|  |  |
| --- | --- |
| Use case name | Login into the system |
| ID | Log\_1.1 |
| Description | Non-logged user has possibility to login into the system. For login user has to put two kinds of information:  Login name  Password  After correct login process, system launch proper module, based on user’s rights |
| Author | Karolina Kapituła |
| Actor | Non-logged user |
| Start conditions | N/A |
| Finish conditions | User logged-in  Appropriate module launched |
| Input data | Login name  Password |
| Output data | N/A |
| Scenarios | Main:   1. User launches system. 2. System opens login window. 3. User puts input data. 4. System checks that data is correct. 5. System launches appropriate module, based on user rights.   Alternative:  Incorrect input data –not more than two times in a row:   1. System checks that input data is incorrect. 2. System views message with error details.   3rd incorrect data input in a row:   1. System checks that input data is incorrect. 2. System lock possibility of login data input (login name and password fields are unavailable). 3. System starts to show time to remain to unlock the system. |
| Notices |  |

# Use case: Login to the system

|  |  |
| --- | --- |
| Use case name | Logout from the system |
| ID | Log\_1.2 |
| Description | Logged user has possibility to logout from the system.  Login user has to click “Logout” button.  After correct logout process, system directs user to the main page (Login).  After success logging-out the system displays a success message in the form of a pop-up window. |
| Author | Karolina Kapituła |
| Actor | Logged user |
| Start conditions | User is logged |
| Finish conditions | User is logged-out  Main page is launched  Pop-up window of succeed logging out. |
| Input data | N/A |
| Output data | Pop-up window |
| Scenarios | Main:   1. User clicks logout button. 2. System opens the window with message about logout attempt. 3. Button “Cancel” is available. 4. User has to confirm that they want to logout. 5. User clicks “OK” button. 6. System logouts the user. 7. System opens “Login” page with empty fields.   Alternative:  Canceling logging out:   1. User clicks “Cancel” button. 2. System closes message window. |
| Notices |  |

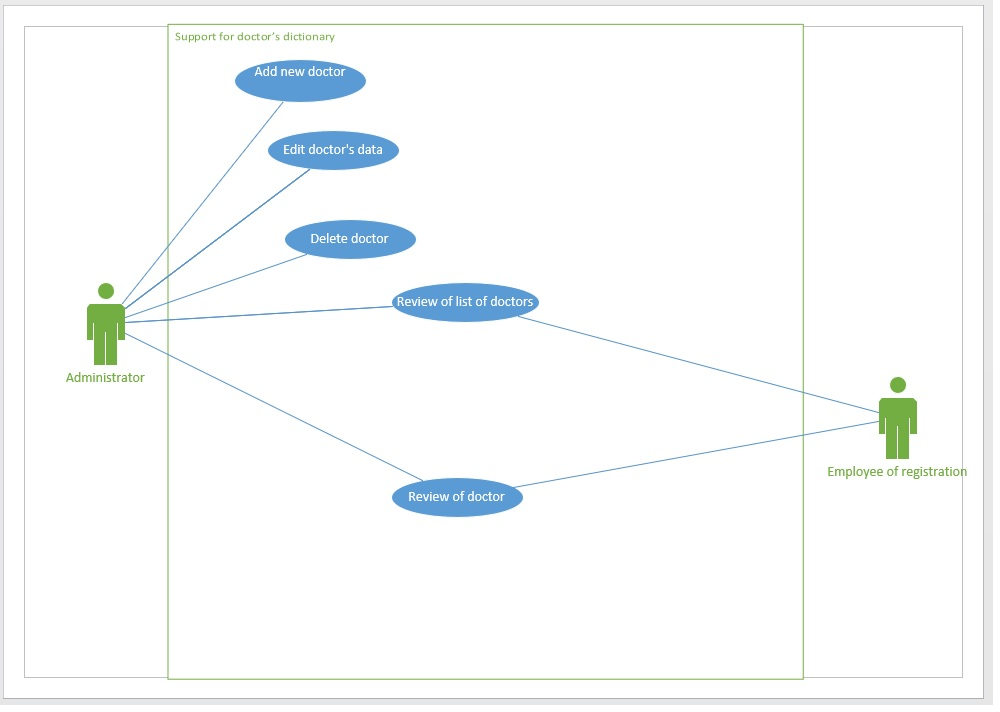
|  |  |
| --- | --- |
| Use case name | Application for changing password |
| ID | Log\_1.1.3 |
| Description | Non-logged user has possibility to submit an application for permission password changing in case of forgotten it  Login user has to put two kinds of information:   * Login name * E-mail   After putting information correctly, system sends link to form of password changing to the e-mail address.  System shows message with information about link. |
| Author | Karolina Kapituła |
| Actor | Non-Logged user |
| Start conditions | N/A |
| Finish conditions | System sends link to form of password changing to the e-mail address. |
| Input data | Login  E-mail |
| Output data | Link |
| Scenarios | Main:   1. User clicks “Forgot password” button. 2. System opens the new form that allows you to enter login and e-mail address for the password change link. 3. User fill out data into the field Login. 4. User fill out data into the field e-mail. 5. Button “OK” comes to be available. 6. User clicks the “OK button”. 7. System checks that login and e-mail are linked. 8. System sends an e-mail with a link with changing password form. 9. System shows a message with information about link to form of password changing. 10. After pressing button “OK” system closes a message window.   Alternative:  Canceling creating new password request:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of change password cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and change password window. 4. If the button “Cancel” is pressed, system closes pop-up window.   User input wrong login:   1. User input login which doesn’t exist in database. 2. System shows the pop-up window of wrong login. 3. After button “OK” is pressed, system closes the pop-up window. 4. System opens empty application for changing password form.   User input wrong e-mail:   1. User input e-mail that is invalid. 2. System shows the pop-up window of wrong e-mail. 3. After button “OK” is pressed, system closes the pop-up window. 4. System opens empty application for changing password form.   User input data that aren’t connected to each other:   1. System checks that login and e-mail aren’t connected to each other. 2. System shows the pop-up window of wrongly typed data. 3. After button “OK” is pressed, system closes the pop-up window. 4. System opens empty application for changing password form. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Password change |
| ID | Log\_1.6 |
| Description | Non logged system user’s has possibility to put and confirm new password.  System changes the password. |
| Author | Karolina Kapituła |
| Actor | Non-logged user |
| Start conditions | The password change form is launched. |
| Finish conditions | System changes the password. |
| Input data | New password  Confirm password |
| Output data | Pop-up window |
| Scenarios | Main:   1. User opens the link he received in the e-mail. 2. System displays password change form. 3. System presents following fields and buttons:    * Login - system shows appropriate login name    * New password - field to filled in by the user    * Confirm password - field to filled in by the user    * Button "OK"    * Button "Cancel" 4. User fills following fields:    * New password    * Confirm password 5. System checks that strings in those fields are the same. 6. Button “OK” comes to be available. 7. System validates correctness of entered password with accordance to password policy.    * Password length: not less than 8 characters and not more than 15 characters.    * Use of each of the following types of marks at least once:      + lowercase      + uppercase      + digital      + special sign: "-", "\_", "!". "#", "$", "\*" 8. System displays a success message in the form of a pop-up window after password meeting the validation criteria. 