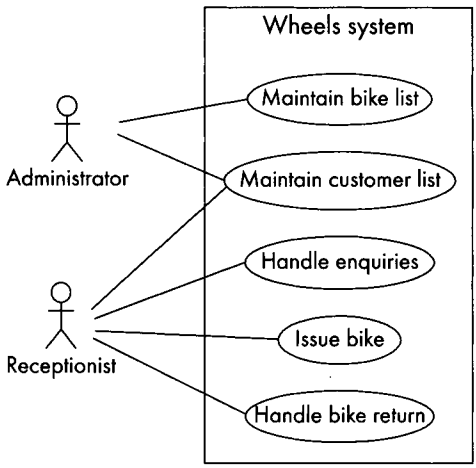

Figure 3.18 Use case diagram for Wheels (Figure 3.2 repeated)

However, what the question often means is: on any given occasion can more than one actor be involved in a use case? For example, in the Wheels system could we model the customer as an actor and then have both Customer and Receptionist involved in the use cases 'Issue bike', 'Handle enquiries' and 'Handle bike return'. The answer is that it is really a question of style (see the discussion in the section on initiating and participating actors). In this book our boundaries represent the automation boundary, i.e. include only the computer hardware and software inside the system. The actors, therefore, are the people who use the system directly, i.e. the Receptionist and the Administrator. Our models, therefore, do not show the Customer as an actor. If we made Wheels a more sophisticated system, where the Customer could make enquiries and hire a bike online, then Customer would also be modelled as an actor.

2 What dictates the granularity of use cases, i.e. how do we know how big or how small to make them?

The first and principal rule that dictates the size of a use case, i.e. what activities are included in it, is what the user sees as a complete job. A use case should represent a complete process; one end-to-end pass through the system, a job that the user sits down at the computer to achieve at one go. Don't confuse steps or stages in a process with the whole use case. For example, we have modelled 'Customer pays amount due (deposit plus rental charge)' as part of the use case 'Issue bike'. This is because we found that making a payment was never an event that was done in isolation: it was always done as part of the hiring process. With 'Handle enquiries', however, we found that, although this