|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU001** | | | |
| **Use Case No.** | STU001 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Profile | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 26/09/2015 | **Priority** | Normal |
| **Actor:**   * Student   **Summary:**   * This use case allows Student to view their own profile.   **Goal:**   * Student can view their own profile.   **Triggers:**   * Student sends command to view profile.   **Preconditions:**   * User must log into the system with role as Student.   **Post Conditions:**   * **Success**: Show Student their profile information * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student goes to view their profile | Display new view that shows Student’s profile including:   * + Student’s name: text   + Student’s code: text   + Student’s identification number: text   + Student’s gender: text   + Student’s date of birth: text   + Student’s address: text   + Student’s phone number: text   + Student’s email address: text   + Phone number of Student’s parent: text   + Email address of Student’s parent: text   + Student’s tuition payments type: text   + Student’s class year: text   + Student’s English level: text |   **Alternative Scenario:** N/A  **Exceptions:**  N/A  **Relationships:** N/A  **Business Rules:**   * Student’s information is loaded from the system’s database | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU002** | | | |
| **Use Case No.** | STU002 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Profile | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 25/09/2015 | **Priority** | Normal |
| **Actor:**   * Student   **Summary:**   * This use case allows Student to update their profile information.   **Goal:**   * Student can update their profile information.   **Triggers:**   * Student sends command to update profile.   **Preconditions:**   * User must log into the system with role as Student.   **Post Conditions:**   * **Success**: Update new profile information to database. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student goes to Update Profile view | Display new view that allows Student to input new information:   * + Student’s phone number: number text input, required, length 8 - 15   + Student’s address: free text input, required, length 3 -250   + Parent’s phone number: number text input, required, length 8 - 15   + Parent’s email address: free text input, required, length 3 -250 | | 2 | Student alters the information in the textboxes |  | | 3 | Student sends command to update profile information | System validates information, request confirmation from Student  [Exception 1] | | 4 | Student sends command to confirm updating profile information | System updates new information to database |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student sends command to update profile information | System shows error message asking user to input information as the fields’ requirements |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Each textbox has pre-inputted text showing current profile information * In successful scenario, new information will be updated to the system | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU003** | | | |
| **Use Case No.** | STU003 | **Use Case Version** | 1.0 |
| **Use Case Name** | Change Password | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 25/09/2015 | **Priority** | Normal |
| **Actor:**   * Student   **Summary:**   * This use case allows Student to change their account’s password.   **Goal:**   * Student can change their account’s password   **Triggers:**   * Student sends command to change password.   **Preconditions:**   * User must log into the system with role as Student   **Post Conditions:**   * **Success**: Update new password to database. * **Fail:** The system shows an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student goes to change password view | Display new view containing textboxes for user to input their current and new password:   * + Current password: free text input, required, length 6-32   + New password: free text input, required, length 6-32   + Confirm new password: free text input, required, length 6-32 | | 2 | User fills in the textboxes |  | | 3 | Student sends command to change password | System validates information, request confirmation from Student  [Exception 1, 2, 3] | | 4 | Student sends command to confirm changing password | System updates new password to database. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Student inputs the wrong current password | Show message to notify that the current password is wrong. | | 2 | If the new password and the confirm password are not matched. | Show message to notify that the confirm password is not matched with the new password. | | 3 | If the new password and the current password are matched | Show message to notify that the new password is the same as the current password |   **Relationships:** N/A  **Business Rules:**   * In successful scenario, new password will be updated to the system * Password is encrypted before sending to the server * Textboxes will display \* character for all inputted characters * If all the textboxes is blank, the system keeps the current password in the database * New password must be different from the current password | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU004** | | | |
| **Use Case No.** | STU004 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Financial Status | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 25/09/2015 | **Priority** | Normal |
| **Actor:**   * Student   **Summary:**   * This use case allows Student to view tuition payments history, retake courses’ payments history, and the tuition fees plan.   **Goal:**   * Student can view their financial status.   **Triggers:**   * Student send command to view their financial status information   **Preconditions:**   * User must log into the system with role as Student   **Post Conditions:**   * **Success**: Show financial status information to Student * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student goes to Financial status view | Display new view showing following information:   * Tuition payments history: link * Retake Courses’ payments history: link * Tuition fees plan: link | | 2 | Student goes to Financial Status view  [Alternative 1, 2] | Display new view showing Tuition payments history in tabular form including:   * + Order number: text   + Semester’s name: text   + Student’s payment status: text   + Transmission date: text   + Transmission deadline: text   + Detail transmission: link | | 3 | Student sends command to view Detail transmission | Display new view showing Detail transmission including:   * + Student’s name: text   + Student’s code: text   + Student’s gender: text   + Student’s identification number: text   + Student’s tuition payments type: text   + Payment detail description: text   + Initial tuition charge: text   + Deduction fees: text   + Actual tuition charge: text   + Payed fees: text   + Balance: text   + Tuition payment status: text   + The transmission date: text   + Bank’s name: text |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Student sends command to view Retake courses’ payments history | Display new view showing Retake courses’ payments history in tabular form including:   * + Order number: text   + Retake course’s code: text   + Retake course’s name: text   + Student’s payment status: text   + Transmission date: text   + Transmission deadline: text   + Detail transmission: link | | 2 | Student sends command to view Tuition fees plan | Display new view showing Tuition fees plan in tabular form including:   * + Order number: text   + Semester’s code: text   + Semester’s name: text   + Tuition fees: text   + Actual fees: text   + Detail transmission: link |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All information is loaded from the system’s database | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU005** | | | |
| **Use Case No.** | STU005 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Studied Subjects | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 25/09/2015 | **Priority** | Normal |
| **Actor:**   * Student   **Summary:**   * This use case allows Student to view their semesters and subjects history, and their incomplete subjects.   **Goal:**   * Student can view their study progress.   **Triggers:**   * Student sends command to view their study progress   **Preconditions:**   * User must log into the system with role as Student.   **Post Conditions:**   * **Success**: Show study progress information to student. * **Fail:** Show an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Student goes to Study Progress view | Display new view showing Study Progress of the latest semester including:  ***Semester’s information***   * Semester’s name: text * Semester’s period: text   ***Subject’s information*** (displayed in tabular form)   * + Order number: text   + Course’s code: text   + Course’s name: text   + Numbers of credits: text   + The half of semester: text   + Course’s grade: text   + Course’s status: text   + Additional note: text | | 2 | Student sends command to view different semester  [Alternative 1] | Display new view showing selected semester information as above |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Student sends command to view Incomplete Subjects | Display new view showing Incomplete subjects in tabular form including:   * + Order number: text   + Course’s code: text   + Course’s name: text   + Numbers of credits: text   + Semester’s name: text   + Reason of incompleteness: text   + Additional note: text |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All information is loaded from the system’s database * The latest semester is determined by Semester’s period. * Only finished semesters are available. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU006** | | | |
| **Use Case No.** | STU006 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Debt | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 26/09/2015 | **Priority** | Normal |
| **Actor:**   * CreditStudent   **Summary:**   * This use case allows CreditStudent to view their own student loans.   **Goal:**   * CreditStudent can view their student loans.   **Triggers:**   * CreditStudent sends command to view their student loans   **Preconditions:**   * User must log into the system with role as CreditStudent   **Post Conditions:**   * **Success**: Show user’s debt information * **Fail:** Show an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | CreditStudent goes to Debt view | If CreditStudent is undergraduate, display new view showing Debt information including:  ***CreditStudent information:***   * CreditStudent’s name: text * CreditStudent’s code: text   ***Debt information:*** (displayed in tabular form)   * + Date: text   + Loan’s description: text   + Tuition fees: text   + Payed fees: text   + Loan: text   + Exchange rate of dollar to Vietnamese dong: text   + Additional note: text   [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | CreditStudent goes to Debt view | If CreditStudent is graduate. display new view showing Debt information including:  ***CreditStudent information:***   * + CreditStudent’s name: text   + CreditStudent’s code: text   + CreditStudent’s date of graduation: text   ***Debt information:*** (displayed in tabular form)   * + Remaining debt: text.   + Interest: text   + Payed amount: text   + After-payed debt: text   + Repayment date: text   + Interest rate: text |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All information is loaded from the system’s database. * After-payed debt is calculated by Payed amount, Interest and Remaining debt * Interest is calculated by Remaining debt and Interest rate. * Repayment dates is a month apart, the day is the exact day of graduation. If graduation date is at the end of the month, then Repayment dates is also at the end of each month | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – STU007** | | | |
| **Use Case No.** | STU007 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Return Money Status | | |
| **Author** | Vu Nhat Anh Khoa | | |
| **Date** | 26/09/2015 | **Priority** | Normal |
| **Actor:**   * InvestedStudent   **Summary:**   * This use case allows Student to view their own student loans.   **Goal:**   * InvestedStudent can view their Return money status.   **Triggers:**   * Student sends command to view their return money status.   **Preconditions:**   * User must log into the system with role as InvestedStudent.   **Post Conditions:**   * **Success**: Show user’s return money status information * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | InvestedStudent goes to Return money status view | Display new view showing Return money status information including:  ***InvestedStudent information:***   * + Student’s name: text   + Student’s code: text   + Student’s date of graduation: text   ***Debt information:***   * + Remaining debt: text.   + Payed amount: text   + Interest rate: text   + Next repayment date: text |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All information is loaded from the system’s database. | | | |