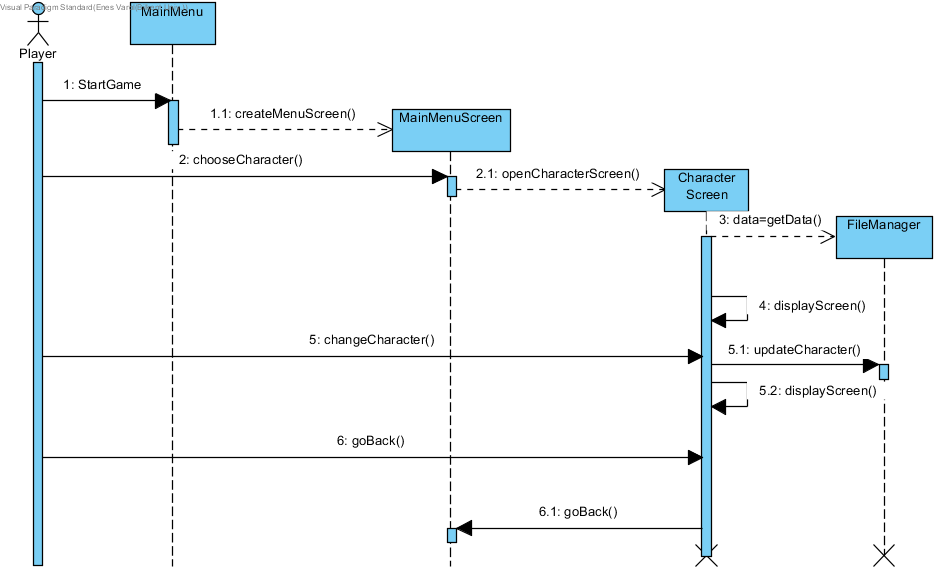
**5.2 Dynamic models**

**Select Character**

**Scenario:** Player wants to select character.

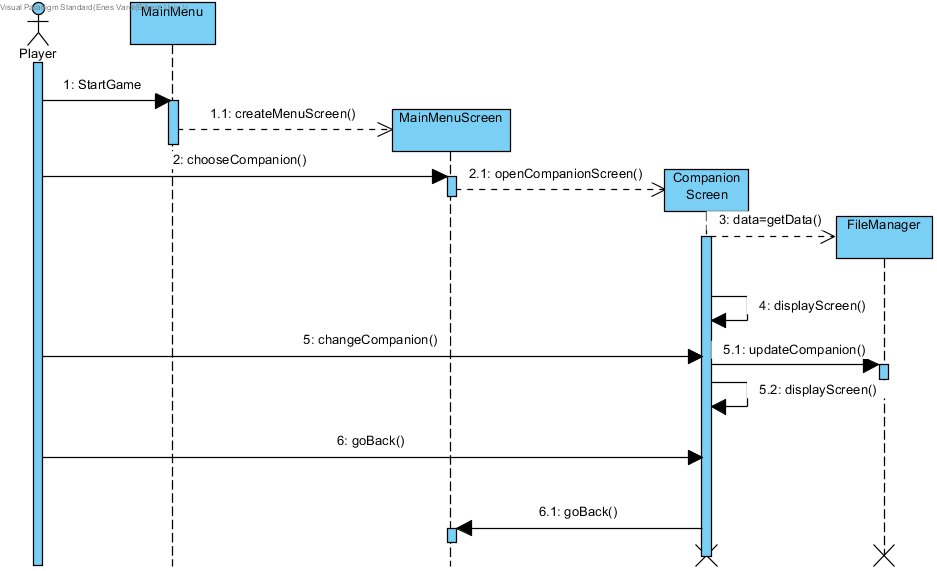
Player wants to select character, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks select character button from MainMenuScreen and chooseCharacter() method will be called. MainMenu screen initializes CharacterScreen via openCharacterScreen(). CharacterScreen gets data from FileManager(data=getData()) after that runs the displayScreen() method and shows it on screen. If player changes the character, then CharacterScreen sends the info with updateCharacter () method to the FileManager. If player decides to go back goBack() method will be called and CharacterScreen goes back to MainMenuScreen.



