KU LEUVEN DEPARTMENT OF COMPUTER SCIENCE

Distributed Systems: Java RMI session 2 & 3

Xavier Dejager
Seppe Duwé

December 4, 2015

1 GAE EXERCISE 3.2

The indirect communication kicks in when we confirm the quotes into actual reservations. This is the point in time where we send the quote to a queue in the back end by means of a deferred task. This task is then executed somewhere in the future by a worker that is assigned to that specific task. The data passed between the front end and the back end are objects of the `QuoteConfirmDeferredTask` (which is an implementation of the `DeferredTask` interface). Inside this `QuoteConfirmDeferredTask` one can find a list of `Quotes` that need to be confirmed. It is not really necessary to persist those objects, because they are discarded once we have a reservation or a failed reservation object. And those latter object will be persisted.

2 GAE EXERCISE 3.3

It is possible for multiple client to make quotes at the same time. But it is not possible to make simultaneous confirmations within the same car rental company unless the system uses more than one worker on a given moment. Then there is a chance that both workers read the same data while they should read different values. Possible solutions could be to use only one worker or another solution could also be to check the database for consistency on regular intervals. If the one worker solution is a bottleneck, a possible improvement could be to use multiple queues, one queue per car type. In that manner throughput is drastically increased without sacrificing consistency.

3 Diagrams

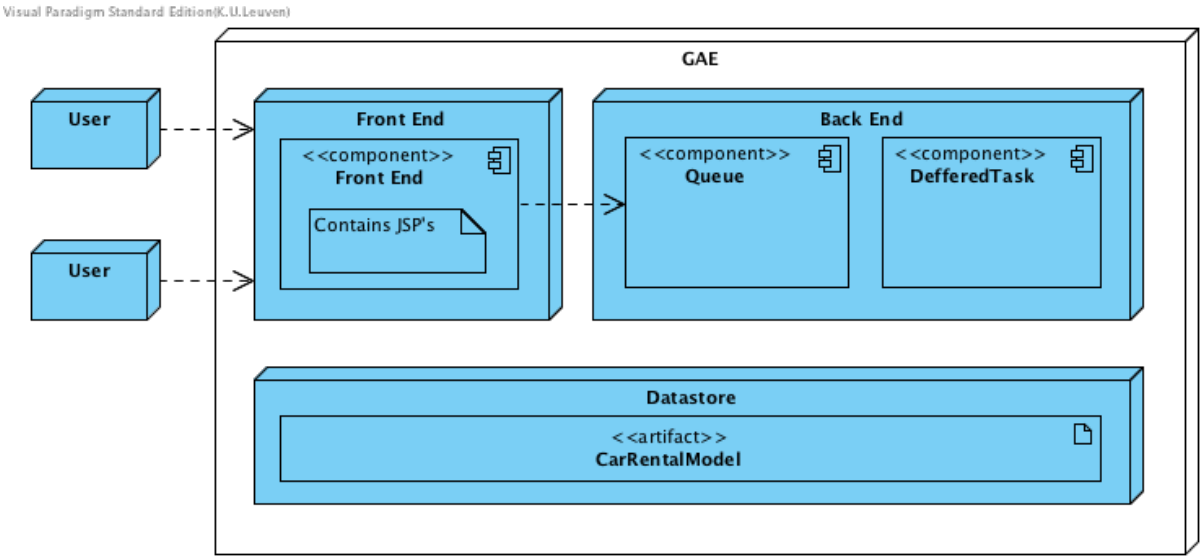

Figure 3.1: Deployment diagram

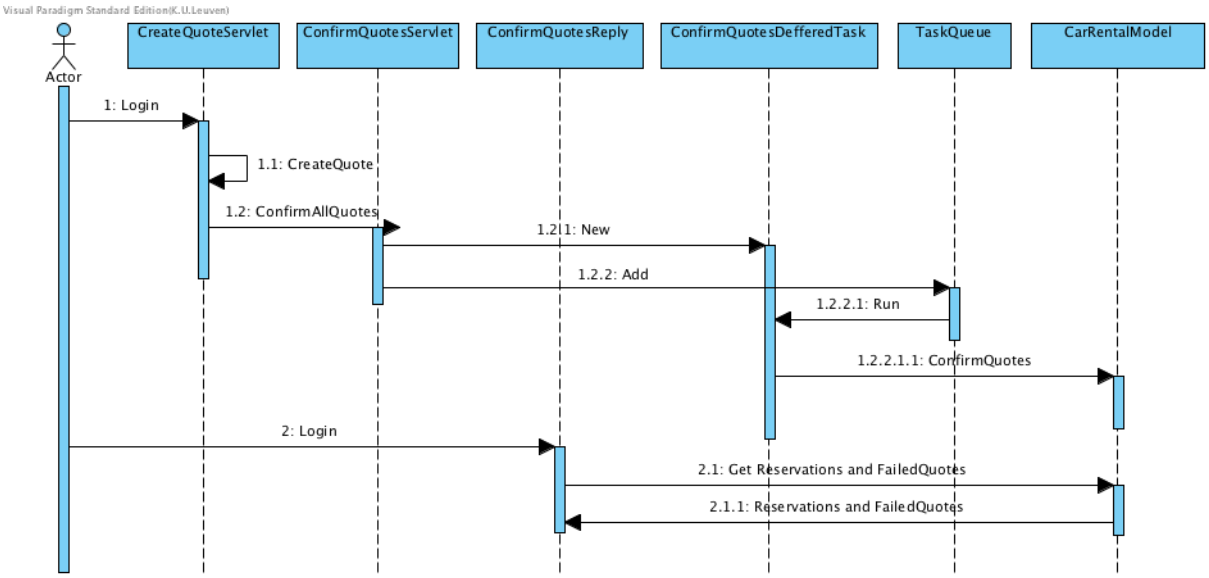

Figure 3.2: Sequence diagram