

# Documentation

## 1) AsteroidSpawningTask

### Description

My game utilises a timer in order to randomly spawn randomly sized asteroids at random intervals, in random positions, traveling at random speeds. The frequency of the asteroids spawning depends on the level the user is currently on (higher level meaning more frequent larger asteroids).

The main difficulty with this class was implementing it along with the game levels feature used in level 2. This was done mainly by using more abstract classes representing the spawn rate for each asteroid. Another challenge was getting the timer to stop while the game was paused and continue once the game was resumed while maintaining the progress of the level. The final major challenge was implementing a 'safe zone' where the asteroids where not able to spawn in, to make the game more playable for the user.

## 2) Save Game/Load game feature.

### Description

My game utilises a text file in order to store current user data and load user data such that the game can be saved and loaded from different sessions.

The main difficulty was figuring out a way to represent the game in a text file, which in the end ended up being in a similar way to how a CSV file works, however using spaces instead of commas. As the games are dynamic (they are always different as all the asteroids are randomly generated), there was some difficulty in storing the number of asteroids along with the direction they were traveling in. Loading the files was significantly more difficult than saving them (creating objects using strings). At first I tried to use the 'reflection' java package in order to generate asteroids, however I was unable to get them to initialise correctly. In the end I used a number of IF statements which give a similar affect.