SOFTWARE ENGINEERING 2021
ASSIGNMENT

Due Date: 10:00pm, Thursday, 30 Dec 2021

Problem Statement

VinaRent is a company in car rental business. It has over 100 branches all over the country. It has about 5,000 cars and makes about 500,000 rentals per year. The rentals spread across 100 branches: branches in major airports, branches in major cities, and branches in local agencies such as hotels and garages. Each branch is identified by a branch number.

VinaRent has several IT systems. We are concerned with one of them. This system is called *Vina Rental System*. Its main purpose is to maintain information of the cars and the rentals of those cars. It is also used to manage the car fleet, e.g. moving cars from branch to branch when necessary.

It is decided that the system will be constructed as a *central* system which has access to the information about every VinaRent branch. Later on, when it is installed at a branch, it will be customized to restrict its access scope to an appropriate level.

VinaRent Cars

Each car has a model, identified by a model number. Useful information about a model includes a short description (which usually includes the model's marketing name), automatic or manual, petrol consumption (such as 1.5 liters or 6 cylinders), number of doors. The cars, on the basis of their model, are divided into five groups, group 'A' to group 'E', and all the cars in a group have the same rental price.

VinaRent Customers

VinaRent estimates that about 15,000 customers per year are served, of whom about 12% rent frequently, 40% rent between 3 to 5 times per year, and 50% are one-off renters in the sense that they use VinaRent once per year or less.

In fact, VinaRent shares a (much larger) customer information base with other businesses including various airlines and hotel providers. However, the sharing of customer information is done through a separate system. This is done so transparently (for both querying and updates) that we think of the shared customer base as part of the current system (for querying and adding customers).

A *blacklist* of customers is maintained by VinaRent. It is updated periodically based on the reports from the branches. The maintenance of the blacklist follows a separate process that we do not need to be concerned with. We can simply assume that the list is available for querying.

A *discount customer list* is also available. Customers on this list are given a 10% discount for their car rents. The list is maintained by a separate system. We can simply assume that the list is available for querying.

Thus, regarding the customers, a list of customers, a blacklist, and a list of customers entitled for discount are available for *querying*. In addition, the branches are also allowed to *add* new customers to the customer list.

VinaRent Car Rental Activities

The main activity at the branches is, of course, to rent cars on customers' requests. Customer can reserve a car. They can also walk-in to request and pick up a car. Thus, the rentals involve the following activities: (1) answering customers' enquiries, (2) making reservations (3) recording walk-in rentals, (4) recording car pick-up, (5) recording returns of cars (may be at a different branch form the pick-up one, late returns possible), etc.

The company's headquarter is responsible for purchasing and disposing off cars. When a car is bought, its information (including its model if necessary) is entered into the system. Depending on their conditions and customer demand, cars may be removed from the fleet to be disposed off (e.g. though sales).

Descriptions of Main Business Processes

The following descriptions of the main business processes have been produced. It is acknowledged that some part of the descriptions might be neither clear nor complete. They are meant to serve as a starting point for further analysis activities. As you carry out the required tasks, you can refine or enhance them with additional details and business processes if you think it is necessary. In such cases, clearly state your assumptions.

Business Process: Renting a Car (BP1)

Starts when: Customer requests for a car

Ends when: The car is returned (normal case)

Description:

1. A customer reserves a car with a branch. This is dealt with in the subprocess Reserving a Car (BP1.1a)

   Alternatively, a customer can request and pick up a car without reservation. See subprocess Picking-up a Car without Reservation (BP1.1b)
2. The branches must make sure that the car is available for pick-up. See subprocess Moving Cars on Request (BP1.2)
3. The customer picks up a car that has been reserved. See subprocess Picking-up a Reserved Cars (BP1.3)
4. The customer returns the car. See subprocess Recording the Returning of a Car (BP1.4)

Business Process: Reserving a Car (BP1.1a)

Starts when: Customer requests for a car

Ends when: A car is reserved for the customer (normal case)

Description:

1. A customer can rent a car by email, by phone or by walking into an office, making an enquiry or a request.
2. The booking clerk gets the customer's requirements. Typical requirements are as shown below:

Location of Travel	
Pick-up Branch: Tan Son Nhat Airport Return Branch: Tan Son Nhat Airport	
When	
Pick-up Date: 20-AUG-2010	Time: AM 10.00
Return Date: 22-AUG-2010	Time: AM 10.00
Type of Car (Make one or more selections)	
Economy: ✓ Compact Car: Medium Size: ✓ Full Size:	Station Wagon: Van: 4 Wheel: Sports/Luxury Car:

3. The clerk finds out from the system if an appropriate car is available at the requested branch or a neighboring branch and the cost (which is often of interest to the customer).

