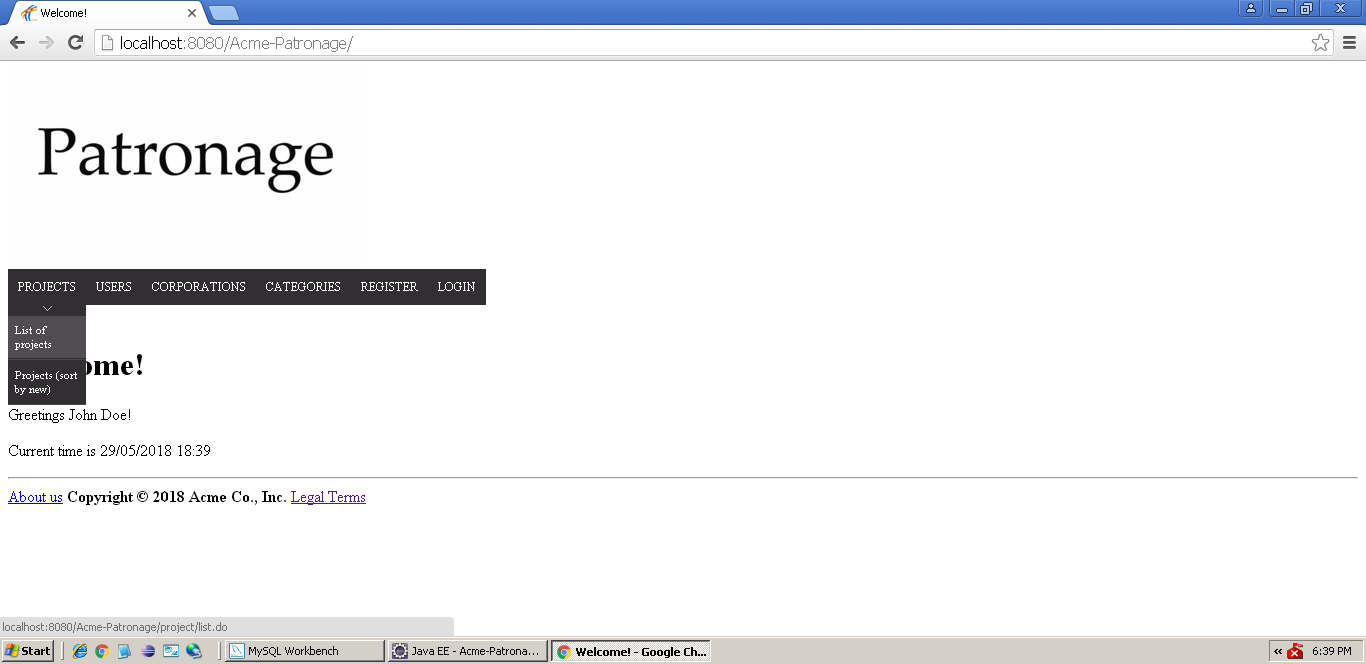
# Use case UC-21.4 List the comments of a project

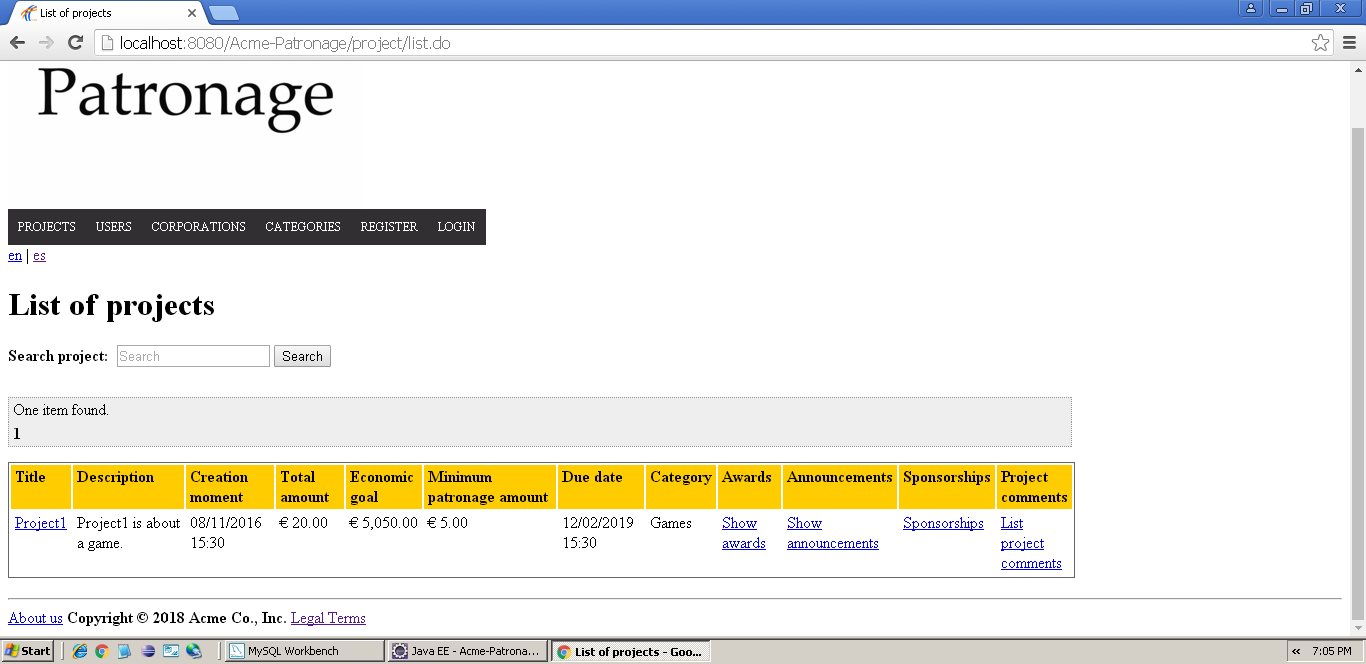
Description

An unauthenticated user wishes to list the comments of a project. To do so, they head to the main menu, place the mouse over the “Projects” button and click the option labelled “List of projects.” Afterwards, they click on “List project comments”.

Access

Main Menu -> Projects -> List of projects -> List project comments





Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user lists all the comments of a project. |
| **Expected** | The system must retrieve a list containing all the project’s comments, displaying the following information about them: the user who posted it, the text, the project it belongs to, its rating and the moment it was created. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user tries to list the comments of a non-existing project by tampering the ID attribute in the URL. |
| **Expected** | The system must redirect the user to the list of projects along with an error message. |
| **Outcome** |  |
| **Notes** |  |

