- A customer, Steve Chen, arrives at the shop with a bike to return
- Annie contacts the mechanics to ask for someone to come and check the bike
- Annie gets the bike number, looks out the relevant bike card and checks Steve's name and address
- Annie makes sure that the bike being returned is the one on the card
- She confirms that the bike is being returned on time by checking the return date against the current date
- One of the mechanics checks the bike and confirms that it has been returned in good condition
- Annie returns Steve's deposit.

Figure 2.4 Simple scenario for the return of a bike in the current Wheels system

questionnaire and instructions on how to return it are stated clearly at the top, and there are different types of question. Most importantly, the questionnaire is relatively short and will only take customers a few minutes to fill in; it could easily be completed while customers are waiting for their bikes to be checked.

Scenarios. Scenarios have been popular as a method of requirements elicitation for many years and have now become closely associated with object-oriented development of systems. A scenario is a sequence of interactions between a user and the system carried out in order to satisfy a specified goal. Scenarios may be recorded in a variety of ways, including diagrams, storyboards or even videos, but they are generally documented in textual form as in the examples in this book. Figure 2.4 shows a scenario from the current Wheels system where the specified goal is to return a bike successfully.

As we can see from the example in Figure 2.4, a scenario is a sequence of particular events, not a general description. This example illustrates the straightforward return of a bike, without any problems. However, this is only one possible scenario for a bike return, and it is the developer's job to find out what happens in all possible cases. Figure 2.5 shows another scenario for bike return, which is slightly more complicated.

It is also important for the developer to find out about what happens when a goal is not achieved, for example a customer may come into Wheels to return a bike, but the bike card is missing. A scenario should be written for this sort of situation in the same way as for normal cases, so that the developer knows how the system should respond when things are not straightforward.