User Interface Project

SMU Access System

Group Member: Yenyi Huang,
Bandar AlTuraif,
Qiang Dai

Date: 2013/5/11

Content

I. Vision 4

a. Description 4

b. Design Decisions 4

c. Assumption..5

d. Future Tasks 6

II. Task Analysis 6

1) Data Model...6

2) Task Analysis..9

i. Eliciting tasks:...9

ii. High-level Tasks:..9

iii. Task descriptions:..10

3) Use case and Scenarios...33

1. Use Case Diagram..33

4) Hierarchical Task Analysis..34

III. Initial Design..44

1) Credo Matrix...46

2) Low Fidelity Prototype...47

4) Gestalts...53

5) Data Presentation..54

6) Usability Testing 55

IV. Detailed Prototype..60

1. Function Design..60

2. Error Message...66

3. Prototype...71

1) Login Interface...71

2) Home Page..72

3) Academic -Class Schedule...73

4) Academic-Class Search..74

5) Academic-Class Catalog...75

6) Academic-Class Drop...76

7) Academic-Grades..77

8) Financial Information-Account Summary...78

9) Financial Information-Financial Aid..79

10) Financial Information-Transaction Detail...80

11）Personal Information-Address..81

12）Personal Information-Address Add..82

13）Personal Information-Name...83

14）Personal Information-Name Add..84

15）Personal Information-Phone Number...85

16）Personal Information-Phone Number Add..86

17）Personal Information-Emergency...87

18）Personal Information-Emergency Add..88

19）Personal Information-Demographic...89

20）Personal Information-Ethnicity...90

I. Vision

a. Description

Students are automatically assigned an Access.SMU account during the initial account creation process. This allows them to view their personal information, emergency contact information, register for AARO (if applicable), view class schedule, enroll in classes, add/drop classes, view grades and veiw financial aid packages.

b. Design Decisions

Requirement: This is first step of the project. We determine the main goal and fundamental functions of the project.
Analysis: We do investigations from our objective users, including students, faculties and parents. After getting the feedbacks from them, we change the initial requirement statement and add more functions according to the requirement of users.
Design prototype: This is the first design of windows for each functional part.
Final UI Design: After we finish the first prototype, we do usability test for it. For convenient and understandable goal, we change partial design of prototype and finish the final user interface design.

Four states above is the whole design procedure, and their relationship can be draw in the diagram as the "Figure 1" below.

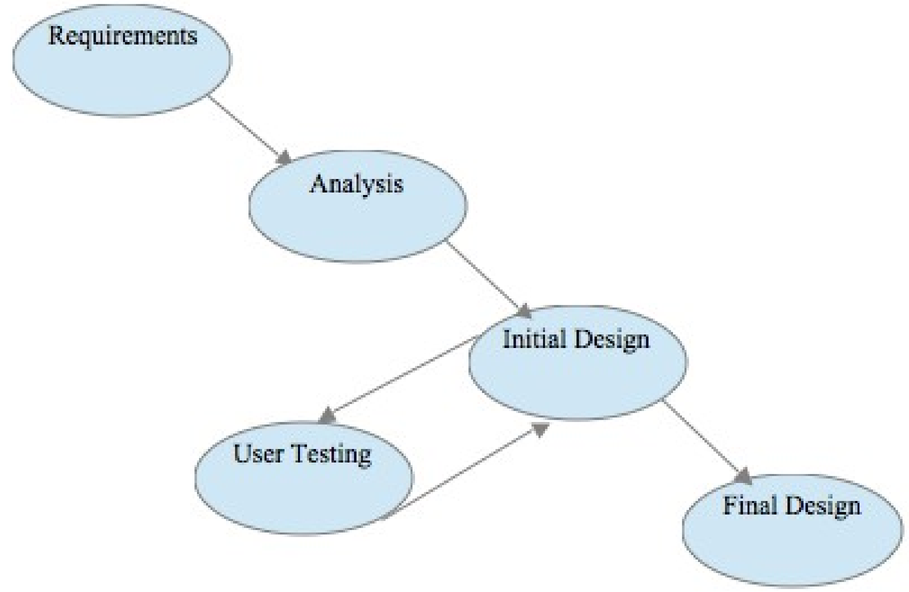


Figure 1

c. Assumption

In this project, we assume that students' account has been established and the password has been set.

Another important assumption is that the account of library and blackboard can cooperate with the access accounts which means once we login the access system, we can directly go to cooperate hyper links without login again.

d. Future Tasks

Only the user interfaces for student role are addressed in this project; therefore, user interfaces for employees, parents, and administrators need to be done in the future. Another future task is to provide new, useful features, such as library access, campus activity information, open hour of gym and offices, parking notice, and access to job postings.

II. Task Analysis

1) Data Model

As can be seen, the data model for this project will consist of four primary Entities: Student, Financial Info, Academics, and Personal Info. The student Entity has five attributes: Student ID, Gender, DOB, Nation ID, and Religion. This student Entity consists of one Financial Info Entity, Academics Entity, and Personal Info Entity.

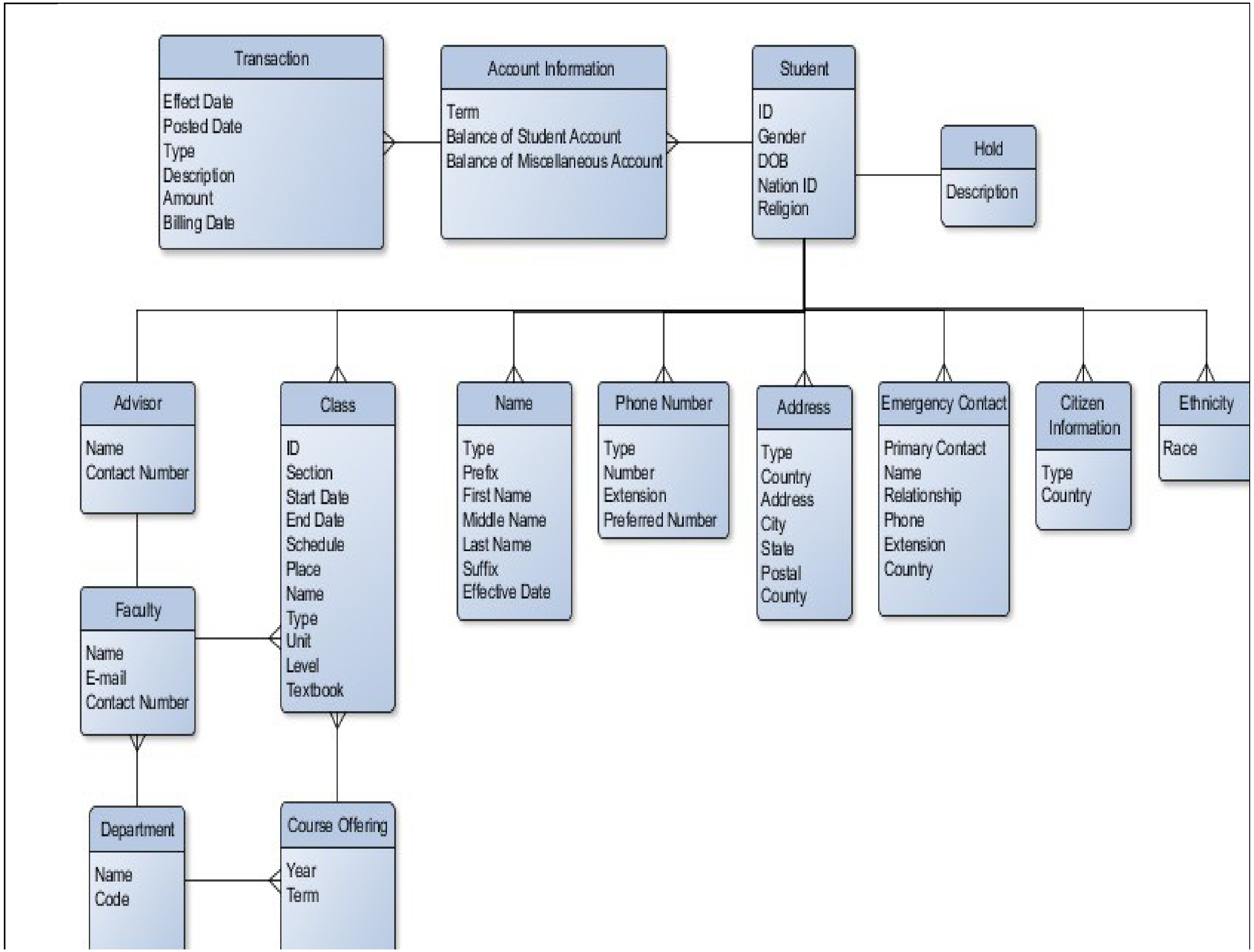
A Financial Info Entity could have multiple financial aid Entities to document financial aid records and multiple account Entities to keep summary of financial information, while each financial aid and account Entity can belong toone specific financial info Entity. An account Entity may have multiple transaction Entities to store detailed data for each transaction, but each transaction Entity could only belong to one account Entity.

A personal info Entity could have multiple name, phone number, address, emergency contact, citizen information, ethnicity Entities. Each of them keeps a specific category of personal data and could only belong to one personal info Entity.

As for academic Entity, it could have multiple class Entities, multiple record Entities, and only one transcript Entity. A class Entity has 15 attributes to store description data for a specific course and each class has a record Entity to store the grading information. Multiple record Entities could be associated to one transcript Entity.

Since a department could have many faculty members and offer many courses, a department Entity has multiple faculty Entities and courses Entities. Each course Entity could have multiple class Entities, for example, one course may have different classes, given each different section IDs.

Besides student Entity, an academic Entity could also belong to Advisor and Hold Entity, which means an advisor or a hold can be assigned to many academics records. For example, several students can have the same advisor. In addition, it is one to one relationship between advisor and faculty.


Transaction
Effect Date
Posted Date
Type
Description
Amount
Billing Date
Account Information
Term
Balance of Student Account
Balance of Miscellaneous Account
Student
ID
Gender
DOB
Nation ID
Religion
Hold
Description
Advisor
Name
Contact Number
Class
ID
Section
Start Date
End Date
Schedule
Place
Name
Type
Unit
Level
Textbook
Name
Type
Prefix
First Name
Middle Name
Last Name
Suffix
Effective Date
Phone Number
Type
Number
Extension
Preferred Number
Address
Type
Country
Address
City
State
Postal
County
Emergency Contact
Primary Contact
Name
Relationship
Phone
Extension
Country
Citizen Information
Type
Country
Ethnicity
Race
Faculty
Name
E-mail
Contact Number
Department
Name
Code
Course Offering
Year
Term

2) Task Analysis

i. Eliciting tasks:

The following is a list of techniques we will use to elicit task information from our prospective customers.

1. Customer interviews:
These can be done on a one on one level or with a group of potential users of this system. These will be informal and consist of five to six questions at the most.

2. Questionnaires and Surveys:

These will be sent to the customer to hand out to perspective users of this system. This can reach a larger number of users and should be less that one page.

3. Customer observation:

Time will be spent at the customer's location observing how the potential users perform some of the defined tasks now.

ii. High-level Tasks:

The system will perform the following high-level tasks:

1 View/Edit personal information

- 1.1 Edit/View Addresses
- 1.2 Edit/View Names
- 1.3 Edit/View Phone Numbers
- 1.4 Edit/View Emergency Contact Information
- 1.5 Edit/View Demographic Data
- 1.6 Edit/View Ethnicity

2 View/Edit Academics Information

- 2.1 View Class Schedule
- 2.2 Search for Classes
- 2.3 Browse Course Catalog
- 2.4 View Grades
- 2.5 Enroll in Classes
  - 2.5.1 Add Classes
  - 2.5.2 Drop Classes

3 View Financial Information

- 3.1 View Account Summary
- 3.2 View Transaction Details
- 3.3 View Financial Aid

iii. Task descriptions:

For the View/Edit Personal Information task we have the following subtasks:

1.1. Edit/View Addresses: personal information in this use case is address.

a) Task description: This use case describes how a Student uses the system to view the personal information held by the University and submit a request to change the personal information.

b) Actors:

1. ***Student***: The Student is registered on a course at the University and can view

personal information about the Student held by the University.

2. ***Student Information System (SIS)***: The system storing the student personal information.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
3. Display Personal Information: The ***SIS*** displays personal information and an option to change personal information.
4. Change Personal Information: The ***Student***selects to change personal information. The ***SIS*** displays editable personal information and options to cancel or submit a change request.
5. Edit Personal Information: The ***Student*** edits the personal information.
6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.
7. Update Personal Information: The ***SIS*** is updated with the changed personal information for the ***Student***.
8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the personal information and an option to change the personal information.

9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.
2. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

1.2. Edit/View Names: same as use case 1.1, personal information in this use case is names.

a) Task description: This use case describes how a Student uses the system to view the names held by the University and submit a request to change the names.

b) Actors:

1. ***Student***: The Student is registered on a course at the University and can view names about the Student held by the University.
2. ***Student Information System (SIS)***: The system storing the student names.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
3. Display Names: The ***SIS*** displays names and an option to change names.
4. Change Names: The ***Student***selects to change names. The ***SIS*** displays editable names and options to cancel or submit a change request.
5. Edit Names: The ***Student*** edits the names.
6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.
7. Update Names: The ***SIS*** is updated with the changed names for the ***Student***.
8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the names and an option to change the names.
9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.
2. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the

communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

1.3. Edit/View Phone Numbers: same as use case 1.1, personal information in this use case is phone numbers.

a) Task description: This use case describes how a Student uses the system to view the phone numbers held by the University and submit a request to change the phone numbers.

b) Actors:
   1. ***Student***: The Student is registered on a course at the University and can view phone numbers about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student phone numbers.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:
   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
   3. Display Phone numbers: The ***SIS*** displays phone numbers and an option to change phone numbers.
   4. Change Phone numbers: The ***Student***selects to change phone numbers. The ***SIS*** displays editable phone numbers and options to cancel or submit a change request.

