1. System administration module

Login/logout

Functionalities of login into the system and logout from the system is part of the administration module but is actually used for getting access to each of the modules.

All users use this part of the system to log in/log out.

After correct login, the system automatically opens the appropriate module based on logged user’s rights.

**Epic: Login**

**User stories:**

Logging into the system

|  |  |
| --- | --- |
| Name | Login into the system |
| ID | LOG\_01 |
| The aim | Login the user into the system |
| Description | Non-logged user has the possibility to login into the system. For the login user has to put two kinds of information:   * Login name * Password   After the correct login process, the system launches the proper module, based on the user’s rights |
| Author | Jarosław Zając |
| Actor | Non-logged user |
| Start conditions | N/A |
| Final 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 then 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 |  |

Resigning of logging in

|  |  |
| --- | --- |
| Name | Resigning of logging in |
| ID | LOG\_02 |
| The aim | Cancel logging in process |
| Description | Non-logged user has the possibility to cancels the login process before confirming the login details.   * Once the "Cancel" button is pressed the login form is closed |
| Author | Mikołaj Uznański |
| Actor | Non-logged user |
| Start conditions | N/A |
| Final conditions | User is not logged-in  Empty login form is closed. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User launches system 2. System opens login window 3. User can enter his login data or leave the fields empty. 4. User presses the Cancel button. 5. The user is taken to an empty login form. |
| Notices |  |

Incorrect login

|  |  |
| --- | --- |
| Name | Incorrect login |
| ID | LOG\_05 |
| The aim | System checks correction of input login data |
| Description | System checking if user login data is correct which enables safes access to the system for proper users only.  Acceptance criteria:   * Once input data is incorrect, the system generates and views a message with information about incorrect input data |
| Author | Mikołaj Uznański |
| Actor | Non-logged user |
| Start conditions | The non-logged user is trying to get access to system. |
| Final conditions | The system presents a message indicating that the user is entering incorrect data. |
| Input data | Login name  password |
| Output data | N/A |
| Scenarios | Main:   1. System opens login window 2. User trying to login with incorrect login data. 3. System verifies the entered data. 4. The system generates and views a message with information about incorrect input data. 5. The system inform user about two more login attempts before system blocking |
| Notices |  |

System blocking

|  |  |
| --- | --- |
| Name | System blocking |
| ID | LOG\_06 |
| The aim | System blocking the possibility of logging again for entering incorrect data for the third time in a row. |
| Description | The system is blocking the possibility of logging again for chosen time (parameter) X when user trying to entering incorrect input data three times in a row, which provides the security of the system will by higher.  Acceptance criteria   * The third incorrect login data entering causes blocking of the possibility of using next attempt of login (entering input data) * The system starts showing the time remaining to unlock the possibility of entering input data |
| Author | Mikołaj Uznański |
| Actor | non-logged user |
| Start conditions | The non-logged user uses incorrect login data three times |
| Final conditions | System blocking for the user the possibility of logging again for X (parameter) minutes |
| Input data | Login name  password |
| Output data | N/A |
| Scenarios | Main:   1. User launches system. 2. System opens login window. 3. User uses incorrect login data. 4. System verifies the entered data. 5. After third unsuccessful attempt the system blocks input data fields. 6. the login button and the system informs the user about the remaining time during which they cannot log in. |
| Notices |  |

Input data unlock

|  |  |
| --- | --- |
| Name | Input data unlock |
| ID | LOG\_07 |
| The aim | System automatically unlocks the possibility of entering input data |
| Description | The system is automatically unlocks the possibility to enter login in data for non-logged user. When locking time is over, user get access to try to login again.  Acceptance criteria   * If the time of system locking is complete, input data fields are available again and information about time to unlock disappears |
| Author | Mikołaj Uznański |
| Actor | System |
| Start conditions | The system is blocked for X (parameter) minutes |
| Final conditions | System automatically unlocks the possibility of entering input data. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. The time for blocking access has ended. 2. System unlocks access. 3. User get access to input data login once again. |
| Notices |  |

**Epic: Password changing**

**User stories:**

Application for changing password

|  |  |
| --- | --- |
| Name | Application for changing password |
| ID | PAS\_01 |
| The aim | non-logged in system’s user have possibility to submit an application to change the password |
| Description | The non-logged in system’s user have the option to submit an application to change the password to recover forgotten password. This operation is based on a request from user.  System verifies and unlock this option, only when address entered into the appropriate field contains exactly one "@", otherwise application for changing password is going to be still blocked. When user meeting the requirements and send request to change password, system inform the user by send message with the link to form of password changing. |
| Author | Mikołaj Uznański |
| Actor | Non-logged in system’s user |
| Start conditions | Non-logged in system’s user report a desire to submit an application password changing |
| Final conditions | When user meeting the requirements, system send the link to change password |
| Input data | User’s login User’s e-mail address |
| Output data | N/A |
| Scenarios | 1. Non-logged in system’s user report a desire to submit an application password changing. 2. Non-logged in system’s user entered the email addresses into the appropriate field 3. The system verifies whether the email address entered into the appropriate field contains exactly one "@" character and if so, the "OK" button becomes available 4. Non-logged in system’s user pressed the “OK" button. 5. The system send a message with information about the link to form of password changing is displayed on the screen.   Alternative:  Incorrect email address entered     1. The system verifies whether the email address entered into the appropriate field contains exactly one "@" character and if not, the "OK" button stay be unavailable 2. The "OK" button is unavailable (there is no possibility to push it) 3. The message with link to form of password changing has been not sent |
| Notices |  |

Password change link

|  |  |
| --- | --- |
| Name | Password change link |
| ID | PAS\_02 |
| The aim | The user receives a link to change the password to the e-mail address provided earlier |
| Description | The system generates a link to password change and send it to previously submitted email address. Non-logged in system user will be able to open appropriate form to password change from this link |
| Author | Mikołaj Uznański |
| Actor | System’s user |
| Start conditions | System’s user send previously submitted email address. |
| Final conditions | System’s user receive password change link |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. System’s user indicate email address 2. System checks and confirm that email address is the same as address stored in the database 3. System sends the link to change password to the appropriate email address   Alternative:  Wrong address email     1. System found that the email address is not found in the database 2. System doesn’t send the link to change password to the provided email address |
| Notices |  |

Password change form launching

|  |  |
| --- | --- |
| Name | Password change form launching |
| ID | PAS\_03 |
| The aim | System display the password change form after clicking link |
| Description | When non-logged system’s user click the received link to the password change form, immediately gets access to put a new password |
| Author | Mikołaj Uznański |
| Actor | Non-logged system’s user |
| Start conditions | Non-logged system’s user click the link received on address email |
| Final conditions | Non-logged system’s user is able to put a new password |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Non-logged system’s user click the password change link 2. The password change form has been shown to the non-logged system’s user 3. Non-logged system’s user is able to set new password 4. Non-logged system’s user must decide to apply new changes |
| Notices |  |

New password entering

|  |  |
| --- | --- |
| Name | New password entering |
| ID | PAS\_04 |
| The aim | The non-logged system's user changing password to new one and confirm it |
| Description | System uses displayed a form “Password change form launching”. Open the form “OK” button is unavailable. User must enter the data into two fields:  - New password  - Confirm password  Once the two are the same, button “OK” becomes available. After pressing this button, system checks if the data meets validation criteria:  - Password length; not less than 8 characters, not more than 15 characters.  Following types of marks are used at least once:  - Lowercase  - Uppercase  - Digital  - Special sing: “-”, “\_”, “!”, “#”, “$”, “\*”  If the validation process is successful, system will displays a message in form of pop-up window. In the window there is “OK” button, which closes window and the password change form. Password is changed successfully. If the valiidation doesn’t meet criteria, system displays a message in form of pop-up window. On the pop-up window there is “OK” button, which closes the window. |
| Author | Mikołaj Uznański |
| Actor | Non-logged system’s user |
| Start conditions | Link to password change form is pressed |
| Final conditions | Password is changed |
| Input data | New password  Confirm password |
| Output data | Pop-up window with message about successful changing of password |
| Scenarios | Main:  Matching passwords and password meets validation criteria   1. User enters data into:   - New password  - Confirm password   1. System verifies that passwords are matching 2. “OK” button is now available 3. System validates correctness of entered data 4. Pop-up window opens up. 5. User presses “OK” button on pop-up window 6. Pop-up window and password change form closes.   Alternative:  Passwords are not matching   1. System verifies that passwords are not matching 2. “OK” button stays unavailable   Alternative:  New password doesn’t meet validation criteria   1. System validates correctness of entered data 2. Pop-up window opens up 3. User presses “OK” button on pop-up window 4. Pop-up window closes. |
| Notices |  |

Password change cancelation

|  |  |
| --- | --- |
| Name | Password change cancelation |
| ID | PAS\_05 |
| The aim | The possibility to resigning from password changing at any time. |
| Description | The non-logged in system's user is able to close password change form at any time user wants. After pressing ”Cancel” button, system presents pop-up window with the request for confirmation, with two buttons “OK” and “Cancel”. After “OK” button is pressed, password change form closes. After “Cancel” button is pressed, system closes pop-up window, but if there was any data entered in password change form, it’s cleared now. |
| Author | Mikołaj Uznański |
| Actor | Non-logged system’s user |
| Start conditions | Password change form is presented |
| Final conditions | Closing password change form |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User presses “Cancel” button on password change form. 2. Pop-up window opens up. 3. User presses “OK” button on pop-up window. 4. Pop-up window and password change form has been closed   Alternative:  Pressing “Cancel” instead of “OK”   1. User presses “Cancel” button on pop-up window. 2. Pop-up window closes 3. All data previously entered in password change form is cleared. |
| Notices |  |

