Visual Paradigm for UML

FROM TEXTUAL DESCRIPTION TO UML USE CASE DIAGRAM

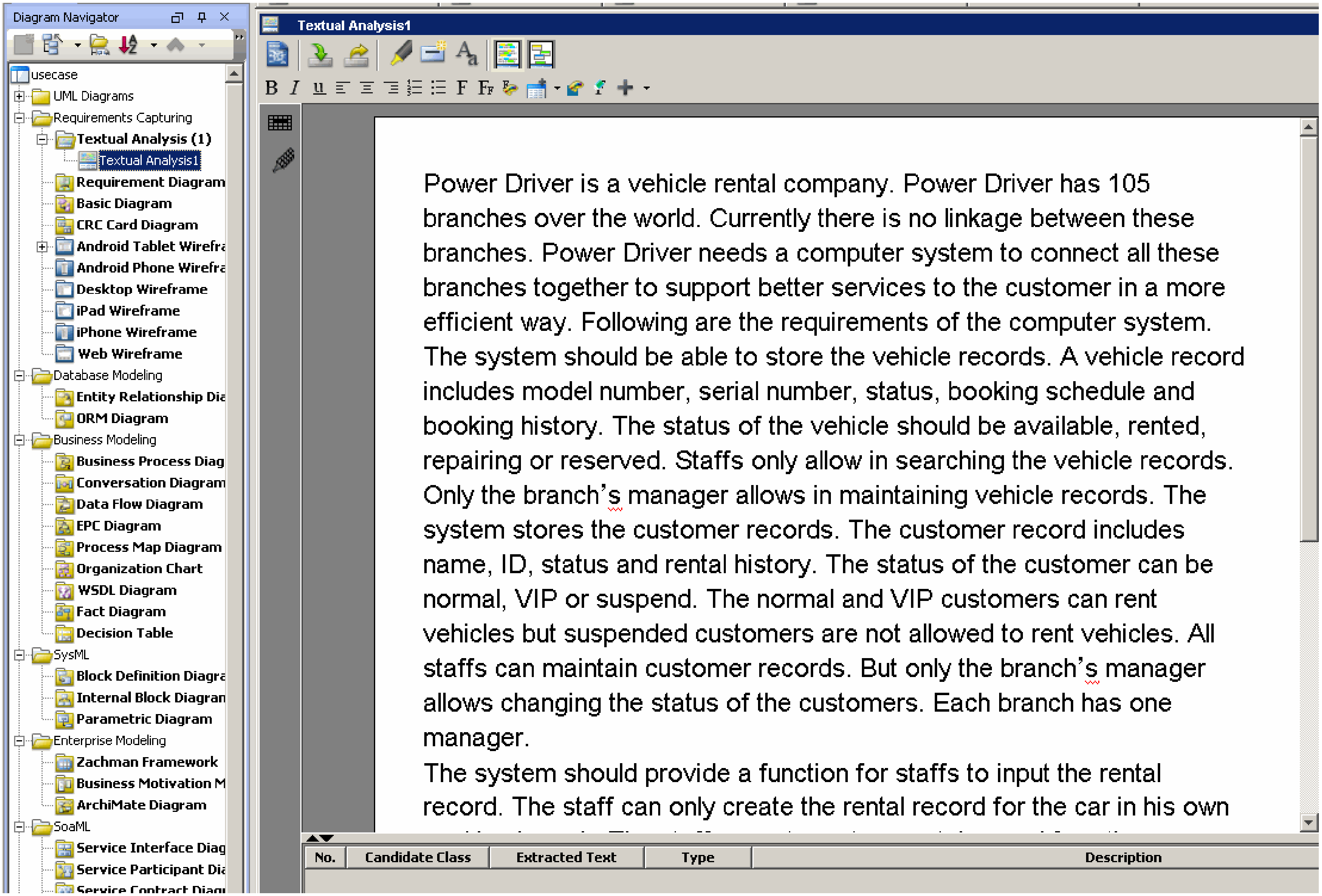
Mario G.C.A. Cimino

Department of Information Engineering

Pisa, Sept-Dec 2018

1. Textual Analysis and candidate items

1. Diagram Navigator \ Requirements Capturing \
2. Right click on Textual Analysis → *New Textual Analysis*
3. Paste or import textual description