   Branches A and B are neighboring branches if any car at A can be requested to be moved to B overnight and vice versa (see subprocess BP1.2). Each branch has a list of neighboring branches.

   To find out if a car is available is *not* a trivial task. We can apply simple search rules as well as quite sophisticated search rules. Currently, VinaRent uses the following simple search rule.

   Simple Search Rule:

   A car is available if

   (1) Its status is RENT-READY

   (2) It matches the types of car the customer asks for

   (3) It is at the requested pick-up branch or one of its neighboring branches

4. If no car is available, the clerk fills out a paper form which contains the requested pick-up branch, pick-up date and time, types of car wanted, and the customer contact phone number and/or email address. The supervisor may negotiate to have a car

delivered to the requested branch and inform the customer about it. (It is not part of the responsibility of the system to support this sub-process.)

5. Otherwise (i.e. one or more cars are available), the clerk selects a car and put it on hold, that is sets it status to “HELD” (so that other operators will not reserve it).

   NOTE ON ALTERNATIVE: If the deal does *not* go ahead as planned the clerk must release the hold on a car.

6. The clerk then asks for the customer’s driver license to check if the customer is blacklisted or not.

7. If the customer is new, the clerk will enter the new customer to the customer base. The following information is entered: the customer’s first name, last name, driver license, email address (if available), and contact phone number (if available).

8. If the customer is not black listed, a new transaction, which is called a *rental*, is created. The new rental has a unique rental number (generated by the system) and has its status set to “RESERVED”.

   In addition to information about the reserved car, the pick-up and the return (date, time, branch), the clerk also enters information about payment.

   A deposit of 10 % of cost (which may involves discount) has to be made by the customer. The payment can be made against the payment item “Deposit Payment” (the other payment item is “Cost Less Deposit Payment”). Payment amount and payment method (cash or credit card) are recorded. If the payment is made by credit card, the credit card’s details are obtained and validated. The validation is performed by a call to another system.

   The car assigned to this rental will have its status set to “RESERVED”.

Business Process: Picking-up a Car without Reservation (BP 1.1b)

Starts when: Customer walks in and requests a car

Ends when: The customer picked up a car (normal case)

Description:

1. A customer walks into the office of a branch and makes a request.

2. The booking clerk gets the customer's requirements (as in subprocess BP1.1a) and see if a car is available at the branch.

3. If it is the clerk asks for the customer’s driver license to check if the customer is blacklisted or not.

4. If the customer is not black listed, the clerk asks for payment and creates a new car rental. The same information recorded is as those described in the subprocess BP1.1a, except that the status of the new rental is marked as “PICKED-UP”.

   In addition, the car status is set to “PICKED-UP” and the mileage of the car is recorded for the rental (which is known as start-mileage).

NOTE ON ALTERNATIVE: The customer may be a new one. In this case, his or her details are entered to the system as described in the previous process.

Business Process: Moving Cars on Request (BP1.2)

Starts when: The supervisor starts arranging to move the cars requested for neighboring branches (done daily at certain time)

Ends when: Cars requested by other branches have been arranged to be moved

Description:

1. The supervisor prints out the list of cars at his/her branch which have been requested by other neighboring branches
2. The supervisor arranges for the cars to be transferred. (It is not part of the responsibility of the system to support this activity.)

NOTE ON SUBSEQUENT ACTIVITIES: When a car requested from a neighboring branch arrives, its new location is updated. This is actually a separate process though it is not currently described in this document.

Business Process: Picking-up a Reserved Car (BP1.3)

Starts when: The customer arrives at a branch to pick-up a car that has been reserved

Ends when: The car has been picked-up

Description:

1. If the reserved car is not available for some reason (e.g. overnight trip failed, which is rarely the case), the clerk will replace it by another car (if possible).

   In this case, the previously reserved car will have its status set to "EXCEPTIONAL", and the substitute car will have its status set to "HELD" (so that no one else can put it on hold).
2. If a substitute cannot be made, the clerk sets the status of the reserved rental to "EXCEPTIONAL" and also ensures that the status of the reserved car is set to "EXCEPTIONAL".