Cancelation of password change cancelation

|  |  |
| --- | --- |
| Name | Cancelation of password change cancelation |
| ID | PAS\_06 |
| The aim | The possibility to cancel password change cancelation |
| Description | The non-logged in system's user is able to cancel the process of “password changing cancelation”. After pressing ”Cancel” button, system closes the pop-up message window of cancelation of password change, otherwise data entered to the password change form will be lost. |
| Author | Mikołaj Uznański |
| Actor | Non-logged system’s user |
| Start conditions | Cancelation Password change form is presented |
| Final conditions | Closing the cancelation of password change cancelation |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User presses “Cancel” button on pop-up window 2. Pop-up window closes 3. All data previously entered in password change form is cleared |
| Notices |  |



**Epic: Logout**

Logging out from the system

|  |  |
| --- | --- |
| Name | Logging out from the system |
| ID | LOG\_03 |
| The aim | Logged-in user have possibility to log out from the system |
| Description | The user has the possibility to ends his work with the system safely and correctly by logging out of it. After that user sees system login form with empty fields. |
| Author | Mikołaj Uznański |
| Actor | Logged-in user |
| Start conditions | The user is logged-in |
| Final conditions | User has been logged out of the system.  User is not logged-in, system presents login form with empty fields |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User is logged in to the system. 2. The user presses the Logout button. 3. The system displays a logout attempt message. 4. The user presses the OK button. 5. User is logged out from the system   Alternative:  User decided to   1. The users presses the cancel button. 2. Message window is closed |
| Notices |  |



System user management – clinic staff

* 1. **Epic: List of users**
  2. **User stories:**

View list of users

|  |  |
| --- | --- |
| Name | View list of users |
| ID | LIS\_01 |
| The aim | Administrator can list the available users |
| Description | The Administrator has access to list of users, and possibility to see available users from this list. Additionally administrator can perform the following operations:   * Add new user * View user’s details * Edit user data * Deactivate activated user * Activate deactivated user |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator is logged into the system |
| Final conditions | The list of users is presented |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator choosing “Users” in the system menu, to launch the form with a list of users 2. System opens form with list of users |

Adding new system user

|  |  |
| --- | --- |
| Name | Adding new system user |
| ID | UMG\_01 |
| The aim | Administrator can create a new system user |
| Description | The Administrator has a possibility to create a new system’s user and confirm it, after that the new user is registered in to the system.   * Button “Add new user” is available on the list of users * Administrator select employee from a list * Pressing the button “Add new user” send a request to system to create a new user. * The system adds the new User to the database. * The system generates login and password and sends an e-mail to the user with information of adding to the system and login data. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator create new user |
| Final conditions | New user is registered in to the system |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator click “add new user” button 2. Administrator must select a employee 3. Administrator click the “Accept” button 4. The System adds the new user to the database 5. The System generate login and password and sends an e-mail to the user with information of adding to the system and login data.   Alternative:  Cancel button is pressed   1. The System closes the adding new user window. |

View user’s details

|  |  |
| --- | --- |
| Name | View user’s details |
| ID | UMG\_02 |
| The aim | Administrator can view users details |
| Description | The Administrator has the possibility to view user details, all information about user details.  Acceptance criteria:   * The "View" button is available on the list of users, besides the user’s name. * Pressing the "View" button system opens a user details form (edit form). * Two buttons are available on the screen: “Save” and “Cancel”. * When the “Cancel” button is pressed, the system closes the user details form. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator open user edit form |
| Final conditions | Users details displayed |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator press the “view” button, to open the users details 2. The system opens edit form 3. The Administrator see the users details in edit form |

Edit of the user’s data

|  |  |
| --- | --- |
| Name | Edit of the user’s data |
| ID | UMG\_03 |
| The aim | Administrator has a possibility to edit user data |
| Description | The Administrator have the possibility to edit user data and confirm changes, to user details were up-to-date  Acceptance criteria:   * On the list of users, the button "Edit" is available, besides the user’s name. * Pressing the "Edit" button opens a user edit form. * All data of the edited user can be changed, including the role of the user in the system * The system sends an e-mail to the user with information about changes. * Cancel button is available to click * Once “Cancel” button is pressed system closes the edit form * If the user made some changes and try to leave the editing form without saving it, the system displays a pop-up with confirmation of leaving. * When the “Save” button is pressed system saves changes in the database and views the list of users. * The “Change password” button is available. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator open user edit form |
| Final conditions | Users details displayed |
| Input data | Changes in user's details |
| Output data | N/A |
| Scenarios | Main:   1. Administrator press the “edit” button, to edit the users details 2. The system opens edit form 3. The Administrator have a possibility to edit users details 4. The system sends an e-mail to the user with information about changes. If there was a changes. 5. The Administrator press the “cancel” button 6. The System closes the edit form.   Alternative:  Unsaved changes   1. The System displays a pop-up with confirmation of leaving button   “Save” button Is pressed   1. The System saves changes in the database and views the list of users. |

User deactivation from the list

|  |  |
| --- | --- |
| Name | User deactivation from the list |
| ID | UMG\_04 |
| The aim | Administrator has a possibility to deactivation user |
| Description | The Administrator have the possibility to deactivate a user from list of users. When Administrator confirm this action user will be unable to perform any action in the system. To do this the Administrator must press a button "Deactivate user". When this button is pressed, the system displays a message in the form of a pop-up window with two buttons “Deactivate” and “Cancel”. When the “Cancel” button is pressed there is nothing changing, but if the “Deactivate” button is pressed, the system changes the user status to “Inactive”. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator press the button to deactivate a user from system |
| Final conditions | The user is deactivated |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator decide to deactivate a user from the system 2. The Administrator press the button "Deactivate user" on the list of user. 3. The Administrator have to decide to press “Deactivate” or “Cancel” button 4. The Administrator pressed “Deactive” button 5. The System changes the user status to “Inactive”.   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The System does not changes the user status |

User deactivation from edit form

|  |  |
| --- | --- |
| Name | User deactivation from edit form |
| ID | UMG\_05 |
| The aim | Administrator has a possibility to deactivation user |
| Description | The Administrator have the possibility to deactivate a user from the details form. When Administrator confirm this action user will be unable to perform any action in the system. To do this the Administrator must press a button "Deactivate user". When this button is pressed, the system displays a message in the form of a pop-up window with two buttons “Deactivate” and “Cancel”. When the “Cancel” button is pressed there is nothing changing, but if the “Deactivate” button is pressed, the system changes the user status to “Inactive”. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator press the button to deactivate a user from system |
| Final conditions | The user is deactivated |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator decide to deactivate a user from the system 2. The Administrator press the button "Deactivate user" on the details forms. 3. The Administrator have to decide to press “Deactivate” or “Cancel” button 4. The Administrator pressed “Deactive” button 5. The System changes the user status to “Inactive”.   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The System does not changes the user status |

User activation from the list

|  |  |
| --- | --- |
| Name | User activation from the list |
| ID | UMG\_06 |
| The aim | Administrator has a possibility to activation user |
| Description | The Administrator have the possibility to activate a user from the list of users. The users gains the same actions which they could perform before being deactivated. When Administrator confirm this action user will be able to perform any action in the system. To do this the Administrator must press a button "Activate user". When this button is pressed, the system displays a message in the form of a pop-up window with two buttons “Activate” and “Cancel”. When the “Cancel” button is pressed there is nothing changing, but if the “Activate” button is pressed, the system changes the user status to “Active”. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator press the button to Activate a user to the system |
| Final conditions | The user is activated |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. Administrator decide to activate a user to the system 2. The Administrator press the button "Activate user" on the list of users. 3. The Administrator have to decide to press “Activate” or “Cancel” button 4. The Administrator pressed “Active” button 5. The System changes the user status to “Active”.   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The System does not changes the user status |

User activation from user details

|  |  |
| --- | --- |
| Name | User activation from user details |
| ID | UMG\_07 |
| The aim | Administrator has a possibility to activation user |
| Description | The Administrator have the possibility to activate a user from the details of users. The users gains the same actions which they could perform before being deactivated. When Administrator confirm this action user will be able to perform any action in the system. To do this the Administrator must press a button "Activate user". When this button is pressed, the system displays a message in the form of a pop-up window with two buttons “Activate” and “Cancel”. When the “Cancel” button is pressed there is nothing changing, but if the “Activate” button is pressed, the system changes the user status to “Active”. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator press the button to Activate a user to the system |
| Final conditions | The user is activated |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. The Administrator decide to activate a user to the system 2. The Administrator press the button "Activate user" on the details of users 3. The Administrator have to decide to press “Activate” or “Cancel” button 4. The Administrator pressed “Active” button 5. The System changes the user status to “Active”.   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The System does not changes the user status |

* 1. **Epic: User password management**

User’s password changing

|  |  |
| --- | --- |
| Name | User’s password changing |
| ID | UPM\_01 |
| The aim | Administrator has a possibility to change the user password |
| Description | If the user forgotten a password the Administrator has a possibility to bring back the access for user by creating the new password.  When The “Change password” button is pressed system presents the following fields and buttons after opening the user password changing form:   * New password - field to filled in by the Administrator * Confirm password - field to filled in by the Administrator * The "Save" button * The "Cancel" button   Systems checks if strings in fields: "New password" and "Confirm password" are the same.  The system validates the correctness of entered password in 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: "-", "\_", "!". "#", "$", "\*" * If the password meets the validation criteria, system doesn’t view error * If the password doesn't meet the validation criteria, system displays an error   about incorrect data * When the user clicks the “Save” button the system saves the new password in the database and closes the form. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator changing user password |
| Final conditions | The password has been changed |
| Input data | New password |
| Output data | N/A |
| Scenarios | Main:   1. The Administrator press the “Change password” button 2. The Administrator press the button "Activate user" on the details of users 3. The Administrator filled the New password 4. The Administrator filled the Confirm password 5. The Administrator press a save button 6. The Systems checks if strings in fields: "New password" and "Confirm password" are the same 7. The system validates the correctness of entered password in with accordance to password policy. 8. The System sets a new password   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The process stops   The password isn’t the same   1. The Systems displays an error about incorrect data detected   Inncorect password policy   1. The Systems displays an error about incorrect data detected |
|  |  |