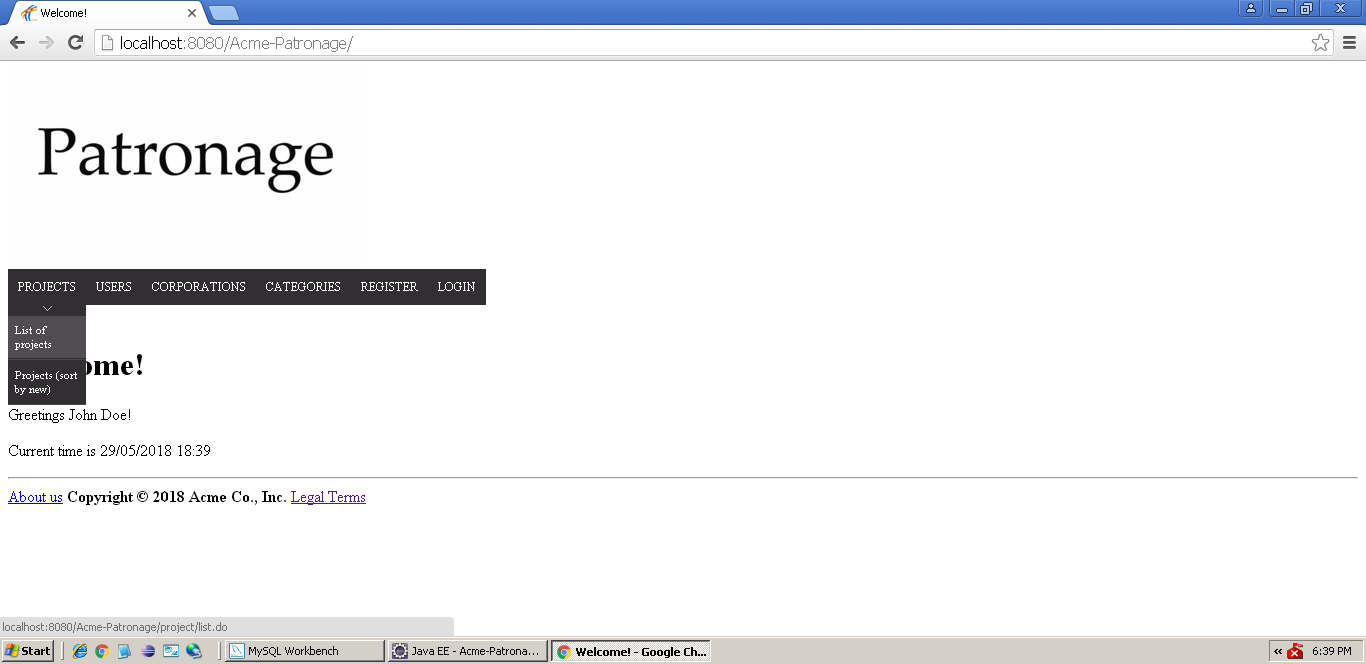
# Use case UC-21.5 List the comments of a award

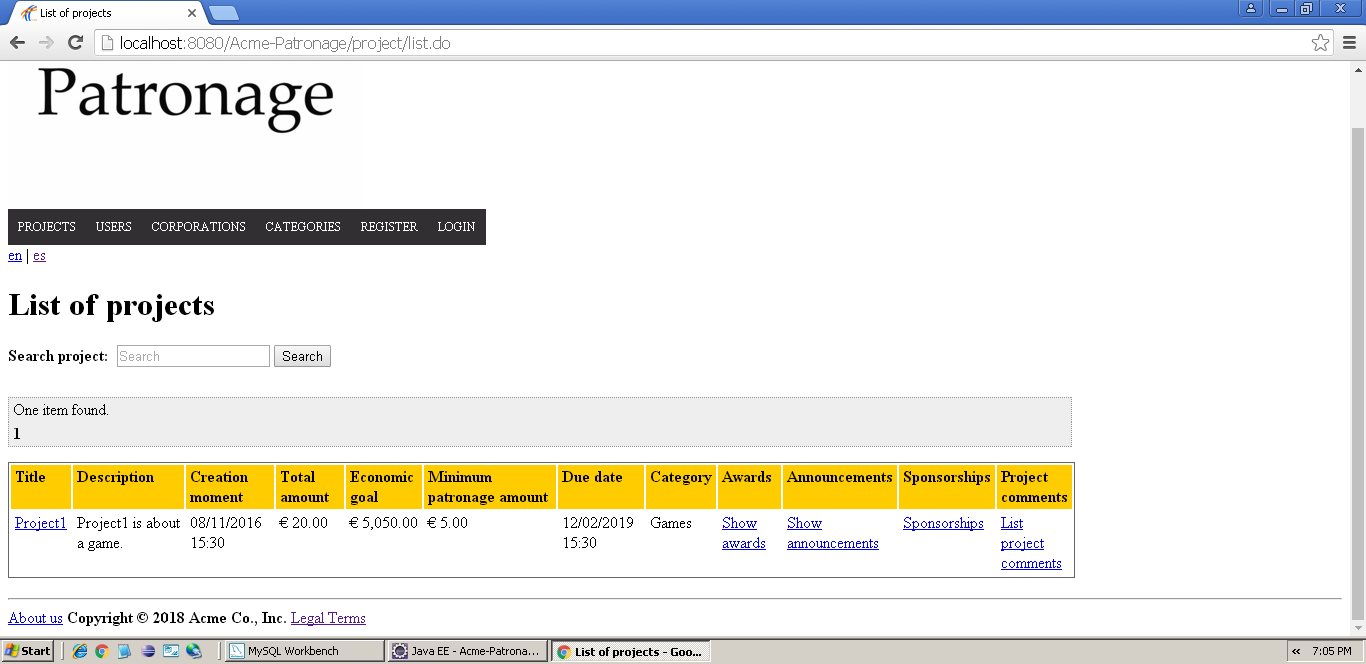
Description

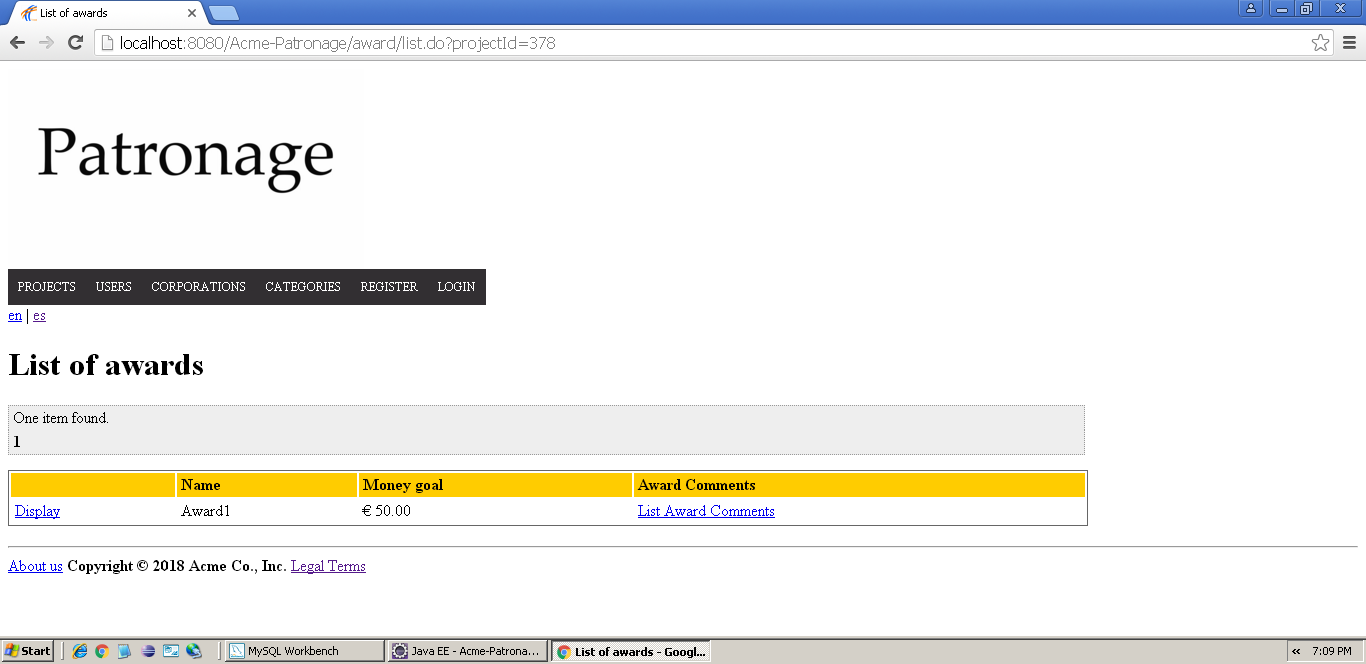
An unauthenticated user wishes to list the comments of an award. To do so, they head to the main menu, place the mouse over the “Projects” button and click the option labelled “List of projects.” Afterwards, they click on “Show awards” and finally head to “List award comments”.

