



Figure 6.19 Sequence diagram for the 'Issue bike' scenario with shading to show processing

- `issueReceipt()` stops processing when it sends the `getDetails()` message to `:Customer`
- Senders regain control as the operations finish processing.

Iteration. The normal assumption on an interaction diagram is that the object icon at the top of a lifeline represents only one object. However, sometimes we want to send the same message to many objects. For example, in the 'Issue bike' use case, when the Receptionist is searching for the details of the bike with a specific bike number, the same message is sent to all of the bike objects until we get a match. This is indicated on the sequence diagram by the iteration marker (*), see Figure 6.20. We can also, optionally, specify how many times the message should be sent. The iteration clause, 'until bike# matched', is shown in square brackets after the asterisk. Bases for iteration are:

- The number of times the message is iterated, e.g. `[i = 1..4]`
- Repetition while an expression is true, e.g. `[while more bikes]`
- A for loop, e.g. `[for all customers]`.

On the collaboration diagram, the UML uses a stacked icon to indicate a plurality of `:Bike` objects, see Figure 6.21. This is known as a *multiobject*.