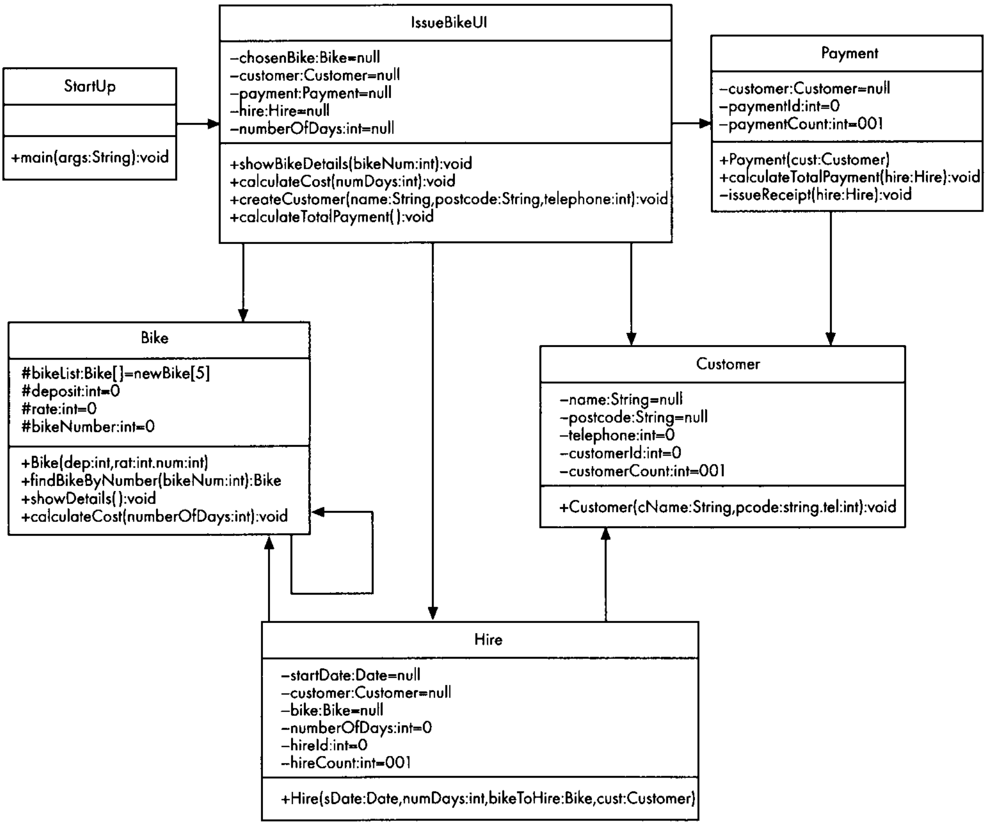

Figure 11.1 The implementation class diagram

end consisting of a welcome menu and other layers of interface screens. The class effectively simulates the function of Receptionist as modelled in the sequence diagram in Figure 10.11, reproduced here as Figure 11.2. The four method calls, showBikeDetails(), calculateCost(), newCust() and calcTotals(), sent by the Receptionist to :IssueBikeUI, are now made from StartUp.

IssueBikeUI class. This class has already been introduced in Chapter 9. It combines the functions of controller and interface class. Its first four attributes, chosenBike, customer, payment and hire are all used to hold the object identifiers of (or references to) objects with which the IssueBikeUI needs to interact. These attributes are used to implement the unidirectional navigable paths shown on the class diagram issuing from IssueBikeUI. The last attribute, numberOfDays, holds the length of a hire. It is set when calculateCost(numDays) is called and subsequently used as a parameter when the Hire object is created.