Access

Main Menu -> Projects -> List of projects -> Show awards -> List award comments







Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user lists all the comments of an award. |
| **Expected** | The system must retrieve a list containing all the award’s comments, displaying the following information about them: the user who posted it, the text, the award it belongs to, its rating and the moment it was created. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user tries to list the comments of a non-existing award by tampering the ID attribute in the URL. |
| **Expected** | The system must redirect the user to the list of projects along with an error message. |
| **Outcome** |  |
| **Notes** |  |

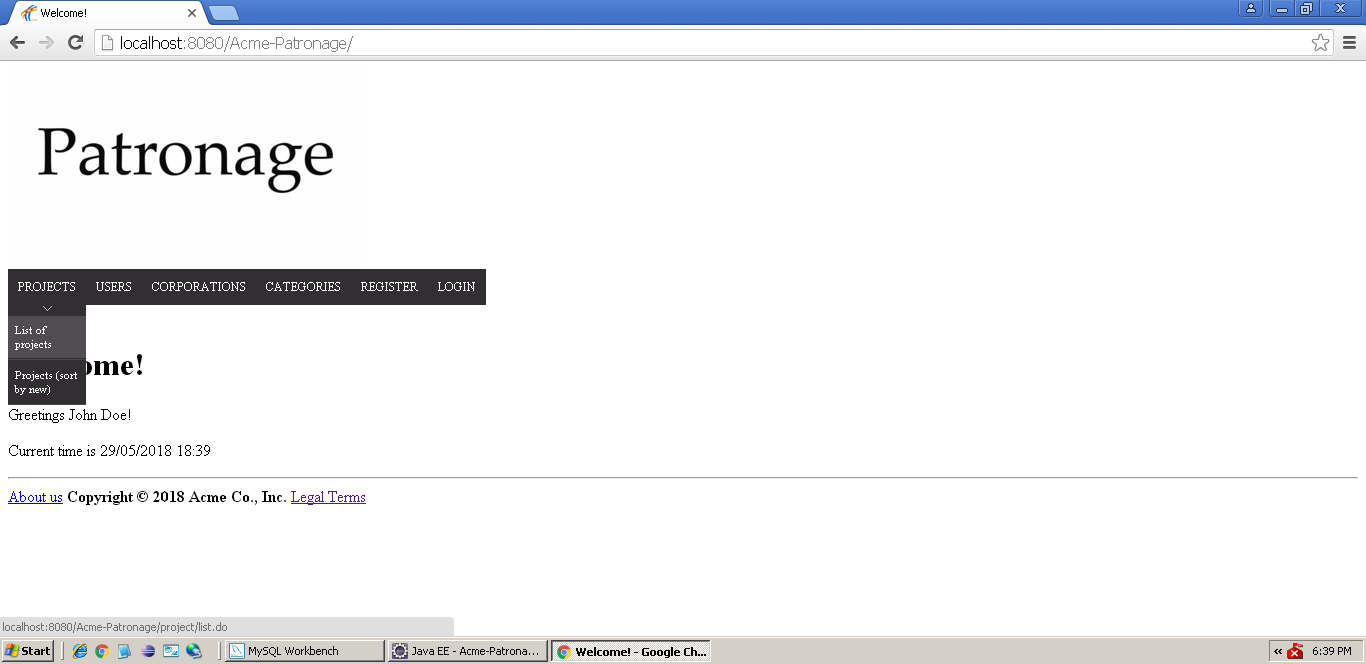
# Use case UC-21.6 List the comments of an announcement