**Select Companion**

**Scenario:** Player wants to select companion.

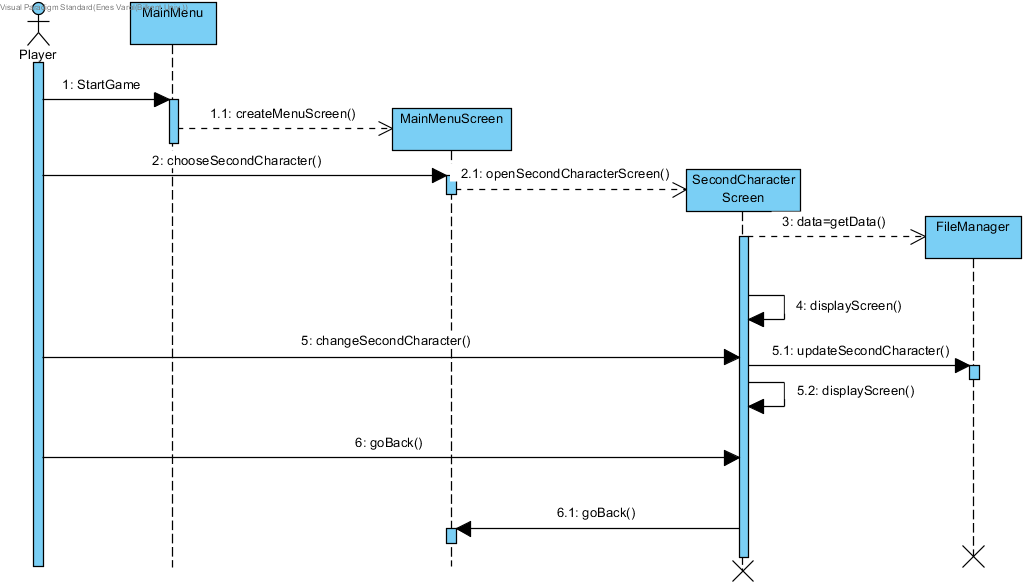
Player wants to select companion, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks select companion button from MainMenuScreen and chooseCompanion() method will be called. MainMenu screen initializes CompanionScreen via openCompanionScreen(). CompanionScreen gets data from FileManager(data=getData()) after that runs the displayScreen() method and shows it on screen. If player changes the companion, then CompanionScreen sends the info with updateCompanion() method to the FileManager. If player decides to go back goBack() method will be called and CompanionScreen goes back to MainMenuScreen.



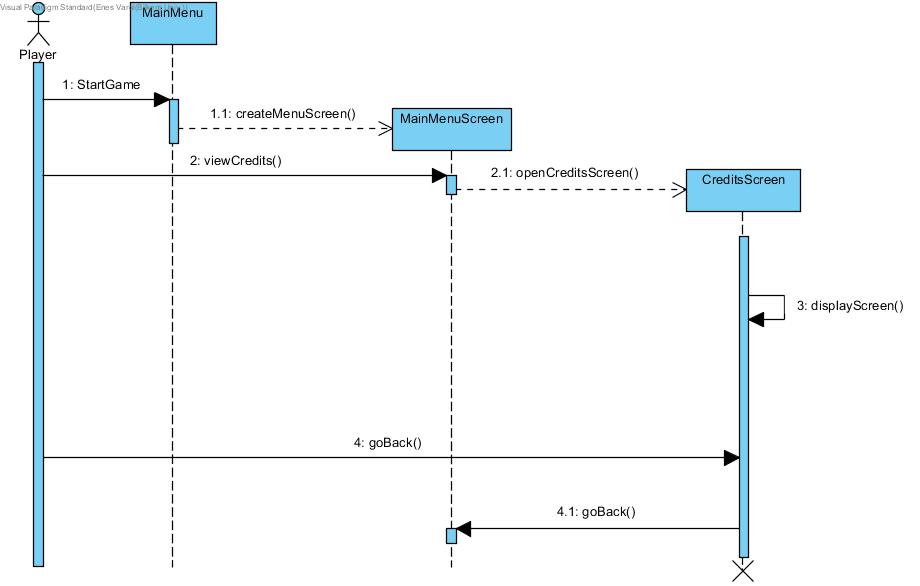
**Select Second Character**

**Scenario:** Player wants to select second character.

Player wants to select second character, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks select second character button from MainMenuScreen and chooseSecondCharacter() method will be called. MainMenuScreen initializes SecondCharacterScreen via openSecondCharacterScreen(). SecondCharacterScreen gets data from FileManager(data=getData()) after that runs the displayScreen() method and shows it on screen. If player changes the companion, then SecondCharacterScreen sends the info with updateSecondCharacter () method to the FileManager. If player decides to go back goBack() method will be called and SecondCharacterScreen goes back to MainMenuScreen.