System parameters and dictionaries management

* 1. **EPIC: doctor’s specializations**

Adding a specialization

|  |  |
| --- | --- |
| Name | Adding a specialization |
| ID | DSP\_01 |
| The aim | Administrator has a possibility to add a new doctor’s specialization |
| Description | The Administrator has a possibility to add a new doctor’s specialization to update the list of specializations.  When the “Add a specialization” button is pressed system opens an adding specialization form with an empty field (specialization name) and two buttons.  The “Add” button is available to click.  The “Add” button is available to click.  When the “Cancel” button is pressed system closes adding form.  If the user entered specialization name and try to leave adding form without saving, the system displays a pop-up with confirmation of leaving.  The system adds a new specialization to the list of specialization.  The system closes adding form and views the list of specializations. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator adds new specializations |
| Final conditions | The new specializations added to the list of specializations |
| Input data | adding a new doctor’s specialization |
| Output data | N/A |
| Scenarios | Main:   1. The Administrator press the “Add a specialization” button 2. The System opens adding specialization forms 3. The Administrator press the "add" button 4. The Administrator filled the specialization name 5. The Administrator save new specialization 6. The Systems adds a new specialization to the list of specialization 7. The system closes adding form and views the list of specializations   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The system closing adding form   The new specialization name has been not saved   1. The system displays a pop-up with confirmation of leaving   Inncorect password policy   1. The Systems displays an error about incorrect data detected |
|  |  |

Editing a specialization

|  |  |
| --- | --- |
| Name | Editing a specialization |
| ID | DSP\_02 |
| The aim | Administrator has a possibility to edit existing doctor specialization |
| Description | The Administrator has a possibility to edit existing doctor specialization, so if it needed to be edited administrator is able to do it  When the “Edit” button is pressed system opens an editing form with an empty field (specialization name) and two buttons.  The “Save” button is available to click.  When the “Cancel” button is pressed system closes the editing form  If the user made some changes name and try to leave the editing without saving, the system displays a pop-up with confirmation of leaving  Once the “Save” button is pressed systems save changes and views list of specializations  The system saves changes and updates them in the database |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator edit existing specializations |
| Final conditions | The specialization has been edited |
| Input data | editing existing doctor specialization |
| Output data | N/A |
| Scenarios | Main:   1. The Administrator press the “Edit” button 2. The Administrator editing existing doctor specialization 3. The Administrator press the “Save” button 4. The System save changes and views list of specializations 5. The system saves changes and updates them in the database   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button 2. The system closing editing form   The process of editing specialization has been not saved   1. The system displays a pop-up with confirmation of leaving |
|  |  |

Removing a specialization

|  |  |
| --- | --- |
| Name | Removing a specialization |
| ID | DSP\_03 |
| The aim | Administrator has a possibility to remove existing doctor specialization from the list of doctors |
| Description | The Administrator has a possibility to remove existing doctor specialization from the list of doctors so if it needed to be removed administrator is able to do it.  When the “Remove” button is pressed system displays a window with confirmation of specialization removing.  If the “Delete” button is pressed system removes a specialization from the list of specializations.  If the “Cancel” button is pressed system cancels specialization removing. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | Administrator remove existing specializations |
| Final conditions | The specialization has been removed |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. The Administrator press the “Remove” button 2. The System displays a window with confirmation of specialization removing. 3. The Administrator press the “delete” button 4. The System removes a specialization from the list of specializations   Alternative:  The “Cancel” button has been pressed   1. The Administrator pressed the “cancel” button |
|  |  |

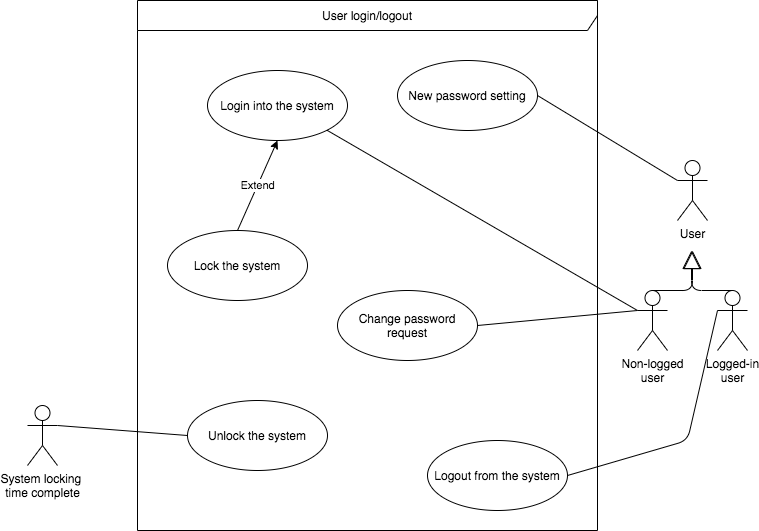
* 1. **EPIC: setting up the time to login page**

Setup the time to unlock login page

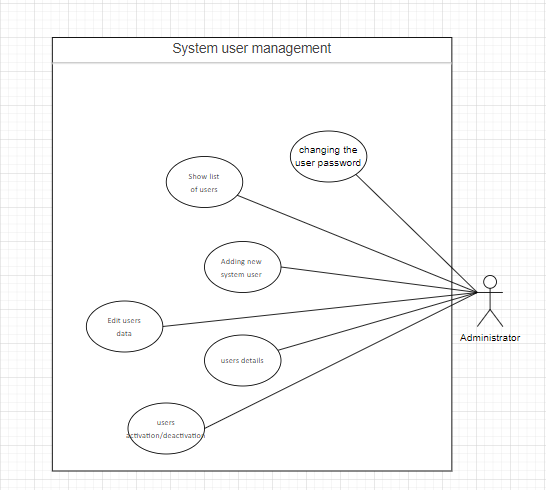
|  |  |
| --- | --- |
| Name | Setup the time to unlock login page |
| ID | STL\_01 |
| The aim | The Administrator has a possibility to set time to unlock login page |
| Description | The Administrator is able to change value of the requested time to unlock login page. They have the possibility to set up time and change value in the “Time to unlock login page” parameter. |
| Author | Mikołaj Uznański |
| Actor | Administrator |
| Start conditions | User logged in as an Administrator User launches the Settings module from the Administrator’s panel |
| Final conditions | Administrator has changed “Time to unlock login page” parameter |
| Input data | A non-negative number corresponding to the page blocking time in minutes |
| Output data | N/A |
| Scenarios | 1. Administrator sets up new time to the “Time to unlock page” parameter 2. Administrator changes the parameter 3. The new parameter of “Time to unlock page” has been changed |
| Notices |  |

Diagrams for System administration module

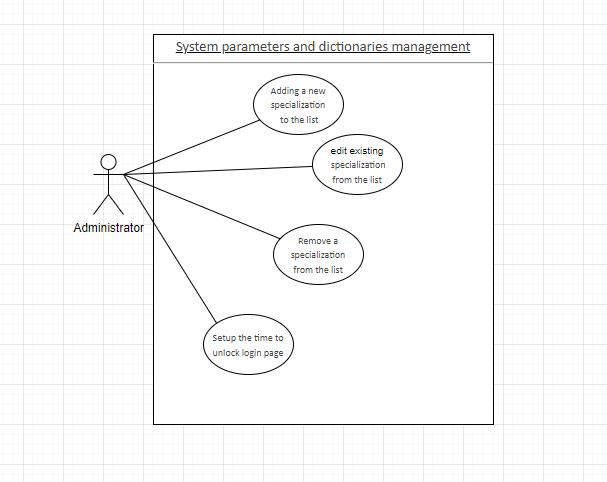
Login/Logout

* 1. 

System user management – clinic staff

* 1. 

System parameters and dictionaries managment

* 1. 

1. Business administration module

Employees management

Epic: List of employees

**User stories:**

Launching the “Employees management” functionality

**As** **an** administrator

**I want** to have a possibility to launch the functionality of employees management

**So that** I will have access to all functions of the system allow to manage staff of the clinic

Acceptance criteria

* The button allowing access to the employees’ management functionalities is available on the system administration module dashboard
* Clicking the button opens a screen with the list of employees

View list of employees

**As an** Administrator

**I want** to have possibility to review the list of employees

**So that** I can have access to all employees in one place.

Acceptance criteria:

* List of employees is presented in tabular form
* The list can be arranged in ascending and descending order
* The list can be filtered by the user
* System presents following buttons after opening the form:
  + Button “Add new employee”
  + Button “Review employee”, next to each employee – after pressing, system opens view employee’s details form.
  + Button “Deactivate employee”, next to each employee – after pressing, system opens deactivating employee form.