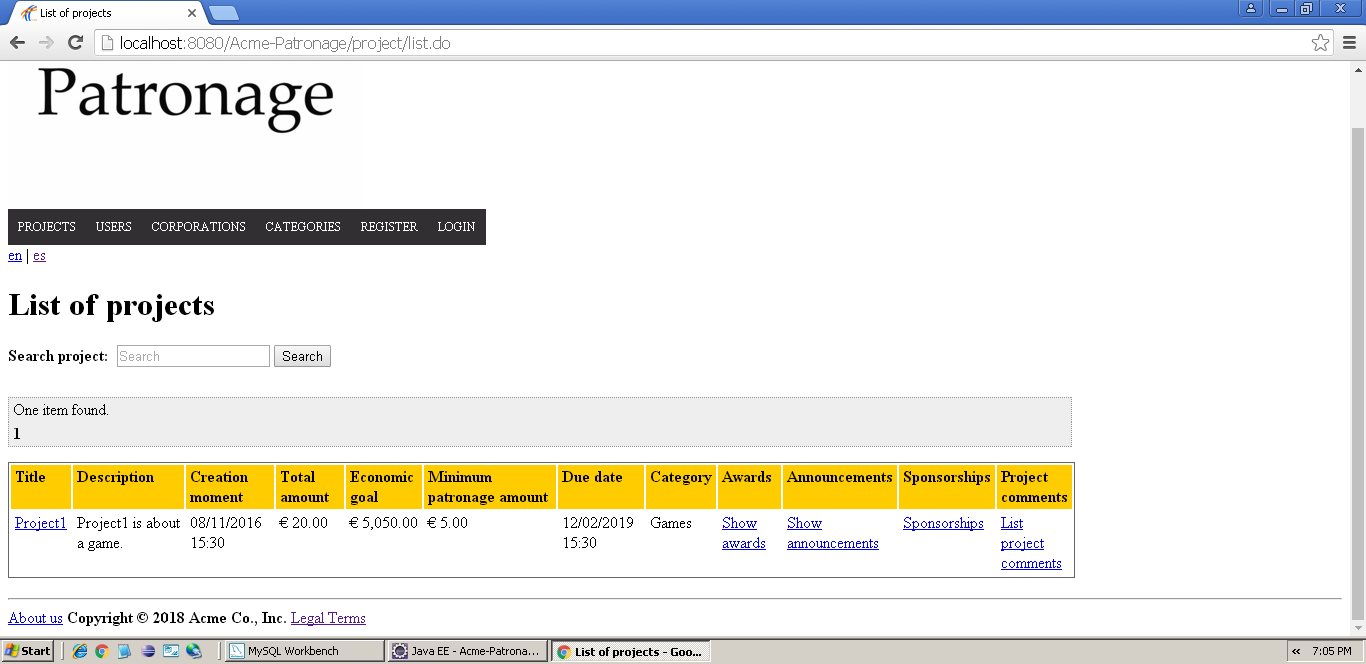
Description

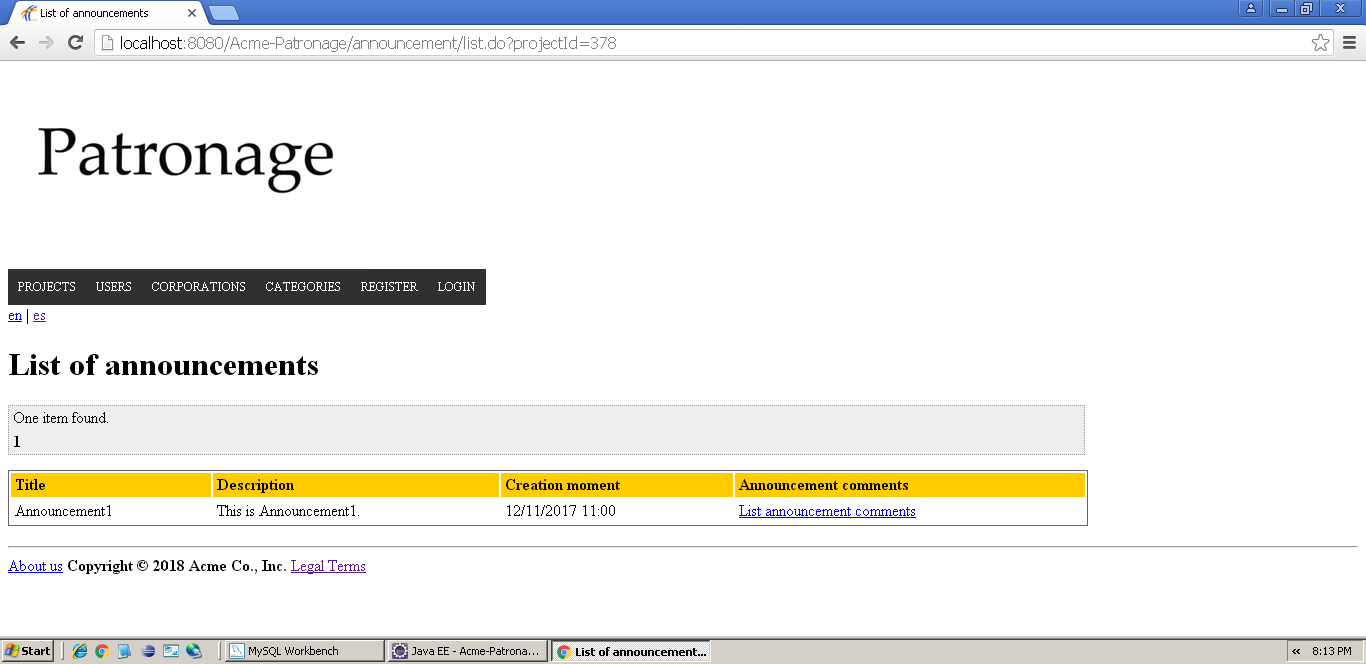
An unauthenticated user wishes to list the comments of an announcement. To do so, they head to the main menu, place the mouse over the “Projects” button and click the option labelled “List of projects.” Afterwards, they click on “Show announcements” and finally head to “List announcement comments”.

Access

Main Menu -> Projects -> List of projects -> Show announcements -> List announcement comments







Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user lists all the comments of an announcement. |
| **Expected** | The system must retrieve a list containing all the announcement’s comments, displaying the following information about them: the user who posted it, the text, the announcement it belongs to, its rating and the moment it was created. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user tries to list the comments of a non-existing announcement by tampering the ID attribute in the URL. |
| **Expected** | The system must redirect the user to the list of projects along with an error message. |
| **Outcome** |  |
| **Notes** |  |

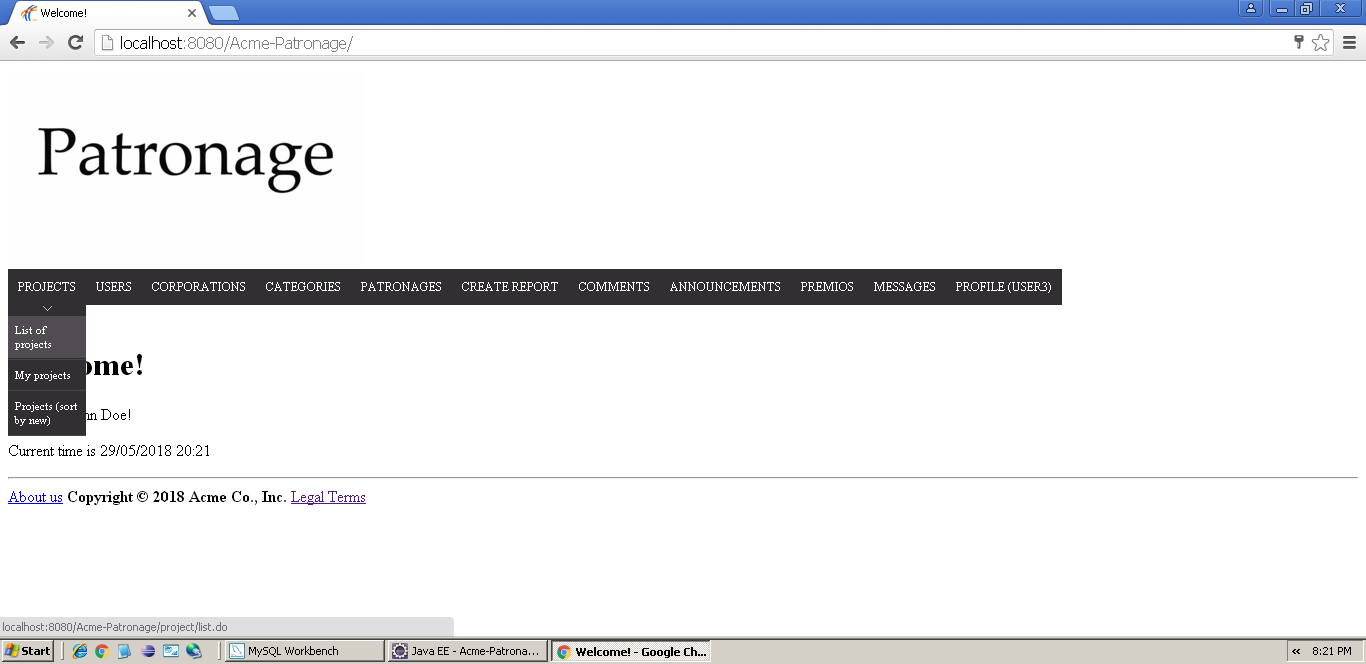
# Use case UC-23.3. Post a comment in a project, even if he or she hasn’t funded it.