9. Button “OK” comes to be available. 10. Once the button “OK” is pressed, the pop-up window and change password window are closed. 11. System changes the password in the database.   Alternative:  Password change cancelation:   1. User presses “Cancel” button. 2. System shows pop-up window with the request of confirmation of change password cancelation message. 3. Buttons “OK” and “Cancel” are available. 4. If the “OK” button is pressed, the pop-up window and change password window are closed. 5. If the “Cancel” button is pressed system closes the message window.   String in “New password” and “Confirm password” are different:   1. Input data in “New password” and “Confirm password” fields are different. 2. Button “OK” is unavailable. 3. System displays pop-up window with message that those fields aren’t the same. 4. Button “OK” is available. 5. Once the “OK” button is pressed, system closes the pop-up window.   Password doesn’t meet the validation criteria:   1. System displays an error message in the form of pop-up window. 2. The “OK” button is available on the pop-up window. 3. Once the “OK” button is pressed, the pop-up window is closed. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Lock the system |
| ID | Log\_1.1.5 |
| Description | Non-logged user inputs incorrect data 2 times in a row. |
| Author | Karolina Kapituła |
| Actor | Non-logged user |
| Start conditions | 2nd incorrect login data entered |
| Finish conditions | User can’t enter input login data and system shows the timer. |
| Input data | Wrong login and password |
| Output data | N/A |
| Scenarios | Main:  3rd incorrect data input in a row:   1. System checks that input data is incorrect. 2. System lock possibility of login data input (login name and password fields are unavailable). 3. System starts to show time to remain to unlock the system. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Unlock the system |
| ID | Log\_1.1.6 |
| Description | After the designated time, the system enables user to log in again. |
| Author | Karolina Kapituła |
| Actor | System locking time complete |
| Start conditions | Possibility of logging is blocked |
| Finish conditions | Data fields are available again |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Time of system locking is complete. 2. Input data fields are available. 3. Information about time to unlock disappears. |
| Notices |  |

# User case: List of doctors



|  |  |
| --- | --- |
| Use case name | Adding new doctor |
| ID | Doc\_1.2.1.1 |
| Description | Administrator has possibility to enter data of a new doctor and add them to the system.  System opens adding new doctor form.  After correct data input, system adds doctor to the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Logged user with adding doctor privilege. |
| Finish conditions | New doctor added to the system. |
| Input data | Mandatory:   * Personal data   + Name   + Last name   + Residential address * E-mail * PESEL * Phone number   Optional:   * Doctor’s specialty * Doctor’s room * Working days * Working hours |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Add new doctor” button. 2. System opens a new form that allows to add new doctor to the system after filling the fields. 3. Administrator needs to fill mandatory fields. 4. System validates that PESEL is correct in accordance with user story. 5. System validates that e-mail is correct in accordance with user story. 6. System validates that phone number is correct in accordance with user story. 7. Button “Add” comes to be available. 8. System allows you to optionally add the following data:  * Doctor’s specialty * Doctor’s room * Working days * Working hours  1. Administrator clicks “Add” button. 2. System sends an e- mail to the doctor with information of adding to the system. 3. System sends an e-mail to the doctor with information of login data. 4. System shows a message window of successful adding a doctor to the database. 5. System adds the doctor to the database and sets their status to active. 6. Once button “Ok” is pressed, system closes the message window and adding new doctor form.   Alternative:  User input wrong PESEL:   1. System using PESEL validator checks that input PESEL is incorrect. 2. System shows the pop-up window of wrong input PESEL. 3. Button “OK” closes the pop-up window.   User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling adding a new doctor:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of adding new doctor cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and adding new doctor form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

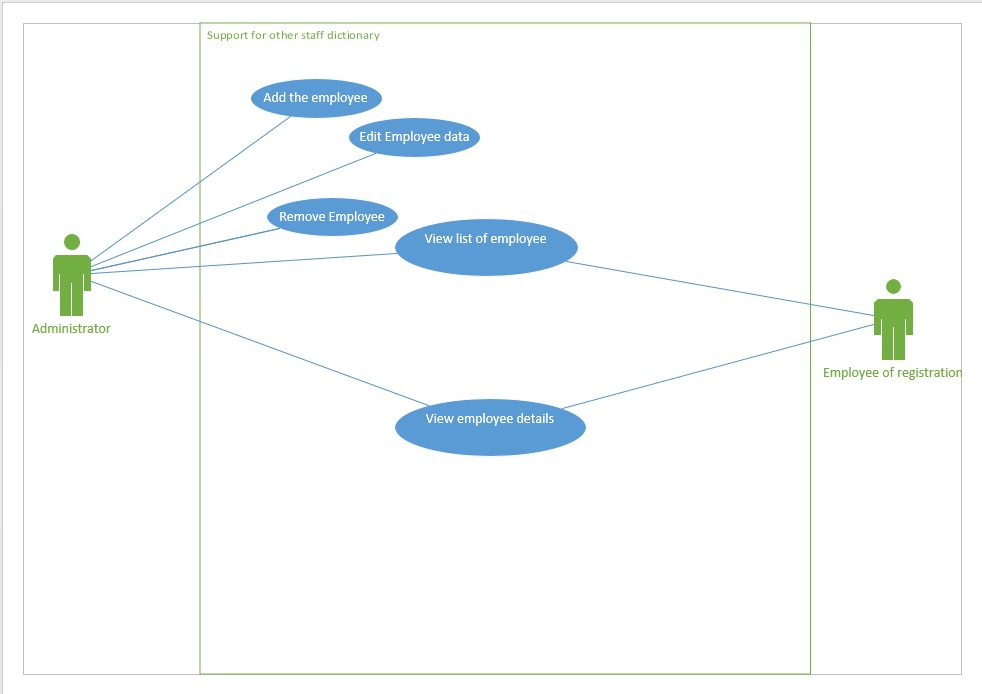
|  |  |
| --- | --- |
| Use case name | Editing doctor’s data |
| ID | Doc\_1.2.1.2 |
| Description | Administrator has possibility to update data of an existing doctor in the system and confirm it.  System opens editing doctor’s data form.  After correct data input, system edits doctor’s data in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of active doctors or view of doctor’s details form. |
| Finish conditions | Doctor’s data change in the system. |
| Input data | * Personal data * E-mail * PESEL * Phone number * Doctor’s specialty * Doctor’s room * Doctor’s status * Working days * Working hours |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Edit doctor’s data” button. 2. System opens a new form that allows to update existing doctor’s data in the system after filling the fields. 3. Administrator selects which field he wants to change.  * E-mail * Phone number  1. After changing one or more fields. 2. Button “Change data” comes to be available. 3. System allows you to optionally edit the following data:  * Doctor’s specialty * Doctor’s room * Working days * Working hours  1. Administrator can change doctor’s status in accordance with user stories. 2. Administrator clicks “Change data” button. 3. System validates that PESEL is correct. 4. System validates that e-mail is correct. 5. System validates that phone number is correct. 6. System sends an e- mail to the doctor with information of editing their data in the system. 7. System shows a message window with an information of successful editing doctor’s data in the database. 8. Once button “Ok” is pressed, system closes the message window and editing doctor’s data form.   Alternative:  User input wrong PESEL:   1. System using PESEL validator checks that input PESEL is incorrect. 2. System shows the pop-up window of wrong input PESEL. 3. Button “OK” closes the pop-up window.   User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling editing doctor’s data:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of editing doctor’s data cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and editing doctor’s data form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Doctor’s deactivation |
