# 

1. SEC
2. REPORT
3. 19744591

# Multi-threading design and architecture

## Threads and responsibility

This project was designed with a heavy focus on multi-threading. The class responsible for starting the threads is the FileCompareApp class inside of package View. This class started two thread pools, and I/O thread which is a single thread pool and one for CPU intensive tasks. The CPU intensive task consisted of most of the available thread pool that the machine would offer. The decision to not give this app all the threads available was due to the machine freezing sometimes on which this app was tested.

A blocking queue was used in the FileLogger class. The blocking queue is not known to the outside world as it only concerns the logger class. Its only responsibility is to provide comparison results to be written to files after they are provided by the outside world.

## Threads communication

The