Textual Analysis is a useful technique that helps you discover useful model data from a given text, such as a problem statement, business case, use case description, domain descriptions, legacy model description, and even legacy code.

In this tutorial, you will learn how to identify classes from a problem description. After that, a class diagram and sequence diagram will be formed.

Problem Description

The following problem description will be used in this tutorial. Let’s take a look.

Saturn Int. management wants to improve their security measures, both for their building and on site. They would like to prevent people who are not part of the company to use their car park.

Saturn Int. has decided to issue identity cards to all employees. Each card records the name, department and number of a company staff, and give them access to the company car park. Employees are asked to wear the cards while on the site.

There is a barrier and a card reader placed at the entrance to the car park. When a driver drives his car into the car park, he/she inserts his or her identity card into the card reader. The card reader then verify the card number to see if it is known to the system. If the number is recognized, the reader sends a signal to trigger the barrier to rise. The driver can then drive his/her car into the car park.

There is another barrier at the exit of the car park, which is automatically raised when a car wishes to leave the car park.

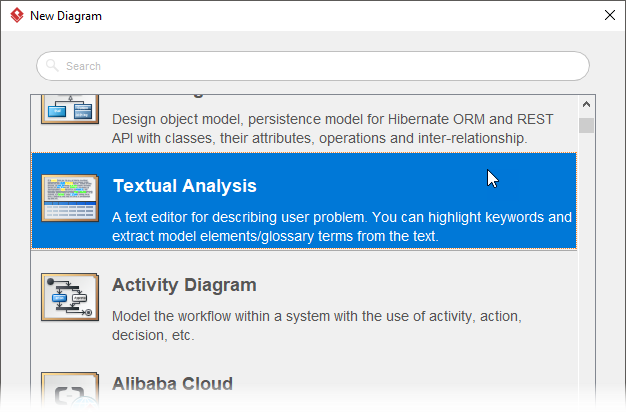
A sign at the entrance display “Full” when there are no spaces in the car park. It is only switched off when a car leaves.

There is another type of card for guests, which also permits access to the car park. The card records a number and the current date. Such cards may be sent out in advance, or collected from reception. All guest cards must be returned to reception when the visitor leaves Saturn Int.

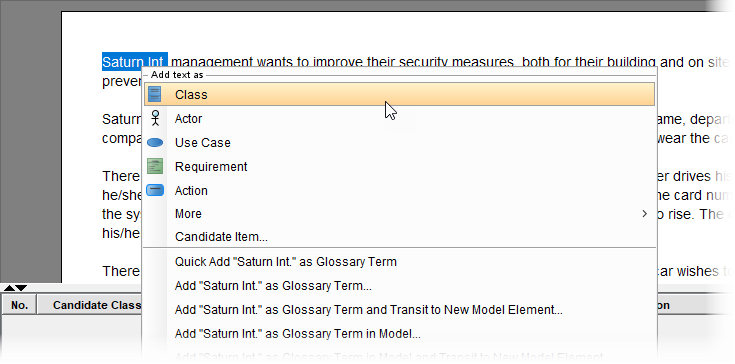
Identifying classes from text

Creating Textual Analysis

1. Create a new project. Select **Project > New** from the main menu. In the **New Project** window, name the project *Tutorial* and click **Create Blank Project** to confirm.
2. Create a new Textual Analysis. Select **Diagram > New** from the main menu.
3. In the **New Diagram** window, select **Textual Analysis** and click **Next**. Enter *Security Improvement* as the diagram name and click **OK**.



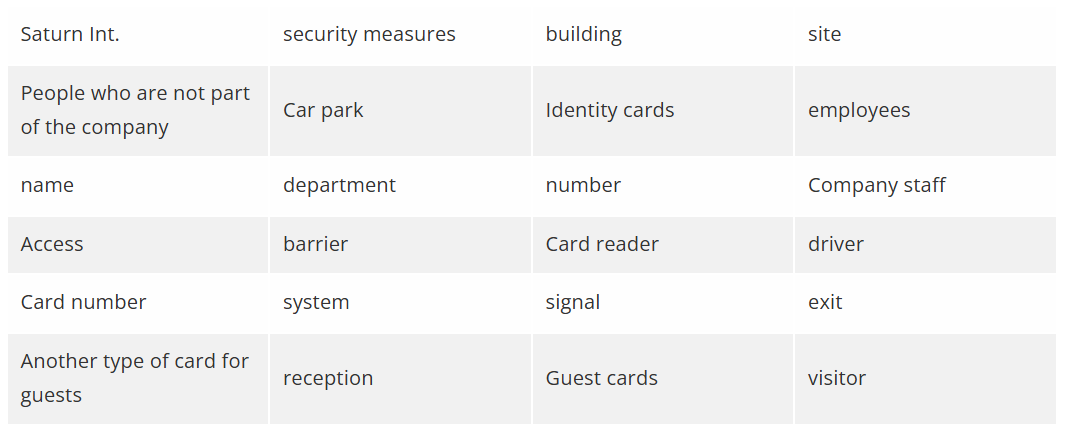
This opens the Textual Analysis editor. Copy the problem description provided above and paste it into the editor. Pasting can be done by pressing **Ctrl-V** or right-clicking on the editor and selecting **Paste**.



