Outline Grading Scheme:

My marking approach will be to evaluate the 4 criteria below and to assign a grade that reflects the average those criteria. Please remember that these are just guidelines, and marking will not simply be a box ticking exercise.

Grade	Technical Requirements	Problem Complexity	Solution Quality	Report Quality
A	Use of additional distributed technology not covered in course.	Complex Problem that requires distribution to solve	Excellent solution that follows best practice.	Excellent reflection on pros- and cons- of approach.
B	Use of REST or Web Services in project	Reasonably complex problem that could be distributed.	Good Solution to Problem	Well written report
C	Use of RMI in project.	Problem that requires distribution	Adequate solution to problem	Adequately written report
D	Use of Basic Networking in project.	Distribution is not necessary for this problem.	You made something that works	A report was submitted.
E	No distribution technology used.		Code does not work	No Report Submitted.

NOTE:

- *Simply using an additional technology is not enough to get the A grade – you need to apply it to an appropriate problem, demonstrate its correct use in solving that problem **and** submit an excellent report.*
- *You can use **more than one** additional technology if you want to!*

Examples of technologies that have been mentioned, but qualify for A grade:

- Zeroconf / jmDNS
- Actors
- GraphQL
- Netflix Eureka (for RESTful service discovery)
- Java Messaging Service (JMS)
- Swagger (for specifying web services)
- Any other distribution technology you want to use (Apache Zookeeper, Spark, …)
- I will also consider the use of alternate data storage techniques, such as NoSQL databases, Key-Value stores, … so long as they have been used in a distributed context.