|  |  |  |  |
| --- | --- | --- | --- |
| **FR-01** |  | **The system must have one admin account by default.** | **MUST** |
|  | R-01.1 | An admin cannot make use of the requirements set for the user. (Ex: creating a *Hero*, playing the game) | Must |
|  | Q-02.1 | To log in the user should provide an existing combination of username and password. |  |
| **FR-02** |  | **An admin can create other admin accounts.** | **MUST** |
|  | Q-02.1 | The admin must provide an email and full name. | Must |
|  | Q-02.2 | The system must generate the password. | Should |
| **FR-03** |  | **An admin can manage content for the game (Items/Dungeons/Quests)** | **MUST** |
|  | Q-03.1 | Admin can create content. | Must |
|  | Q-03.2 | Admin can read all content. | Must |
|  | Q-03.3 | Admin can update content. | Must |
|  | Q-03.4 | Admin can delete content. | Must |
|  | Q-03.5 | Admin must provide full information over the content before it can be added or edited. | Must |
| **FR-04** |  | **A user must be able to create an account.** | **Must** |
|  | Q-04.1 | An account must be created using an email, username and password. | Must |
|  | R-04.2 | A username must exist of at least 6 characters. | Should |
|  | R-04.3 | A password must exist of at least 6 characters of which minimal one capital letter, one small letter and one number. | Should |
| **FR-05** |  | **An user must be able to log in.** | **MUST** |
|  | Q-05.1 | To log in the user should provide an existing combination of username and password. | Must |
| **FR-06** |  | **An account must be able to create a *Hero*.** | **Must** |
|  | R-06.1 | Each account can have a maximum of 3 *Heroes*. | Must |
|  | Q-06.2 | Statistics of the created *Hero* can be edited while creating. | Must |
| **FR-07** |  | **A *Hero* must be able to buy items.** | **Must** |
|  | Q-07.1 | The *Hero* must have enough gold in the bank to buy the item. | Must |
|  | R-07.2 | The *Hero* must have at least one empty space in the backpack. | Must |
| **FR-08** |  | **A *Hero* must be able to sell an item.** | **Must** |
|  | R-08.1 | A *Hero* can only sell items that are in the backpack. | Must |
| **FR-09** |  | **A *Hero* must be able to enter a dungeon from a list.** | **Must** |
|  | Q-09.1 | The list must consist of all available dungeons at that moment. | Should |
|  | Q-09.2 | The list must be first sorted on level (difficulty) and then on alphabet. | Should |
| **FR-10** |  | **A *Hero* must be able to combat other creatures in a dungeon.** | **Must** |
|  | Q-10.1 | The *Hero* can earn gold and experience from battling these creatures. | Should |
|  | Q-10.2 | The *Hero* will earn a bonus of gold and experience when the dungeon is completed. | Must |
| **FR-11** |  | **A *Hero*****should be able to accept quests.** | **Should** |
|  | R-11.1 | A *Hero* can only have one active quest at the same time. | Should |
| **FR-12** |  | **A *Hero*****should be able to complete quests.** | **Should** |
|  | Q-12.1 | When a quest is completed the *Hero* is giving gold, experience and/or items as reward. | Should |
| **FR-13** |  | **A *Hero* should be able to cancel an active quest.** | **Should** |
|  | R-13.1 | This is only possible when the quest is not yet completed. | Should |
| **FR-14** |  | **Every user can watch the leaderboards.** | **Should** |
|  | R-14.1 | Only non-blocked users will be appearing on the leaderboards. | Must |
| **FR-15** |  | **An admin can block an existing active account.** | **Must** |
|  | R-15.1 | An admin cannot block another admin. | Must |
|  | Q-15.2 | An admin must provide a length of the block. (Ex: one week or permanent) | Should |
|  | Q-15.3 | An admin must provide a message for the user. | Should |
| **FR-16** |  | **An admin can unblock an existing blocked account.** | **Must** |