5. Edit Phone numbers: The ***Student*** edits the phone numbers.

6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.

7. Update Phone numbers: The ***SIS*** is updated with the changed phone numbers for the ***Student***.

8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the phone numbers and an option to change the phone numbers.

9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.
2. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

1.4. Edit/View Emergency Contact Information: same as use case 1.1, personal information in this use case is emergency contact information.

a) Task description: This use case describes how a Student uses the system to view the emergency contact information held by the University and submit a request to change the emergency contact information.

b) Actors:
   1. ***Student***: The Student is registered on a course at the University and can view emergency contact information about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student emergency contact information.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:
   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
   3. Display Emergency contact information: The ***SIS*** displays emergency contact information and an option to change emergency contact information.
   4. Change Emergency contact information: The ***Student***selects to change emergency contact information. The ***SIS*** displays editable emergency contact information and options to cancel or submit a change request.
   5. Edit Emergency contact information: The ***Student*** edits the emergency contact information.
   6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.
   7. Update Emergency contact information: The ***SIS*** is updated with the changed

emergency contact information for the ***Student***.

8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the emergency contact information and an option to change the emergency contact information.

9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.

i) Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

1.5. Edit/View Demographic Data: same as use case 1.1, personal information in this use case is demographic data.

a) Task description: This use case describes how a Student uses the system to view the demographic data held by the University and submit a request to change the demographic data.

b) Actors:
   1. ***Student***: The Student is registered on a course at the University and can view demographic data about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student demographic data.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:
   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
   3. Display Demographic data: The ***SIS*** displays demographic data and an option to change demographic data.
   4. Change Demographic data: The ***Student***selects to change demographic data. The ***SIS*** displays editable demographic data and options to cancel or submit a change request.
   5. Edit Demographic data: The ***Student*** edits the demographic data.
   6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.
   7. Update Demographic data: The ***SIS*** is updated with the changed demographic data

for the ***Student***.

8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the demographic data and an option to change the demographic data.

9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.

    i) Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

1.6. Edit/View Ethnicity: same as use case 1.1, personal information in this use case is ethnicity.

a) Task description: This use case describes how a Student uses the system to view the ethnicity held by the University and submit a request to change the ethnicity.

b) Actors:
   1. ***Student***: The Student is registered on a course at the University and can view ethnicity about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student ethnicity.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:
   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
   3. Display Ethnicity: The ***SIS*** displays ethnicity and an option to change ethnicity.
   4. Change Ethnicity: The ***Student***selects to change ethnicity. The ***SIS*** displays editable ethnicity and options to cancel or submit a change request.
   5. Edit Ethnicity: The ***Student*** edits the ethnicity.
   6. Submit Change Request: The ***Student*** selects to submit a change request to the ***SIS***.
   7. Update Ethnicity: The ***SIS*** is updated with the changed ethnicity for the ***Student***.
   8. Acknowledge Change Request Submission: The ***SIS*** displays a message to inform the ***Student*** that a change request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The

system displays the ethnicity and an option to change the ethnicity.

9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Change Request Cancelled: At ***step 5*** or ***step 6*** if the ***Student*** selects to cancel the change request, the use case resumes from ***step 3***.

i) Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

For the View/Edit Academics Information task we have the following subtasks:

2.1. View class schedule

a) Task description: This use case describes how a Student uses the system to view their class schedule held by the University.

b) Actors:
   1. ***Student***: The Student is registered on a course at the University and can view class schedule about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student class schedule.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems. The student must have enrolled in at least one course at any semester past or present..

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:
   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
   3. Select academic semester-year: The ***Student*** selects the academic semester-year of courses taken.
   4. Display Class Schedule: The ***SIS*** displays class schedule for academic semester-year chosen.
   5. Use Case Ends: The use case ends.

f) Alternative Flows:
   1. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the

communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 5***.

2.2. Search for classes

a) Task description: This use case describes how a Student uses the system to search for classes.

b) Actors:

1. ***Student***: The Student must be registered as a student with the University.
2. ***Student Information System (SIS)***: The system storing classes.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
3. Select academic semester-year: The ***Student*** selects the current academic semester-year or future academic semester-year to search.
4. Select course subject: The ***Student*** can select the course subject to limit results.
5. Select course number: The ***Student*** can select the course number to limit results.
6. Select course career: The ***Student*** can select the course career to limit results.
7. Select university curriculum: The ***Student*** can select the university curriculum to limit results.
8. Display Classes: The ***SIS*** displays classes for academic semester-year chosen.
9. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Handle communication failures: At ***step 8***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 8***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 9***.

2.3. Browse course catalog

a) Task description: Students may view the course catalog description of a course before registering. They are especially likely to view it if they are prevented from enrolling in a course because of a missing prerequisite.

b) Actors:

1. ***Student***: The Student must be registered as a student with the University.
2. ***Student Information System (SIS)***: The system storing classes.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.

3. Select course subject: The ***Student*** can select the course subject to limit results.
4. Display Classes: The ***SIS*** displays classes for the course subject chosen.
5. Select course number: The ***Student*** can select the course number they want to view.
6. Display Class Information: The ***SIS*** displays class information.
7. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Handle communication failures: At ***step 4 or step 6*** if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 4 or step 6***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 7***.

2.4. View grades

a) Task description: This use case describes how a Student uses the system to view their grades held by the University.

b) Actors:

1. ***Student***: The Student is registered on a course at the University and can view grades about the Student held by the University.
2. ***Student Information System (SIS)***: The system storing the student grades.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems. The student must have completed at least one semester.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
2. Authenticate Student: The ***SIS*** will authenticate the user name an password entered by the ***Student***.
3. Select academic semester-year: The ***Student*** selects the academic semester-year of courses taken.
4. Display Grades: The ***SIS*** displays grades for courses taken.
5. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 5***.

2.5. Enroll in classes

2.5.1 Add classes

a) Task description: This use case describes how a Student uses the system to add classes.

b) Actors:

1. ***Student***: The Student is registered at the University and eligible to enroll in classes.
2. ***Student Information System (SIS)***: The system storing the classes.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems. The student must be eligible to enroll in classes.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. Search for classes: Do ***use case 2.2***.
2. Add class: The student selects the class they want to add.
3. Acknowledge Add Class Request: The ***SIS*** displays a message to inform the ***Student*** that the class addition request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the class schedule and an option to change the class schedule.
4. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Add Request Cancelled: At ***step 2,*** if the ***Student*** selects to cancel the add request, the use case resumes from ***step 1***.
2. Add Request Illegal: At ***step 2,*** if pre-requests classes are not met or registration stop for non-payment of outstanding bill, the use case resumes from ***step 1***.
3. Handle communication failures: At ***step 2***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications

are re-established, the basic flow is resumed at ***step 2***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 4***.

2.5.2 Drop classes

a) Task description: This use case describes how a Student uses the system to drop classes.

b) Actors:

1. ***Student***: The Student is registered at the University and eligible to enroll in classes.
2. ***Student Information System (SIS)***: The system storing the classes.
3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems. The student must be eligible to enroll in classes.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

1. View class schedule: Do ***use case 2.1***.
2. Drop class: The student selects the class they want to drop.
3. Acknowledge Drop Class Request: The ***SIS*** displays a message to inform the ***Student*** that the class addition request has been submitted and an acknowledgement option. The ***Student*** selects to acknowledge the change request message. The system displays the class schedule and an option to change the class schedule.
4. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Drop Request Cancelled: At ***step 2,*** if the ***Student*** selects to cancel the drop

request, the use case resumes from ***step 1***.

2. Drop Request Illegal: At ***step 2,*** if registration stop for non-payment of outstanding bill, the use case resumes from ***step 1***.
3. Handle communication failures: At ***step 2***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, And if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 2***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 4***.

For the View financial information task we have the following subtasks:

3.1 View Account Summary

a) Task description: This use case describes how a Student uses the system to view his financial information held by the University
b) Actors:
    1. ***Student***: The Student is registered on a course at the University and can view his financial information about the Student held by the University.
    2. ***Student Information System (SIS)***: The system storing the student financial information.
c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.
d) Postconditions: The transaction terminates normally while the financial information is displayed with no system failures.
e) Basic Flow of Events:
    1. Launch System: The use case begins when the actor ***Student*** launches the URL of

the system in a web browser.

2. Authenticate Student: The ***SIS*** will authenticate the user name and password entered by the ***Student***.
3. Display Financial Information: The ***SIS*** displays **Student**'s financial summary information
4. Use Case Ends: The use case ends.

f) Alternative Flows:

1. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, and if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 4***.

3.2 View account detail

a) Task description: This use case describes how a Student uses the system to view his financial information held by the University

b) Actors:

1. Student: The Student is registered on a course at the University and can view his financial information.

2. ***Student Information System (SIS)***: The system storing the student financial

information.

g) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

h) Postconditions: The transaction terminates normally while the financial detail information is displayed with no system failures.

i) Basic Flow of Events:

  1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
  2. Authenticate Student: The ***SIS*** will authenticate the user name and password entered by the ***Student***.
  3. Display Financial Information: The ***SIS***displays detail descriptions and charged amount for each payment.
  4. Use Case Ends: The use case ends.

j) Alternative Flows:

  1. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, and if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 4***.

3.3 View financial aid

a) Task description: This use case describes how a Student uses the system to view the personal information held by the University and submit a request to change the personal information.

b) Actors:

   1. ***Student***: The Student is registered on a course at the University and can view personal information about the Student held by the University.
   2. ***Student Information System (SIS)***: The system storing the student personal information.
   3. ***Service Administrator***: User of University MIS system who receives reports of system failures and can fix them.

c) Preconditions: The Student must be registered as a student with the University. The Student must have a username and password to access University systems.

d) Postconditions: The transaction terminates normally with no system failures.

e) Basic Flow of Events:

   1. Launch System: The use case begins when the actor ***Student*** launches the URL of the system in a web browser.
   2. Authenticate Student: The ***SIS*** will authenticate the user name a password entered by the ***Student***.
   3. Request Financial Aid Information: The ***Student*** sends a request asking the ***SIS*** to display financial aid information.
   4. Display Financial Aid Information: The ***SIS*** displays the financial aid information.
   5. Use Case Ends: The use case ends.

f) Alternative Flows:

   1. Handle communication failures: At ***step 3***, if the ***SIS*** cannot be contacted or does not reply within the set communication time-out period, and if the communication link has failed more times than the communication retry number, then communication is abandoned and the system informs the ***Student*** of the communication error. The ***SIS*** will be contacted until the number of retry attempts, indicated by the communication retry number, has completed. If communications are re-established, the basic flow is resumed at ***step 3***. If there is still no response

from the ***SIS***, an event log entry will be created to record the failure of the communications link to the ***SIS***. The event log entry includes the type of failure. The event log will be sent to the ***Service Administrator*** to inform them that communication with the ***SIS*** has been lost. Resume the basic flow at ***step 4***.

3) Use case and Scenarios

The following section describes the typical use case and scenarios for this system.

1. Use Case Diagram

The use case for this system is illustrated in figure 3. This was derived from the tasks listed in section 2.

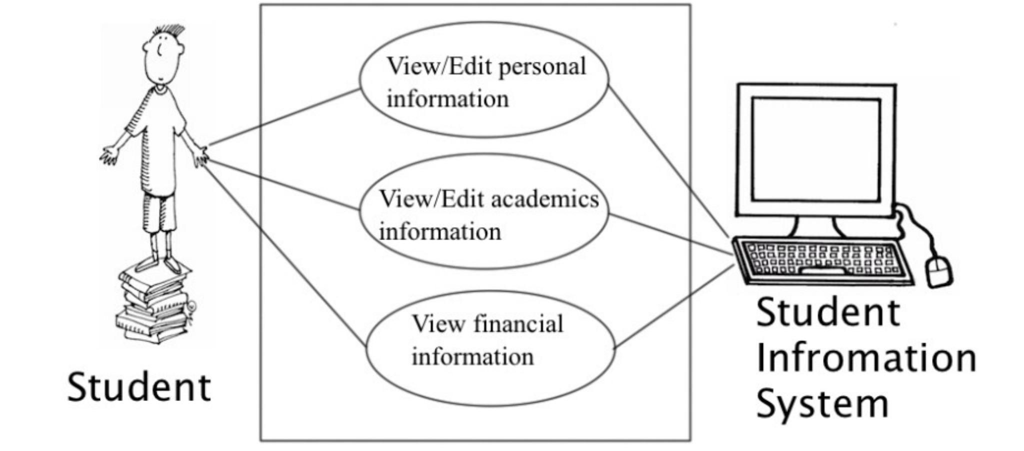
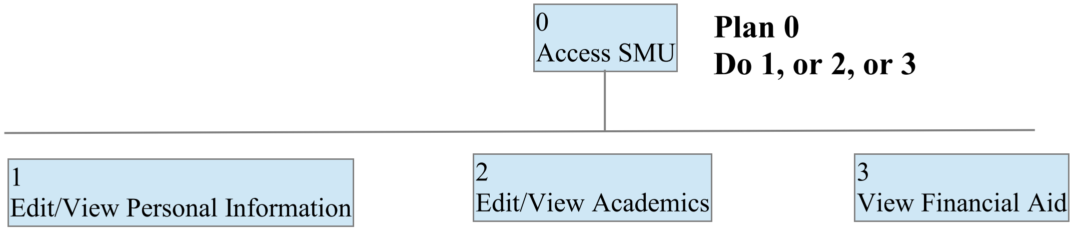