   The case is then referred to the supervisor to arrange for a refund. (It is not part of the responsibility of the system under study to support this process.)
3. Otherwise, the clerk checks the driver license to see if it matches the license used for the reservation. If a different driver license is used and the new driver is not blacklisted, a change of driver can be made.
4. The clerk asks for the rest of the payment and enters the payment details is entered. The payment item is "Cost Less Deposit Payment".
5. The clerk also enters the *actual* pickup date and time. The status of the rental is changed to PICKED-UP. The status of the car is also changed to PICKED-UP. The car's mileage is recorded again the rental record.
6. The customer picks up the car.

NOTE ON ALTERNATIVE: A customer may cancel a reservation or may simply not turn up. In either case, the rental will be cancelled and the customer losses their deposit. The rental's status is set to "EXCETIONAL", and the car's status set to "RENT-READY".

Business Process: Recording the Returning of a Car (BP1.4)

Starts when: A customer returns a car

Ends when: Details have been entered for the return

Description:

1. The clerk records end mileage, and the actual return date and time, and changes the status of the rental to "RETURNED".
2. If the car is returned to different branch (i.e. not the "return" branch recorded at the time of reservation or pick-up), this has to be recorded against the rental record.
3. The clerk changes the status of the car to RETURNED, and updates the car's residing branch (i.e. where the car is).

NOTE ON SUBSEQUENT ACTIVITIES: The car will need to be inspected *before* returning to the "active" pool ready to be rented. The car can also be set aside for services or to be removed from the fleet..

Business Process: Arranging for Car Maintenance (BP2)

Starts when: One or more cars need to be inspected

Ends when: Decisions have been made about the cars

Description:

1. The supervisor prints the list of cars to be inspected if necessary (for example, the list of cars that were returned to the current branch).
2. The supervisor inspects the cars. The outcome for a car can be that it is OK, or that it needs servicing, or that it is to be removed, and the supervisor will have its status updated accordingly, i.e. RENT-READY, SERVICE NEEDED, or REMOVED respectively.
3. Supervisor arranges for necessary actions. (It is not part of the responsibility of the system under study to support this activity.)

NOTE ON SUBSEQUENT ACTIVITIES: Later when a car has been serviced, and becomes available at a branch, the information about the car is updated.

Business Process: Adding a New Car to the Active Pool (BP3)

Starts when: A new car is acquired by the company and to be added to the active pool

Ends when: The car's details have been entered

Description:

1. If the model is new, the clerk enter the model's details, which include (a) the model number and a short description of the model (which usually includes the model's marketing name), (b) automatic or manual, (c) petrol consumption (such as 1.5 liters or 6 cylinders) , and (d) number of doors. The clerks also specifies which group ("A" to "E") the model is classified into
2. The clerk enters details of the car, including registration number, color, year of production, and the car's initial residing branch. The car's status will be RENT-READY.

Notes on Other Processes

Perhaps some other related processes which are not described here (for example the processes to generate various kinds of reports) will not be concerned in this assignment.

Tasks

For this assignment, you are required to do the following:

Task 1 – Use Cases [20%]

(a) Identify all the use cases, at the system level, to support the business processes described above.

The business processes you have to consider include those given as NOTE ON ALTERNATIVE or NOTE ON SUBSEQUENT ACTIVITIES (regardless of whether these processes are described separately or not).

If for certain processes to make sense, you think additional details or even additional processes need to be added, then by all means do so. In such cases, clearly describe your additional details or processes.

You *must* list your use cases against the steps in business process descriptions (given or added by you). Each step of the business process corresponds to zero or more use cases. Set up your answer in a table in the format shown below:

Business Process	Step	Use case	Remarks
<name of business process> (BP n)	1	<name of use case> (UC 1)	
		<name of use case> (UC 2)	
	2		
…	…	…	…
		…	…

(b) Describe each of the use cases identified in Part (a) of Task 1 using the Event Flow Format (with Main Flow/Extensions)

For ease of reference, give each use case a number.

Though there are dependencies among the use cases, you can *ignore* the specification of invocation constraints and post obligations. This is done to keep the descriptions simple. We do not seem to lose much information by not specifying those constraints and obligations.

SAMPLE SOLUTION:

The business process BP2 (Arranging for Car Maintenance) has been filled in the table as a sample solution for you as follows.

Business Process	Step	Use case	Remarks
…	…	…	…
Arranging for Car Maintenance (BP2)	1	UC 16: Print the List of Cars to be Inspected	
	2	UC 17: Record that a Car Needs to be Serviced	
		UC 18: Record that a Car is to be Removed	
…	…	…	…

Task 2 – Structural Domain Modeling [20%]

Construct a structural domain model in terms of one or more (analysis) class diagrams. Clearly state any assumptions you make.