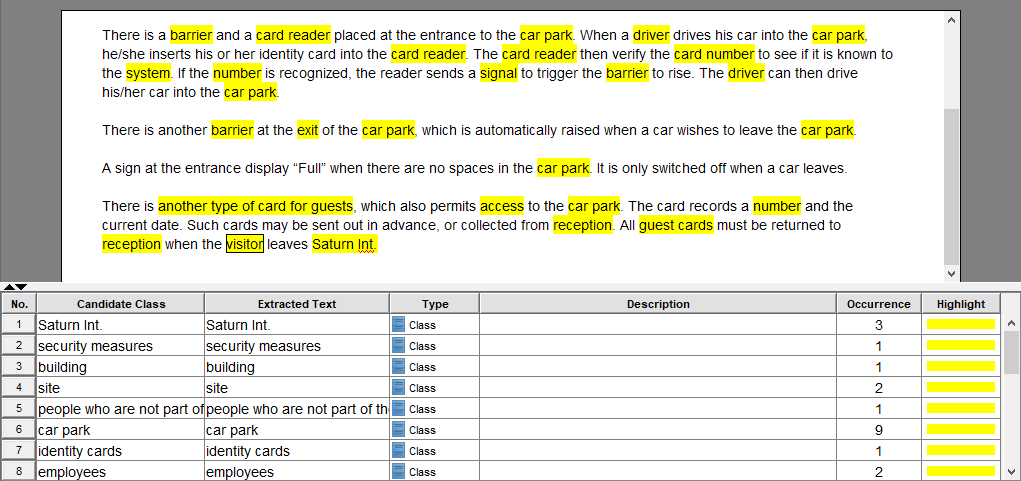
dentifying candidate objects from text

Read through the problem description to identify the candidate classes. When a candidate is found, right click on the segment of text and select **Add text as Class** from the popup menu.

A list of candidate class is obtained below:



Up to now, the Textual Analysis editor should look like this:



Rejecting candidate objects

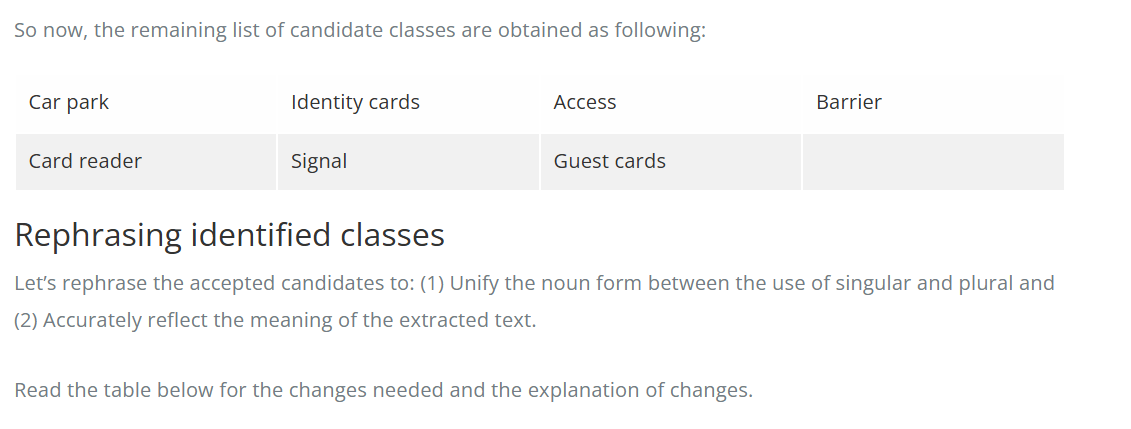
In this section, those candidate classes will be reviewed against a list of rejection rules. By the end of this section, a list of class models will be obtained.