View/Edit personal information
View/Edit academics information
View financial information
Student
Student Infromation System

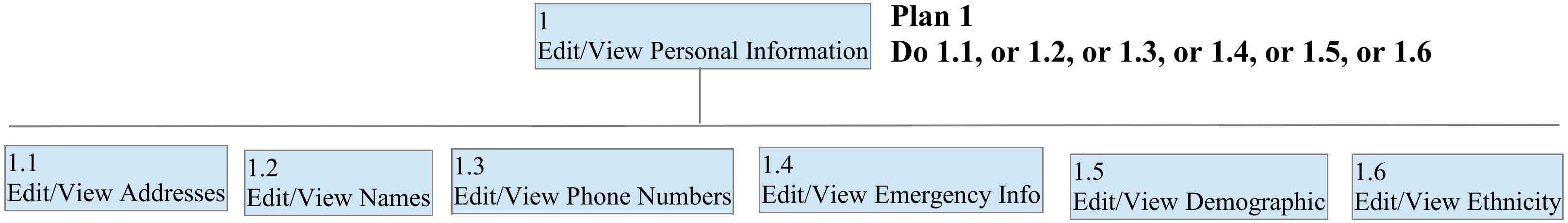
Figure 3

4) Hierarchical Task Analysis

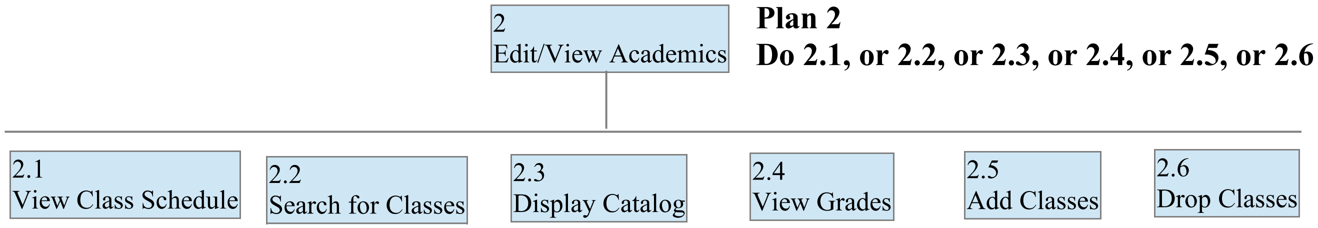
Because this project includes three main tasks: “View/Edit personal information”, “View/Edit Academics Information” and “View Financial Information”, three hierarchical task analyses are made for each main task



Plan 0
Do 1, or 2, or 3



Plan 1
Do 1.1, or 1.2, or 1.3, or 1.4, or 1.5, or 1.6



2
Edit/View Academics

Plan 2
Do 2.1, or 2.2, or 2.3, or 2.4, or 2.5, or 2.6

2.1
View Class Schedule

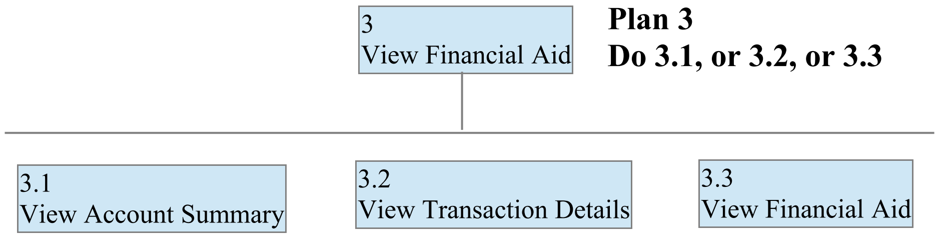
2.2
Search for Classes

2.3
Display Catalog

2.4
View Grades

2.5
Add Classes

2.6
Drop Classes



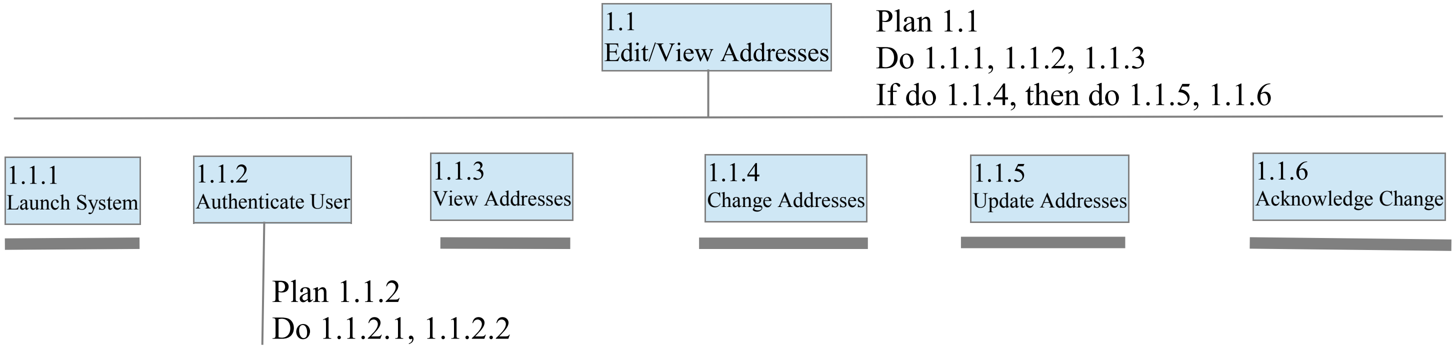
3
View Financial Aid

Plan 3
Do 3.1, or 3.2, or 3.3

3.1
View Account Summary

3.2
View Transaction Details

3.3
View Financial Aid



1.1
Edit/View Addresses

Plan 1.1
Do 1.1.1, 1.1.2, 1.1.3
If do 1.1.4, then do 1.1.5, 1.1.6

1.1.1
Launch System

1.1.2
Authenticate User

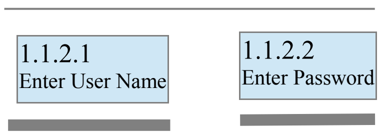
1.1.3
View Addresses

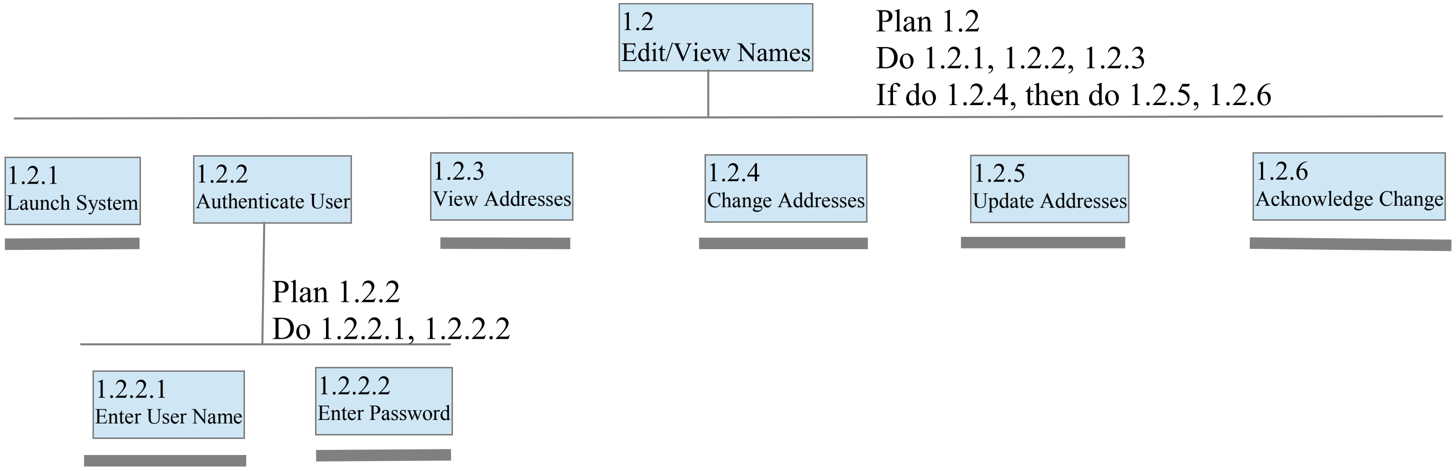
1.1.4
Change Addresses

1.1.5
Update Addresses

1.1.6
Acknowledge Change

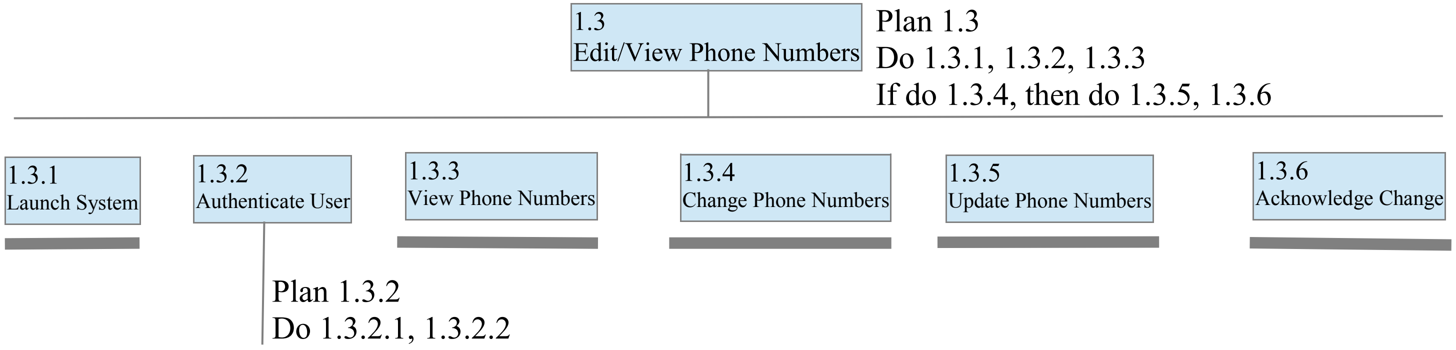
Plan 1.1.2
Do 1.1.2.1, 1.1.2.2





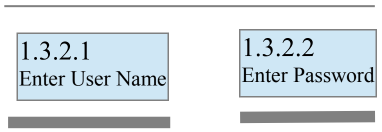
Plan 1.2
Do 1.2.1, 1.2.2, 1.2.3
If do 1.2.4, then do 1.2.5, 1.2.6