FR = Functional Requirement

Q = Quality Requirement

R = Restrictive Requirement

|  |  |
| --- | --- |
| UC01 | Starting with an admin |
| Description | Before starting the application for the first time a admin should already exist. |
| Actors | Developer |
| Preconditions | Application has not launched before. |
| Flow | 1. Application is started. 2. Actor will face normal login view. 3. Actor logs in using the standard admin account. 4. System will check credentials. Q-01.2 5. System grants access to the admin interface. Q-01.1 |
| Exception(s) | 1. Incorrect credentials were entered. Show message and return to step 2. |
| Result | Actor will log into the standard admin account. FR-01 |

|  |  |
| --- | --- |
| UC02 | Creating admin account |
| Description | An logged in admin created another admin account for someone else to use. |
| Actors | Developer |
| Preconditions | One admin account already exists. FR-01 |
| Flow | 1. Application is started. 2. Actor logs into admin account. 3. Actor faces admin interface. 4. Actor navigates to the correct view. 5. Actor fills all necessary data. Q-02.1 6. The system creates a new admin account. Q-2.2 |
| Exception(s) | 1. Not all necessary data was entered. Show message and back to step 5. |
| Result | The system has created a new admin account with a generated password. FR-02 |

|  |  |
| --- | --- |
| UC03 | Creating new content |
| Description | An actor on a admin account creates a new item for the game |
| Actors | Admin |
| Preconditions | One admin account already exists. FR-01 |
| Flow | 1. Application is started. 2. Actor logs into admin account. 3. Actor face admin interface. 4. Actor navigates to the correct view. 5. Actor select what kind of content it wants to add to the game. 6. Actor fills in all necessary data. Q-03.5 7. The system will create the new content. Q-03.1 |
| Exception(s) | 1. Not all necessary data was entered. Show message and return to step 6. |
| Result | The system has created the new content and added to the game. FR-03 |

|  |  |
| --- | --- |
| UC04 | Editing content |
| Description | An actor on a admin account edits existing content. |
| Actors | Admin |
| Preconditions | One admin account already exists. FR-01 |
| Flow | 1. Application is started. 2. Actor logs into admin account. 3. Actor face admin interface. 4. Actor navigates to the correct view. 5. Actor select what kind of content it wants to edit. 6. Actor selects the existing content. Q-03.2 7. Actor fills in all necessary data. Q-03.5 8. The system will edit the existing content. Q-03.3 |
| Exception(s) | 1. Not all necessary data was entered. Show message and return to step 6. |
| Result | The system has edited the content. FR-03 |

|  |  |
| --- | --- |
| UC05 | Deleting content |
| Description | An actor on a admin account deletes existing content. |
| Actors | Admin |
| Preconditions | One admin account already exists. FR-01 |
| Flow | 1. Application is started. 2. Actor logs into admin account. 3. Actor face admin interface. 4. Actor navigates to the correct view. 5. Actor select what kind of content it wants to deleted. 6. Actor selects the existing content. Q-03.2 7. The system will delete the selected content. Q-03.4 |
| Exception(s) | 1. The content is already deleted (by another admin). Show error message and return to step 6. |
| Result | The system has deleted the content. FR-03 |

|  |  |
| --- | --- |
| UC06 | Registering |
| Description | A user creates an account to play with. |
| Actors | User |
| Preconditions |  |
| Flow | 1. Application is started. 2. Actor navigates to the correct view. 3. Actor fills in all necessary data. Q-04.1 4. The system creates an account for the actor. |
| Exception(s) | 1. Not all necessary data was entered. Show message and return to step 3. Q-04.1 2. The entered email is incorrect. Show message and return to step 3. Q-04.1 3. The entered username is too short. Show message and return to step 3. R-04.2 4. The entered password does not fulfill all expectations. Show message and return to step 3. R-04.3 |
| Result | The system has created an account for the actor. FR-04 |

|  |  |
| --- | --- |
| UC07 | Logging in |
| Description | The actor logs in using its account credentials. |
| Actors | User |
| Preconditions | The actor already has an existing account. |
| Flow | 1. Application is started. 2. Actor navigates to the correct view. 3. Actor fills in username and password. 4. The system logs the user in and returns start view. |
| Exception(s) | 1. The username does not exist in the system. Show message and return to step 3. Q-05.1 2. The password does not match the username. Show message and return to step 3. Q-05.1 |
| Result | The actor is logged into its account. FR-05 |

|  |  |
| --- | --- |
| UC08 | Creating a hero |
| Description | Creating a hero to play with |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Actor changes attributes of the hero. Q-06.2 3. Actor confirms changes. 4. System creates a hero for the actor. |
| Exception(s) | 1. Actor already has 3 existing heroes. R-06.1 |
| Result | A hero is creating on the account of the actor. FR-06 |