| ID | Doc\_1.2.1.3 |
| Description | Administrator has possibility to move data of an existing doctor to archive and confirm it.  System opens doctor’s deactivation form.  After correct password input, system moves doctor’s data to the archive.  System blocks adding appointments if doctor has inactive status. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Doctor is active. |
| Finish conditions | Doctor’s status is changed to inactive.  Doctor’s data is moved to the archive. |
| Input data | Administrator’s password |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Deactivate doctor” button. 2. System opens a new form that allows to deactivate doctor in the system after typing administrator’s password. 3. Administrator has to input his password. 4. Button “Deactivate” comes to be available. 5. Administrator clicks “Deactivate” button. 6. The system checks if the doctor has any appointments 7. System moves doctor's data to archive. 8. System changes doctor's status to inactive. 9. System shows a message window with an information of successful deactivating the doctor. 10. Once button “Ok” is pressed, system closes the message window and doctor’s deactivating form.   Alternative:  Doctor has appointments:   1. Pop-up window appears with message that you can’t deactivate doctor because they have planned appointments. 2. Button “OK” closes the pop-up window and deactivating doctor form.   Canceling deactivating doctor:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of deactivating doctor cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and deactivating doctor form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Administrator typed wrong password.   1. System shows pop-up window that the password is incorrect. 2. Button “OK” closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View list of doctors |
| ID | Doc\_1.2.1.4 |
| Description | Administrator has possibility to view list of doctors  System opens a form with list of doctors.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Logged in administrator |
| Finish conditions | Form with list of doctors. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View list of doctors” button. 2. System opens a new form that allows to view list of doctors.    * Data shown in the list:      + Name      + Surname      + Specialty      + E-mail address 3. Following buttons are available:    * “Cancel” button    * “View doctor” button    * “Deactivate doctor” button   Alternative:  Closing view list of doctors form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of doctor’s cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view list of doctor’s form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Filtering doctors based on:   1. Status    * Administrator picks which status and doctors assigned to it they want to see.    * System filters the list of doctors, showing the ones with chosen status. 2. Specialty    * Administrator picks which specialty and doctors assigned to it they want to see.    * System filters the list of doctors, showing the ones with chosen role.   Sorting doctors based on:   1. Specialty    * Administrator picks how he wants to sort doctors based on their specialty:      + Alphabetically ascending      + Alphabetically descending    * System sorts the list of doctors, showing the ones with chosen filter first.   Pressing “View employee” button   1. Once the “View employee” button is clicked, deactivate employee form is opened. 2. System launches the “ View employee’s details” functionality.   Pressing “Deactivate doctor” button   1. Once the “Deactivate doctor” button is clicked, deactivate doctor form is opened. 2. System launches the “ Deactivate doctor” functionality. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View doctor’s details |
| ID | Doc\_1.2.1.5 |
| Description | Administrator has possibility to see significant information as free terms, personal data, room he admits patients and move to other forms.  System opens a view of doctor’s details form.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | View list of doctors |
| Finish conditions | Form with view of doctor’s details |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View of doctor” button. 2. System opens a new form that allows to view doctor’s details.  * Free terms * Personal data * Room they admits  1. Following buttons are available:    * “Cancel” button    * “Review calendar” button    * “Edit doctor’s data” button    * “Deactivate doctor” button   Alternative:  Closing view list of doctors form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of doctor’s details cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view of doctor’s details form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Pressing “Review calendar” button   1. Once the “Review calendar” button is clicked, view of calendar form is opened. 2. System launches the “Review calendar” functionality.   Pressing “Edit doctor’s data” button   1. Once the “Edit doctor’s data” button is clicked, edit of doctor’s data form is opened. 2. System launches the “ Edit doctor’s data” functionality.   Pressing “Deactivate doctor” button   1. Once the “Deactivate doctor” button is clicked, deactivate doctor form is opened. 2. System launches the “ Deactivate doctor” functionality. |