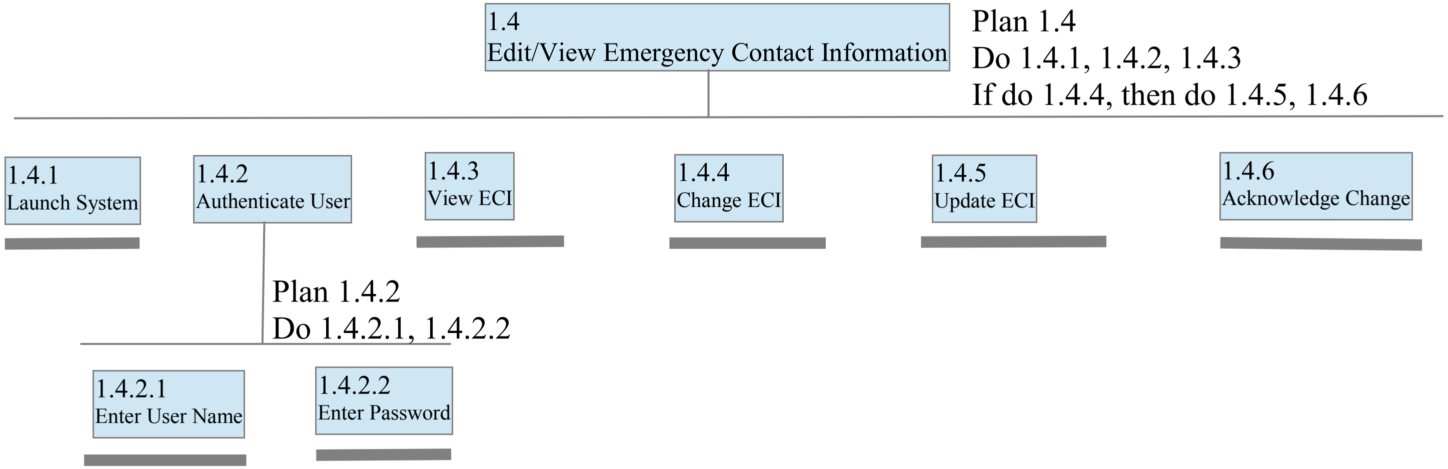
Plan 1.2.2
Do 1.2.2.1, 1.2.2.2

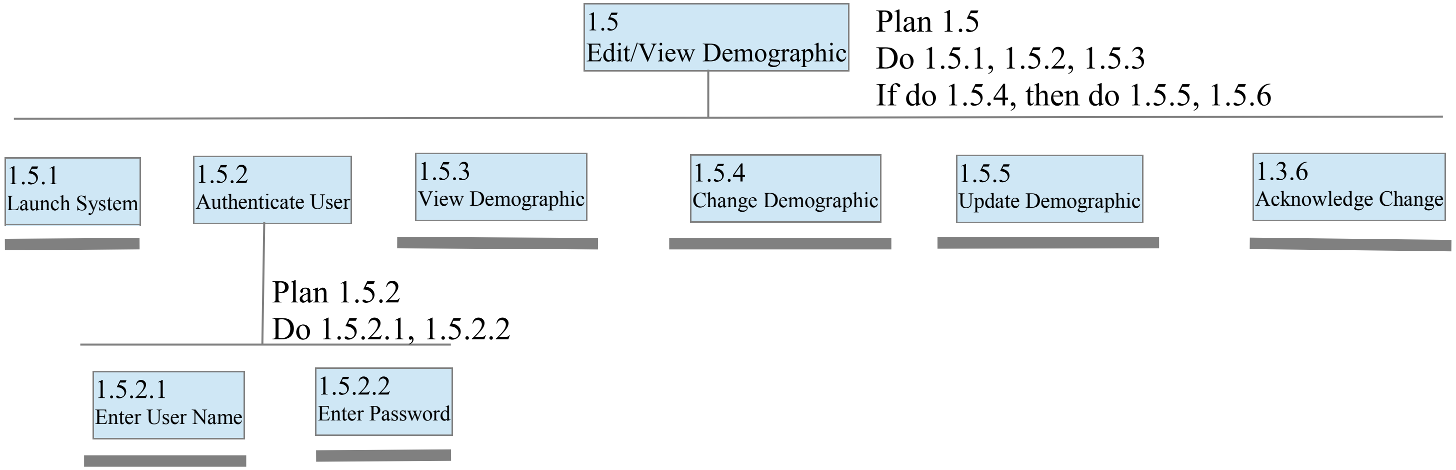


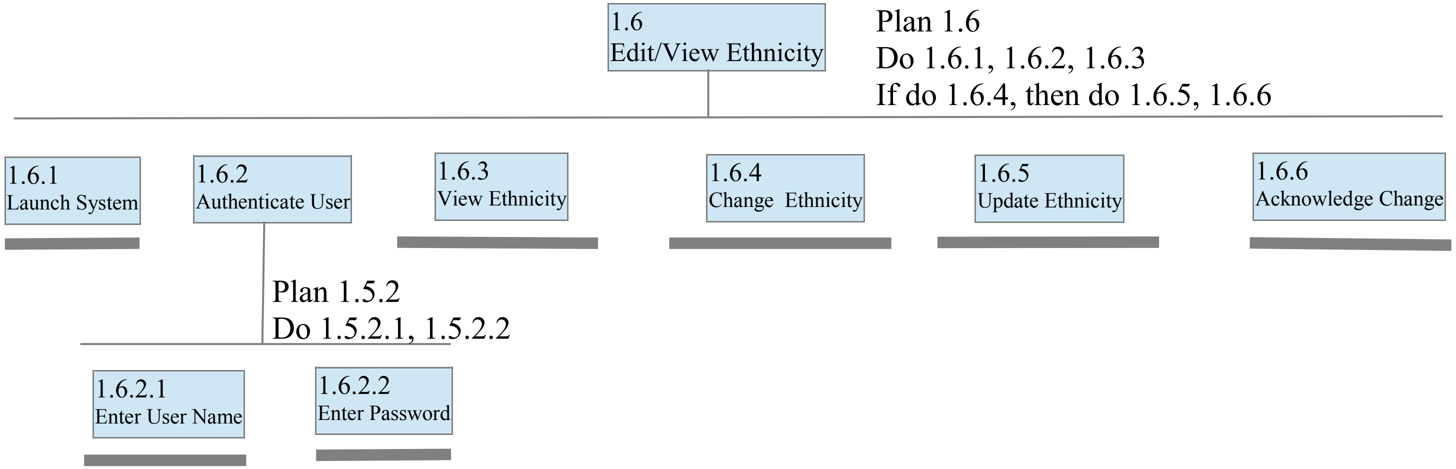
Plan 1.3
Do 1.3.1, 1.3.2, 1.3.3
If do 1.3.4, then do 1.3.5, 1.3.6

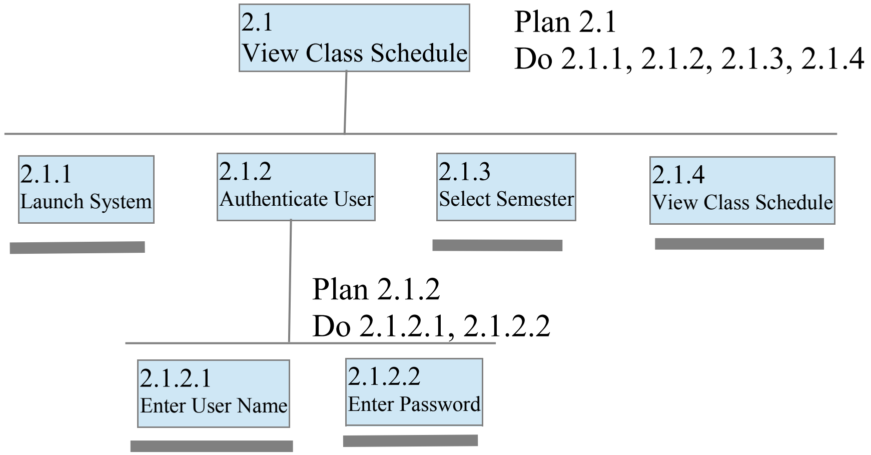
Plan 1.3.2
Do 1.3.2.1, 1.3.2.2

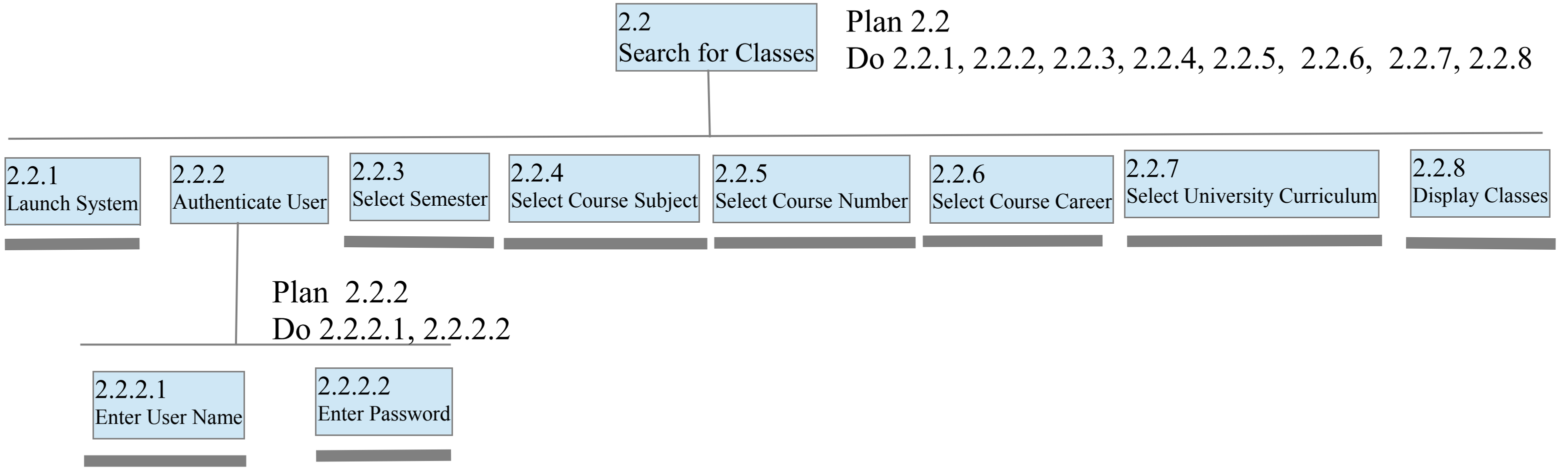

1.3.2.1
Enter User Name
1.3.2.2
Enter Password

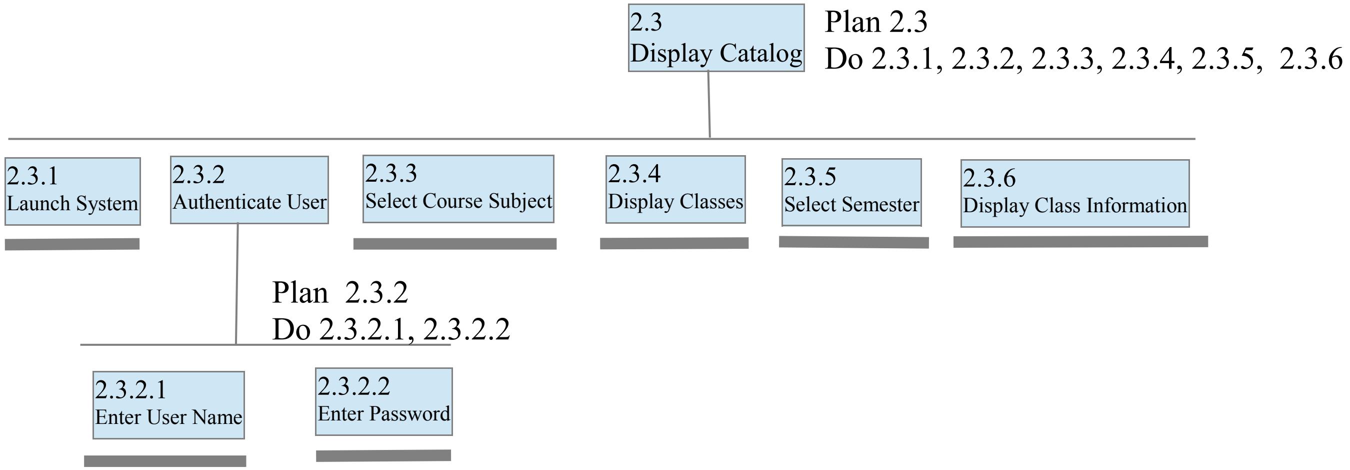

1.4
Edit/View Emergency Contact Information
Plan 1.4
Do 1.4.1, 1.4.2, 1.4.3
If do 1.4.4, then do 1.4.5, 1.4.6
1.4.1
Launch System
1.4.2
Authenticate User
1.4.3
View ECI
1.4.4
Change ECI
1.4.5
Update ECI
1.4.6
Acknowledge Change
Plan 1.4.2
Do 1.4.2.1, 1.4.2.2
1.4.2.1
Enter User Name
1.4.2.2
Enter Password

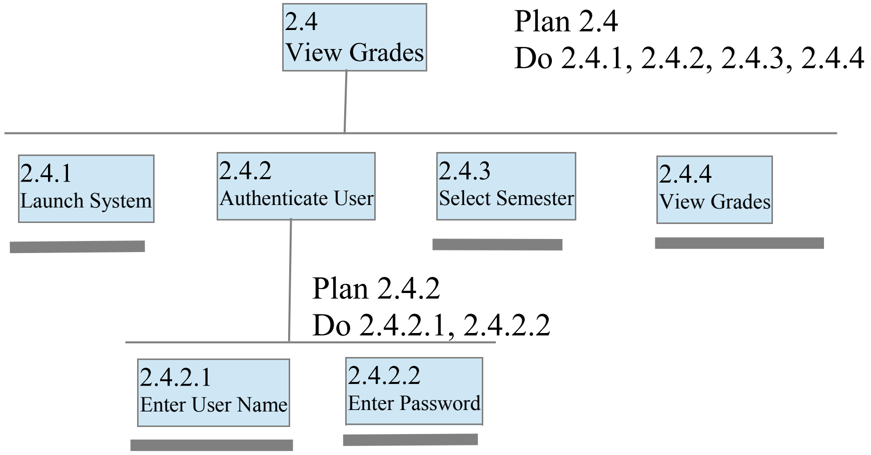

1.5
Edit/View Demographic
Plan 1.5
Do 1.5.1, 1.5.2, 1.5.3
If do 1.5.4, then do 1.5.5, 1.5.6
1.5.1
Launch System
1.5.2
Authenticate User
1.5.3
View Demographic
1.5.4
Change Demographic
1.5.5
Update Demographic
1.3.6
Acknowledge Change
Plan 1.5.2
Do 1.5.2.1, 1.5.2.2
1.5.2.1
Enter User Name
1.5.2.2
Enter Password

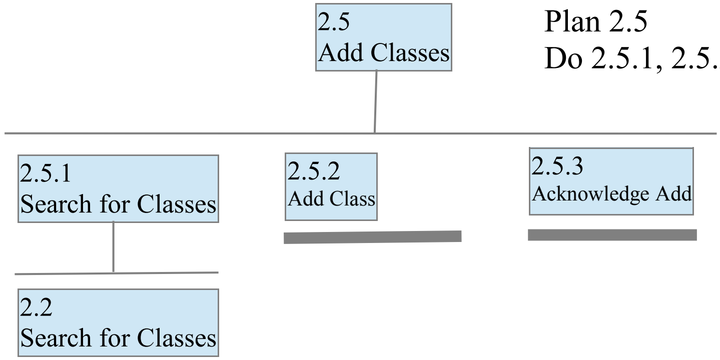

1.6
Edit/View Ethnicity
Plan 1.6
Do 1.6.1, 1.6.2, 1.6.3
If do 1.6.4, then do 1.6.5, 1.6.6
1.6.1
Launch System
1.6.2
Authenticate User
1.6.3
View Ethnicity
1.6.4
Change Ethnicity
1.6.5
Update Ethnicity
1.6.6
Acknowledge Change
Plan 1.5.2
Do 1.5.2.1, 1.5.2.2
1.6.2.1
Enter User Name
1.6.2.2
Enter Password