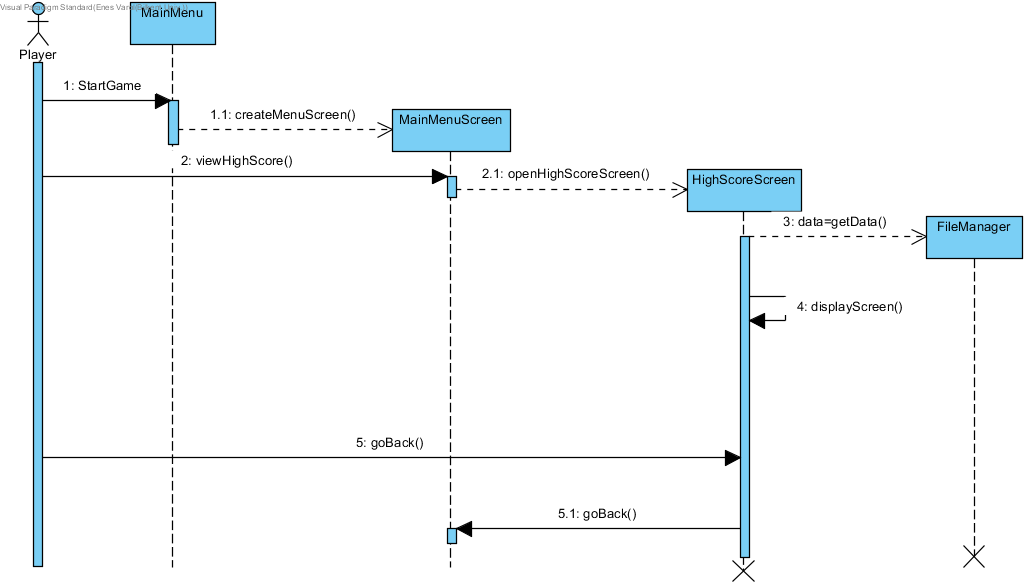
**View Credits**



**Scenario:** Player wants to see the credits.

Player wants to see the credits, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks credits button from MainMenuScreen and viewCredits() method will be called. MainMenu screen initializes CreditsScreen via openCreditsScreen(). CreditsScreen runs the displayScreen() method and shows it on screen. If player decides to go back goBack() method will be called and CreditsScreen goes back to MainMenuScreen.

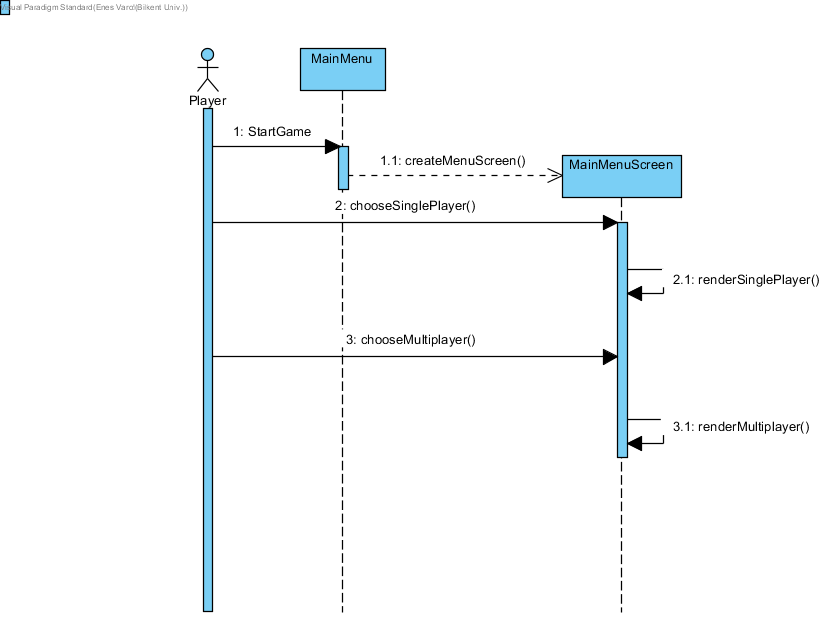
**View High Score**



**Scenario**

Player wants to see highscores, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks Highscores button from MainMenuScreen and viewHighScore() method will be called. MainMenu screen initializes HighScoreScreen via openHighScoreScreen(). HighScoreScreen gets data from FileManager(data=getData()) after that runs the displayScreen() method and shows it on screen. If player decides to go back goBack() method will be called and HighScoreScreen goes back to MainMenuScreen.