Task 3 – Design Class Diagrams [20%]

Based on the analysis class diagram in Task 2, your task is to construct the design class diagrams to show the complete design of the system. If there are more than 1 design option, present all of these design options, and present your final design decision (if any).

This class diagrams must be completed with attributes (including attributes representing navigations), attribute types, methods with complete operation signature. Include all – and only - methods that are required for the atomic use cases listed in Task 4 (including the constructors).

Note: You could suppress the signatures (i.e., without parameters and data types) on a class diagram and provide each class separately where the methods' signatures are shown.

Task 4 – Atomic Use Case Specifications [20%]

Assume that we have identified the following atomic use cases:

1. Add a branch
2. Make a pair of branches neighbors to each other
3. Add a car rental group
4. Add a model
5. Add a car
6. Add a customer
7. List cars that are available at a specified branch and belong to a specified rental group (do not include the cars at neighbor branches)
8. Record the return of a car

Your task is to specify the atomic use cases listed above, using the formal specification language introduced in the course. In your answer, number the use cases as shown.

The following points must be observed:

- Your specifications must be based on *your design class diagrams*.
- Your specification must also be based on the *problem description given in the description*, except where we make explicit changes for the prototype (e.g. Customer has attributes to indicate if they are blacklisted or entitled to discount) or simplifications (e.g. we treat dates as integers)

Task 5 – Protype and Testing the Prototype [20%]

a. Prototype all the atomic use cases in Java (or other object-oriented programming language of your interest).

Your implementation of the prototype must be done in a systematic manner. In particular, for each use case, the preconditions should be checked first, and the postconditions should then be satisfied.

b. For each atomic use case, design the test cases and include them in a Java program, called VinaRentSystemTest, to carry out the testing. Screenshots of these test cases should be attached for their execution demo.

The nature of the problem statement (in reality) is not necessarily 100% correct or complete at the beginning. It is your team's responsibility to collect sufficient requirments from your customer, and clearly state reasonable assumptions if any.

SUBMISSION

A. Your team is required to submit an MS Word report with the following contents:

1. Cover page (with the **contribution percentages** of all the members, each member associated with one single percentage number)
2. Task 1: (a) all use cases, (b) use case descriptions
3. Task 2: analysis class diagram(s)
4. Task 3: design class diagram(s)
5. Task 4: atomic use cases
6. Task 5: a prototype and testing with screenshots.
7. Appendix (if any)

B. Your MS Word report must be named as: **TeamName-Name1Name2Name3.docx,**

where Name1, Name2, Name3: are names of your team members.

C. Your report (.docx) and other files (e.g., source code, etc) should be zipped into a single file with the file name **"TeamName-Name1Name2Name3.zip".**

a) This zip file must be submitted to our course's Blackboard under the Assignment folder.

b) As a backup solution, your team leader must email your zip file to the instructor: iu.subjects@gmail.com, AND email (cc) it to ALL OF YOUR TEAM MEMBERS. This aims to **ensure all the team members acknowledge their real contributions to the work**, which will eventually affect their scores.

D. IMPORTANT:

Please make sure your work **not plagiarize** others' work. Committing plagiarism will lead to a severe penalty based on The University's academic misconduct policy.