|  |  |
| --- | --- |
| UC09 | Buying items |
| Description | The hero of the actor buys an item from the shop. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. The actor selects the items it wants to buy. 3. The actor confirms the selection. 4. The system retracts the money from the actor’s hero’s account 5. The system gives the hero its item. |
| Exception(s) | 1. The hero has not enough money to buy the item. Show message and return to step 2. Q-07.1 2. The hero has not enough empty space in its backpack to save the item. Show message and return to step 2. R-07.2 |
| Result | The hero has its money withdrawn and receives the item in its backpack. |

|  |  |
| --- | --- |
| UC10 | Selling items |
| Description | The hero of the actor sells an item to the shop. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. The actor selects the item from its backpack that it wants to sell. 3. The actor confirms the selection. 4. The system retrieves the item from the actor’s backpack. 5. The system deposits the correct amount of gold on the hero’s balance. |
| Exception(s) | 1. The hero has no items in its backpack, and thus cannot select any items. |
| Result | The hero loses the item from the backpack and receives gold for it. |

|  |  |
| --- | --- |
| UC11 | Entering a dungeon |
| Description | The hero enters a dungeon from a list. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. The actor finds a list with all available dungeons, sorted on alphabet. Q-09.1, Q-09.2 3. The actor selects a dungeon. 4. The system will transfer the hero to the selected dungeon. |
| Exception(s) |  |
| Result | The hero is transferred to the selected dungeon. FR-09 |

|  |  |
| --- | --- |
| UC12 | Combat in dungeon |
| Description | The hero combats creeps in a joined dungeon. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Hero attacks enemy creeps. 3. Hero kill enemy creeps. 4. Hero receives gold and experience. Q-10.1 |
| Exception(s) | 1. The hero dies while in the dungeon. Go back to main view and lose some gold. |
| Result | The hero gained gold and experience from defeating enemy creeps. FR-10 |

|  |  |
| --- | --- |
| UC13 | Completing a dungeon |
| Description | The hero combats creeps in a joined dungeon. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Hero attacks enemy creeps. 3. Hero kill enough enemy creeps. 4. Hero completed the dungeon. 5. Hero receives bonus experience and gold. Q-10.2 |
| Exception(s) | 1. The hero dies while in the dungeon. Go back to main view and lose some gold. |
| Result | The hero gained gold and experience from defeating the dungeon. FR-10 |

|  |  |
| --- | --- |
| UC14 | Accepting a quest |
| Description | The hero accepts a quest. |
| Actors | User |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Actor selects a quest from a list. 3. Actor confirms selecting. 4. The system assigns the quest to the hero. |
| Exception(s) | 1. The actor already has a quest assigned. Ask the actor if it wants to cancel current quest. Then return to step 2. R-11.1 |
| Result | The hero now has a quest active. FR-11 |

|  |  |
| --- | --- |
| UC15 | Completing a quest |
| Description | The hero completes a quest. |
| Actors | User |
| Preconditions | The actor is already logged in and as a quest active. |
| Flow | 1. Actor navigates to the correct view. 2. Actor selects ‘complete quest’ button 3. Hero receives bonus experience and gold. Q-12.1 4. The system removes the quest from the hero. |
| Exception(s) | 1. The hero has not completed the quest yet. Show message and return to step 1. |
| Result | The hero received experience and gold for completing a quest. FR-12 |

|  |  |
| --- | --- |
| UC16 | Canceling a quest |
| Description | The hero cancels a quest. |
| Actors | User |
| Preconditions | The actor is already logged in and as a quest active. |
| Flow | 1. Actor navigates to the correct view. 2. Actor selects ‘cancel quest’ button. 3. Actor confirms the choice. 4. The system removes the quest from the hero. |
| Exception(s) | 1. The quest is completed. Show message and go to step 1 of UC16. R-13.1 |
| Result | The quest is removed from the user. FR-13 |

|  |  |
| --- | --- |
| UC17 | Leaderboards |
| Description | Every actor can watch the leaderboards. |
| Actors | Admin, user |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Actor can search for active players or R-14.1 3. Actor can watch top players in the leaderboards. |
| Exception(s) |  |
| Result | Actor can view all active players on the leaderboards. FR-14 |

|  |  |
| --- | --- |
| UC18 | Blocking an account |
| Description | An actor can block an existing account and thus preventing the account from logging in. |
| Actors | Admin |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Actor selects the account from a list of all non-admin accounts. R-15.1 3. Actor clicks a button to block the account. 4. Actor must set a message which states the reason for the block. Q-15.3 5. Actor must select the duration of the block. Q-15.2 6. The system will block the account from logging in. |
| Exception(s) | 1. The user is already blocked. Button should not be clickable. Return to step 2. |
| Result | The selected account is blocked and cannot login. FR-15 |