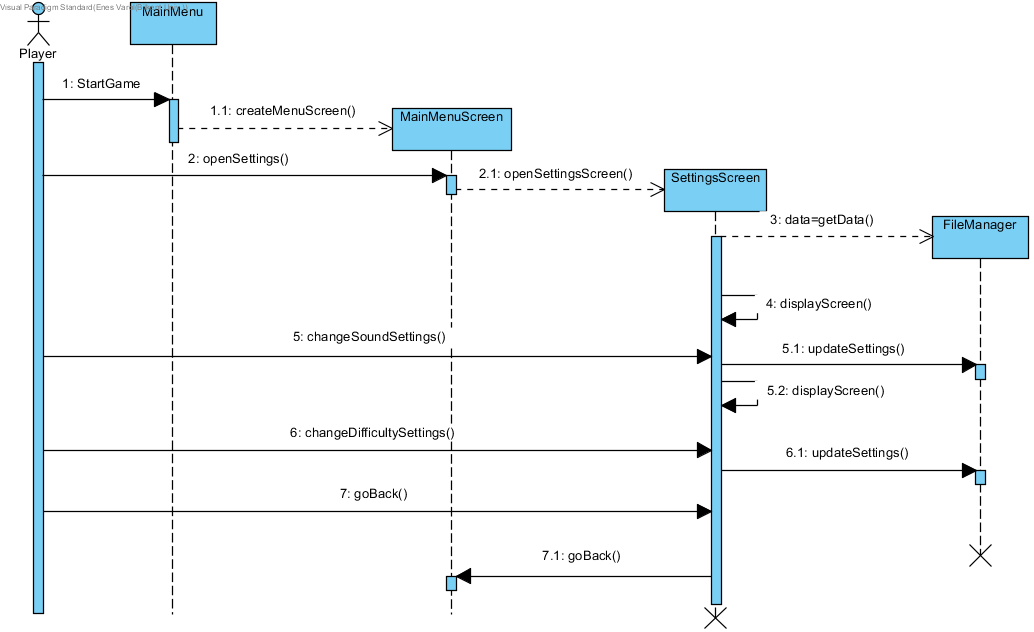
**Choose Game Mode**



**Scenario**

Player wants to choose a game mode, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks single player button from MainMenuScreen and screen runs renderSinglePlayer() method via chooseSinglePlayer(). Player clicks multiplayer button from MainMenuScreen and screen runs renderMultiplayer() method via chooseMultiplayer().

**Change Settings**



**Scenario:** Player wants to change the settings

Player wants to change settings, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player clicks settings button from MainMenuScreen and openSettings() method will be called. MainMenu screen initializes SettingsScreen via openSettingsScreen(). SettingsScreen gets data from FileManager(data=getData()) after that runs the displayScreen() method and shows it on screen. If player changes the sound level or the music level, then SettingsScreen sends the information with updateSettings() method to the FileManager. If player changes the difficulty, then SettingsScreen sends the information with updateSettings() method to the FileManager. If player decides to go back goBack() method will be called and SettingsScreen goes back to MainMenuScreen.

**Open Shop**

**Scenario:** Player wants to open shop and buy items.

Player wants to open shop and buy items, he/she starts to game. MainMenu initializes MainMenuScreen via createMenuScreen() method. Player presses the proper button from main menu in order to enter shop page and openShop() method will be called. MainMenuScreen initializes the ShopScreen via openShopScreen(). ShopScreen gets data from FileManager(data=getDATA()) and displays it on the screen with displayScreen() method where player has different options to buy an item for improving the attributes of his/her character during the game if item is equipped. Items in the shop is being bought by the game currency, via buyItem() method and the information sends to FileManager with updateShop() method after that ShopScreen updates itself with displayScreen() method. Initially the player has no items, by playing and gaining game currency, player might decide whether buying an item or not. Game currency is also saved in the file manager therefore, when user tries to buy an item which does not match with the balance of the player, then the system does not allow player to buy that item. Player has also an option to only look for items and go back without buying any, in that case if the player presses go back button, goBack() method will be called and MainMenuScreen initializes.