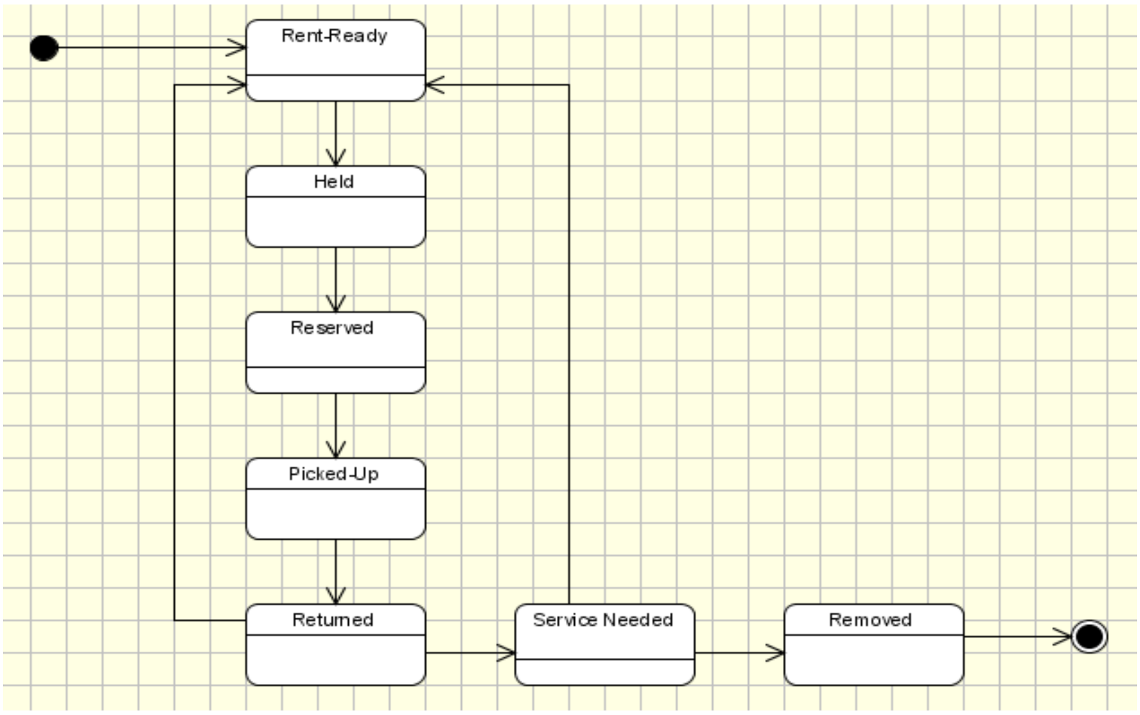
APPENDICES

THE BELOW STATECHARTS FOR REFERENCE ONLY:

To accomplish all the tasks above, it may be useful for you to consider the statecharts below for (a) Car, (b) Rental with Reservation, and (c) Rental without Reservation.