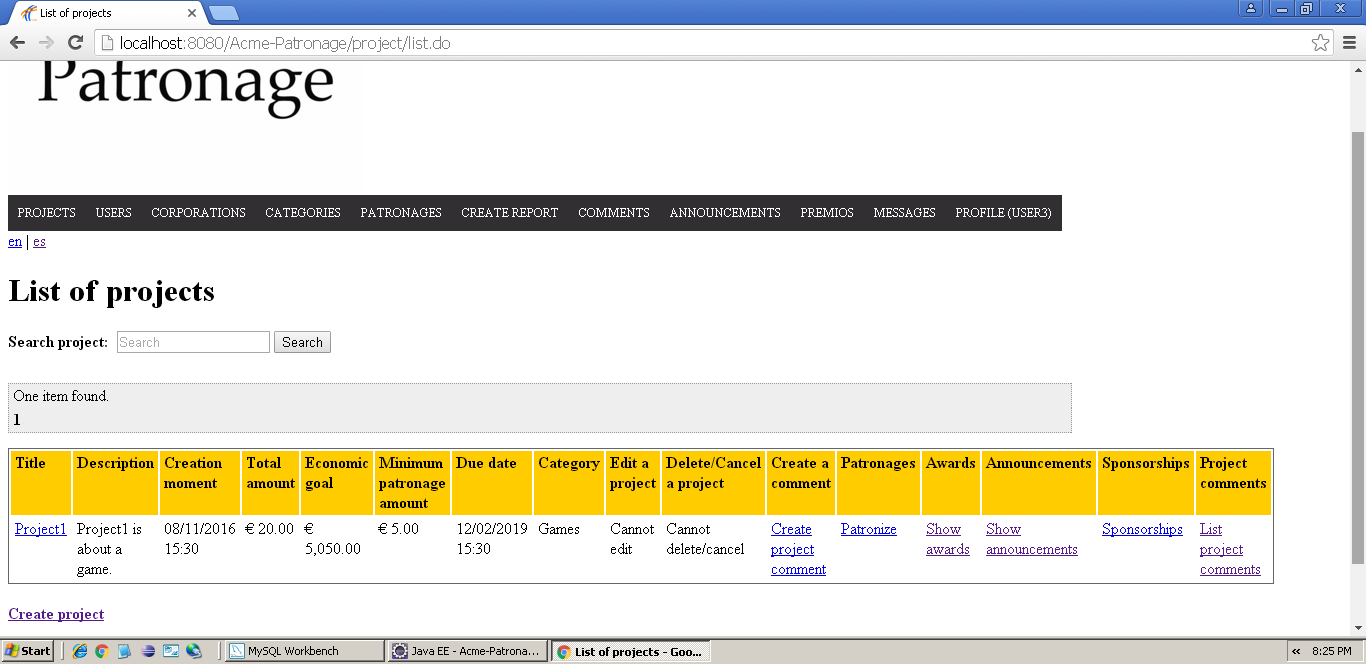
Description

An authenticated user wishes to post a comment in a project. In order to do so, they click the button labelled “List of projects” in the Projects section, then click on the link marked as “Create project comment”.

Access

Main Menu -> Projects -> List of projects -> Create project comment





Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user posts a new comment in a project, filling all the required fields. |
| **Expected** | The system saves the comment into the system and the user is brought to a list of project comments published by him or her. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user attempts to save a comment without filling the fields. |
| **Expected** | The system must stop the operation and inform the user about the errors. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#003>** | |
| **Description** | The user attempts to save a comment in a non-existent project by tampering the URL. |
| **Expected** | The system must stop the operation and redirect the user to the list of projects, along with an error message. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#004>** | |
| **Description** | The user attempts to save a comment with a rating lower than 1 or higher than 5. |
| **Expected** | The system must stop the operation and show an error message telling the user about the rating’s range. |
| **Outcome** |  |
| **Notes** |  |

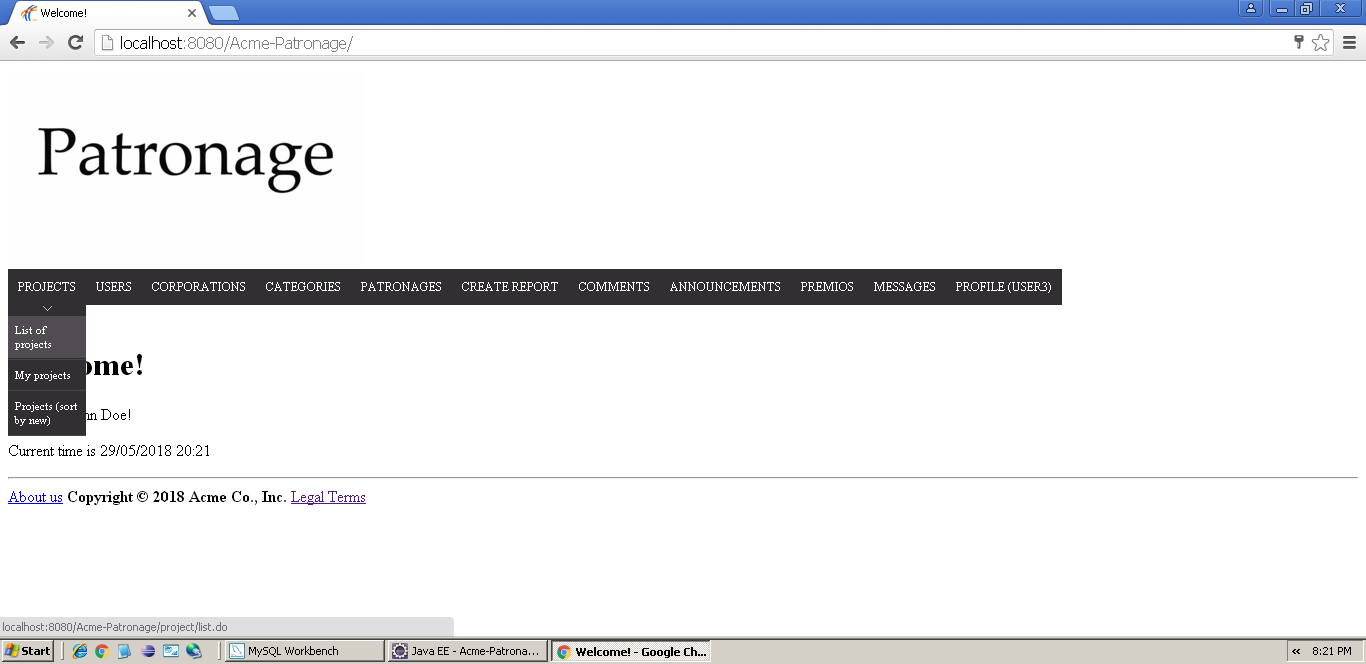
# Use case UC-23.4. Post a comment in an award, even if he or she hasn’t funded it.