|  |  |
| --- | --- |
| UC19 | Logging in with blocked account |
| Description | The actor tries to log in with an account that is blocked by an admin. |
| Actors | User |
| Preconditions | The account is blocked. |
| Flow | 1. Actor navigates to the correct view. 2. Actor fills in the correct credentials. 3. The system will show a message which states the reason of the block and the duration of the block. Q-15.2, Q-15.3 4. Return to step 1. |
| Exception(s) |  |
| Result | The actor cannot login and will receive a message. |

|  |  |
| --- | --- |
| UC20 | Unblocking an account |
| Description | An actor can unblock an existing blocked account. |
| Actors | Admin |
| Preconditions | The actor is already logged in. |
| Flow | 1. Actor navigates to the correct view. 2. Actor selects the account from a list of all non-admin accounts. R-15.1 3. Actor clicks a button to unblock the account. 4. The system will unblock the account. |
| Exception(s) | 1. The user is not blocked. Button should not be clickable. Return to step 2. |
| Result | The selected account is unblocked and can login again. FR-16 |

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Use case | Input | Expected output |
| TC01 | UC01 | List<Admins> = *null* | One admin is added automatic before the application is started. |
| TC02 | UC01 | List<Admin>.count = 1 | No admin is added. |
| TC03 | UC02 | FirstName = Ruud  LastName = Deenen  DateOfBirth = 03-07-1998 | Admin is created by system. |
| TC04 | UC02 | FirstName = Ruud  LastName = *null*  DateOfBirth = 03-07-1998 | Admin is not created, incorrect LastName. |
| TC05 | UC02 | FirstName = *null*  LastName = Deenen  DateOfBirth = 03-07-1998 | Admin is not created, incorrect FirstName. |
| TC06 | UC02 | FirstName = Ruud  LastName = Deenen  DateOfBirth = *null* | Admin is not created, incorrect DateOfBirth. |
| TC07 | UC02 | FirstName = Ruud  LastName = Deenen  DateOfBirth = 18-04-2620 | Admin is not created, DateOfBirth is not in the past. |
| TC08 | UC06 | Email = player@mail.com  FirstName = *null*  LastName = *null*  Username = player123  Password = Test@123  ConfirmPassword = Test@123 | User is created. All required fields are filled. |
| TC09 | UC06 | Email = player@mail.com  FirstName = *null*  LastName = *null*  Username = *null*  Password = Test@123  ConfirmPassword = Test@123 | User is not created. Username is missing. |
| TC10 | UC06 | Email = player@mail.com  FirstName = Super  LastName = Player  Username = player123  Password = Test@123  ConfirmPassword = Test@123 | User is created. All required fields are filled. |
| TC11 | UC06 | Email = player@mail.com  FirstName = Super  LastName = Player  Username = player123  Password = Test@1234  ConfirmPassword = Test@123 | User is not created. Passwords do not match. |
| TC12 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = player123  Password = Test@1234  ConfirmPassword = Test@123 | User is not created. Email is incorrect. |
| TC13 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = ABC  Password = Test@1234  ConfirmPassword = Test@123 | User is not created. Username is too short. |
| TC14 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = player123  Password = TestOne  ConfirmPassword = TestOne | User is not created. Password is too easy (requires at least one extra number). |
| TC15 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = player123  Password = test123  ConfirmPassword = test123 | User is not created. Password is too easy (requires at least one extra capital letter). |
| TC16 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = player123  Password = TEST123  ConfirmPassword = TEST123 | User is not created. Password is too easy (requires at least one small letter). |
| TC17 | UC06 | Email = player@mail  FirstName = Super  LastName = Player  Username = player123  Password = TEST123  ConfirmPassword = TEST123 | User is not creating. Password is too short. |
| TC18 | UC07 | Username: Player123  Password: Test@123 | User is logged in. |
| TC19 | UC07 | Username: Player123  Password: Test | User is not logged in. Passwords do not match. |
| TC20 | UC07 | Username: Player  Password: Test@123 | User is not logged in. Username does not exist. |
| TC21 | UC18 | User is not blocked. | User gets blocked. |
| TC22 | UC18 | User is blocked. | Nothing changes. |
| TC23 | UC19 | User is blocked. | User does not get logged in and sees message. |
| TC24 | UC20 | User is blocked. | User gets unblocked. |
| TC25 | UC20 | User is not blocked. | Nothing changes. |