2.1
View Class Schedule
Plan 2.1
Do 2.1.1, 2.1.2, 2.1.3, 2.1.4
2.1.1
Launch System
2.1.2
Authenticate User
2.1.3
Select Semester
2.1.4
View Class Schedule
Plan 2.1.2
Do 2.1.2.1, 2.1.2.2
2.1.2.1
Enter User Name
2.1.2.2
Enter Password


2.2
Search for Classes
Plan 2.2
Do 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.2.5, 2.2.6, 2.2.7, 2.2.8
2.2.1
Launch System
2.2.2
Authenticate User
2.2.3
Select Semester
2.2.4
Select Course Subject
2.2.5
Select Course Number
2.2.6
Select Course Career
2.2.7
Select University Curriculum
2.2.8
Display Classes
Plan 2.2.2
Do 2.2.2.1, 2.2.2.2
2.2.2.1
Enter User Name
2.2.2.2
Enter Password


2.3
Display Catalog
Plan 2.3
Do 2.3.1, 2.3.2, 2.3.3, 2.3.4, 2.3.5, 2.3.6
2.3.1
Launch System
2.3.2
Authenticate User
2.3.3
Select Course Subject
2.3.4
Display Classes
2.3.5
Select Semester
2.3.6
Display Class Information
Plan 2.3.2
Do 2.3.2.1, 2.3.2.2
2.3.2.1
Enter User Name
2.3.2.2
Enter Password


2.4
View Grades
Plan 2.4
Do 2.4.1, 2.4.2, 2.4.3, 2.4.4
2.4.1
Launch System
2.4.2
Authenticate User
2.4.3
Select Semester
2.4.4
View Grades
Plan 2.4.2
Do 2.4.2.1, 2.4.2.2
2.4.2.1
Enter User Name
2.4.2.2
Enter Password



2.5
Add Classes

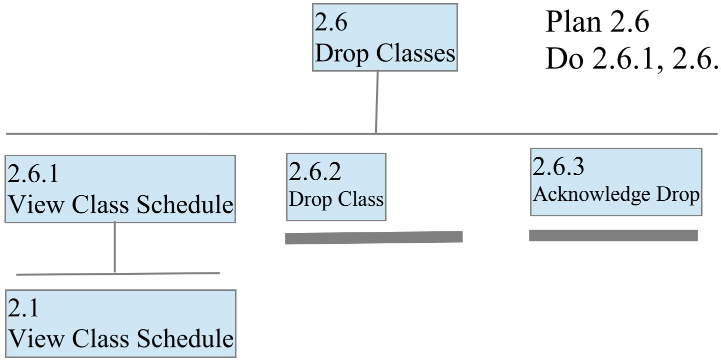
2.5.1
Search for Classes

2.5.2
Add Class

2.5.3
Acknowledge Add

2.2
Search for Classes

Plan 2.5
Do 2.5.1, 2.5.2, 2.5.3, 2.5.4



2.6
Drop Classes

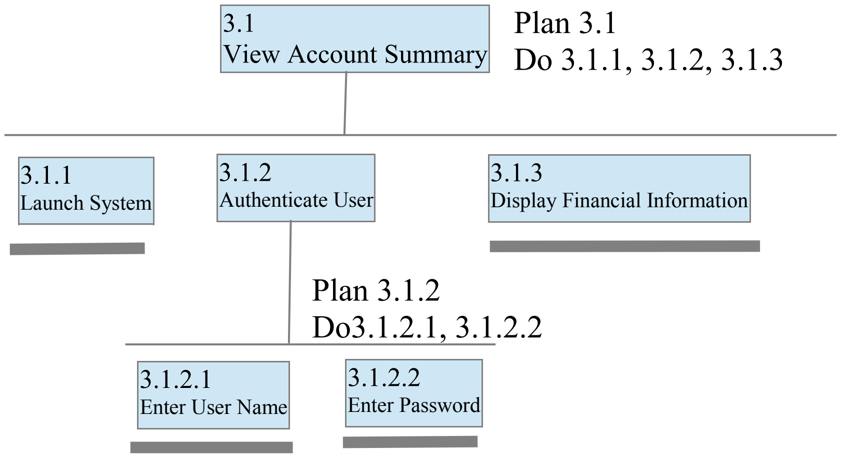
2.6.1
View Class Schedule

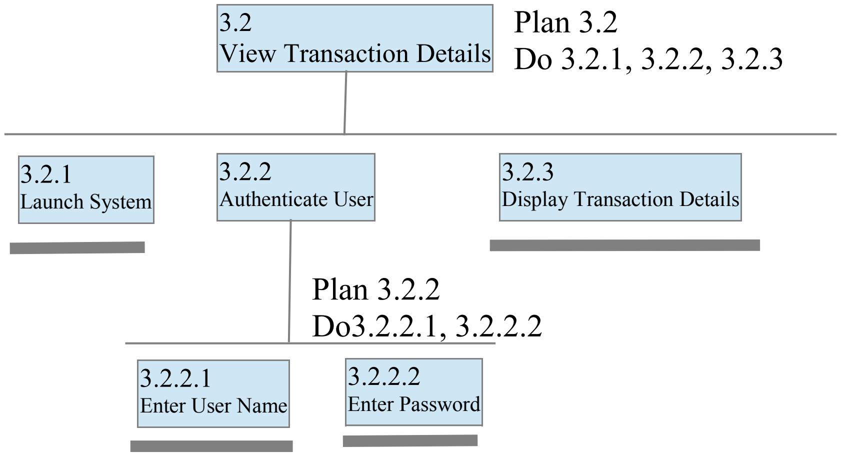
2.6.2
Drop Class

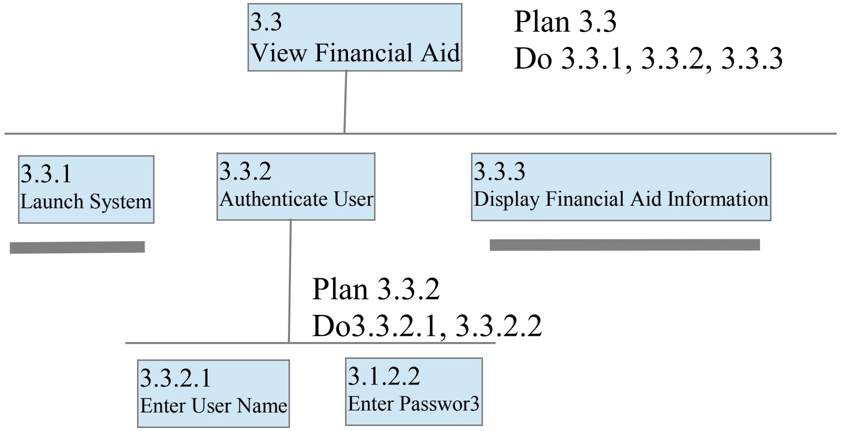
2.6.3
Acknowledge Drop

2.1
View Class Schedule

Plan 2.6
Do 2.6.1, 2.6.2, 2.6.3, 2.6.4

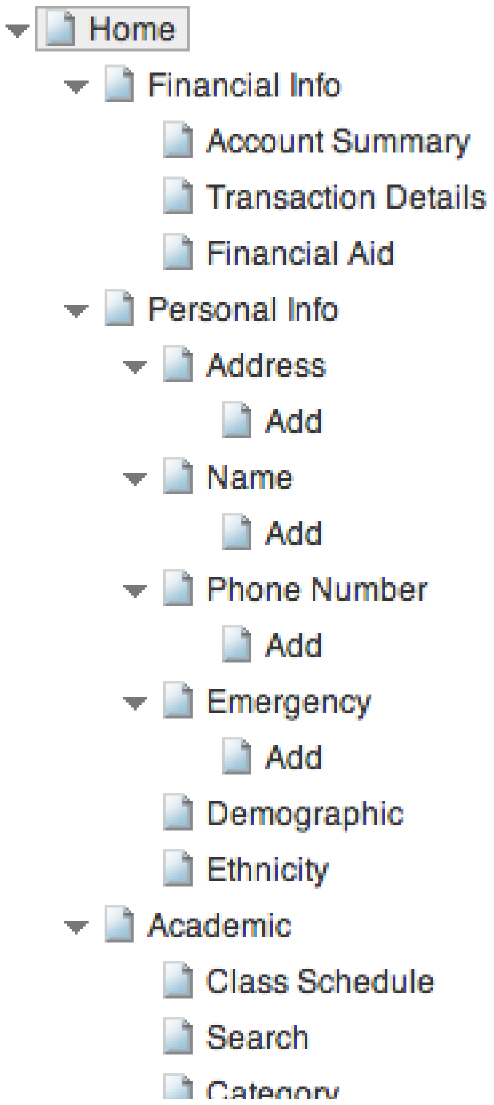

3.1
View Account Summary
Plan 3.1
Do 3.1.1, 3.1.2, 3.1.3
3.1.1
Launch System
3.1.2
Authenticate User
3.1.3
Display Financial Information
Plan 3.1.2
Do3.1.2.1, 3.1.2.2
3.1.2.1
Enter User Name
3.1.2.2
Enter Password


3.2
View Transaction Details
Plan 3.2
Do 3.2.1, 3.2.2, 3.2.3
3.2.1
Launch System
3.2.2
Authenticate User
3.2.3
Display Transaction Details
Plan 3.2.2
Do3.2.2.1, 3.2.2.2
3.2.2.1
Enter User Name
3.2.2.2
Enter Password


3.3
View Financial Aid
Plan 3.3
Do 3.3.1, 3.3.2, 3.3.3
3.3.1
Launch System
3.3.2
Authenticate User
3.3.3
Display Financial Aid Information
Plan 3.3.2
Do3.3.2.1, 3.3.2.2
3.3.2.1
Enter User Name
3.1.2.2
Enter Passwor3

III. Initial Design

An sitemap below give a overview ofthe whole website pages including three main part: "Financial Information", "Personal Information" and "Academic".

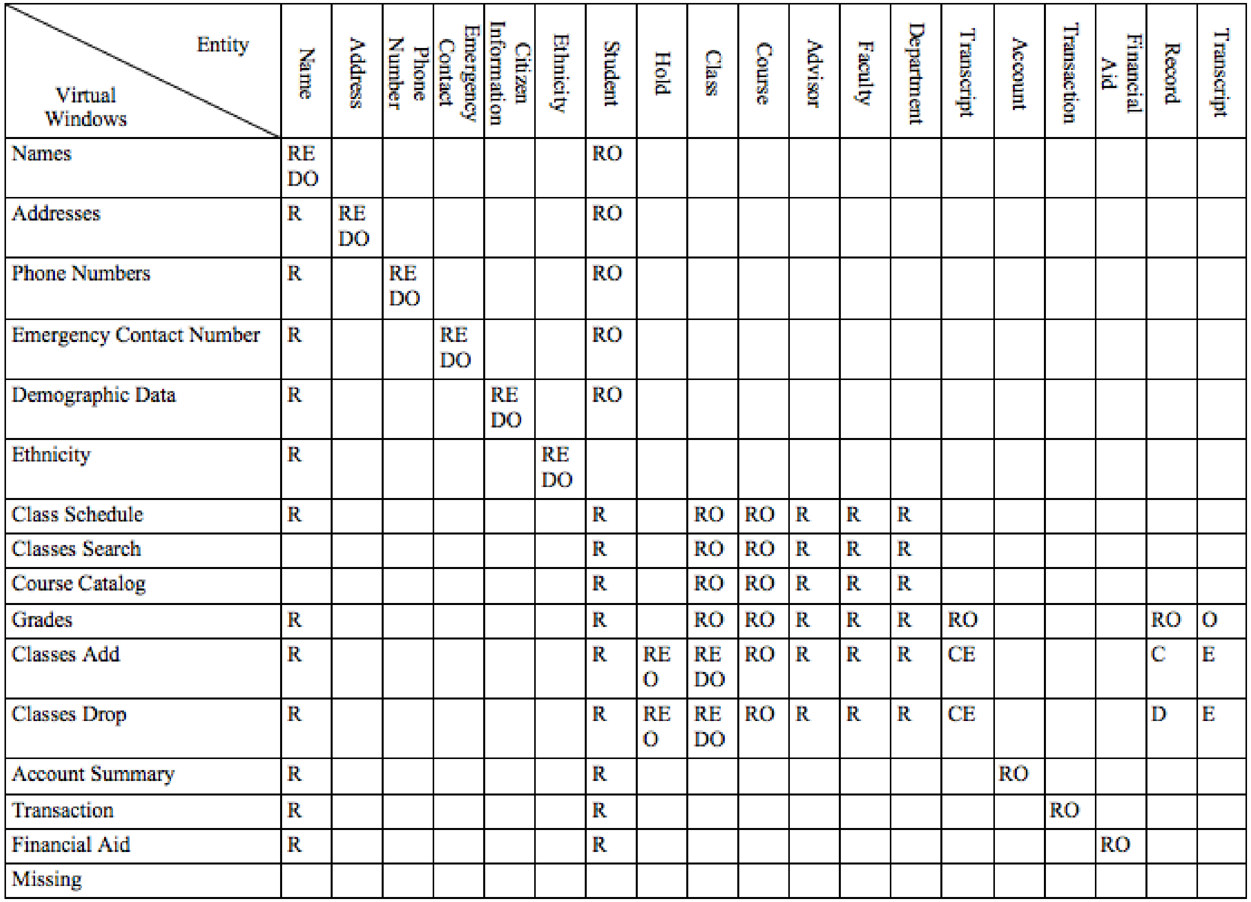


- Home
  - Financial Info
    - Account Summary
    - Transaction Details
    - Financial Aid
  - Personal Info
    - Address
      - Add
    - Name
      - Add
    - Phone Number
      - Add
    - Emergency
      - Add
    - Demographic
    - Ethnicity
  - Academic
    - Class Schedule
    - Search
    - Category

1) Credo Matrix

We can also check the internal consistency of our design. Here is an example: a check that all relevant data in the database can be handled by the user through some virtual window (assuming reasonable functions).