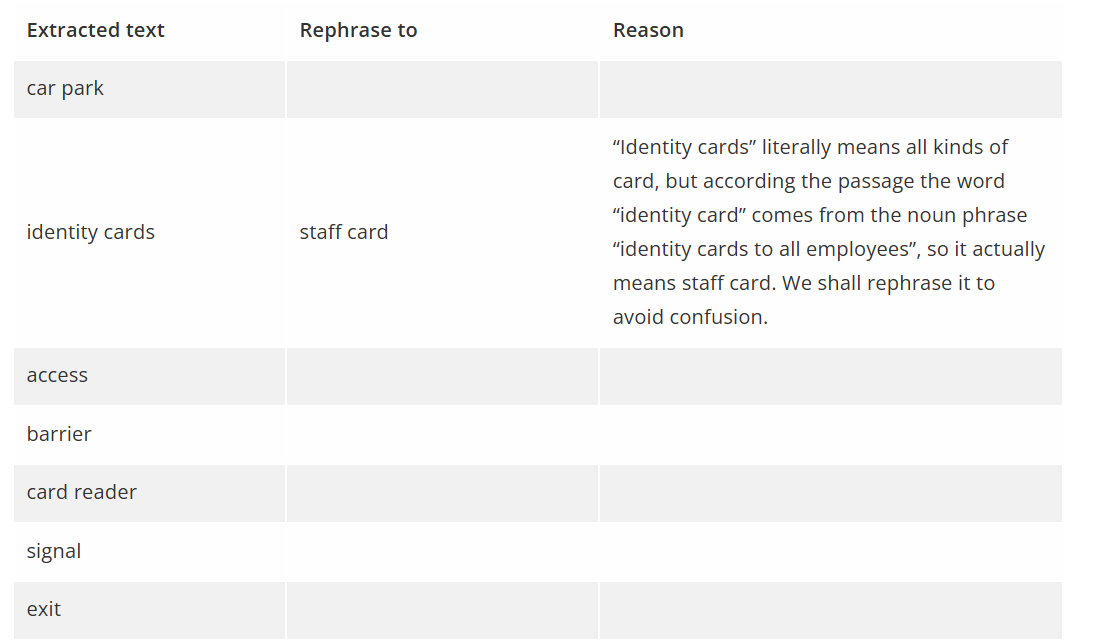
Now, review the list of candidate classes against the rejection rules below:

* Duplicates: if two or more objects are simply different names for the same thing.
* Irrelevant: objects which exists in the problem domain, but which are not intended.
* Vague: when considering words carefully it sometimes becomes clear that they do not have a price meaning and cannot be the basis of a useful in the system.
* General: the meaning is too broad.
* Attributes: as the attribute of objects.
* Associations: actually represents the relationships between objects.
* [**Roles**](https://www.visual-paradigm.com/scrum/what-are-the-three-scrum-roles/): sometimes objects referred to by the role they play in a particular part of the system.

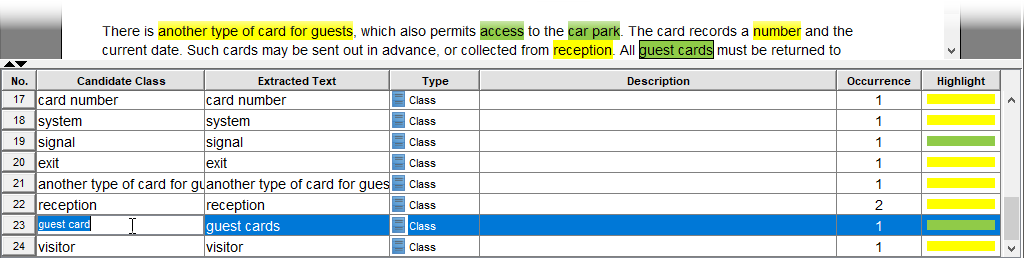


o differentiate the accepted and rejected candidate classes, let’s change the highlight color to green for those accepted. To do this, click on the cell with yellow highlight and select green from the color picker.

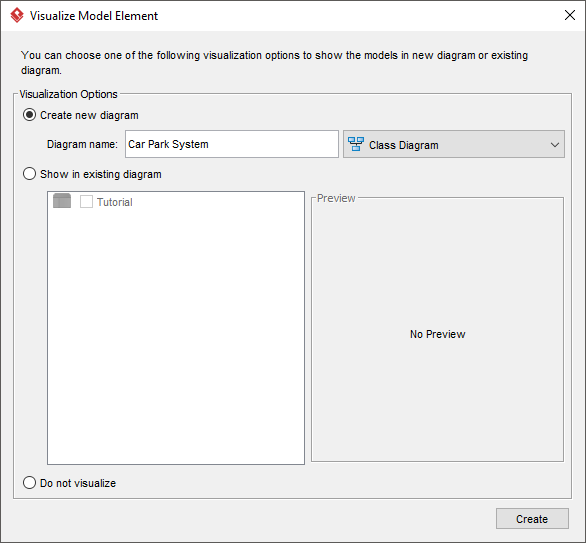




Update the name of the candidate classes accordingly:



You are prompted to visualize the created elements in a diagram. Keep the option **Create new diagram** selected. Rename the class diagram to *Car Park System* and click **Create**.



This creates a class diagram.