| Notices |  |

# User cases: Employees



|  |  |
| --- | --- |
| Use case name | Add new employee |
| ID | Emp\_1.2.2.1 |
| Description | Administrator has possibility to enter data of a new employee and them to the system.  System opens adding new employee form.  After correct data input, system adds an employee to the database.  System sends and e-mail with information of successful adding employee to the system and login data to the employee’s e-mail address. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of employees form or main page. |
| Finish conditions | New employee added to the system. |
| Input data | Mandatory:   * Personal data   + Name   + Last name   + Residential address * E-mail * PESEL * Phone number * Role |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Add new employee” button. 2. System opens a new form that allows to add new employee to the system after filling the fields. 3. Administrator needs to fill mandatory fields. 4. Button “Add” comes to be available. 5. System validates that PESEL is correct in accordance with user story. 6. System validates that e-mail is correct in accordance with user story. 7. System validates that phone number is correct in accordance with user story. 8. Administrator clicks “Add” button. 9. System sends an e- mail to the employee with information of adding to the system. 10. System sends an e-mail to the employee with information of login data. 11. System shows an information of successful adding an employee to the database. 12. System adds the employee to the database and sets their status to active. 13. Once button “Ok” is pressed, system closes the message window and adding new employee form.   Alternative:  User input wrong PESEL:   1. System using PESEL validator checks that input PESEL is incorrect. 2. System shows the pop-up window of wrong input PESEL. 3. Button “OK” closes the pop-up window.   User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling adding a new employee:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of adding new employee cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and adding new employee form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

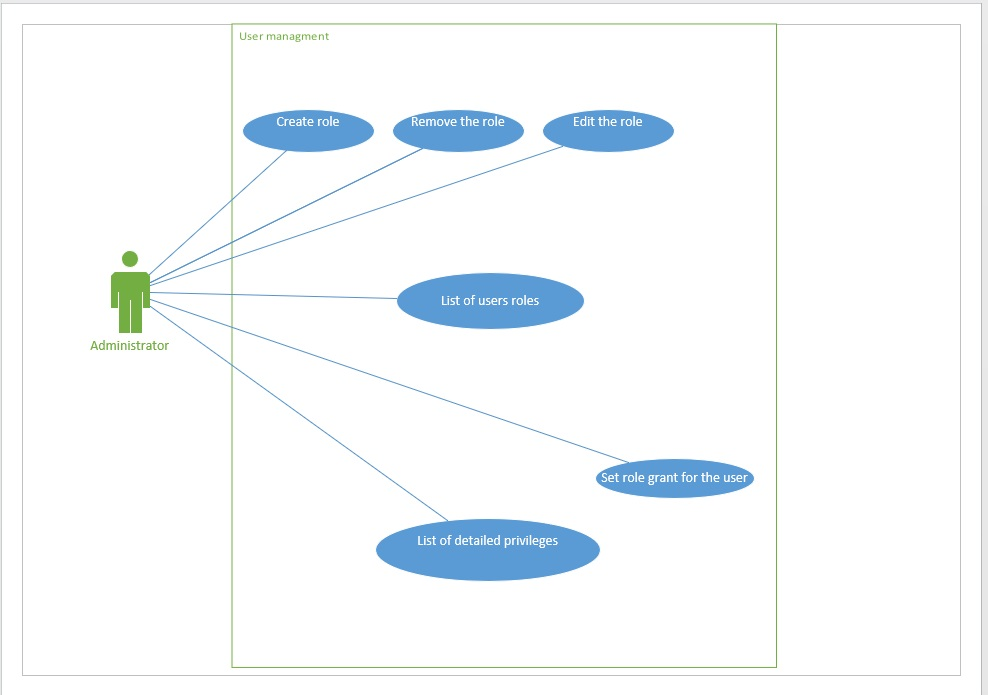
|  |  |
| --- | --- |
| Use case name | Editing employee’s data |
| ID | Emp\_1.2.2.2 |
| Description | Administrator has possibility to update employee’s data in the system and confirm it.  System opens editing employee’s data form.  After correct data input, system edits employee’s data in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of employees or view of employee’s details form. |
| Finish conditions | Employee’s data change in the system. |
| Input data | Mandatory:   * Personal data * E-mail * PESEL * Phone number * Role |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Edit employee’s data” button. 2. System opens a new form that allows to update existing employee’s data in the system after filling the fields. 3. Administrator selects which field he wants to change.    * E-mail    * Phone number 4. After changing one or more fields button “Change Data” comes to be available. 5. Administrator can change employee’s status in accordance with user stories. 6. Administrator clicks “Change data” button. 7. System validates that PESEL is correct. 8. System validates that e-mail is correct. 9. System validates that phone number is correct. 10. System sends an e- mail to the employee with information of editing data in the system. 11. System shows a message window with an information of successful editing employee’s data in the database. 12. Once button “Ok” is pressed, system closes the message window and editing employee’s data form.   Alternative:  User input wrong PESEL:   1. System using PESEL validator checks that input PESEL is incorrect. 2. System shows the pop-up window of wrong input PESEL. 3. Button “OK” closes the pop-up window.   User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling editing employee’s data:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of editing employee’s data cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and editing employee’s data form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Status is changed from inactive to active:   1. System restores data from the archive. 2. System sets employee’s status to active.   Status is changed from active to inactive:   1. System launches employee’s deactivating form. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Employee’s deactivation |
| ID | Emp\_1.2.2.3 |
| Description | Administrator has possibility to move data of an existing employee to archive and confirm it.  System opens employee’s deactivation form.  After correct password input, system moves employee’s data to the archive. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Employee is active. |
| Finish conditions | Employee’s status is changed to inactive.  Employee’s data is moved to the archive. |
| Input data | Administrator’s password |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Deactivate employee” button. 2. System opens a new form that allows to deactivate employee in the system after typing administrator’s password. 3. Administrator has to input their password. 4. Button “Deactivate” comes to be available. 5. Administrator clicks “Deactivate” button. 6. System checks if employee has any planned work. 7. System moves employee’s data to archive. 8. System changes employee’s status to inactive. 9. System shows a message window with an information of successful deactivating the employee. 10. Once button “Ok” is pressed, system closes the message window and employee’s deactivating form.   Alternative:  Canceling deactivating employee:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of deactivating employee cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and deactivating employee form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Administrator typed wrong password.   1. System shows pop-up window that the password is incorrect. 2. Button “OK” closes pop-up window. 3. Empty employee’s deactivation form is launched. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View list of employees |
