Distributed Systems (H0N08A)

Report: JAVA RMI

Dries Janse *(r06)*
Steven Ghekiere *(r0626062)*

October 28, 2019

1. How would a client complete one full cycle of the booking process, for both a successful and failed case? Base yourself on the example scenarios in Figure 1. Create sequence drawings to illustrate this. Answer 1

2. When do classes need to be serializable? You may illustrate this with an example class. Answer 2

3. When do classes need to be remotely accessible (Remote)? You may illustrate this with an example class. Answer 2

4. What data has to be transmitted between client and server and back when requesting the number of reservations of a specific renter? Answer 2

5. What is the reasoning behind your distribution of remote objects over hosts? Show which hosts execute which classes, if run in a real distributed deployment (not a lab deployment where everything runs on the same machine). Create a component/deployment diagram to illustrate this: highlight where the client and server are. Answer 2

6. How have you implemented the naming service, and what role does the built-in RMI registry play? Why did you take this approach? Answer 2

7. Which approach did you take to achieve life cycle management of sessions? Indicate why you picked this approach, in particular where you store the sessions. Answer 2

8. Why is a Java RMI application not thread-safe by default? How does your application of synchronization achieve thread-safety? Answer 2

9. How does your solution to concurrency control aspect the scalability of your design? Could synchronization become a bottleneck? Answer 2