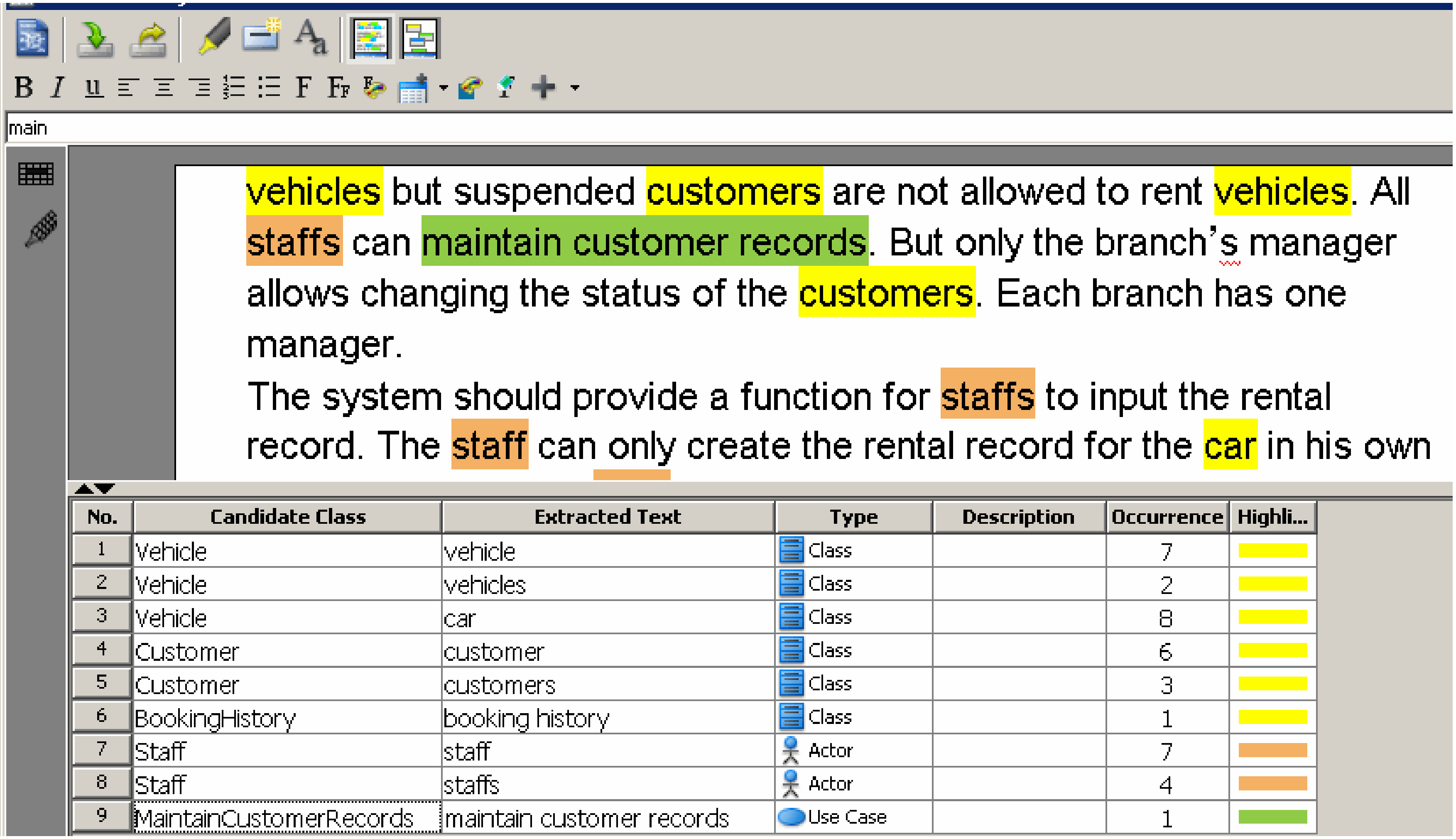


Power Driver is a vehicle rental company. Power Driver has 105 branches over the world. Currently there is no linkage between these branches. Power Driver needs a computer system to connect all these branches together to support better services to the customer in a more efficient way. Following are the requirements of the computer system. The system should be able to store the vehicle records. A vehicle record includes model number, serial number, status, booking schedule and booking history. The status of the vehicle should be available, rented, repairing or reserved. Staffs only allow in searching the vehicle records. Only the branch's manager allows in maintaining vehicle records. The system stores the customer records. The customer record includes name, ID, status and rental history. The status of the customer can be normal, VIP or suspend. The normal and VIP customers can rent vehicles but suspended customers are not allowed to rent vehicles. All staffs can maintain customer records. But only the branch's manager allows changing the status of the customers. Each branch has one manager.