| ID | Emp\_1.2.2.4 |
| Description | Administrator has possibility to view list of employees  System opens a form with list of employees.  System allows you to filter employees on active or inactive status.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Logged in administrator |
| Finish conditions | Form with list of employees. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View list of employees” button. 2. System opens a form with list of employees.    * Name    * Surname    * Role    * E-mail address 3. Following buttons are available:    * “Cancel” button    * “View employee” button    * “Deactivate employee” button   Alternative:  Closing view list of employees form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of employee’s cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view list of employee’s form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Filtering employees based on:   1. Status    * Administrator picks which status and employees assigned to it they want to see.    * System filters the list of employees, showing the ones with chosen status. 2. Roles    * Administrator picks which role and employees assigned to it they want to see.    * System filters the list of employees, showing the ones with chosen role.   Sorting employees based on:   1. Role    * Administrator picks how he wants to sort employees based on their role:      + Alphabetically ascending      + Alphabetically descending    * System sorts the list of employees, showing the ones with chosen filter first.   Pressing “View employee” button   1. Once the “View employee” button is clicked, deactivate employee form is opened. 2. System launches the “ View employee’s details” functionality.   Pressing “Deactivate employee” button   1. Once the “Deactivate employee” button is clicked, deactivate employee form is opened. 2. System launches the “ Deactivate employee” functionality. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View employee’s details |
| ID | Emp\_1.2.2.5 |
| Description | Administrator has possibility to see significant information as personal data, working days and hours and move to other forms.  System opens a view of employee’s details form.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | View list of employees |
| Finish conditions | Form with view of employee’s details |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View of employee” button. 2. System opens a new form that allows to view employee’s details.    * Personal data 3. Following buttons are available:    * “Cancel” button    * “Edit employee’s data” button    * “Deactivate employee” button   Alternative:  Closing view list of employees form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of employee’s details cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view of employee’s details form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Pressing “Edit employee’s data” button   1. Once the “Edit employee’s data” button is clicked, edit of employee’s data form is opened. 2. System launches the “ Edit employee’s data” functionality.   Pressing “Deactivate employee” button   1. Once the “Deactivate employee” button is clicked, deactivate employee form is opened. 2. System launches the “ Deactivate employee” functionality. |
| Notices |  |

# User case: Users management



|  |  |
| --- | --- |
| Use case name | Add new user |
| ID | Use\_1.4.1.1 |
| Description | Administrator has possibility to enter data of a new user and add them to the system.  System opens adding new user form.  After correct data input, system adds an user to the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of users form or main page. |
| Finish conditions | New user added to the system. |
| Input data | * Personal data * E-mail * Phone number * Role |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Add new user” button. 2. System opens a new form that allows to add new user to the system after filling the fields. 3. Administrator needs to fill the fields. 4. Button “Add” comes to be available. 5. Administrator clicks “Add” button. 6. System validates if e-mail is correct. 7. System validates if phone number is correct. 8. System sends an e- mail to the user with information of adding to the system. 9. System sends an e-mail to the user with information of login data. 10. System shows an information of successful adding an user to the database. 11. System adds the user to the database and sets their status to active. 12. Once button “Ok” is pressed, system closes the message window and adding new user form.   Alternative:  User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling adding a new user:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of adding new user cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and adding new user form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Editing user’s data |
| ID | Use\_1.4.1.2 |
| Description | Administrator has possibility to update user’s data in the system and confirm it.  System opens editing user’s data form.  After correct data input, system edits user’s data in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of users or view of user’s details form. |
| Finish conditions | User’s data change in the system. |
| Input data | * Personal data * E-mail * Phone number * Role |
| Output data | Information window |