Filter list of employees

**As an** administrator

**I want** to filter data

**So that** I can see only the chosen data on the list.

Acceptance criteria:

* The system allows to filter employees on:
  + Active or inactive status.
  + Roles

Sort list of employees

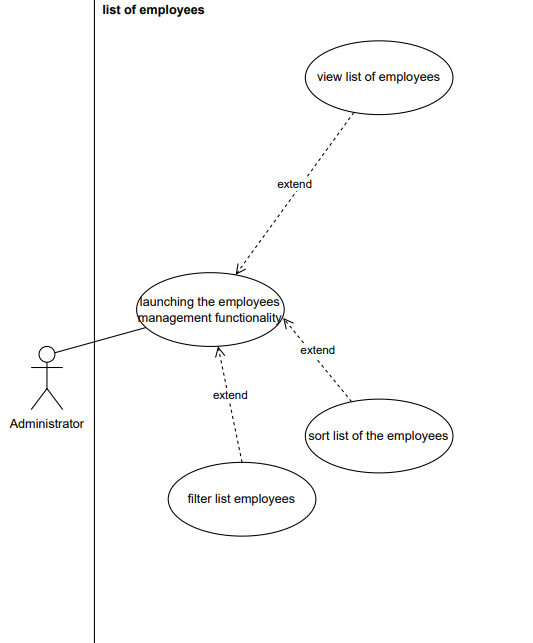
**As an** administrator

**I want** to have possibility to organize data on the list

**So that** the order of presented data will be changing.

Acceptance criteria:

* The system allows you to sort from:
  + Alphabetically ascending
  + Alphabetically descending



|  |  |
| --- | --- |
| Name | Functionality "Employees management" |
| ID | Emp\_manage\_run\_1.1 |
| The aim | Launch of the "Employee Management" functionality |
| Description | The system enables the launch of the "Employee Management" functionality. |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | Logged Administrator |
| Finish conditions | Activation of functionality |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. The user starts the program 2. The user correctly logs on to the Administrator account 3. The administrator clicks on the "Employees management" button |
| Notice |  |

|  |  |
| --- | --- |
| Name | View list of employees |
| ID | Emp\_manage\_viev\_1.2 |
| The aim | Employee list view |
| Description | Displaying a list of employees |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | "Employees Management" functionality launched |
| Finish conditions | List of employees |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. The administrator clicks on the "Employees management" button 2. The system displays a list of employees |
| Notice |  |

**Epic: Employee management**

Adding the employee

**As an** Administrator

**I want** to have possibility to add employee to the system

**So that** they will be able to start work in the clinic.

Acceptance criteria:

* Button “Add new employee” is available on the list of employees and pressing it opens a new form that allows to enter data of a new employee and confirm it.
* Button “Next” is disabled as long as mandatory fields aren’t filled.
* Button “Cancel” is available.
* Once the "Cancel" button is pressed the adding form is closed.
* Once input data is incorrect, the system generates and presents the message with information about incorrect input data.
* The form can be filled in the following kinds of data:
  + Mandatory:
    - Personal data
      * First name
      * Last name
      * The PESEL
    - Date of birth – chosen from calendar
    - Role – chosen from the drop-down list
  + Optionally:
    - Correspondence address
    - E-mail address
    - Phone number
    - Sex
* Validation rules:
  + PESEL policy:
    - The number needs to have 11 numbers
    - The First 6 numbers contain the short date of birth (DDMMYY)
    - The penultimate number must match the sex of the employee:
      * Even or zero in the case of female
      * Odd in the case of male
  + E-mail policy
    - E-mail needs to contain @.
    - E-mail needs to contain valid syntax.
    - E-mail needs to contain valid top-level domain.
    - E-mail needs to contain recipient name.
  + Mobile number needs to have 9 numbers.
* If the chosen role is the Medical Doctor then a list of specialties is shown up.
* If any of the following data doesn’t meet the data policy there’s a pop-up window with information of putting the data wrongly.
* Button “OK” closes the window and allows an administrator to correct the data.
* After clicking the “Next” button system opens a new form which allows you to optionally add a new user to the employee.
* You can skip this step by pressing the “Skip” button and the system adds new employee to the system.
* An administrator needs to fill following data:
  + Login
  + Permissions
* System shows a pop-out window message: “Employee added”.
* System adds the new employee to the database and sets their status to active.
* System sends an e-mail to the user with information of login data.
* System shows an information of successful adding an employee to the database.
* Button “OK” closes information window and adding new employee form.

View of employee’s details

**As an** Administrator

**I want** to have the possibility to view employee’s details

**So that** I can see significant information as personal data.

Acceptance criteria:

* Button “Review employee” is available and pressing it opens a new form that allows you to view and change data of an employee and confirm it.
  + Button is available on the view list of employees.
* Button “Cancel” is available.
* Once the "Cancel" button is pressed the view of employee’s details form is closed.
* System allows you to see all employee’s data.
* System presents following buttons after opening the form:
  + Button “Edit employee” – after pressing, the system launches edit mode of the employee data.

Editing of employee’s data

**As an** Administrator

**I want** to have the possibility to change the data of an existing employee and confirm it

**So that** I will be able to update their date in the system

Acceptance criteria:

* Button “Edit employee” is available and pressing it opens a new form that allows to update data of an employee and confirm it.
  + The button is available on the view list of employees and view employee’s details.
* Button “Confirm” is disabled as long as all fields aren’t filled.
* Button “Cancel” is available.
* Once the "Cancel" button is pressed the editing form is closed.
* Once input data is incorrect, the system generates and views the message with information about incorrect input data.
* Validation rules are the same as for the “Add new employee” user story
* If any of the following data doesn’t meet the data policy there’s a pop-up window with information of putting the data wrongly.
* Button “OK” closes the window and allows an Administrator to correct the data.
* System shows a message window with an information of “Employee’s data changed.”
* Button “OK” closes message window.

Employee’s deactivation

**As an** Administrator

**I want** to have the possibility to change employee’s status to Inactive f

**So that** employee can no longer work

Acceptance criteria:

* Button “Deactivate employee” is available and pressing it opens a new form that allows to deactivate employee and confirm it.
  + The button is available in the list of employees.
  + Form can also be launched by changing employee’s status from active to inactive in updating employee’s data form.
* Button “Confirm” is unavailable as long as fields aren’t filled.
* Button “Cancel” is available.
* Once the "Cancel" button is pressed the deactivation form is closed.
* Field to fill is:
  + Administrator’s password
* If password is wrong, system shows pop-up window.
  + Button “OK” closes pop-up window.
* If the employee is an active user of the system, the user account is automatically deactivated.

Employee reactivation

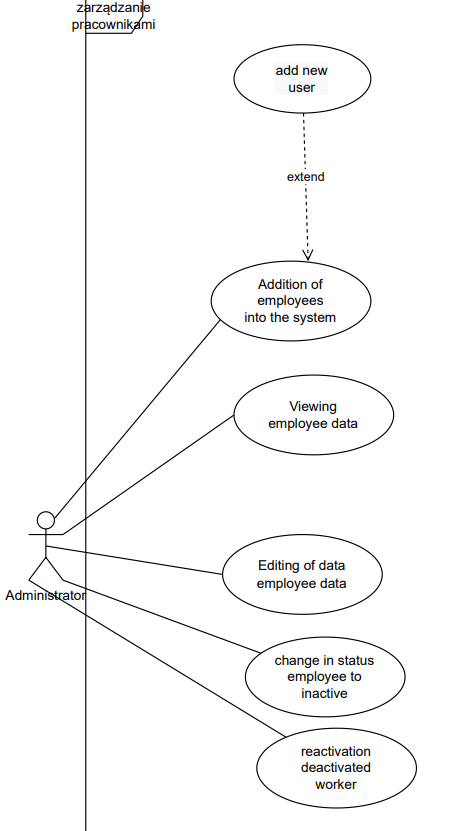
**As an** Administrator

**I want** to have the possibility to reactivate deactivated early employee

**So that** the employee will be able to back to work

Acceptance criteria:

* There is the possibility to find employees with the “Deactivated” status.
* On the list of employees with the “Disactivated” status the “Reactivate” button is available
* Clicking on the “Reactivated” button changes status of the employee into “Active”.
* The operation needs to be confirmed by the Administrator.
* Once an inactive employee had access to the system (was associated with the system user), the user is automatically reactivated, too.



|  |  |
| --- | --- |
| Name | Addition of employees into the system |
| ID | Emp\_manage\_add\_2.1 |
| The aim | Adding an employee to the system |
| Description | Adding a new employee |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | Displayed list of employees |
| Finish conditions | New employee added |
| Input data | First name Last name PESEL  Date of birth |
| Output data | N/A |
| Scenarios | 1. The administrator goes to the "Manage Employees" function. 2. Click on the "Add new employee" button. 3. The system displays the form window 4. The administrator fills in the form with the mandatory data of the employee 5. can fill in the optional data 6. Indicates the employee's role 7. clicks the "Next" button. 8. A new form is displayed to add a new user 9. Click the "Skip" button. 10. the system displays a window with the confirmation of the administrator login and password 11. The administrator types in the correct login and password 12. Click the "OK" button. 13. the system adds the employee to the database and sets their status to active 14. System sends an e-mail to the user with information of login data. 15. the system displays a window with a message about the successful addition of the employee to the database 16. click the "OK" button   Alternative: chosen role: doctor  6. the system displays a form with a list of specialties  7. select the specialty  8. click "OK"  Alternative: The data entered do not comply with the data rules.  8.The system displays a window with a message about incorrect data entry  9. clicks the "Ok" button and corrects the data |
| Notice |  |

|  |  |
| --- | --- |
| Name | Viewing of employee data |
| ID | Emp\_manage\_view\_2.2 |
| The aim | Viewing of employee data |
| Description | The system allows you to view the data of an existing employee |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | displayed list of employees |
| Finish conditions | View all employee data |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. clicks on the "Review employee" button next to the employee on the list 2. the system displays a form with all employee data |
| Notice |  |

|  |  |
| --- | --- |
| Name | Editing of data employee data |
| ID | Emp\_manage\_edit\_2.3 |
| The aim | Edit employee data |
| Description | The system allows editing the data of an existing employee |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | Logged Administrator and Existing employee |
| Finish conditions | Edit employee data |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The user starts the program 2. The user logs into the administrator account 3. The administrator goes to the " Employees management" function 4. He clicks on the button "Review employee" next to the employee on the list 5. Click on the "Edit employee" button 6. The administrator edits the data 7. clicks the "Confirm" button on the form 8. the system displays a window with the message "Employee data changed" 9. clicks the "OK" button on the message window   Alternative: The data is incorrect.  1.The system displays a window with information about incorrect data  2. clicks the "Ok" button and corrects the data |
| Notice |  |

|  |  |
| --- | --- |
| Name | Change in status employee to inactive |
| ID | Emp\_manage\_inactive\_2.4 |
| The aim | deactivating a worker |
| Description | The system enables the employee's status to be changed to inactive |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | employee list view and existing employee |
| Finish conditions | Deactivating a worker |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Click on the button "Deactivate employee" 2. system displays the form 3. administrator sets the employee's status to deactivated 4. administrator enters his password 5. clicks the "Confirm" button 6. system deactivates the employee and their user account   Alternative: wrong administrator password.   1. system displays a window with a message about incorrect data 2. clicks the 'OK' button and corrects the data |
| Notice |  |

|  |  |
| --- | --- |
| Name | Reactivation deactivated worker |
| ID | Emp\_manage\_reactive\_2.5 |
| The aim | Change of employee status to active |
| Description | The system makes it possible to change the employee's status to active |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | employee list view and deactivated employee |
| Finish conditions | Change of employee status to active |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. administrator searches the list for a user with the status "Deactivated" 2. clicks the "Reactivate" button 3. the system displays a window with a field for the administrator's password 4. The administrator enters his password 5. clicks the "confirm" button 6. the system changes the employee status to active and reactivates the user account |
| Notice |  |

Offices management

* 1. **Epic: Offices in the clinic management**

**User stories**

View of offices list

**As an** Administrator

**I want** to have the possibility to review the offices list

**So that** I know which offices are available.

Acceptance criteria:

* Button "Offices" is available on the main form.
* When the button "Offices" is pressed, the system displays a form with a list of rooms.
* Once no one office is registered, the empty list is presented.

