*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Feature Document

User Story #**885**

**Team Member:**

Daylin Gonzalez Miguel

**Product Owner(s)**:

Eddy Garcia

**Mentor(s)**:

NA

**Instructor**: Masoud Sadjadi

**User Story** Modify registration page

* As a developer I need to modify the registration page so users can register in a sequence of steps
* As a developer I need to connect the registration page to the current database

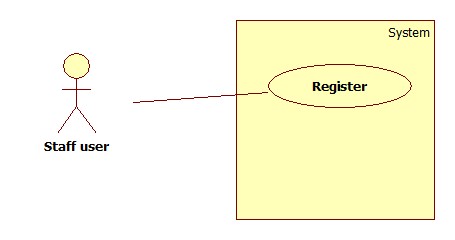
Acceptance Criteria

* Registration page is divided in different pages
* Registration Page is connected to the current database

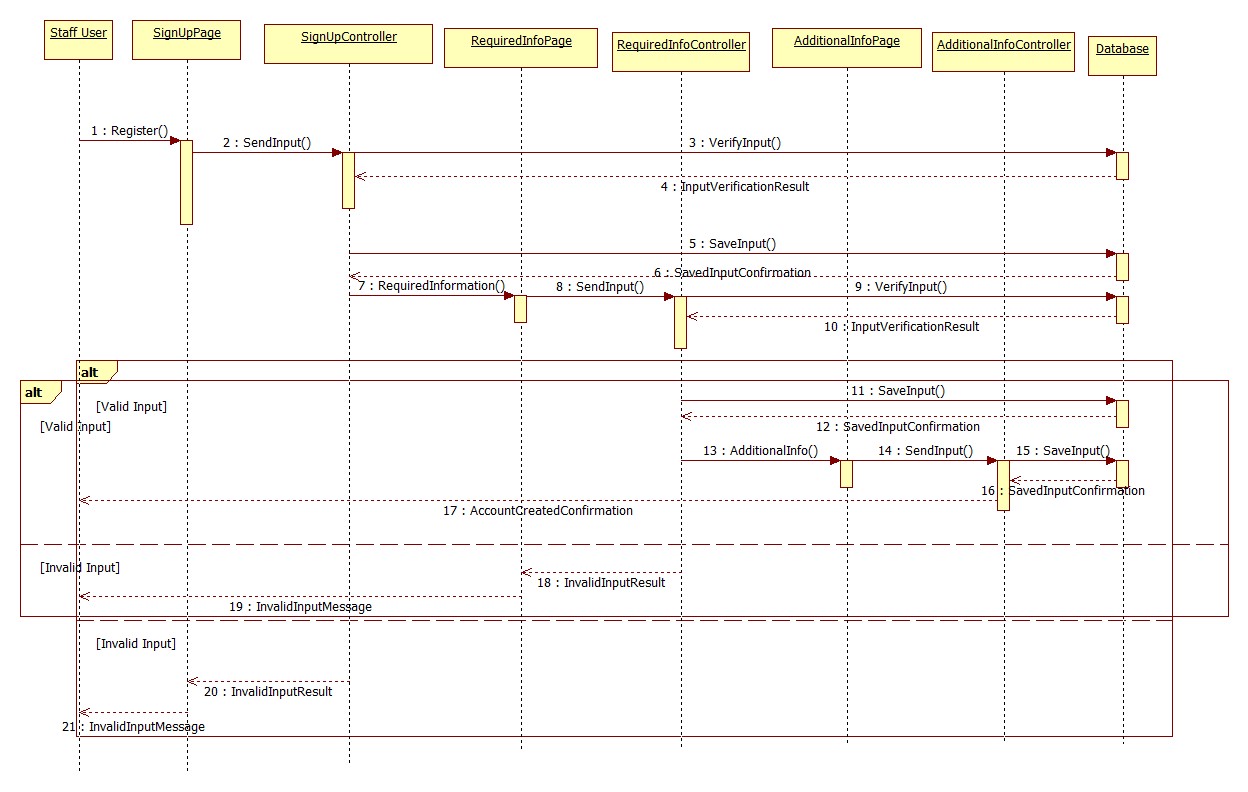
**Use Case** #**885 – Modify registration page**