The system should provide a function for staffs to input the rental record. The staff can only create the rental record for the car in his own

No.	Candidate Class	Extracted Text	Type	Description

4. Right click on the term *vehicle, vehicles, car, cars* → Class
5. Right click on the term *customer* , *customers* → Class
6. Right click on the text *booking history* → Class
7. Right click on the term *staff, staffs* → Actor *Staff*
8. In the underlying table, make uniform the field *candidate class* for singular/plural forms and synonyms by using the *camel case format* (e.g. *booking history* → *BookingHistory*)
9. Right click on the text *maintain customer records* → Use case *MaintainCustomerRecords*



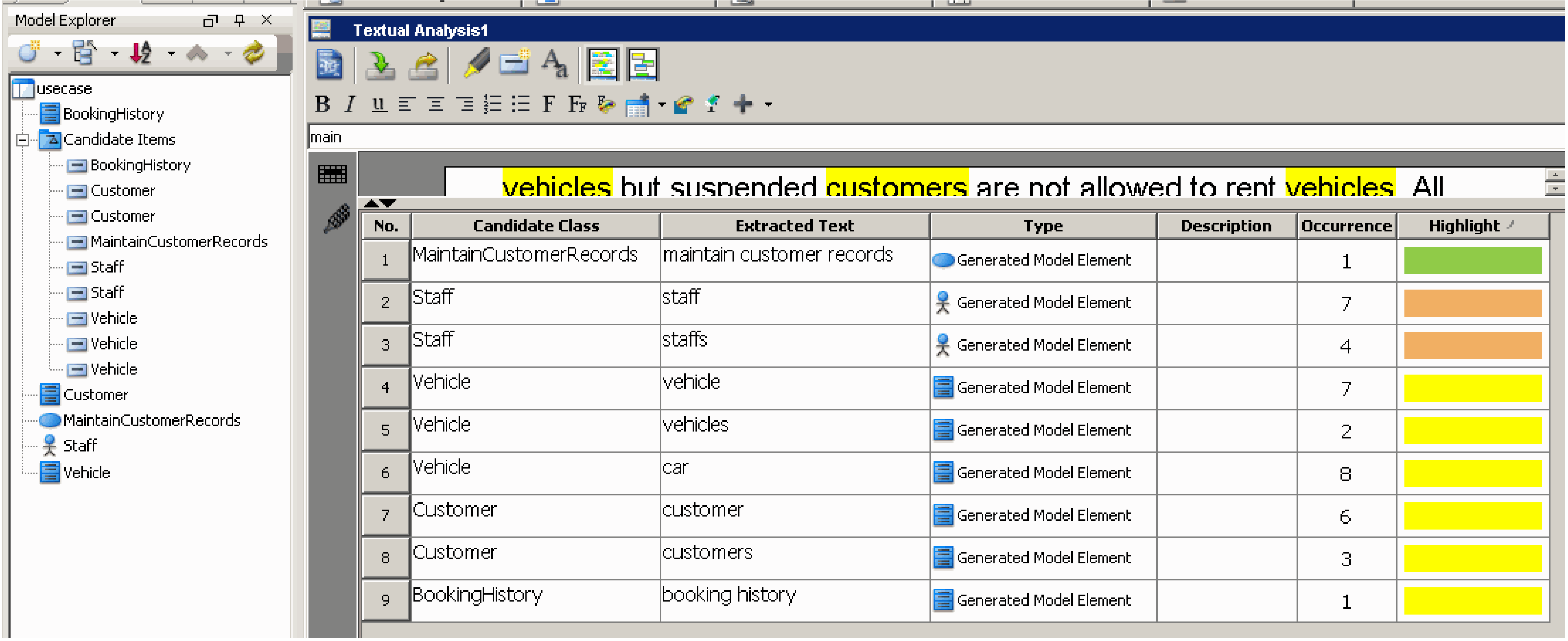
vehicles but suspended customers are not allowed to rent vehicles. All staffs can maintain customer records. But only the branch's manager allows changing the status of the customers. Each branch has one manager.