Adding new office

**As an** Administrator

**I want** to have the possibility to enter data of a new office and confirm it

**So that** I will be able to assign room to doctor.

Acceptance criteria:

* There is the possibility to add new office from the list of offices.
* Each office must have a number, specialization – chosen from the list, and information that can be used for general meetings or specialized only.

Edit office’s data

**As an** Administrator

**I want** to have possibility to edit data of an office and confirm it

**So that** the office’s data is up-to-date

Acceptance criteria:

* The office can be:
  + Deactivated – being active
  + Activated – being inactive
  + Renumerated
  + Changed into another specialization
  + Checked/unchecked as a general (dedicated for general meetings)

Deleting office

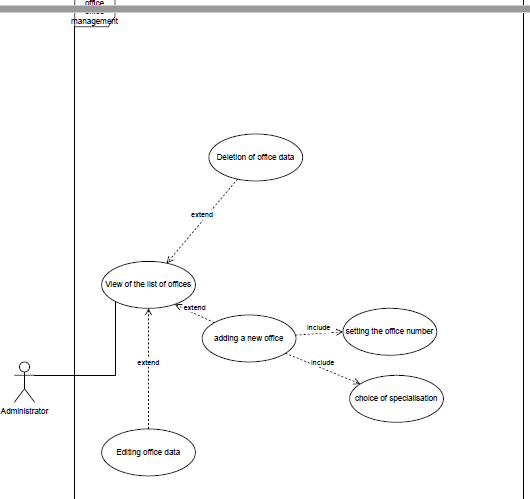
**As an** Administrator

**I want** to have possibility to delete data of an office and confirm it

**So that** I will be able to delete them from the list of offices

Acceptance criteria:

* On the list of offices, there is a button "Delete", besides the rooms name.
* When the “Delete” button is pressed, the system displays a message in the form of a pop-up window with two buttons “Delete” and “Cancel”.
  + When the “Delete” button is pressed, the system deletes room from the database.
  + The only office not planned for a future visit may be deleted.
  + When the “Cancel” button is pressed, the system closes the pop-up window.



|  |  |
| --- | --- |
| Name | Office management |
| ID | Office\_view\_3.1 |
| The aim | View of the list of offices |
| Description | The system offers the possibility to browse the list of offices |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | Logged Administrator |
| Finish conditions | View of the list of offices |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The administrator clicks the "Offices" button 2. The system displays a list of offices |
| Notice |  |

|  |  |
| --- | --- |
| Name | Office management |
| ID | Office\_add\_3.2 |
| The aim | Adding a new office |
| Description | The system offers the possibility of assigning a room to a doctor from a list |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | list of offices |
| Finish conditions | Adding a new office |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. administrator selects an office from the list of offices 2. the administrator sets the number 3. selects a specialty from the list 4. clicks the "OK" button |
| Notice |  |

|  |  |
| --- | --- |
| Name | Office management |
| ID | Office\_edit\_3.3 |
| The aim | Editing office data |
| Description | The system offers the possibility to edit the office data |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | list of offices |
| Finish conditions | Edit office data |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The administrator selects an office from the list of offices 2. clicks the "Edit" button 3. the system displays the form 4. selects a status from the list 5. Click "OK". |
| Notice |  |

|  |  |
| --- | --- |
| Name | Office management |
| ID | Office\_remov\_ 3.4 |
| The aim | Removal of the office |
| Description | The system offers the possibility to delete the office |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | list of offices |
| Finish conditions | Removal of the office |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The system displays a list of offices 2. The administrator clicks the "Delete" button next to the selected office 3. the system displays a window with the message "Do you want to delete?" 4. Clicks the "Delete" button |
| Notice |  |

* 1. **Epic: calendar generation**
  2. **User stories**
  3. Generation of a new calendar based on a blank form
  4. **As an** Administrator
  5. **I want** to generate new calendar based on bank form
  6. **So that** I will be able to fill in all data of the calendar manually
  7. Acceptance criteria:
* After generation, the form of the calendar is empty (no term if filled in)
* The calendar is filled in according to the following flow and conditions:
  + An administrator chooses the office
  + An administrator schedule available terms for the office:
    - If the selected office is specialized and is not dedicated to general meetings, the administrator can only schedule available terms for doctors with the same specialization as the office
    - If the selected office is specialized and is dedicated to general meetings, the administrator can schedule available terms for doctors with the same specialization and for doctors with the “Family medicine”.
    - If the selected office is not specialized, which means that is dedicated to any kind of doctor’s specialization, the administrator can schedule terms for all doctors
  + Only one meeting in the same office can be scheduled at a time.
  + Meetings for many doctors can be scheduled on the same day in the same office.
  1. Generation of a new calendar based on the previous month
  2. **As an** Administrator
  3. **I want** to have the possibility to generate a new calendar based on the previous month
  4. **So that** the new calendar will be filled in the same terms as the template one and the process of calendar generation will be facilitated.
  5. Acceptance criteria:
* As a template month, any earlier month can be selected
* The system automatically fills in the calendar
* An Administrator can make changes to the generated calendar.
  1. Update a calendar
  2. **As an** Administrator
  3. **I want** to have a possibility to change the calendar
  4. **So that** the calendar can be updated
  5. Acceptance criteria:
* Updated can be only not activated yet calendar
* The operation can be performed from the list of calendars or details of the selected calendar.
  1. Calendar removing
  2. **As an** Administrator
  3. **I want** to have the possibility to remove existing calendar
  4. **So that** the calendar will not be able to use for schedule appointments
  5. Acceptance criteria:
* Removed can be an only inactive calendar.
* The operation can be performed from the list of calendars or details of the selected one.
  1. Calendar sending for approval
  2. **As an** Administrator
  3. **I want** to have the possibility to send a calendar for approval to the manager
  4. **So that** the manager will be able to activate or reject it
  5. Acceptance criteria:
* For the approval can be sent only calendar accepted by all doctors
* The system allows sending for approval more than one calendar for the same month
  1. Calendar approval by the manager
  2. **As a** manager
  3. **I want** to have the possibility to approve the selected calendar
  4. **So that** the calendar can be activated
  5. Acceptance criteria:
* The calendar can be approved from the list of calendars available for a manager or details of the selected calendar
* A manager can approve only one calendar for the same month (means as a number on the name of the month and the year)
  1. Rejection of single calendar by the manager
  2. **As a** manager
  3. **I want** to have the possibility to reject the selected calendar
  4. **So that** the calendar can be updated or removed by the administrator
  5. Acceptance criteria:
* The calendar can be rejected from the list of calendars or details of the selected one
  1. Multiple rejections of selected calendars
  2. **As a** manager
  3. **I want** to have the possibility to select and reject more than the calendar at the same time
  4. **So that** I will be able to operate in mass form
  5. Acceptance criteria:
* A manager can select all calendars by clicking on the “global” checkbox
* A manager can select each of calendar manually
* A manager can unselect any of position on the list
* Calendars can be rejected from the list when, at least, one calendar is selected
  1. View list of calendars by the manager
  2. **As a** manager
  3. **I want** to have the possibility to view the list of calendars to approval
  4. **So that** I will be able to choose the better one
  5. Acceptance criteria:
* The list of calendars can be filtered and sorted

Filter list of calendars by the manager

**As a** manager

**I want** to filter list of calendars

**So that** I can see only the chosen data on the list.

Acceptance criteria:

* The system allows to filter calendars on:
  + Calendar name
  + Status

Sort list of calendars by the manager

**As a** manager

**I want** to have the possibility to organize data on the list

**So that** the order of presented data will be changing.

Acceptance criteria:

* The system allows sorting the list from:
  + Alphabetically ascending
  + Alphabetically descending
  1. View details of the selected calendar by the manager
  2. **As a** manager
  3. **I want** to have the possibility to view details of the selected calendar
  4. **So that** I will be able to recognize calendar details and approved the best one
  5. Acceptance criteria:
* The system presents details of the calendar after clicking on the selected on the list of calendars
  1. View list of calendars by the administrator
  2. **As an** Administrator
  3. **I want** to have the possibility to view the list of calendars
  4. **So that** I will be able to select the proper one
  5. Acceptance criteria:
* The list of calendars can be filtered and sorted

Filter list of calendars by the administrator

**As an** Administrator

**I want** to filter list of calendars

**So that** I can see only the chosen data on the list.

Acceptance criteria:

