the Views are how the game is displayed. The game uses a surface view to render and draw objects.

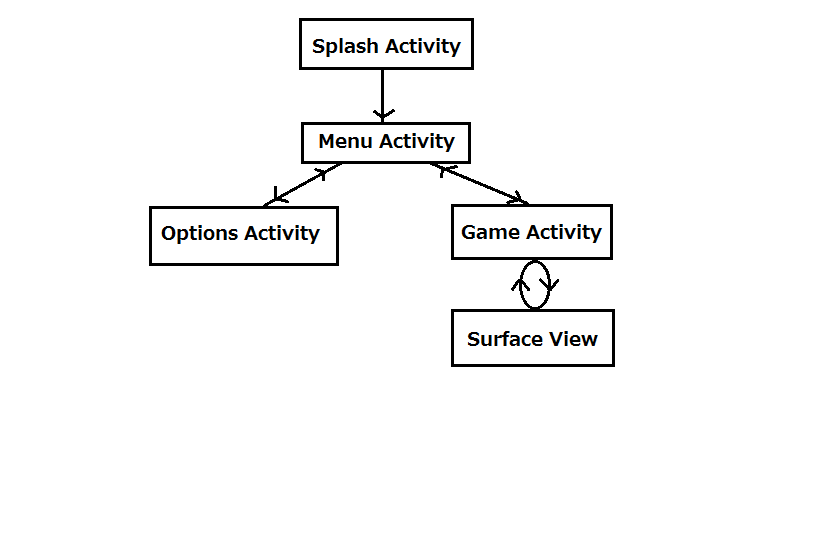
## Activities

The currently planned activities are:

* Splash Activity – This will be the starting screen, which is used to load certain parts of the game, or prepare files for the game
* Menu Activity – This is the main menu the player will see, and will allow access to other activities
* Game Activity – The game itself, this will then have a surface view to render and play the game in
* Options Activity – This will have the editable options for the player, such as changing the volume of music or sound effects, or changes to controls of the game

### Activity Life-cycle

Starting up the game enters the Splash Activity. This splash screen allows us to start loading parts of the game, hiding this from the user with a loading screen. From here we enter the Menu Activity. This is the main hub from which we can access other activities. The Splash Activity will not be accessible anymore. Trying to return to the previous activity (the Splash Activity) from the Menu Activity will instead close the game, showing a toast message to confirm if the user wants to quit the game.

We can branch to either the Game Activity or the Options Activity. In the Options Activity, the user is able to set certain settings, such as the volume of the music and sound effects, as well as an option to reset these to default settings. Unless the user confirms these settings, leaving this activity and returning to the Menu Activity will return to settings to the last used settings. The Game Activity is linked to a Surface View, which is how the 3D objects are rendered to the screen, as well as updated over time. The interaction between the Game Activity and Surface View is important, for example when the player touches the screen, the Game Activity takes this input, then passes it to the Surface View to process. When the game ends, the Surface View will return this to the Game Activity, and it will return to the Menu Activity.