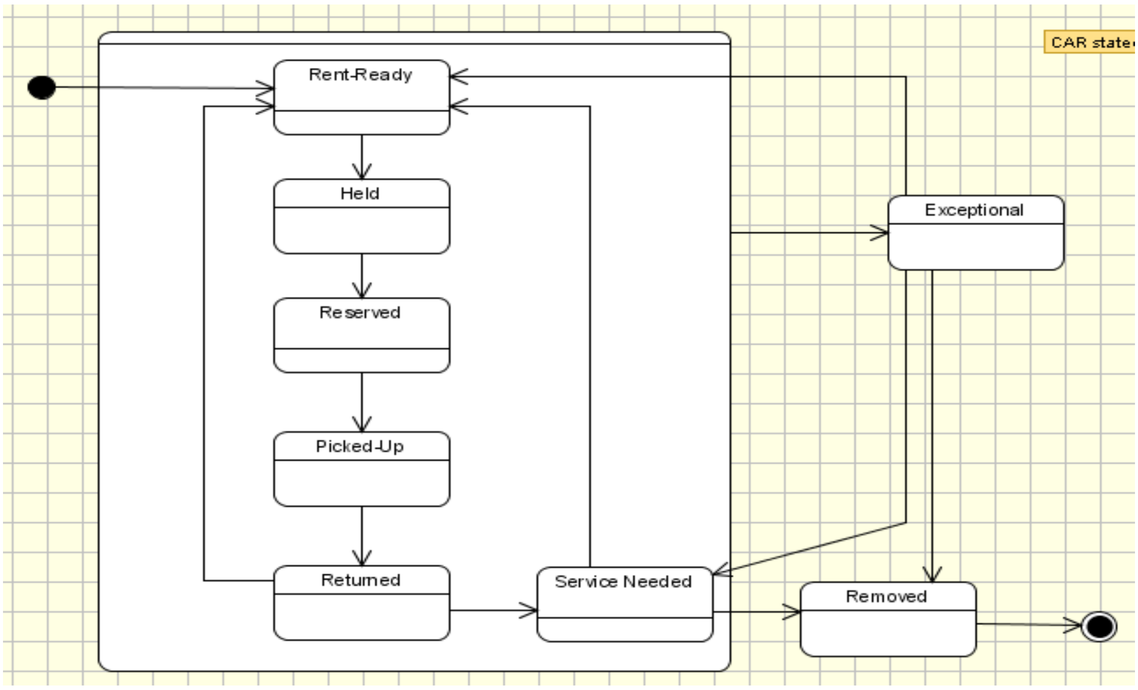
Statecharts for Car

Basic Statechart – with exceptions being left out



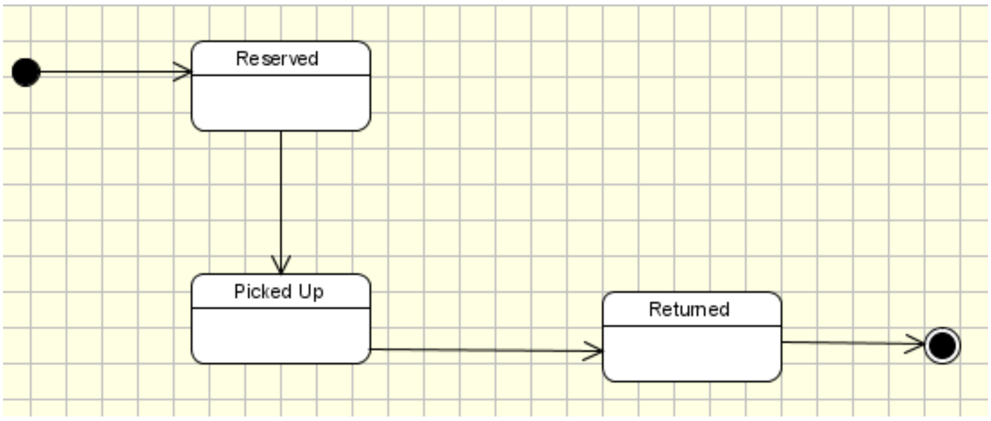
From any state – except “REMOVED” – we can go to the special catch-all state “EXCEPTIONAL”. This state covers the cases: Customer cancels reservation, or Reserved car is not available, or VinaRent cancels a reservation, or a Picked-up car has an accidence, etc. Each time, one of these events happen, the car is moved to the state of “EXCEPTIONAL”. From this state, it can go to: RENT-READY, SERVICE NEEDED, or REMOVED.

Statechart – With exceptions being modeled by a catch-all EXCEPTIONAL state

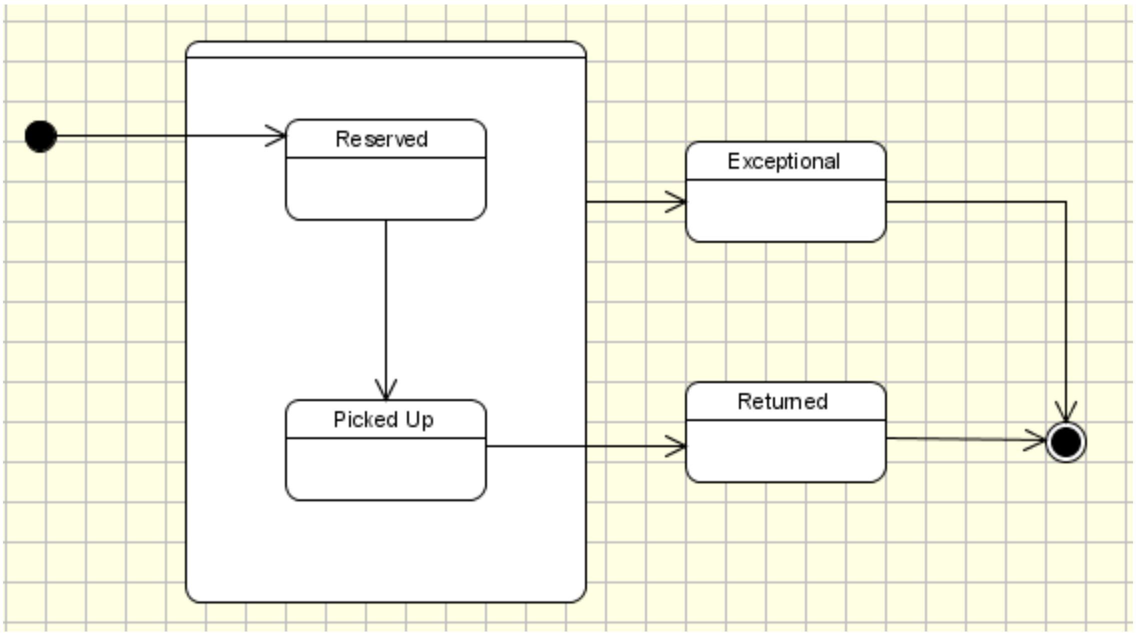


Statecharts for Rental with Reservation

Basic Statechart - With exceptions being left out



Statechart - With exceptions modeled by the Exceptional state



For Rental without Reservation

The statecharts are the same as for Rental with Reservation, except that they do not have Reserved state.

■