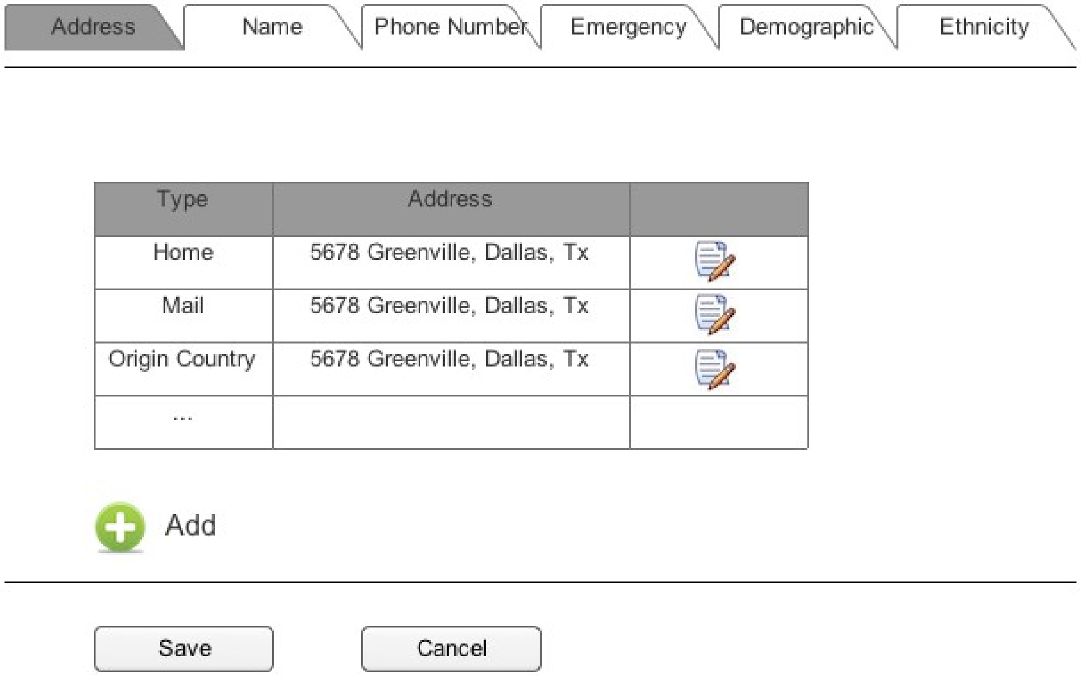
C means that the data can be Created, R read, E edited, D deleted, and O that the user gets an overview of this data. The last line shows what is missing.



Entity Virtual Windows	Name	Address	Phone Number	Emergency Contact	Citizen Information	Ethnicity	Student	Hold	Class	Course	Advisor	Faculty	Department	Transcript	Account	Transaction	Financial Aid	Record	Transcript
Names	RE DO						RO												
Addresses	R	RE DO					RO												
Phone Numbers	R		RE DO				RO												
Emergency Contact Number	R			RE DO			RO												
Demographic Data	R				RE DO		RO												
Ethnicity	R					RE DO													
Class Schedule	R						R		RO	RO	R	R	R						
Classes Search							R		RO	RO	R	R	R						
Course Catalog							R		RO	RO	R	R	R						
Grades	R						R		RO	RO	R	R	R	RO				RO	O
Classes Add	R						R	RE O	RE DO	RO	R	R	R	CE				C	E
Classes Drop	R						R	RE O	RE DO	RO	R	R	R	CE				D	E
Account Summary	R						R								RO				
Transaction	R						R									RO			
Financial Aid	R						R										RO		
Missing																			

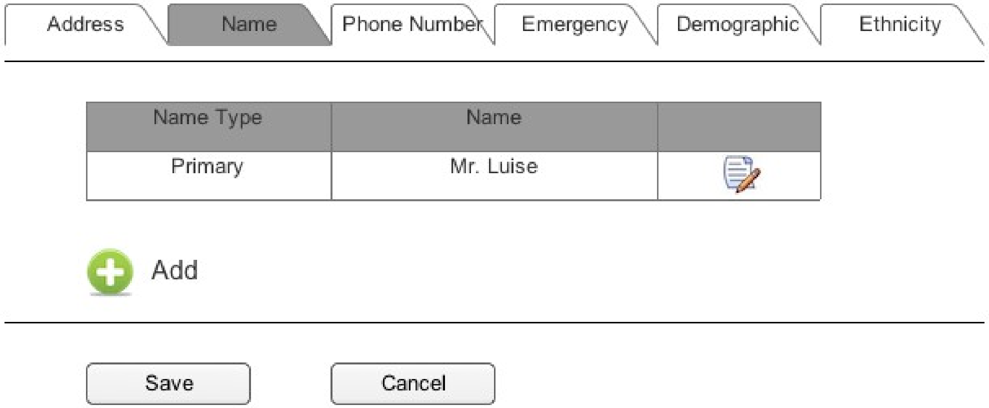
2) Low Fidelity Prototype

Edit/View Address



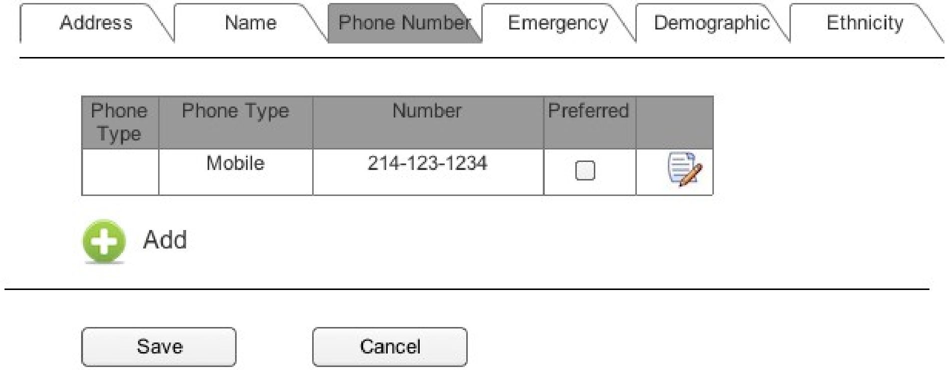
Type	Address	
Home	5678 Greenville, Dallas, Tx	✎
Mail	5678 Greenville, Dallas, Tx	✎
Origin Country	5678 Greenville, Dallas, Tx	✎
...		

Edit/View Name



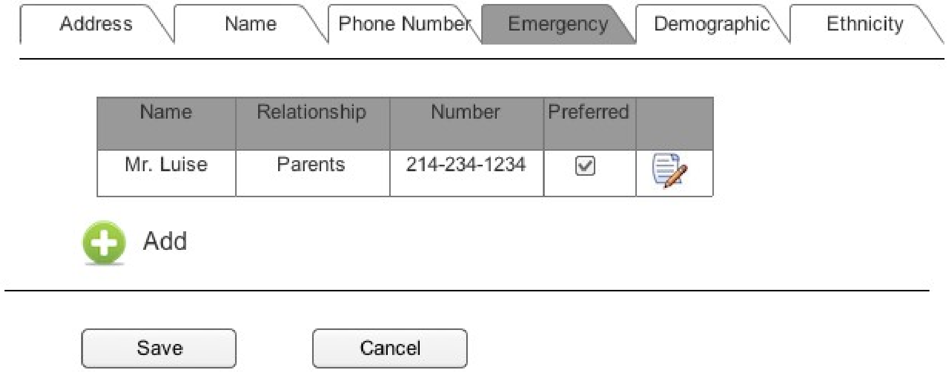
Name Type	Name	
Primary	Mr. Luise	✎

Edit/View Phone Numbers



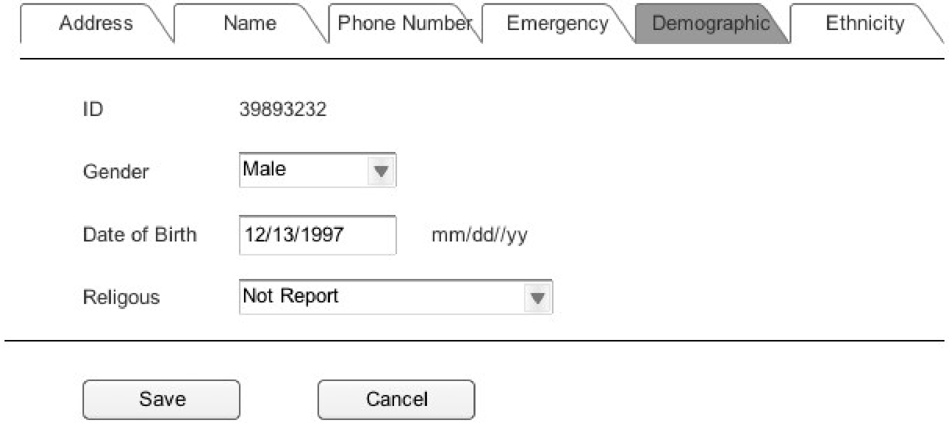
Phone Type	Phone Type	Number	Preferred	
	Mobile	214-123-1234	□	✎

Edit/View Emergency Contact Information



Name	Relationship	Number	Preferred	
Mr. Luise	Parents	214-234-1234	☑	✎

Edit/View Demographic Data



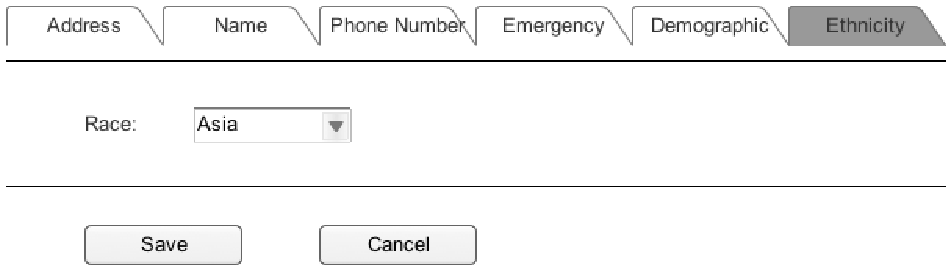
ID 39893232

Gender Male

Date of Birth 12/13/1997 mm/dd//yy

Religous Not Report

Edit/View Ethnicity



Race: Asia

4) Gestalts

Proximity
At Home page, all external links are put together at the leftmost column to allow users to access useful online resources.

Similarity
At Search page, the search fields are put together at left and the search results are put together in the table at right. The use of tabs is also another example how similarity looks like in the design of SMU.Access.

Continuity

At Search page, the label and the corresponding input field are put close to each other, which meet the definition of Gestalt continuity.

Closure
At Academic-Class Schedule page, the enrolled courses are grouped together in a bracket if their schedules are on the same date.

Symmetry
SMU.Access does not use Gestalt symmetry technique.

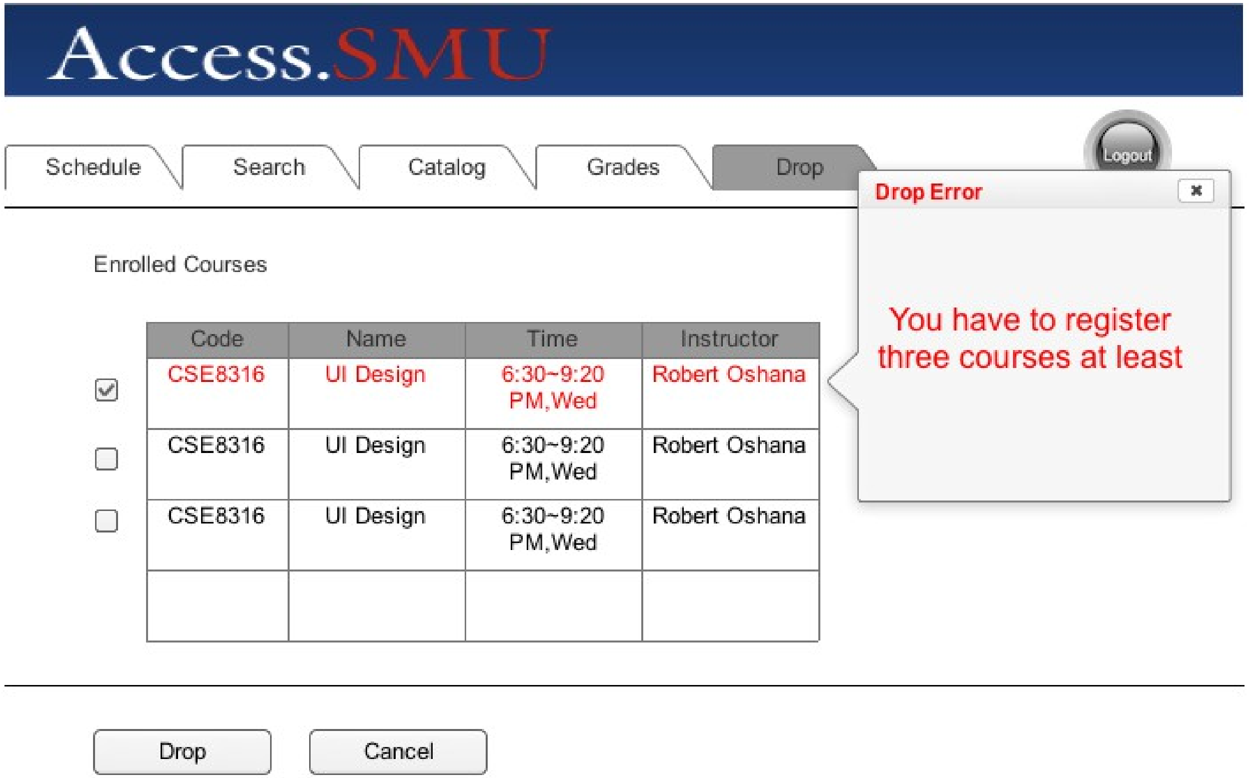
Figure/Ground
At Academic-Class Schedule page, while the course schedule has conflicts, the non-conflicted schedule will be greyed out and the conflicted one will stand out in a pop color.