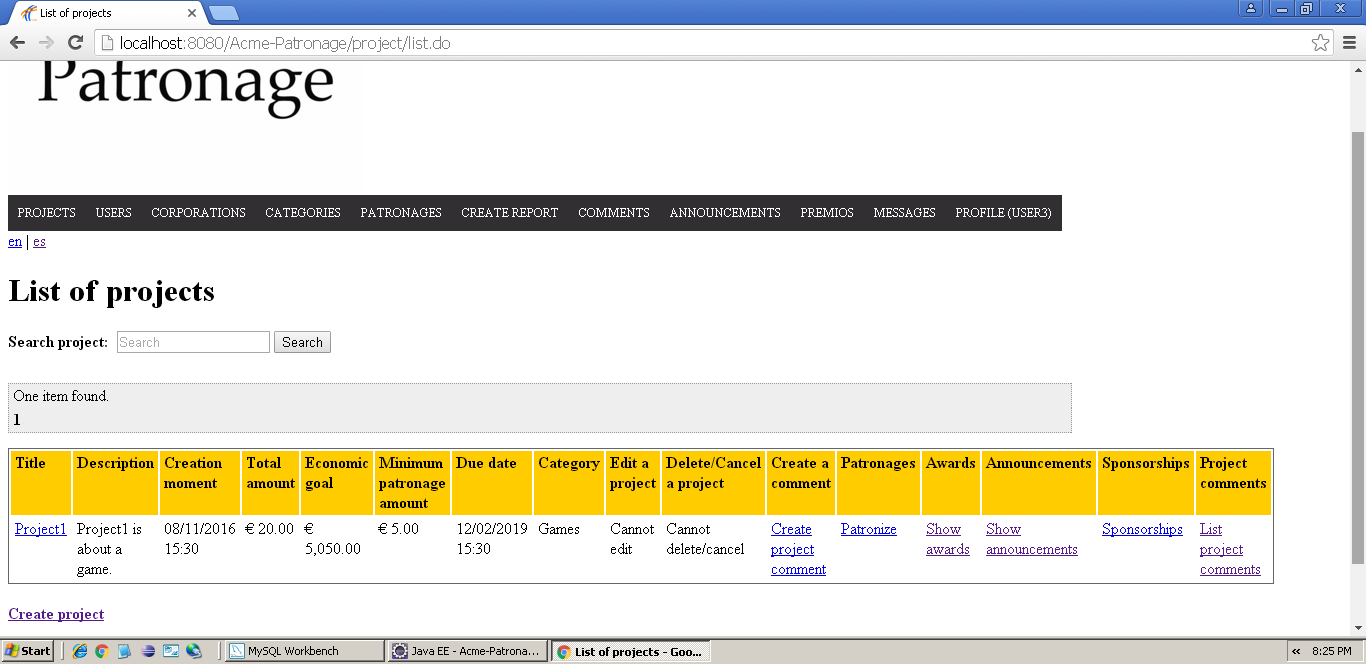
Description

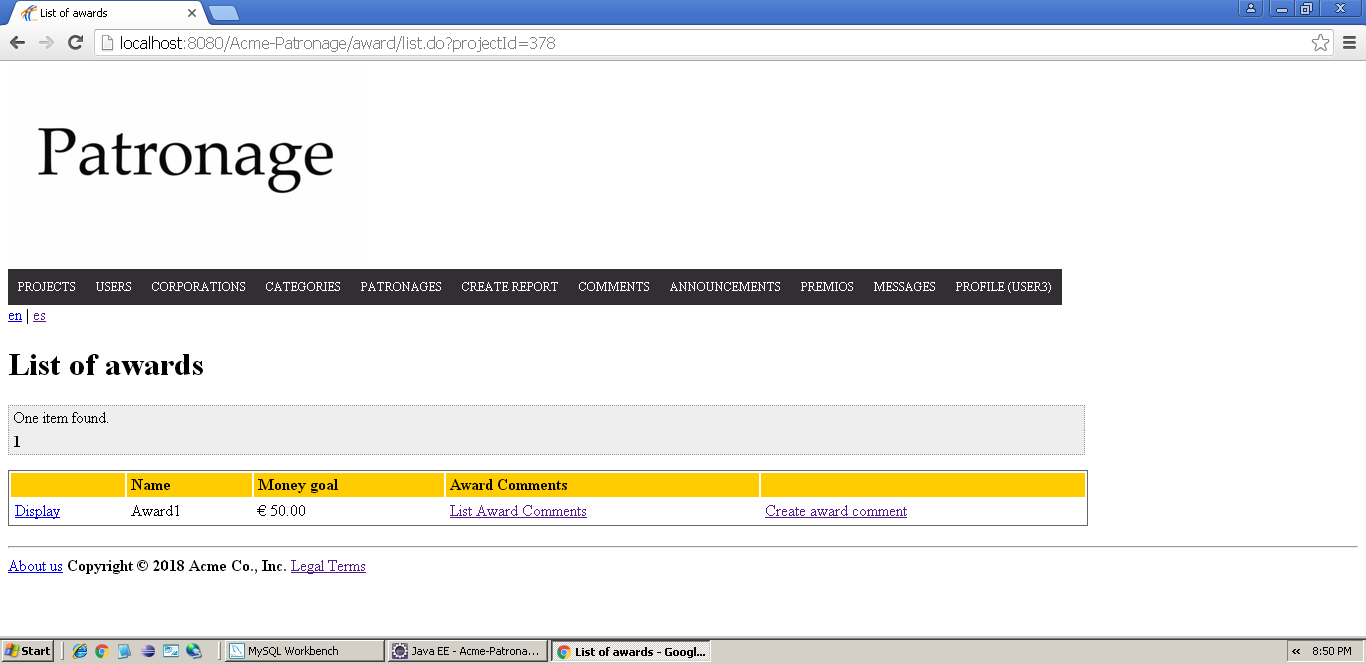
An authenticated user wishes to post a comment in an award. In order to do so, they click the button labelled “List of projects” in the Projects section, then click on the link marked as “Show awards” and then head to “Create award comment”.

Access

Main Menu -> Projects -> List of projects -> Show awards -> Create award comment







Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user posts a new comment in an award, filling all the required fields. |
| **Expected** | The system saves the comment into the system and the user is brought to a list of award comments published by him or her. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user attempts to save a comment without filling the fields. |
| **Expected** | The system must stop the operation and inform the user about the errors. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#003>** | |
| **Description** | The user attempts to save a comment in a non-existent project by tampering the URL. |
| **Expected** | The system must stop the operation and redirect the user to the list of projects, along with an error message. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#004>** | |
| **Description** | The user attempts to save a comment with a rating lower than 1 or higher than 5. |
| **Expected** | The system must stop the operation and show an error message telling the user about the rating’s range. |
| **Outcome** |  |
| **Notes** |  |

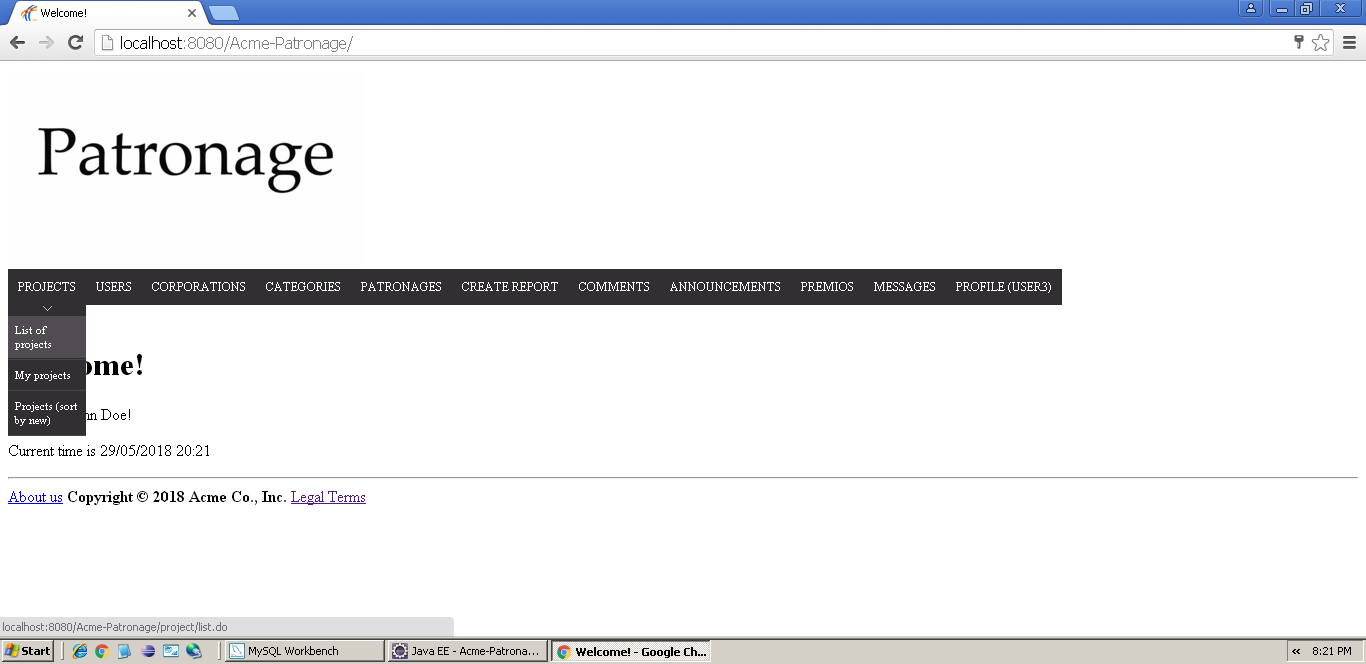
# Use case UC-23.5. Post a comment in an announcement, only if he or she has funded the related project.