|  |  |
| --- | --- |
| **Name** | **Create Account** |
| **Brief Description** | **The Create Account use case allows the User to create a login and become a Registered User.** |
| **Actor(s)** | **Guest** |
| **Flow of Events** | |
| **Basic Flow** | |
| **This use case starts when the User accesses the system feature that enables him/her to create an account by entering information that is maintained in the User’s account.**  **1. The User enters the required *User Account* information values and requests that the system saves the entered values.**  **2. The system validates the entered *User Account* information.**  ***3.* The values for the *User Account* information are stored in the User’s account. The system notifies the User that the account has been created.**  ***4.* The use case ends** | |
| **Alternate Flows** | |
| **Title** | **Description** |
| **User Enters Invalid User Account Information** | **If during Create Account, the system determines that the User entered invalid *User Account* information, the following occurs:**  **1. The system describes which entered data was invalid.**  **2. The system prompts the User to re-enter the invalid information.**  **3. The User re-enters the information and the system re-validates it.**  **4. If valid information is entered, the User Account Information is stored.**  **5. If invalid information is entered, the Entered Information is Invalid alternative flow is executed again. This continues until the User enters valid information.**  **Invalid *User Account* information:**  **- Missing information items**  **- Username already exists in the system**  **- *User Account* information entered does not comply to its definition in the glossary**  **- Not well formed e-mail address** |
| **Pre-Conditions** | |
| **Title** | **Description** |
| **None** |  |
| **Post-Conditions** | |
| **Title** | **Description** |
| **Success** | **The User entered data is stored in the user account.** |
| **The user account was not created** | **The User entered invalid data. No account will be created.** |
| **Extension Points** | |
| **None** | |

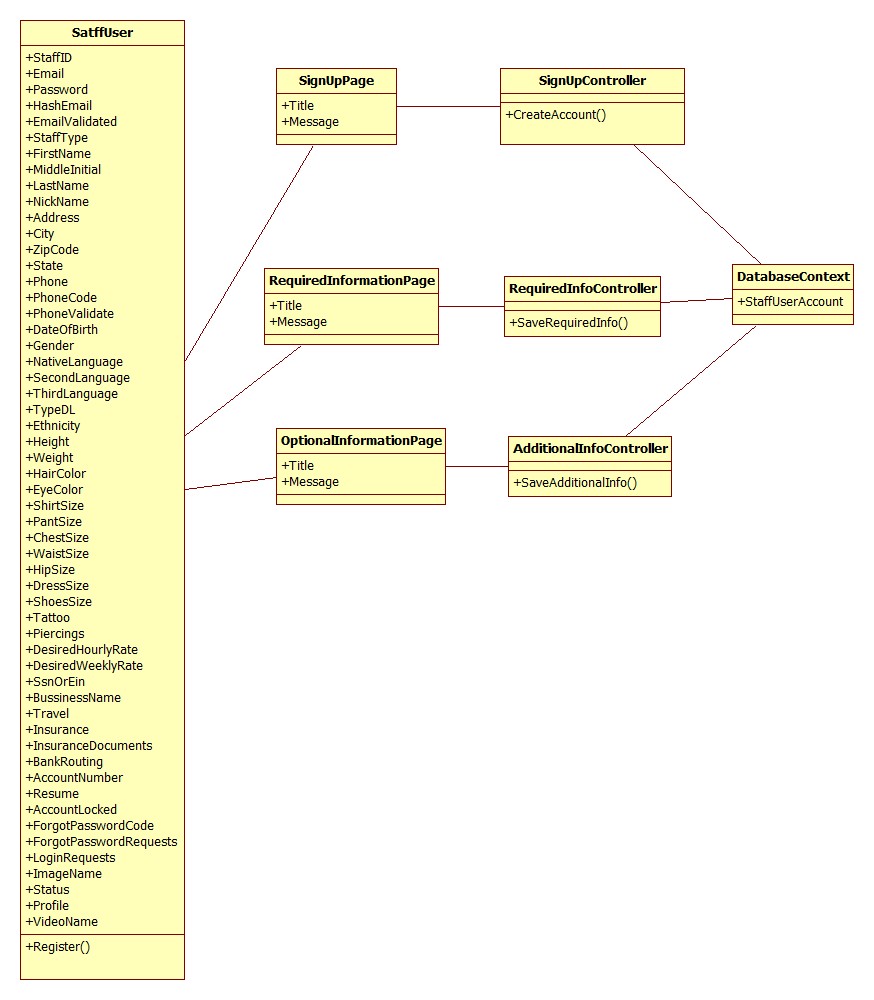
**Use Case Diagram**



**Sequence Diagram**



**Class Diagram**



**Unit Test**

|  |
| --- |
| **Test ID: UT-002** |
| **Purpose: Test that the Add(entry) and SaveChanges() methods**  **in the OurDatabaseContext class updated the database** |
| **Preconditions: Application is running** |
| **Input: new registered\_staff() { StaffID = 85, FirstName = "John", LastName = "Winston" };** |
| **Expected Output: The new registered\_staff is added to the database** |

