Capacity Planning Documentation

**<REDACTED>– <REDACTED>– James Ryan – <REDACTED>**

University of Victoria

SENG 468

Our ability to capacity plan is hinged on the hardware that we have access to for production deployment. Our transaction servers are very multithreading heavy, therefore if we deploy on machines with higher CPU core count, and higher number of threads, we would see better performance and would need a fewer number of transaction servers along with a fewer number of total machines because more transaction server could be deployed on a single machine.

As seen in the graphs below in this document, we found that the best performance was gained by our system for up to 1200 users was for one transaction server on one machine. One reason for this, is that because of the multi-threading intensive design of our transaction server, when introducing an addition server on a machine in the lab environment the CPU scheduling becomes too much, and bottle necks the performance of both transaction servers.

We also found that too many servers spread over too many physical machines caused too much network overhead when used for even the 1200 user workloads. Future development and testing would be necessary to find the sweet spot of servers to machines to user count of our system.

Further testing would need to be done to narrow down the specifics of how many transaction servers and physical machines would be necessary for a given workload throughout the day. Using that information, we could then configure docker-compose to run specific numbers of transaction servers for a given period where we predict higher or lower usage, such as at market open or close.