| Scenarios | Main:   1. Administrator clicks “Edit user’s data” button. 2. System opens a new form that allows to update existing user’s data in the system after filling the fields. 3. Administrator selects which field he wants to change.  * E-mail * Phone number  1. After changing one or more fields button “Change data” comes to be available. 2. Administrator can change user’s status. 3. Administrator clicks “Change data” button. 4. System validates that e-mail is correct. 5. System validates that phone number is correct. 6. System sends an e- mail to the user with information of editing data in the system. 7. System shows a message window with an information of successful editing user’s data in the database. 8. Once button “Ok” is pressed, system closes the message window and editing user’s data form.   Alternative:  User input wrong e-mail:   1. System validates that input e-mail is incorrect. 2. System shows the pop-up window with information of wrong e-mail. 3. Button “OK” closes the pop-up window.   User input wrong phone number:   1. System validates that input phone number is incorrect. 2. System shows the pop-up window of wrong input phone number. 3. Button “OK” closes the pop-up window.   Canceling editing user’s data:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of editing user’s data cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and editing user’s data form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Status is changed from inactive to active:   1. System restores data from the archive. 2. System sets user’s status to active.   Status is changed from active to inactive:   1. System launches user’s deactivating form. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Deleting user |
| ID | Use\_1.4.1.3 |
| Description | Administrator has possibility to move data of an existing user to archive and confirm it.  System opens user’s deactivation form.  After correct password input, system moves user’s data to the archive. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Administrator logged in |
| Finish conditions | Data of user moves to archive in the system. |
| Input data | Administrator’s password |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Deactivate user” button. 2. System opens a new form that allows to deactivate user in the system after typing administrator’s password. 3. Administrator has to input their password. 4. Button “Deactivate” comes to be available . 5. Administrator clicks “Deactivate” button. 6. System moves user’s data to archive. 7. System changes user’s status to inactive. 8. System shows a message window with an information of successful deactivating the user. 9. Once button “Ok” is pressed, system closes the message window and user’s deactivating form.   Alternative:  Canceling deactivating user:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of deactivating user cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and deactivating user form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Administrator typed wrong password.   1. System shows pop-up window that the password is incorrect. 2. Button “OK” closes pop-up window. 3. Empty user’s deactivation form is launched.   User has planned work:   1. Pop-up window appears with message that you can’t deactivate user because they have planned work. 2. System doesn’t change user’s status. 3. Button “OK” closes the pop-up window and deactivating user form. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View list of users |
| ID | Use\_1.4.1.4 |
| Description | Administrator has possibility to view list of users  System opens a form with list of users.  System allows you to filter users on active or inactive status.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being on the main page. |
| Finish conditions | Form with list of users. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View list of users” button. 2. System opens a form with list of users.    * Data shown in the list:      + Name      + Surname      + E-mail address      + Role 3. Following buttons are available:    * “Cancel” button    * “View user” button    * “Deactivate user” button    * “Set role” button    * “Change password” button   Alternative:  Closing view list of user form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of users cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view list of users form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Pressing “View user” button   1. Once the “View user” button is clicked, view of user’s details form is opened. 2. System launches the “View user’s details” functionality.   Pressing “Deactivate user” button   1. Once the “Deactivate user” button is clicked, deactivate user form is opened. 2. System launches the “ Deactivate user” functionality.   Pressing “Set role” button   1. Once the “Set role” button is clicked, the role grant for the user form is opened. 2. System launches the “The role grant for the user” functionality.   Pressing “Change role” button   1. Once the “Change role” button is clicked, changing password form is opened. 2. System launches the “User password changing by administrator” functionality.   Filtering users based on:   1. Status    * Administrator picks which status and users assigned to it they want to see.    * System filters the list of users, showing the ones with chosen status. 2. Roles    * Administrator picks which role and users assigned to it they want to see.    * System filters the list of users, showing the ones with chosen role.   Sorting users based on:   1. Roles    * Administrator picks how he wants to sort users based on their roles:      + Alphabetically ascending      + Alphabetically descending    * System sorts the list of users, showing the ones with chosen filter first. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | View user’s details |
| ID | Use\_1.4.1.5 |
| Description | Administrator has possibility to see significant information as personal data, working days and hours and move to other forms.  System opens a view of user’s details form.  Opened form allows you to go to further forms. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | View list of users |
