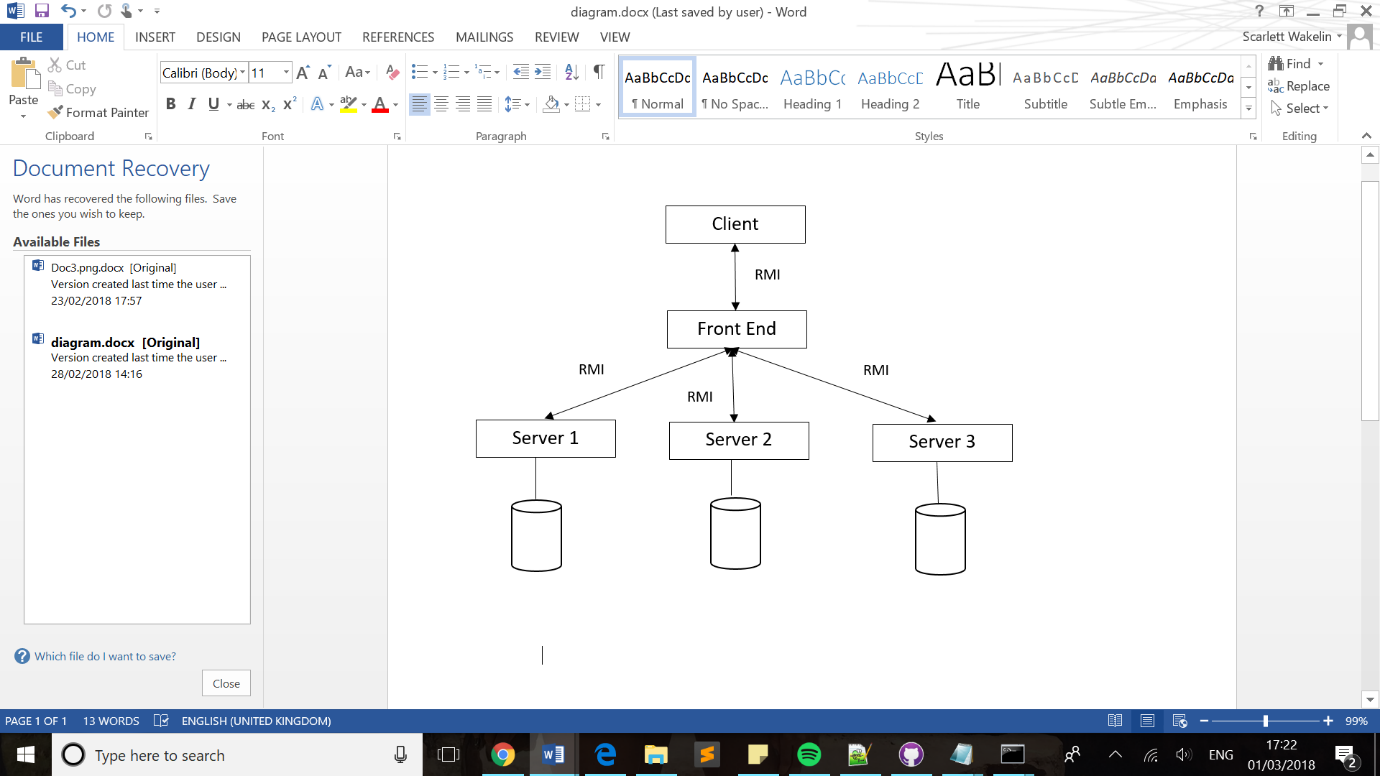
Networks and Distributed Systems Summative- Section B

1. a) Remote Method Invocation (RMI) is Java’s implementation of object-to-object communication among Java objects to make a distributed system. It allows us to distribute our objects on various machines, and invoke methods on the objects located on remotes sites. An advantage of this is it dynamically invocates new versions of remote objects. RMI does not require developers to deal with socket programming when implementing programs that support remote communication. Socket programming is a way of connecting two nodes on a network to communicate with each other, one socket listens at a particular port at an IP, while the other socket reaches out to the other to form a connection. RMI makes it easy and simple to connect separate pieces of software together, independent of their location, connectivity mechanism and technology used to develop them. Sockets are low-level and require one to either use an existing protocol or invent one, whereas RMI wraps sockets with an object-orientated interface so one can concentrate on programming rather than socket detail. Also services comprising multiple servers are easier to build and all services can be acquired through the object request broker. Another advantage of RMI over socket programming is that the geographical complexity and changes of services are hidden.

b) The Front End is a part of the distributed system that makes replication transparent, it monitors and maintains replica availability and performs request distribution and collates responses. The distributed system should be perceived as a single entity by the user rather than a collection of cooperating systems, the users should be unaware of where the services are located and sthe number of services.

2.a)