Common Fate
No animation is used in the SMU.Access design; hence, no common fate is used.

5) Data Presentation

1. Use of Color

Color is used in the interface to draw attention to certain data. Specifically, we use red to represent an error operation on specific data row.



Enrolled Courses

	Code	Name	Time	Instructor
☑	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana
□	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana
□	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana

Drop Error

You have to register three courses at least

6) Usability Testing

1. Heuristic Evaluation

Visibility of System Status

Conformance Questions:

☐ Exact information provided at the exact moment
☐ Natural mapping between system and user concepts
☐ Appropriately direct attention
☐ Group related objects
☐ Alignment
☐ Balance and white space
☐ Few fonts and colors (5 to 7 colors max.)
☐ address color blind issues
☐ Too much information displayed.

Evidence of Conformance: all functions are clearly named and related objects are grouped together. Alignment, balance and white space are addressed appropriately. Fonts and colors are elegant and big enough to read with ease. Only the information requested is displayed. Website colors are used to address color blindness.

Match between the System and the Real World

Conformance Questions:
☐ Accommodate specific vocabulary for users
☐ View from user‘s perspective, not system.
☐ Adopt words as proposed by users
☐ Good mappings between conceptual model and user‘s mental model.
☐ Employ real-world reference system

Evidence of Conformance: The system uses language that is relative to its users. A weekly calendar and tables are used. “Back” command button to takes users back to the previous window. “Cancel” command button cancels the action. “Add” command button adds information.

Minimize User Memory Load

Conformance Questions:
☐ Use recognition, not recall.

☐ Exploit visibility.

☐ Provide implicit clues and rules for required input.

☐ Easy to navigate.

Evidence of Conformance: The user does not have to remember anything from a previous page. The user ids guided via the input stage by drop-down list boxes to choose from and by masked edit fields (e.g. date)

Consistency and Standards

Conformance Questions:

☐ Facilitates exploratory learning.

☐ Consistency of format.

☐ Consistency of task and functionality structure of system.

☐ Consistency of effects

☐ Consistency of language and graphics

Evidence of Conformance: The interface uses the same terminology for all action types and objects. For instance, on all windows that allows you to make changes to data in the system or add data the word "cancel" is used as the action to do so. Also, the tab theme is kept where you can see a tab for each subtask under any of the three main tasks. "Cancel" command button always takes the user back from the subtask window to the main task window at any subtask window.

Feedback

Conformance Questions:

☐ Continuous information on user‘s actions and their interpretation, including positive and partial feedback

☐ Avoid abstract, generalized feedback.

☐ Various levels of persistence of feedback information.

☐ System Failure

Evidence of Conformance: The grayed tab always shows the user where he is at relative to the other subtasks and the main tasks. Command buttons are all consistent, clear, and helpful.

Clearly Marked Exits

Conformance Questions:

□ Empower users and promote exploratory learning
□ Strategies:
□ Cancel
□ Universal Undo
□ Quit
□ Exits visible

Evidence of Conformance: The cancel command button always gives the user a clean exit from the subtask window to the subtask main window. If some windows do not have a "cancel" command button they will have a "back" command button to take them back to the previous window.

Shortcuts

Conformance Questions:

□ Support expert use beyond novice.
□ Accelerators, templates (styles), macros.
□ Type-ahead
□ Reuse of interaction history.
□ Bookmarks, Explorer shortcuts.
□ Judicious use of default values.

Evidence of Conformance: Some data is automatically retrieved for the user. Before the user edits any data, they can see what the old data was. The user always can hit the Home button from any window and go back to the website main window.

Good Error Messages

Conformance Questions:

- □ Opportunity for system to assist and educate user.
- □ Phrase messages in clear language.
- □ Precise rather than vague or general.
- □ Help in the resolution of the problem.
- □ Polite, non-intimidating messages.
- □ Provide good recovery.
- □ Multiple-level messages.

Evidence of Conformance: Recovery is done in the form of going back to the main window. Drop-down list oxes limit the user errors. Error messages are informative, clear, polite, and precise. Error message do not consist of error codes , debug information, or system data.

Error Prevention

Conformance Questions:

□ 'Design out' errors

□ Recognition, not recall. (Select from list.)

□ Use 'Confirmation' boxes sparingly.

Evidence of Conformance:Only relevant data boxes can be modified, they are grayed to stand out. All others are disabled so the user can not enter any value and wont be distracted. Drop-down list boxes are used to limit the user input to only what is displayed. Only generalized description of the error is displayed and possible ways of fixing it.

Help and Documentation

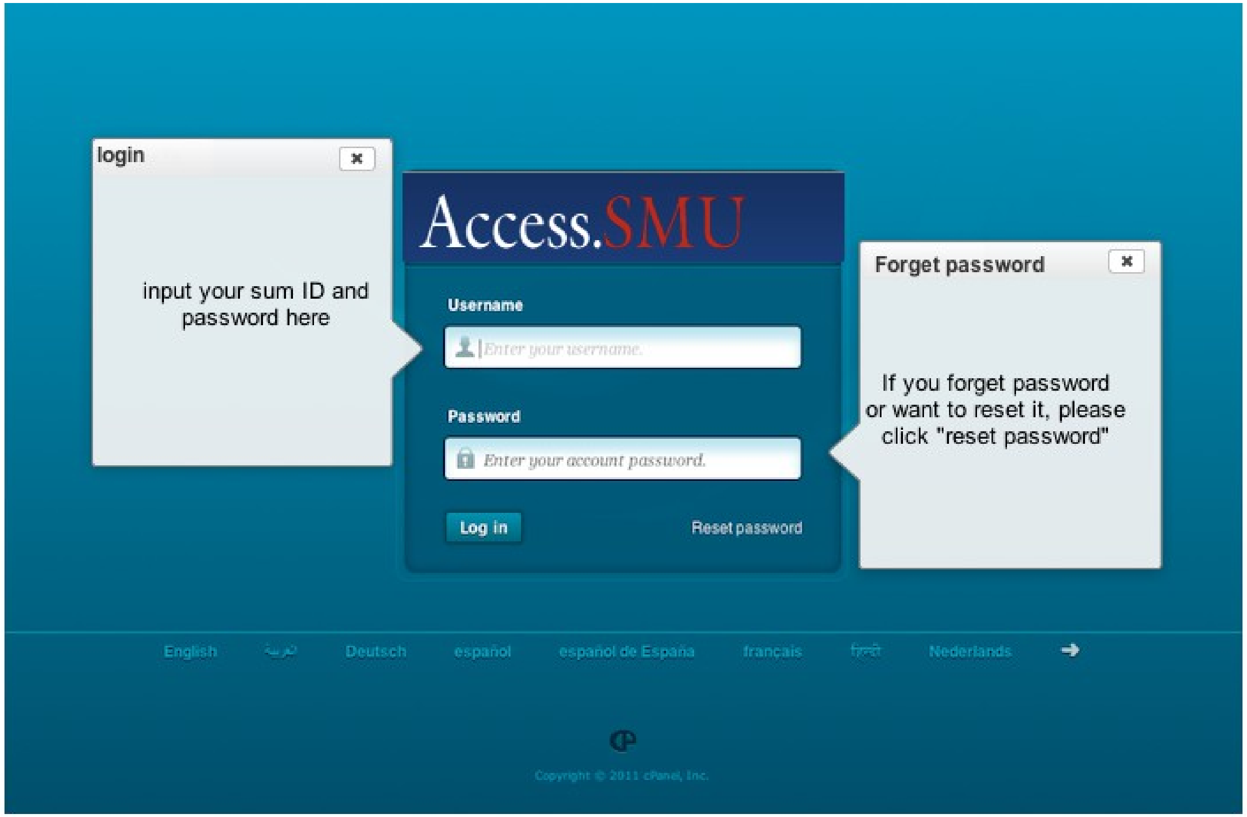
Conformance Questions:

□ Not for improving a poor interface.

Evidence of Conformance: The functionality is not handled at this phase.

IV. Detailed Prototype

1. Function Design

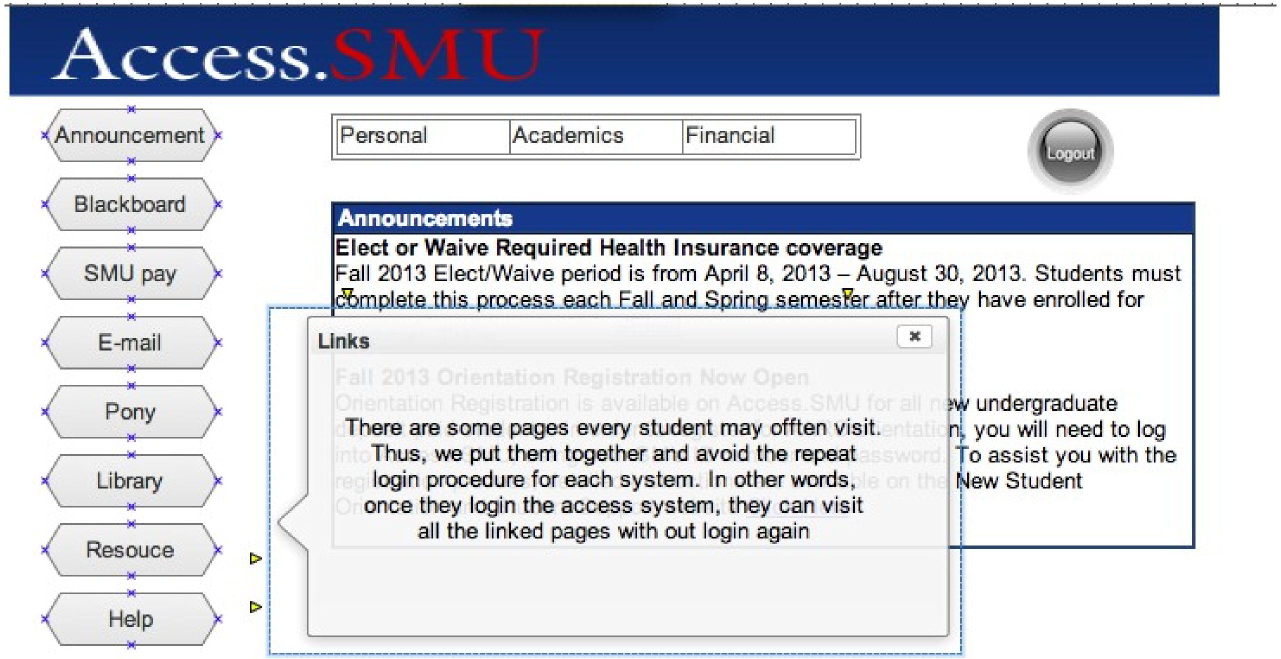


login

input your sum ID and password here

Forget password

If you forget password or want to reset it, please click "reset password"



Elect or Waive Required Health Insurance coverage

Fall 2013 Elect/Waive period is from April 8, 2013 – August 30, 2013. Students must complete this process each Fall and Spring semester after they have enrolled for