| Finish conditions | Form with view of user’s details |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “View of user” button. 2. System opens a new form that allows to view user’s details.    * Personal data 3. Following buttons are available:    * “Cancel” button    * “Edit user’s data” button    * “Deactivate user” button    * “Set role” button    * “Change password” button   Alternative:  Closing view list of user’s form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of view list of user’s details cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and view of user’s details form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Pressing “Edit user’s data” button   1. Once the “Edit user’s data” button is clicked, edit of user’s data form is opened. 2. System launches the “ Edit user’s data” functionality.   Pressing “Deactivate user” button   1. Once the “Deactivate user” button is clicked, deactivate user form is opened. 2. System launches the “ Deactivate user” functionality.   Pressing “Set role” button   1. Once the “Set role” button is clicked, the role grant for the user form is opened. 2. System launches the “The role grant for the user” functionality.   Pressing “Change role” button   1. Once the “Change role” button is clicked, changing password form is opened. 2. System launches the “User password changing by administrator” functionality. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | List of detailed privileges |
| ID | Use\_1.4.4.1 |
| Description | Administrator has possibility to see all privileges.  System opens list of privileges form.  System allows you to filter them. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the list of user’s role. |
| Finish conditions | List of user’s role is launched. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “Manage privileges” button. 2. System opens a new form that allows to see the list of detailed privileges. 3. Button “Cancel” is available.   Alternative:  Closing list of privileges form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of list of privileges cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and list of privileges form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Sorting user’s based on:   1. Name    * Administrator picks how he wants to sort users based on their role:      + Alphabetically ascending      + Alphabetically descending    * System filters the list of users, showing the ones with chosen filter first. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | List of users’ roles |
| ID | Use\_1.4.4.2 |
| Description | Administrator has possibility to see all roles.  System opens list of users’ roles form with privileges assigned to each role. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of users or main page. |
| Finish conditions | View list of users or main page is launched. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator clicks “Manage roles” button. 2. System opens a new form that allows to see the list of users’ roles with privileges assigned to them. 3. Following buttons are available:    * “Cancel” button    * “Create role” button    * “Edit role” button   Alternative:  Closing list of users’ roles form:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of list of users’ roles cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and list of users’ roles form. 4. If the button “Cancel” is pressed, system closes pop-up window   Pressing “Create role” button   1. Once the “Create role” button is clicked, creating role form is opened. 2. System launches the “Create role” functionality.   Pressing “Edit role” button   1. Once the “Edit role” button is clicked, editing role form is opened. 2. System launches the “Edit role” functionality. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Role’s creation |
| ID | Use\_1.4.4.3 |
| Description | Administrator has possibility to create new role in the system.  System opens adding new role form.  After pressing a button, system adds role to the system.  System shows an information of successful adding a role in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the list of users’ roles |
| Finish conditions | New role added to the system |
| Input data | Name of the role  Picked privileges from the list |
| Output data | Message window |
| Scenarios | Main:   1. Administrator click “Create role” button. 2. System opens adding new role form. 3. Administrator needs to fill the data.  * Administrator create new name role * Administrator can set privileges to this role  1. Button “Add role” comes to be available. 2. Administrator clicks confirming button. 3. System adds the role to the database. 4. System shows an information of successful adding a role to the database. 5. Once button “Ok” is pressed, system closes the message window and adding role form.   Alternative:  Canceling adding a new role:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of adding new role cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and adding new role form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Editing the role |
| ID | Use\_1.4.4.4 |
| Description | Administrator has possibility to edit existing role in the system.  System opens editing role form.  After pressing the button, system edits the role in the database.  System shows an information of successful editing a role in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the list of users’ roles |
| Finish conditions | Role edited in the system  List of users’ roles launched.  Message window of succeed editing the role to the system. |
| Input data | N/A |
| Output data | Message window |
| Scenarios | Main:   1. Administrator clicks “Edit role” button. 2. System opens editing role form that allows to edit existing role in the system after filling the fields. 3. Administrator needs to fill the fields.  * Administrator can change name role * Administrator can set name role * Administrator can assign privileges to this role  1. Button “Change data” comes to be available. 2. Administrator clicks “Change data” button. 3. System edits a role in the database. 4. System shows a message window with an information of successful editing role in the database. 5. Once button “Ok” is pressed, system closes the message window and editing role form.   Alternative:  Canceling editing role:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of editing role cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and editing role form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | Role removing |