The system should provide a function for staffs to input the rental record. The staff can only create the rental record for the car in his own

No.	Candidate Class	Extracted Text	Type	Description	Occurrence	Highli...
1	Vehicle	vehicle	Class		7	🟨
2	Vehicle	vehicles	Class		2	🟨
3	Vehicle	car	Class		8	🟨
4	Customer	customer	Class		6	🟨
5	Customer	customers	Class		3	🟨
6	BookingHistory	booking history	Class		1	🟨
7	Staff	staff	Actor		7	🟧
8	Staff	staffs	Actor		4	🟧
9	MaintainCustomerRecords	maintain customer records	Use Case		1	🟩

2. Model Elements

1. Right click on the row number corresponding to a candidate element → *Create Class Model Element → Do not visualize → close.*
2. If a model element with the same name already exists, the system asks whether the existing model element can be used. For synonym, plural/singular forms, select *yes*
3. Generated elements appear in the *Model Explorer* tab (on the left).
4. A model element which does not appear in any diagram is called ***hidden*** element. A classifier which appears in a diagram without a corresponding model element is called ***ghost*** *element*.



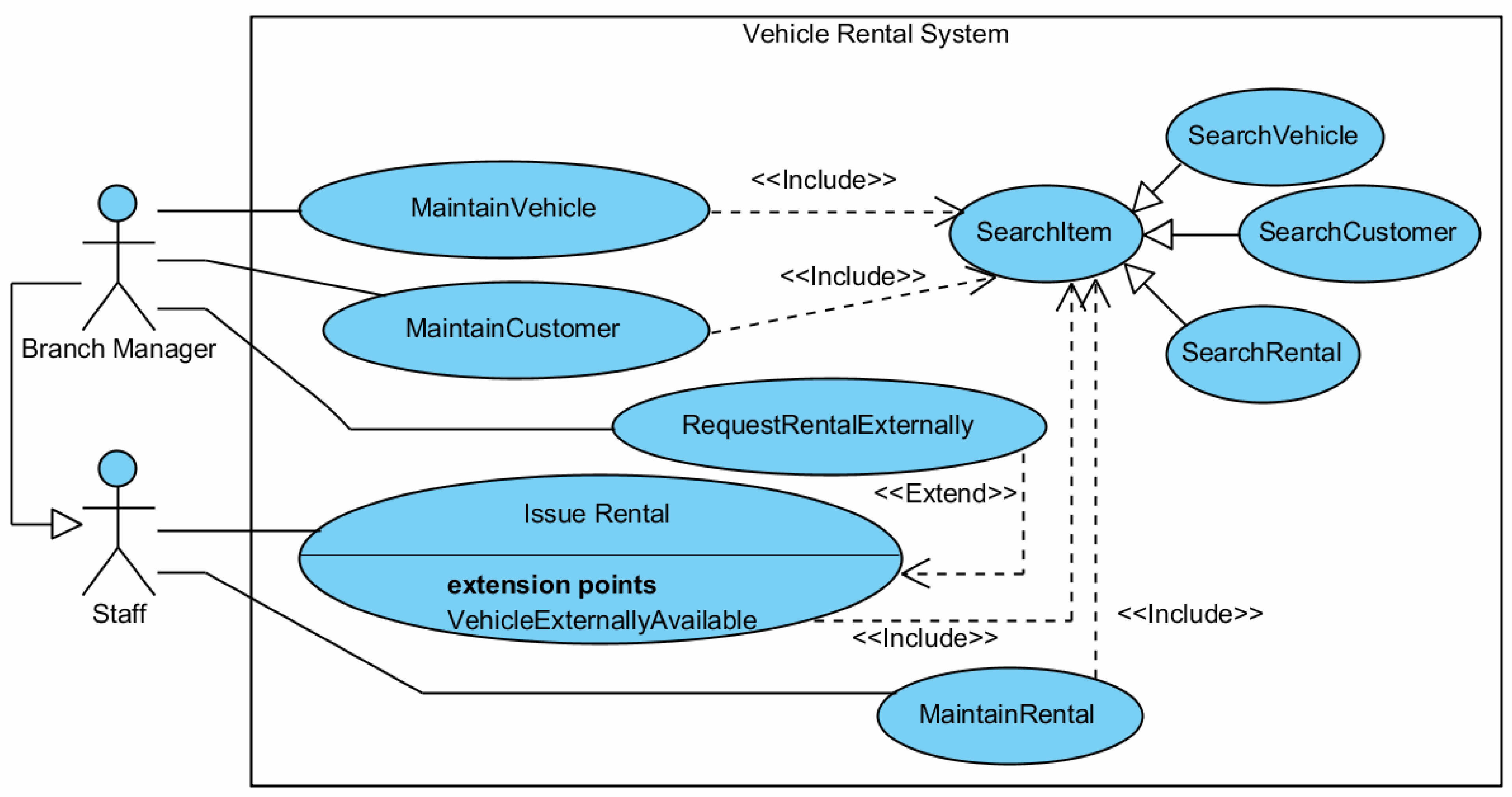
vehicles but suspended customers are not allowed to rent vehicles. All

No.	Candidate Class	Extracted Text	Type	Description	Occurrence	Highlight
1	MaintainCustomerRecords	maintain customer records	Generated Model Element		1	🟩
2	Staff	staff	Generated Model Element		7	🟧
3	Staff	staffs	Generated Model Element		4	🟧
4	Vehicle	vehicle	Generated Model Element		7	🟨
5	Vehicle	vehicles	Generated Model Element		2	🟨