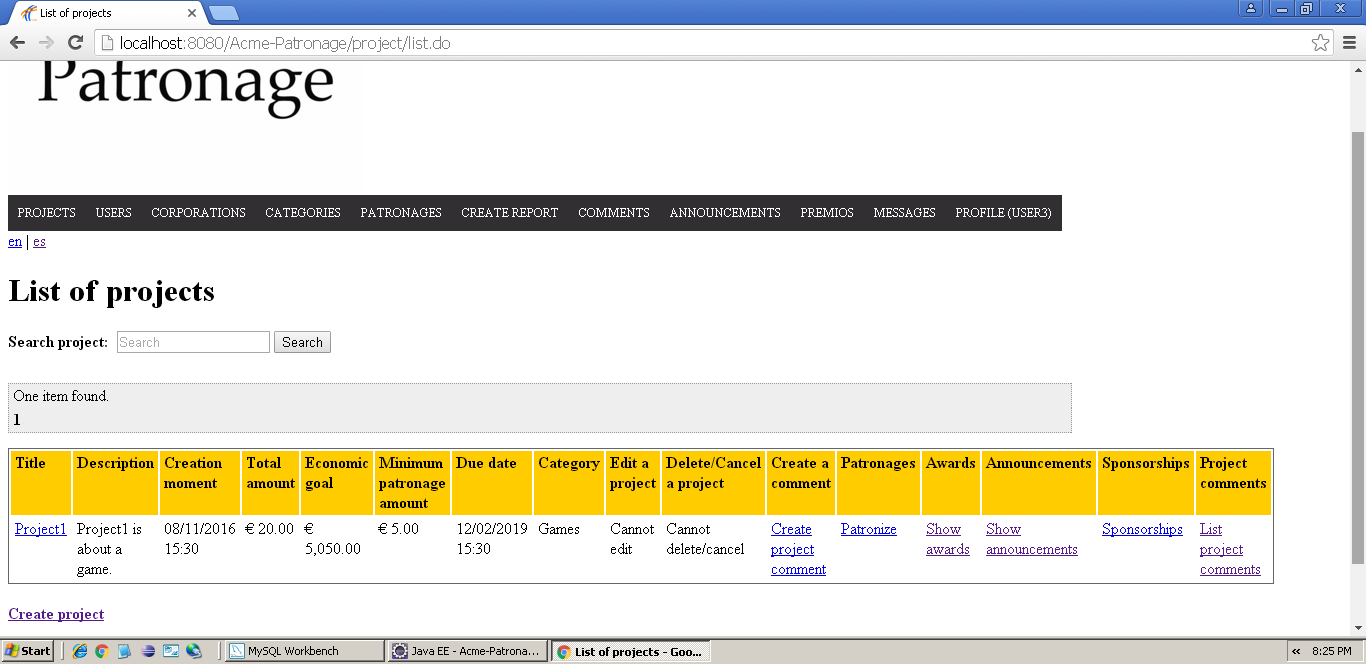
Description

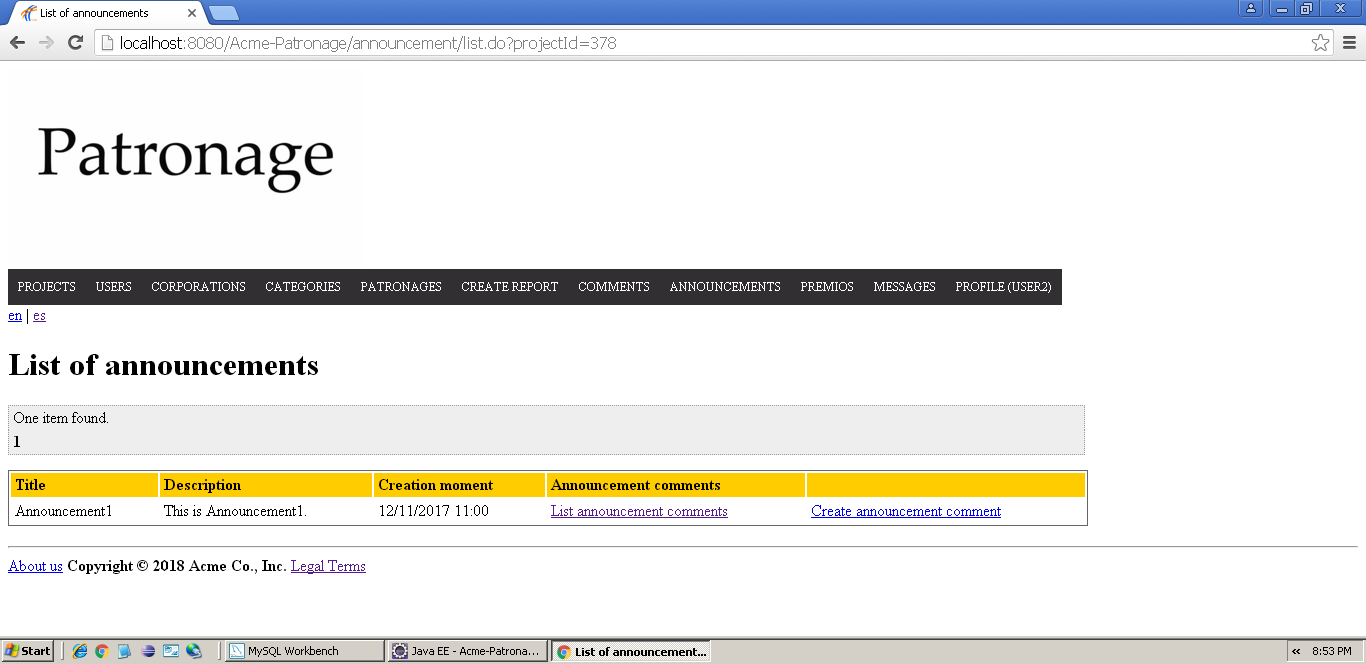
An authenticated user wishes to post a comment in an announcement. In order to do so, they click the button labelled “List of projects” in the Projects section, then click on the link marked as “Show announcements” and then head to “Create announcement comment”.