* The system allows to filter calendars on:
  + Calendar name
  + Status

Sort list of calendars by the administrator

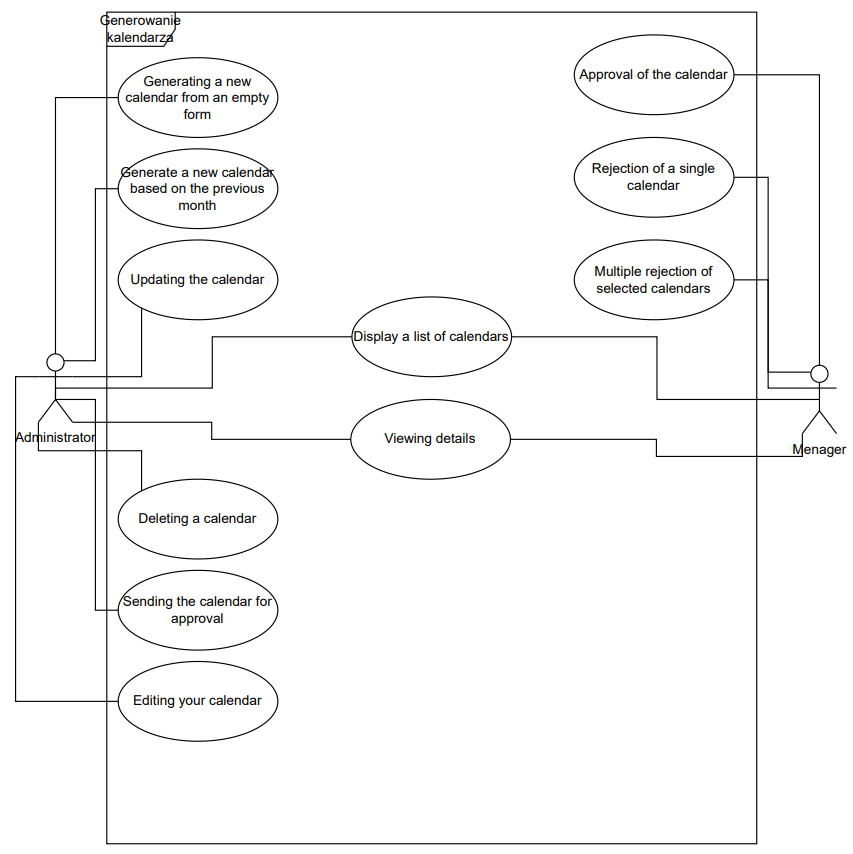
**As an** Administrator

**I want** to have the possibility to organize data on the list

**So that** the order of presented data will be changing.

Acceptance criteria:

* The system allows sorting the list from:
  + Alphabetically ascending
  + Alphabetically descending
  1. View details of selected calendar by the administrator
  2. **As an** Administrator
  3. **I want** to have the possibility to view details of the selected calendar
  4. **So that** I will be able to check the details and make decision about editing
  5. Acceptance criteria:
* The system presents details of the calendar after clicking on the selected on the list of calendars
* There is the possibility to edit the calendar
  1. Edit calendar by the administrator
  2. **As an** Administrator
  3. **I want** to have the possibility to edit selected calendar
  4. **So that** the calendar data will be up-to-date
  5. Acceptance criteria:

The operation can be performed from the list of calendars or the calendar’s details  


|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_gen\_based\_blank\_form |
| The aim | Generate a new calendar from an empty form |
| Description | The system allows the generation of a new calendar based on a blank form |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | N/A |
| Finish conditions | Generate a new calendar |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The administrator clicks the "Generate Calendar" button  2. the system displays an empty calendar and a form  3. the administrator selects an office from the list in the form  4. administrator selects available dates |
| Notice |  |
| Name | Calendar generation |
| ID | Cal\_gen\_prev\_month |
| The aim | Generate a new calendar based on the previous month |
| Description | The system allows the generation of a new calendar based on the previous month |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | N/A |
| Finish conditions | Generate a new calendar |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. the administrator selects any previous months' template 2. the system automatically fills in the calendar 3. the administrator can make corrections |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_update |
| The aim | Updating the calendar |
| Description | the system allows the calendar to be updated |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | calendar not yet activated |
| Finish conditions | updating of the calendar |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. the administrator starts the calendar list  2. start the operation  3. the system displays a window with possible changes  4. perform the changes  Alternative 1:  1. the administrator clicks on the selected calendar  2. Goes to details |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_del |
| The aim | Deleting calendar |
| Description | the system allows the removing calendar so that it cannot be used to plan visits |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | not active calendar |
| Finish conditions | calendar removing |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. the administrator selects a calendar from the list of inactive calendars  2. click the "delete" buton  Alternative 1:  1. the administrator clicks on the selected calendar  2. Goes to details |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_send\_approv |
| The aim | Sending the calendar to approval |
| Description | the system allows the calendar to be sent for approval |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | calendar accepted by all doctors |
| Finish conditions | sending the calendar for approval |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. the administrator selects a calendar from the list   2. clicks the "Send for approval" button  3. the system displays the message "Calendar has been sent for approval" |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_approv |
| The aim | Approval of the calendar |
| Description | the system allows the calendar to be approved |
| Author | Krystian Śmiertka |
| Actor | Manager |
| Start conditions | calendar accepted by all doctors |
| Finish conditions | sending the calendar for approval |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Manager enters the list of calendars  2. clicks on the selected calendar  3. clicks the "Approve" button  4. the system displays the message "Calendar has been approved"  5. clicks the "OK" button |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_singl\_reje |
| The aim | Rejection of single calendar |
| Description | the system shall allow the rejection of a single calendar |
| Author | Krystian Śmiertka |
| Actor | Manager |
| Start conditions | calendar for approval |
| Finish conditions | rejection of the calendar |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Manager enters the list of calendars  2. clicks on the selected calendar  3. clicks the "Reject" button  4. the system displays the message "Calendar has been rejected"  5. clicks the "OK" button |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_multi\_reje |
| The aim | Multiple rejections of selected calendars |
| Description | the system allows multiple calendars to be selected and rejected |
| Author | Krystian Śmiertka |
| Actor | Manager |
| Start conditions | calendar for approval |
| Finish conditions | rejection of the calendar |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Manager enters the list of calendars  2. ticks the box “global”  3. clicks the "Reject" button  4. the system displays the message "Calendars have been rejected "  5. clicks the "OK" button  Alternative 1: 2   1. the manager marks each calendar manually |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_viev\_list\_man |
| The aim | Display a list of calendar |
| Description | the system allows the display of a list of calendars for approval |
| Author | Krystian Śmiertka |
| Actor | Manager |
| Start conditions | calendar for approval |
| Finish conditions | list of calendars |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Manager enters the list of calendars |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_viev\_list\_adm |
| The aim | Display a list of calendar |
| Description | the system allows the display of a list of calendars for approval |
| Author | Krystian Śmiertka |
| Actor | Administrator |
| Start conditions | existing calendar |
| Finish conditions | list of calendars |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. Administrator enters the list of calendars |
| Notice |  |

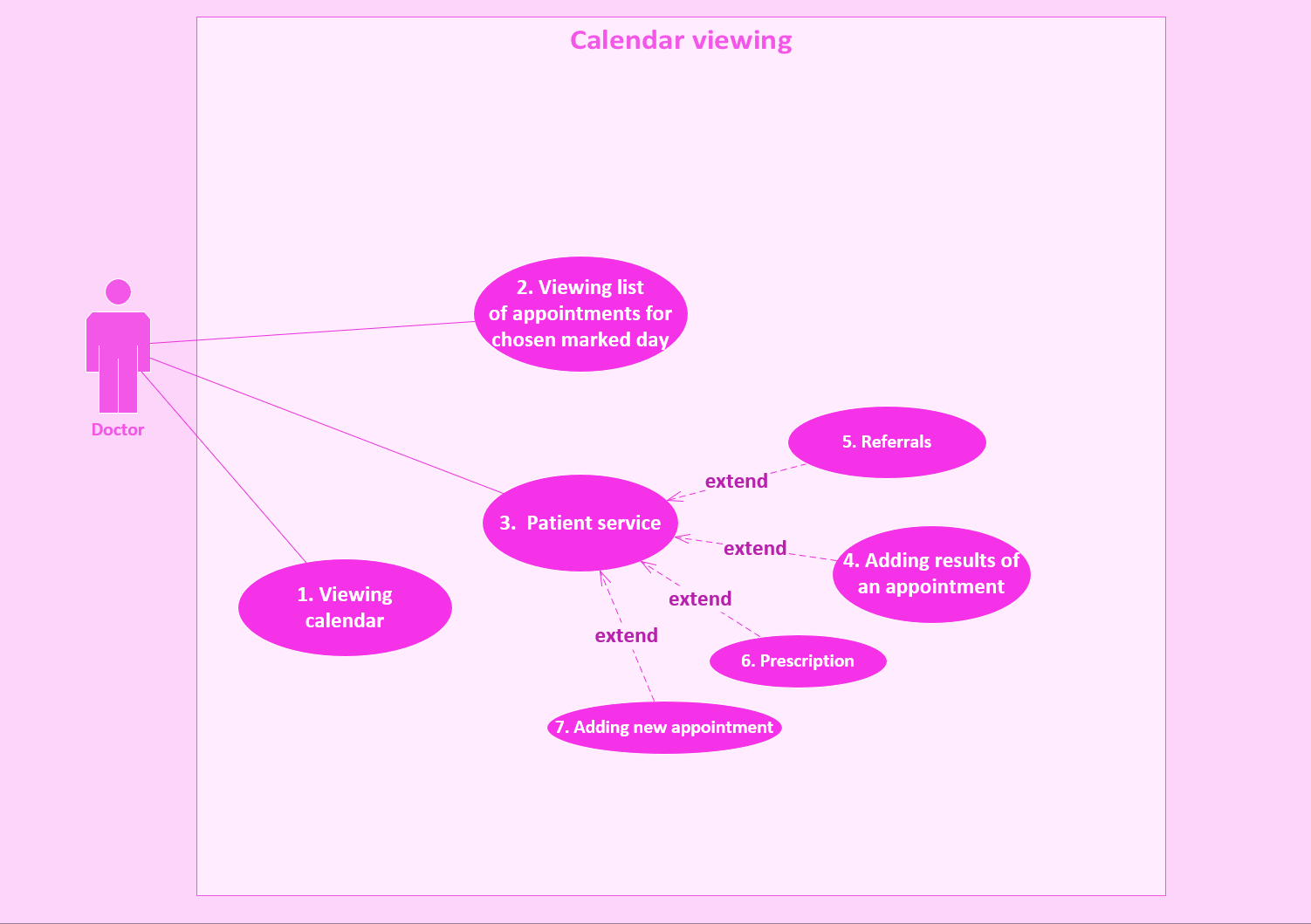
|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_viev\_details\_man |
| The aim | Vieving details |
| Description | the system allows you to view the details of the selected calendar |
| Author | Krystian Śmiertka |
| Actor | Manager |
| Start conditions | existing calendar |
| Finish conditions | calendar details view |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. the manager enters the list of calendars to approve  2. select a calendar by clicking  3. the system displays a window with the details of the calendar |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_viev\_details\_adm |
| The aim | Vieving details |
| Description | the system allows you to view the details of the selected calendar |
| Author | Krystian Śmiertka |
| Actor | Administator |
| Start conditions | existing calendar |
| Finish conditions | calendar details view |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The administrator enters the list of calendars to approve  2. select a calendar by clicking  3. the system displays a window with the details of the calendar |
| Notice |  |