6	Vehicle	car	Generated Model Element		8	🟨
7	Customer	customer	Generated Model Element		6	🟨
8	Customer	customers	Generated Model Element		3	🟨
9	BookingHistory	booking history	Generated Model Element		1	🟨

5. Model elements can be dragged to diagrams, generating views.

3. Use Case Diagram

1. Right click on the Diagram Navigator \ Use case Diagram → *New Use Case Diagram.*
2. Drag the actor and the use case model elements *Staff* and *MaintainCustomerRecords* from the Model Explorer to the Use Case Diagram
3. Rename *MaintainCustomerRecords* to *MaintainCustomer*
4. Create new elements in the diagram using the toolkit (on the left).
5. To create a relationship between elements: hover mouse over an element, choose the relationship, drag the chosen relationship to the other element.
6. Define *BranchManager* as an extension of *Staff* (a *manager* is as an employee with other additional capabilities. A manager can take the place of any employee (*substitutability principle* in object-oriented models). As an effect, ***redundancy*** in the diagram is reduced.
7. Define an extension point over the *IssueRental* use case, entitled *VehicleExternallyAvailable* connected to *RequestRentalExternally*
8. Add a *SearchItem* use case, included by the other use cases, with three specializations *SearchVehicle, SearchCustomer* and *SearchRental*.



J. Arlow, *UML and the Unified Process*, pages to read:

- from page 58 (Section 4.3.2) to page 62 (Section 4.3.3)
- from page 79 (Section 5.2) to page 82 (Section 5.3)
- from page 84 (Section 5.4) to page 88 (Section 5.5.1)