Access

Main Menu -> Projects -> List of projects -> Show announcements -> Create announcement comment







Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The user posts a new comment in an announcement, filling all the required fields. |
| **Expected** | The system saves the comment into the system and the user is brought to a list of announcement comments published by him or her. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The user attempts to save a comment without filling the fields. |
| **Expected** | The system must stop the operation and inform the user about the errors. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#003>** | |
| **Description** | The user attempts to save a comment in a non-existent project by tampering the URL. |
| **Expected** | The system must stop the operation and redirect the user to the list of projects, along with an error message. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#004>** | |
| **Description** | The user attempts to save a comment in an announcement of a project they haven’t patronaged by tampering the URL. |
| **Expected** | The system must stop the operation and redirect the user to the list of their announcement comments, along with an error message. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#005>** | |
| **Description** | The user attempts to save a comment with a rating lower than 1 or higher than 5. |
| **Expected** | The system must stop the operation and show an error message telling the user about the rating’s range. |
| **Outcome** |  |
| **Notes** |  |

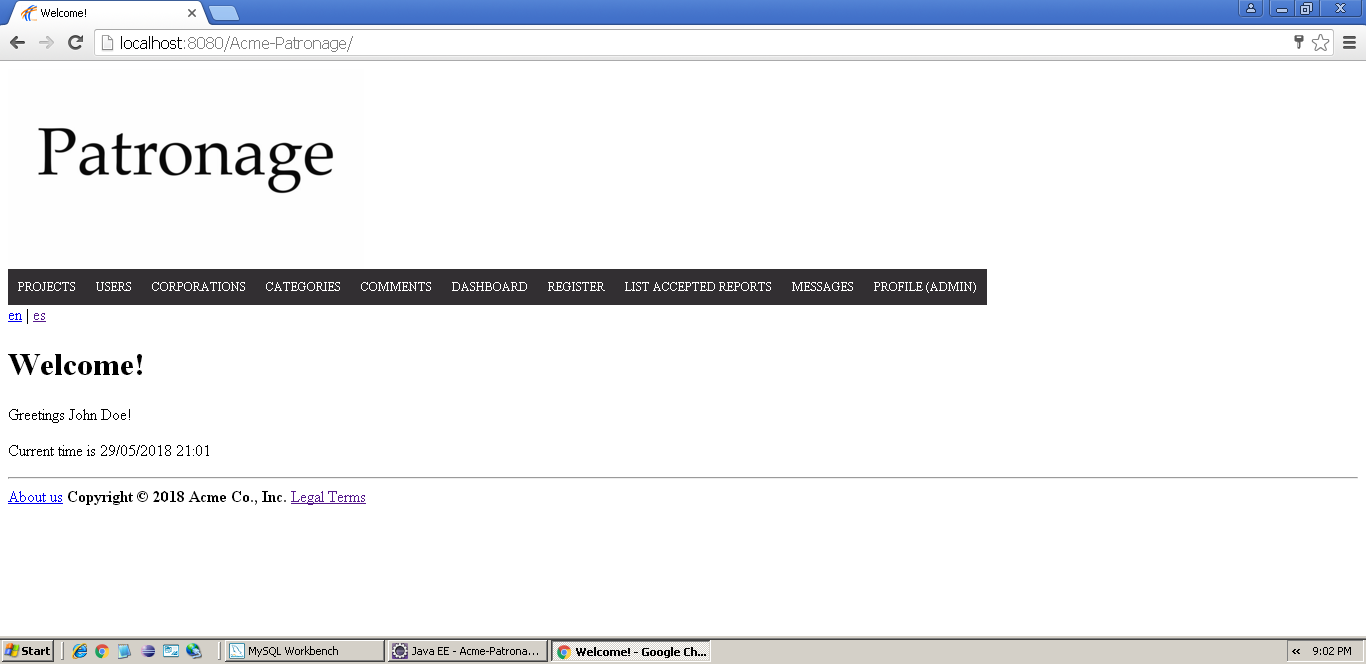
# Use case UC- 25.1. Delete a comment if he or she finds it inappropriate.

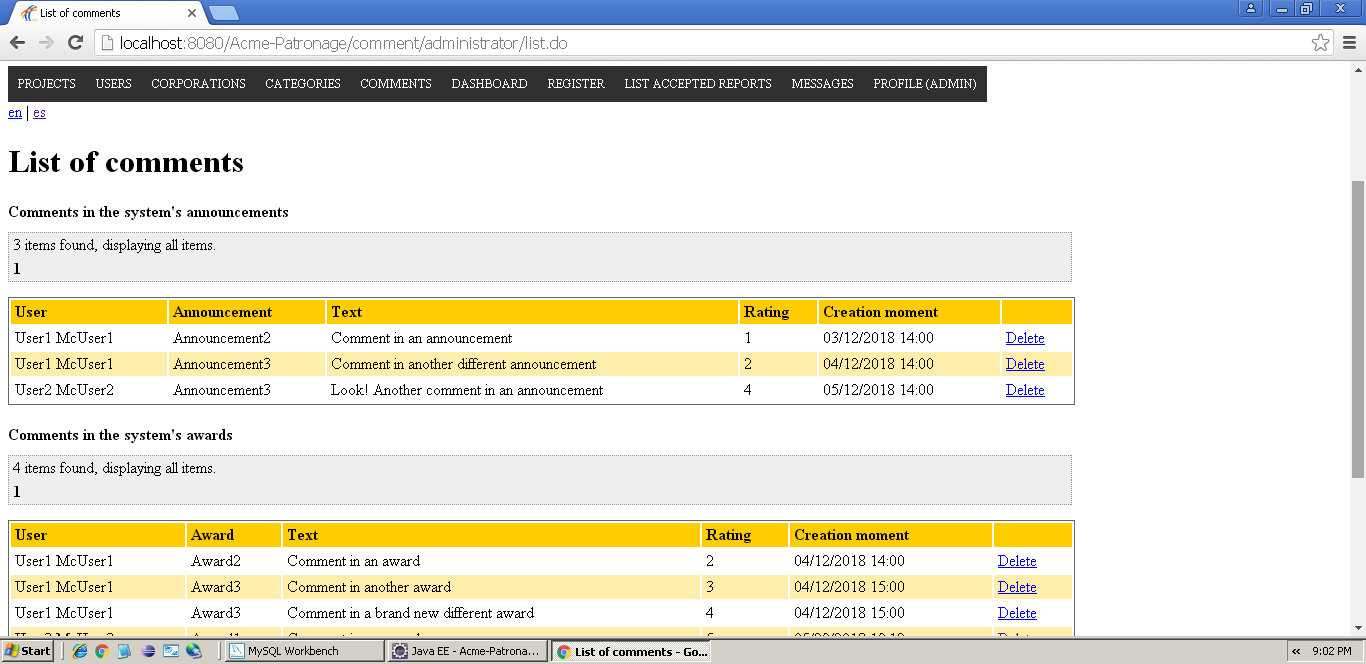
Description

An administrator wishes to delete a comment he or she considers inappropriate. In order to do so, they click the button labelled “Comments” in the Main menu and click on the “delete” link of the desired comment.

Access

Main Menu -> Comments -> Delete link of the comment he or she wants to delete





Tests

|  |  |
| --- | --- |
| **Test <#001>** | |
| **Description** | The administrator clicks the “Delete” link of any comment. |
| **Expected** | The system deletes the comment from the database and takes the administrator back to the same page, but the selected comment must be missing. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#002>** | |
| **Description** | The administrator attempts to delete a non-existent comment by tampering the URL. |
| **Expected** | The system must stop the operation and inform the user about the errors. |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| **Test <#003>** | |
| **Description** | The user attempts to save a comment in a non-existent project by tampering the URL. |
| **Expected** | The system must stop the operation and redirect the user to the list of projects, along with an error message. |
| **Outcome** |  |
| **Notes** |  |