|  |  |
| --- | --- |
| Name | Calendar generation |
| ID | Cal\_edit |
| The aim | Editing your calendar |
| Description | the system allows the calendar to be edited |
| Author | Krystian Śmiertka |
| Actor | Administator |
| Start conditions | existing calendar |
| Finish conditions | calendar changes |
| Input data | N/A |
| Output data | N/A |
| Scenarios | 1. The administrator enters the list of calendars  2. select a calendar by clicking  3. the system displays a window with the details of the calendar  4. the administrator clicks the "Edit" button  5. makes changes  6. clicks the "Save" button  7. the system displays the message "The calendar has been saved" |
| Notice |  |

3 Doctor’s portal handling

3.1 Calendars management

* 1. 
  2. **User stories**

**3.1.1 Viewing the calendar**

|  |  |
| --- | --- |
| Name | Viewing calendar |
| ID | Cal\_1.1 |
| The aim | Viewing calendar |
| Description | The doctor can view a calendar of appointments by clicking a button. |
| Author | Ewelina Danielska |
| Actor | Doctor |
| Start conditions | The user logged in as a doctor. |
| Finish conditions | The user sees the calendar. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User open page: DoctorPage. 2. User clicking the “Calendar” button. 3. System opens a calendar for the entire month with the days of visits marked. |
| Notices |  |

|  |  |
| --- | --- |
| Name | View calendar of appointments. |
| ID | Cal\_2.1 |
| The aim | Viewing a list of appointments for the chosen marked day. |
| Description | User Doctor can view the calendar of list appointments by clicking marked. |
| Author | Ewelina Danielska |
| Actor | Doctor |
| Start conditions | The user is on the displayed calendar. |
| Finish conditions | The user sees the list of appointments. |
| Input data | N/A |
| Output data | N/A |
| Scenarios | Main:   1. User clicks marked day. 2. System opens a list of appointments for the entire day   Alternative:   1. User clicks an unmarked day. 2. System does nothing. |
| Notices |  |

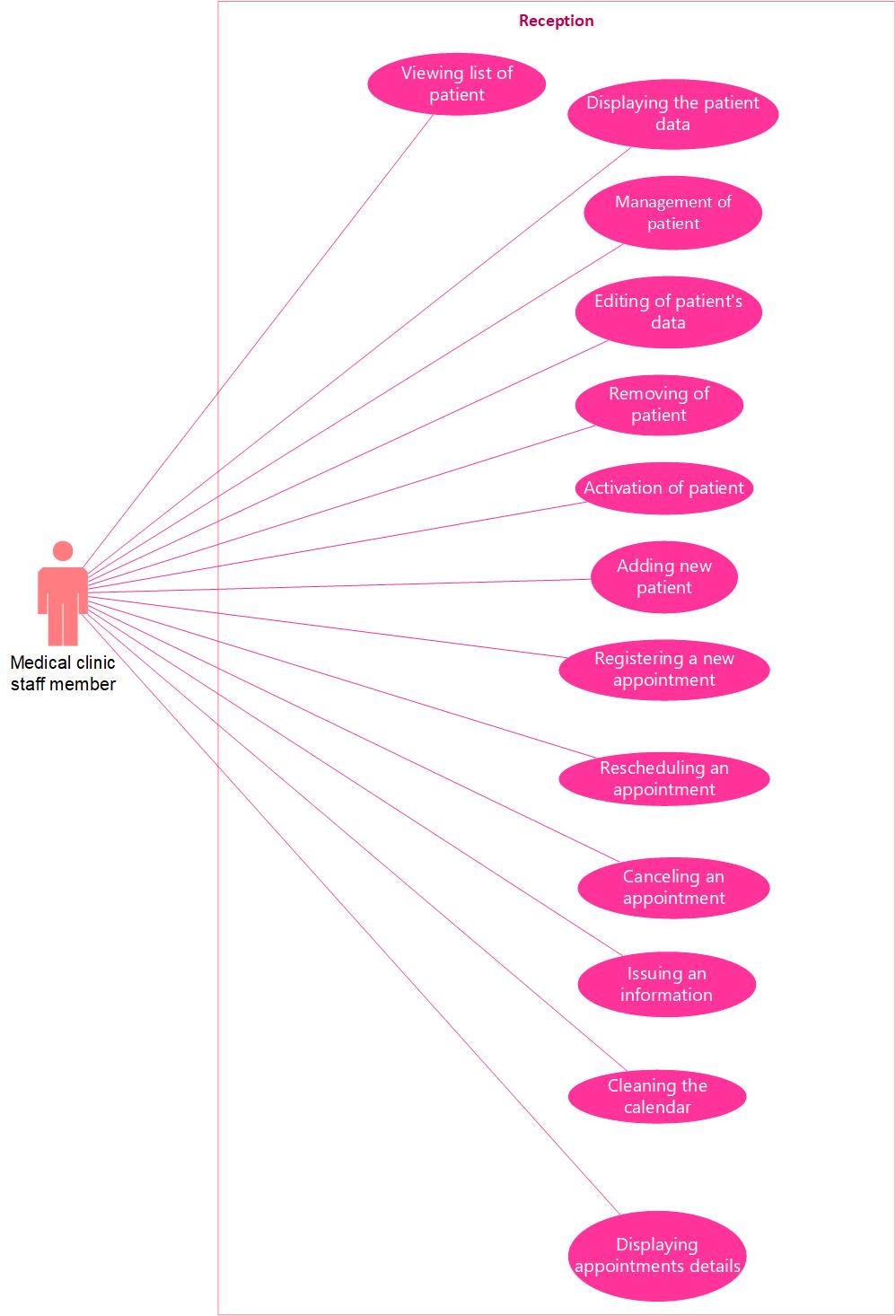
### **3.1.2 Viewing list of appointments for chosen marked day.**

3.2 Medical visit managemen

### **3.2.1 Patient service.**

|  |  |
| --- | --- |
| Name | Patient service |
| ID | Cal\_3.1 |
| The aim | Displaying patient service and adding results and recommendations. |
| Description | The Doctor can add results of visits, prescribed medications, referrals for treatment, and new appointments. |
| Author | Ewelina Danielska |
| Actor | Doctor |
| Start conditions | The user sees a list of appointments. |
| Finish conditions | Visit details have been added. |
| Input data | Results of appointments. Prescribed medications (prescription).  Referral.  New appointment (date, time, patient). |
| Output data | N/A |
| Scenarios | Main:   1. User double–click chosen appointment. 2. The system displays the visit management panel. 3. The Doctor adds the results of the appointment by writing information in the window of the description visit. 4. The Doctor accepts changes by clicking “Add” or canceling by “Cancel”.   Alternative:   1. The Doctor adds a referral by writing the data in the window of “Referral”. 2. The Doctor accepts changes by clicking “Add” or canceling by “Cancel”.   Alternative:   1. The Doctor adds a new appointment by writing data in the window of “New appointment”. 2. The Doctor accepts changes by clicking “Add” or canceling by “Cancel”. |
| Notices |  |

4 Clinic staff's module



|  |  |
| --- | --- |
| Name | View of patient list |
| ID | PAT\_LIST |
| The aim | Have the possibility to display a patient list. |
| Description | Logged user has the possibility to display list of patient |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing list of patients |
| Finish conditions | The employee has the opportunity to view the patient list |
| Input data | N/A |
| Output data | Displayed correct list of available patients |
| Scenarios | Main:  1.The “List of patients” button is available in the main menu.  2.The user clicks the "List of patients" button.  3.the system displays a form with a list of patients. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Filter list of patients |
| ID | PAT\_FILTER |
| The aim | Filtering the patient list |
| Description | Check the correct filtering of the patient list. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing list of patients |
| Finish conditions | Filtered list of patients |
| Input data | Selected patient data |
| Output data | Filtered patient based on the given criteria |
| Scenarios | Main:  1. The user clicks the "List of patients" button.  2. The User enters data in filter field, that he want list of patient to be filtered by.  3.The user clicks “filter” button.  4. The system displays the patient with the given argument.  .  Alternative:  1. The user clicks the "List of patients" button.  2. The User didn’t enter or entered incorrect data to the filter field.  3.The user clicks “filter” button.  4.The system displays an empty list. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Sort list of patients |
| ID | PAT\_SORT |
| The aim | Sorting the patient list |
| Description | Check the correct sorting of the patient list. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing list of patients |
| Finish conditions | Displayed list according to the selected sort option. |
| Input data | N/A |
| Output data | Sorted patient list based on selected sort option. |
| Scenarios | Main:  1. The user clicks the "List of patients" button.  2. The user clicks on “Name”.  3. The patient list is sorted alphabetically ascending or descending. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Adding a new patient |
| ID | PAT\_ADD |
| The aim | Have possibility to enter data of a new patient and confirm it |
| Description | The system allows you to enter the data of a new patient and give him access to the clinic. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Patient is unregistered in database |
| Finish conditions | The new patient is added to the database |
| Input data | Patient name, surname, Date of birth, PESEL, Email, Phone number, Correspondence address |
| Output data | Added data of patient |
| Scenarios | Main:  1.Staff member clicks the “add patient” button  2.System opens the form of new patient addition.  3.Staff member enters the new patient data  4.Entering correct data in form fields  5.Clicking confirm button  6.System will add the patient to the database.  Alternative:  1.Staff member clicks the “add patient” button  2.System opens the form of new patient addition.  3.Staff member enters the new patient data  4.Entering incorrect data or leaving blank fields  5. System will display an error message: “incorrect data” or “you left empty field” |
| Notices |  |