| ID | Use\_1.4.4.5 |
| Description | Administrator has possibility to remove existing role from the system.  System opens a form to delete a role.  After correct password input, system removes the role from the system.  System shows an information of successful deleting a role. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the list of users’ roles |
| Finish conditions | Role deleted from the system. |
| Input data | Administrator’s password |
| Output data | Message window |
| Scenarios | Main:   1. Administrator click “Delete role” button. 2. System opens a new form that allows to remove a role from the system after typing administrator’s password. 3. Administrator has to input their password. 4. Button “Remove” comes to be available. 5. Administrator clicks “Remove” button. 6. System shows a message window with an information of successful removing the role. 7. Once button “Ok” is pressed, system closes the message window and role removing form.   Alternative:  Canceling removing a role:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of removing a role cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and removing a role form. 4. If the button “Cancel” is pressed, system closes pop-up window.   Administrator typed wrong password.   1. System shows pop-up window that the password is incorrect. 2. Button “OK” closes pop-up window. 3. Empty role removing form is launched.   Role is assigned to the user:   1. Pop-up window appears with message that you can’t remove a role because it has users assigned to. 2. System doesn’t remove it. 3. Button “OK” closes the pop-up window and removing a role form. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | The role grant for the user |
| ID | Use\_1.4.4.6 |
| Description | Administrator has possibility to set roles with privileges to the user.  System opens a form to grant someone a role.  After correct password input, system assign someone a chosen role.  System shows an information of successful granting a role in the database. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Being in the view list of users or view of user’s details |
| Finish conditions | Role granted to the user.  List of users’ roles or view list of users launched.  Message window of successful granting role to a user. |
| Input data | Administrator’s password |
| Output data | Message window |
| Scenarios | Main:   1. Administrator click “Set role” button. 2. System opens a new form that allows you to grant a role to a user after typing administrator’s password. 3. Administrator selects the user. 4. Administrator selects the role from the list. 5. Administrator clicks “Set” button. 6. System shows a message window with an information of successful granting the role. 7. Once button “Ok” is pressed, system closes the message window and role granting form.   Alternative:  Canceling removing a role:   1. User clicks “Cancel” button, which is available all the time. 2. System shows the pop-up window with the request of confirmation of removing a role cancelation message. 3. If the button “OK” is pressed, system closes the pop-up window and removing a role form. 4. If the button “Cancel” is pressed, system closes pop-up window. |
| Notices |  |

|  |  |
| --- | --- |
| Use case name | User password changing by administrator |
| ID | Use\_1.4.5 |
| Description | Administrator has possibility to change user’s password.  System opens changing password form.  After successful changing password, the system logs out the user and pops-up the window of succeed password change. |
| Author | Karolina Kapituła |
| Actor | Administrator |
| Start conditions | Proper user already exists. |
| Finish conditions | System changes the password |
| Input data | Login  New password  Confirm password |
| Output data | Pop-up window |
| Scenarios | Main:   1. Administrator clicks “Change user’s password” button. 2. System opens a new form that allows to change user’s password in the system after filling the fields. 3. Administrator fills following fields:    * New password    * Confirm password 4. System checks that strings in those fields are the same. 5. Button “Change” comes to be available . 6. System validates correctness of entered password with accordance to password policy.    * Password length: not less than 8 characters and not more than 15 characters.    * Use of each of the following types of marks at least once:      + lowercase      + uppercase      + digital      + special sign: "-", "\_", "!". "#", "$", "\*" 7. System displays a success message in the form of a pop-up window after password meeting the validation criteria. 8. Button “OK” comes to be available . 9. Once the button “OK” is pressed, the pop-up window and change password window are closed. 10. System changes the password in the database.   Alternative:  Password change cancelation:   1. User pushes “Cancel” button. 2. System shows pop-up window with the request of confirmation of change password cancelation message. 3. Buttons “OK” and “Cancel” are available. 4. If the “OK” button is pressed, the pop-up window and change password window are closed. 5. If the “Cancel” button is pressed system closes the message window.   String in “New password” and “Confirm password” are different:   1. Input data in “New password” and “Confirm password” fields are different. 2. Button “OK” is unavailable. 3. System displays pop-up window with message that those fields aren’t the same. 4. Button “OK” is available. 5. Once the “OK” button is pressed, system closes the pop-up window.   Password doesn’t meet the validation criteria:   1. System displays an error message in the form of pop-up window. 2. The “OK” button is available on the pop-up window. 3. Once the “OK” button is pressed, the pop-up window is closed. |
| Notices |  |