|  |
| --- |
| **Test ID: UT-003** |
| **Purpose: Test that the View(view) method**  **in the SignUpController class returned the view specified** |
| **Preconditions: Application is running** |
| **Input: View(“RequiredInfo”);** |
| **Expected Output: The new RequiredInfo view is displayed** |

|  |
| --- |
| **Test ID: UT-004** |
| **Purpose: Test that the View(view) method**  **in the RequiredInfoController class returned the view specified** |
| **Preconditions: Application is running** |
| **Input: View(“AdditionalInfo”);** |
| **Expected Output: The new AdditionalInfo view is displayed** |

|  |
| --- |
| **Test ID: UT-005** |
| **Purpose: Test that the View(view) method**  **in the AdditionalInfoController class returned the view specified** |
| **Preconditions: Application is running** |
| **Input: View(“AccountCreated”);** |
| **Expected Output: The new AccountCreated view is displayed** |

|  |
| --- |
| **Test ID: UT-006** |
| **Purpose: Test that the SignUp() method**  **in the SignUpController class created a new account** |
| **Preconditions: Application is running and model state is valid** |
| **Input: SignUp(email, password, confirmPassword);** |
| **Expected Output: The new account is created** |

|  |
| --- |
| **Test ID: UT-007** |
| **Purpose: Test that the RequiredInfo() method**  **in the RequiredInfoController class added info to the account** |
| **Preconditions: Application is running and model state is valid** |
| **Input: RequiredInfo(FirstName,MiddleInitial,LastName,Address,City,ZipCode,State,Phone,DateOfBirth,Gender );** |
| **Expected Output: The new account is updated with the new info** |

|  |
| --- |
| **Test ID: UT-008** |
| **Purpose: Test that the AdditionalInfo() method**  **in the AdditionalInfoController class added info to the account and returned AccountCreated view** |
| **Preconditions: Application is running and model state is valid** |
| **Input: AdditionalInfo(StaffType,NativeLanguage,SecondLanguage,ThirdLanguage,TypeDL,Ethnicity,Height,Weight,HairColor,EyeColor,ShirtSize,PantSize,ChestSize,WaistSize,HipSize, DressSize,ShoeSize,Tattoos,Piercings,DesiredHourlyRate,DesiredWeeklyRate,SsnOrEin, BusinessName,Travel,Insurance, BankRouting, AccountNumber );** |
| **Expected Output: The new account is updated with the new info and AccountCreated view is returned** |

**Integration Test**

|  |
| --- |
| **Test ID: IT-002** |
| **Purpose: Verify whether the user is able to create an account** |
| **Preconditions: Internet browser is open and application is running** |
| **Input: { Email = daylin2512@yahoo.es, FirstName = "John", MiddleInitial = “”, LastName = "Winston", Address = “40 Sidonia Ave”, City = “Coral Gables”, ZipCode = “33134”, State = “FL”, Phone = “7865518745”, DateOfBirth = “12/15/85”, Gender = “Female”, StaffType = “Dancer”, NativeLanguage = “Spanish”, SecondLanguage = “English”, ThirdLanguage = “”, TypeDL = “”, Ethnicity = “”, Height = “5’6””, Weight = “130”, HairColor = “Black”, EyeColor = “Green”, ShirtSize = “S”, PantSize = “27”, ChestSize = “”, WaistSize = “”, HipSize = “”, DressSize = “M”, ShoeSize = “ 37”, Tattoos = “No”, Piercings = “Yes”, DesiredHourlyRate = “”, DesiredWeeklyRate = “”, SsnOrEin = “458785412”, BusinessName = “”, Travel = “ Yes”, Insurance = “No”, BankRouting = “”, AccountNumber = “” }** |
| **Expected Output: The account is created and the confirmation page shows up** |

**Visual User Guide**