|  |  |
| --- | --- |
| Name | Edit of patient’s data |
| ID | PAT\_EDIT |
| The aim | Have possibility to edit data of a patient and confirm it |
| Description | Check the possibility to edit data of a patient. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing patient in database |
| Finish conditions | Patient data are updated and displayed on the list |
| Input data | Patient name, surname, PESEL, Date of birth, Email, Phone number, Correspondence address |
| Output data | New values of data |
| Scenarios | Main:  1. The user clicks the "List of patients" button.  2. The user chooses patient.  3. The user clicks the "Edit a patient" button.  4. System opens the form.  5. The user edits patient data.  6. The user clicks the "Save" button.  7. Patient data were edited.  1. The user clicks the "List of patients" button.  2. The user chooses patient.  3. The user clicks the "Edit a patient" button.  4. System opens the form.  5. The user edits patient data.  6. The user data were edited incorrectly or user left blank fields  7. System will display an error message: “incorrect data” or “you left empty field”. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Activation of patient |
| ID | PAT\_ACTIVE |
| The aim | Have possibility to activate existing patient in database. |
| Description | Check the possibility to activate a patient in the database. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Unactive patient in database |
| Finish conditions | Patient is activated |
| Input data | Patient from the database |
| Output data | Activated patient |
| Scenarios | Main:  1.Staff member clicks the “activate patient” button  2.Staff member choose the patient to activate  3.System display pop-up window  4.Staff member clicks active button  5.Patient is being activated in the database  Alternative:  1.Staff member clicks the “active patient” button  2.Staff member choose the patient to active  3.System display pop-up window  4.Staff member clicks cancel button  5.Patient status wasn’t changed |

|  |  |
| --- | --- |
| Name | Deactivation of patient |
| ID | PAT\_DEACTIVE |
| The aim | Have possibility to deactivate existing patient in database. |
| Description | Check the possibility to deactivate a patient in the database. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | active patient in database |
| Finish conditions | Patient is deactivated |
| Input data | Patient from the database |
| Output data | deactivated patient |
| Scenarios | Main:  1.Staff member clicks the “deactivate patient” button  2.Staff member choose the patient to deactivate  3.System display pop-up window  4.Staff member clicks deactivate button  5.Patient is being deactivated in the database  Alternative:  1.Staff member clicks the “deactivate patient” button  2.Staff member choose the patient to deactivate  3.System display pop-up window  4.Staff member clicks cancel button  5.Patient status wasn’t changed |

|  |  |
| --- | --- |
| Name | Remove the patient |
| ID | PAT\_REMOVE |
| The aim | Have possibility to remove patient chosen from the list |
| Description | Check the possibility to remove a patient and confirm it |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing patient in database |
| Finish conditions | Patient is deleted from the list |
| Input data | N/A |
| Output data | Deleted data of patient |
| Scenarios | Main:  1.Staff member clicks the “delete patient” button  2.Staff member choose the patient to delete  3.System display pop-up window  4.Staff member clicks delete button  5.Patient is being deleted from the database  Alternative:  1.Staff member clicks the “delete patient” button  2.Staff member choose the patient to delete  3.System display pop-up window  4.Staff member clicks cancel button  5.Patient is not deleted from the database |
| Notices |  |
| Notices |  |

|  |  |
| --- | --- |
| Name | Registering a new appointment |
| ID | APP\_REGISTER |
| The aim | Have possibility to register a new appointment |
| Description | Check the possibility to register a new appointment |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing patient in database |
| Finish conditions | New appointment is registered |
| Input data | Patient data, doctor data and date of appointment |
| Output data | New appointment is added |
| Scenarios | Main:  1. The user clicks the “Add an appointment” button.  2. The user chooses patient and doctor from drop-down list, specialization from drop-down list, date of appointment from calendar, time of the visit and number of office from drop-down list to create an appointment.  3. All fields have been filled.  4. The user clicks “Add” button.  5.New appointment is added to the database and calendar  Alternative:  1. The user clicks the “Add an appointment” button.  2. The user choose(patient and doctor from drop-down list and date of appointment from calendar) to create an appointment  3.All fields are not filled  4.Staff member can’t click “add and appointment” button  5.New appointment is not added to the database and calendar |
| Notices |  |

|  |  |
| --- | --- |
| Name | Rescheduling an appointment |
| ID | APP\_RESCHEDULE |
| The aim | Have possibility to reschedule an appointment |
| Description | Check the possibility to reschedule an appointment |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing appointment in database and calendar |
| Finish conditions | Appointment is rescheduled |
| Input data | New date of appointment |
| Output data | Rescheduled appointment |
| Scenarios | Main:  1.Staff member clicks the “reschedule an appointment” button  2.Staff member choose the appointment to reschedule  3.System display forms of appointment data  4.Staff member changes the date of appointment in calendar  5.Staff member clicks the “confirm reschedule” button  6.Date of appointment have been changed |
| Notices |  |

|  |  |
| --- | --- |
| Name | Canceling an appointment |
| ID | APP\_CANCEL |
| The aim | Have possibility to cancel an appointment |
| Description | Check the possibility to cancel an appointment |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing appointment in database and calendar |
| Finish conditions | Appointment is canceled |
| Input data | N/A |
| Output data | Canceled appointment |
| Scenarios | Main:  1.Staff member choose the appointment to cancel  2.System display forms of appointment data  3.Staff member clicks the “cancel an appointment” button  4.Staff member clicks the “confirm cancellation” button  5.appointment is canceled from database and calendar |
| Notices |  |

|  |  |
| --- | --- |
| Name | Issuing an information |
| ID | APP\_ISSUE |
| The aim | Have possibility to issue the confirmation |
| Description | Check the possibility to issue the confirmation |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing appointment in database and calendar |
| Finish conditions | Confirmation has been generated |
| Input data | N/A |
| Output data | Generated PDF confirmation |
| Scenarios | Main:  1.Staff member clicks the “issue a confirmation” button.  2.System displays forms calendar with scheduled visits  3.Staff member choose date of visit to issue a confirmation  4.Staff member clicks the “generate pdf” button  5.Pdf file with appointment confirmation and payment details has been generated |
| Notices |  |

|  |  |
| --- | --- |
| Name | Cleaning the calendar from outdated and past visits |
| ID | APP\_CLEAN |
| The aim | Have possibility to clean calendar from outdated and past visits |
| Description | Check the possibility to clean calendar from outdated and past visits |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing database of appointments |
| Finish conditions | Calendar is being cleared from outdated and past visits |
| Input data | N/A |
| Output data | Calendar is cleaned from outdated and past visits |
| Scenarios | Main:  1.Staff member clicks the “Clean calendar” button.  2.System display forms with two buttons: “Clean from outdated visit”, ”Clean from past visit”  3.Staff member choose one of the two following options.  4.Staff member clicks the “confirm” button  5.Calendar is being cleaned from outdated or past visit. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Displaying appointment details |
| ID | APP\_DISPLAY |
| The aim | Have possibility to display an appointment details |
| Description | Check the possibility to display an appointment details |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing appointment in database and calendar |
| Finish conditions | Displayed details of appointment |
| Input data | N/A |
| Output data | Displayed details of appointment |
| Scenarios | Main:  1.The user choose an appointment to display  2.The user clicks the "Display appointments" button.  3.System display a list of all visits |
| Notices |  |

|  |  |
| --- | --- |
| Name | Displaying the list of appointments |
| ID | APP\_LIST |
| The aim | Check the possibility to display the list of appointments |
| Description | Check the correct handling of the appointments list. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing appointment in database and calendar |
| Finish conditions | Displayed list of appointment |
| Input data | N/A |
| Output data | Displayed list of appointment |
| Scenarios | Main:  1. The user clicks on the “Display appointments” button.  2. The system displays all patient visits. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Filter list of appointments |
| ID | APP\_FILTER |
| The aim | Filtering the appointments list |
| Description | Check the correct filtering of the appointments list. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing list of appointments |
| Finish conditions | Filtered list of appointments |
| Input data | Selected appointment data |
| Output data | Filtered appointment based on the given criteria |
| Scenarios | Main:  1. The User clicks the "Display appointments" button.  2. The User enters data in filter field, that he want list of patient to be filtered by.  3. The user clicks on “Filter” button.  4. The system displays the patient with the given argument.  Alternative:  1. The User clicks the "Display appointments" button.  2. The User enters data in filter field, that he want list of patient to be filtered by.  3. The User didn’t enter or entered incorrect data to the filter field.  3.The user clicks “filter” button.  4.The system displays an empty list. |
| Notices |  |

|  |  |
| --- | --- |
| Name | Sort list of appointments |
| ID | APP\_SORT |
| The aim | Sorting the appointments list |
| Description | Check the correct sorting of the appointments list. |
| Author | Adam Majda |
| Actor | Medical clinic staff member |
| Start conditions | Existing list of appointments |
| Finish conditions | Displayed list according to the selected sort option. |
| Input data | N/A |
| Output data | Sorted appointment list based on selected sort option. |
| Scenarios | Main:  1. The user clicks the "Display appointments" button.  2. The user clicks on selected data  3. The appointment list is sorted alphabetically ascending or descending. |
| Notices |  |