Fall 2013 Orientation Registration Now Open

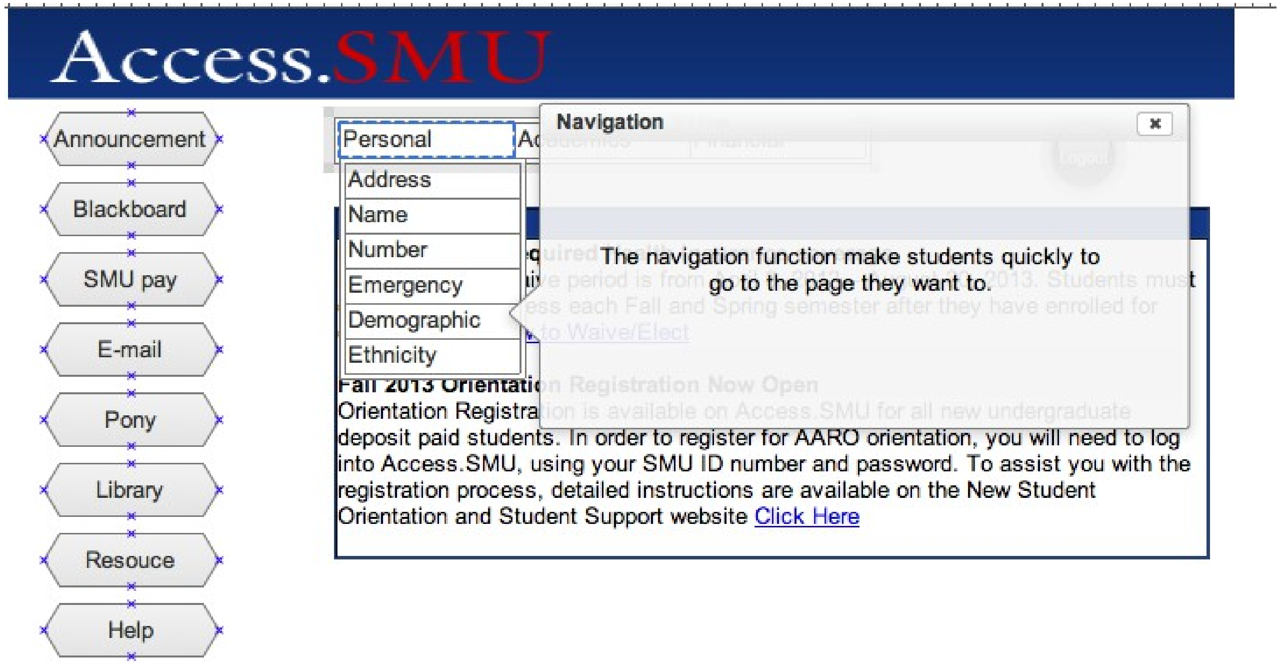
Orientation Registration is available on Access.SMU for all new undergraduate

entation, you will need to log

into assword. To assist you with the

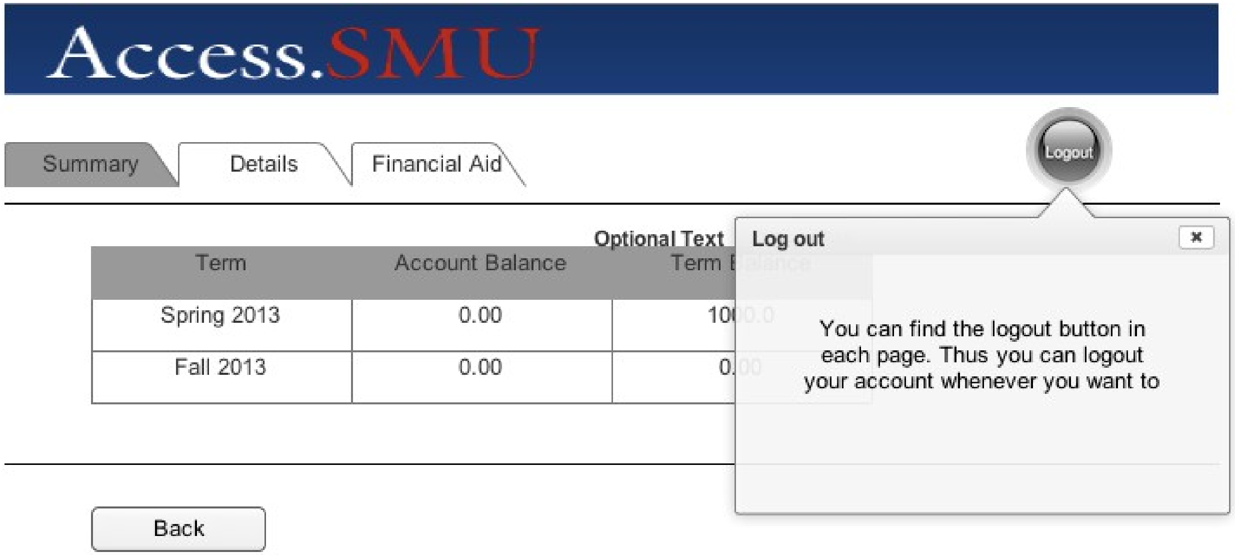
regi ble on the New Student

There are some pages every student may offten visit. Thus, we put them together and avoid the repeat login procedure for each system. In other words, once they login the access system, they can visit all the linked pages with out login again



The navigation function make students quickly to go to the page they want to.

Fall 2013 Orientation Registration Now Open
Orientation Registration is available on Access.SMU for all new undergraduate deposit paid students. In order to register for AARO orientation, you will need to log into Access.SMU, using your SMU ID number and password. To assist you with the registration process, detailed instructions are available on the New Student Orientation and Student Support website Click Here

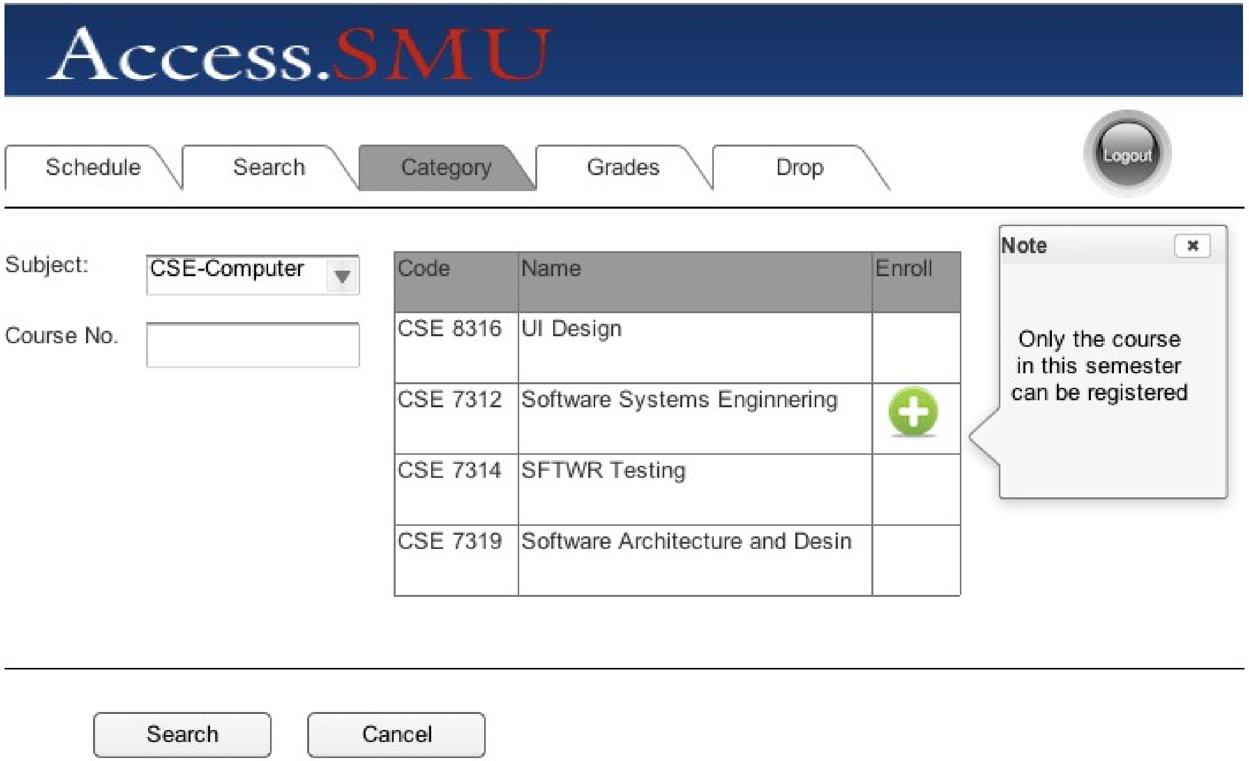


Optional Text

Term	Account Balance	Term Balance
Spring 2013	0.00	1000.0
Fall 2013	0.00	0.00

Log out

You can find the logout button in each page. Thus you can logout your account whenever you want to



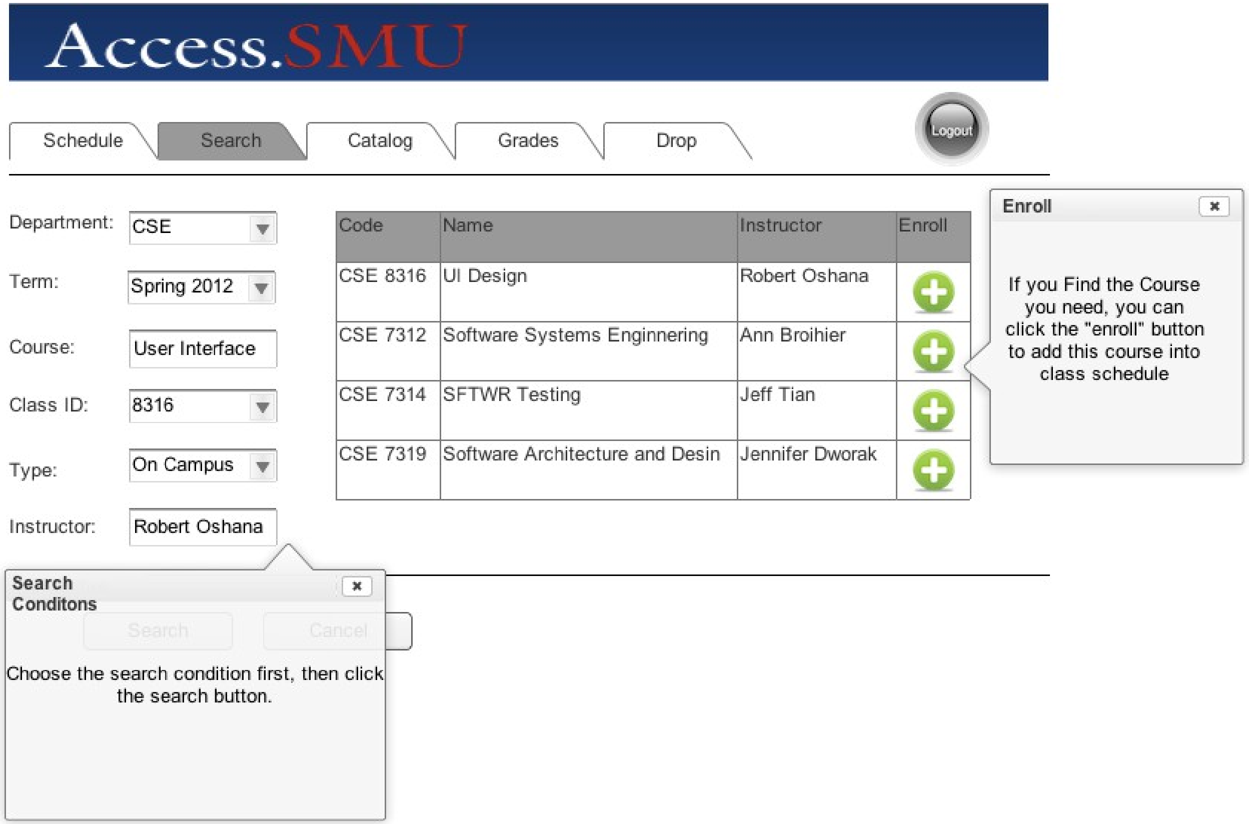
Subject: CSE-Computer

Course No.

Code	Name	Enroll
CSE 8316	UI Design	
CSE 7312	Software Systems Enginnering	⊕
CSE 7314	SFTWR Testing	
CSE 7319	Software Architecture and Desin	

Note

Only the course in this semester can be registered



Department:	CSE
Term:	Spring 2012
Course:	User Interface
Class ID:	8316
Type:	On Campus
Instructor:	Robert Oshana

Code	Name	Instructor	Enroll
CSE 8316	UI Design	Robert Oshana	⊕
CSE 7312	Software Systems Enginnering	Ann Broihier	⊕
CSE 7314	SFTWR Testing	Jeff Tian	⊕
CSE 7319	Software Architecture and Desin	Jennifer Dworak	⊕

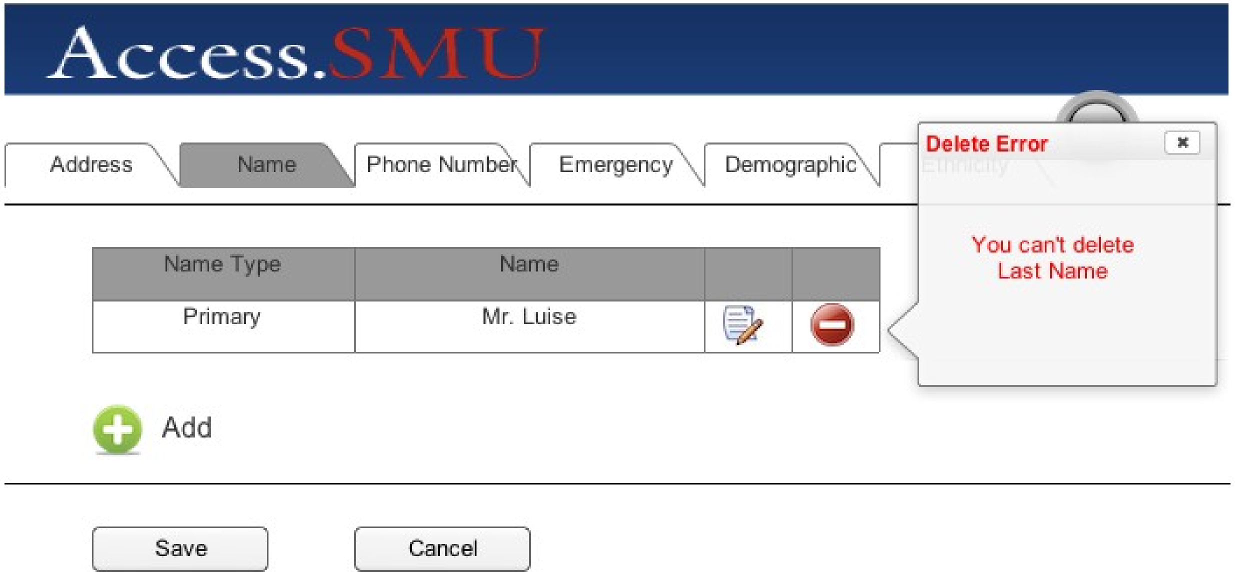
Enroll

If you Find the Course you need, you can click the "enroll" button to add this course into class schedule

Search Conditons

Choose the search condition first, then click the search button.

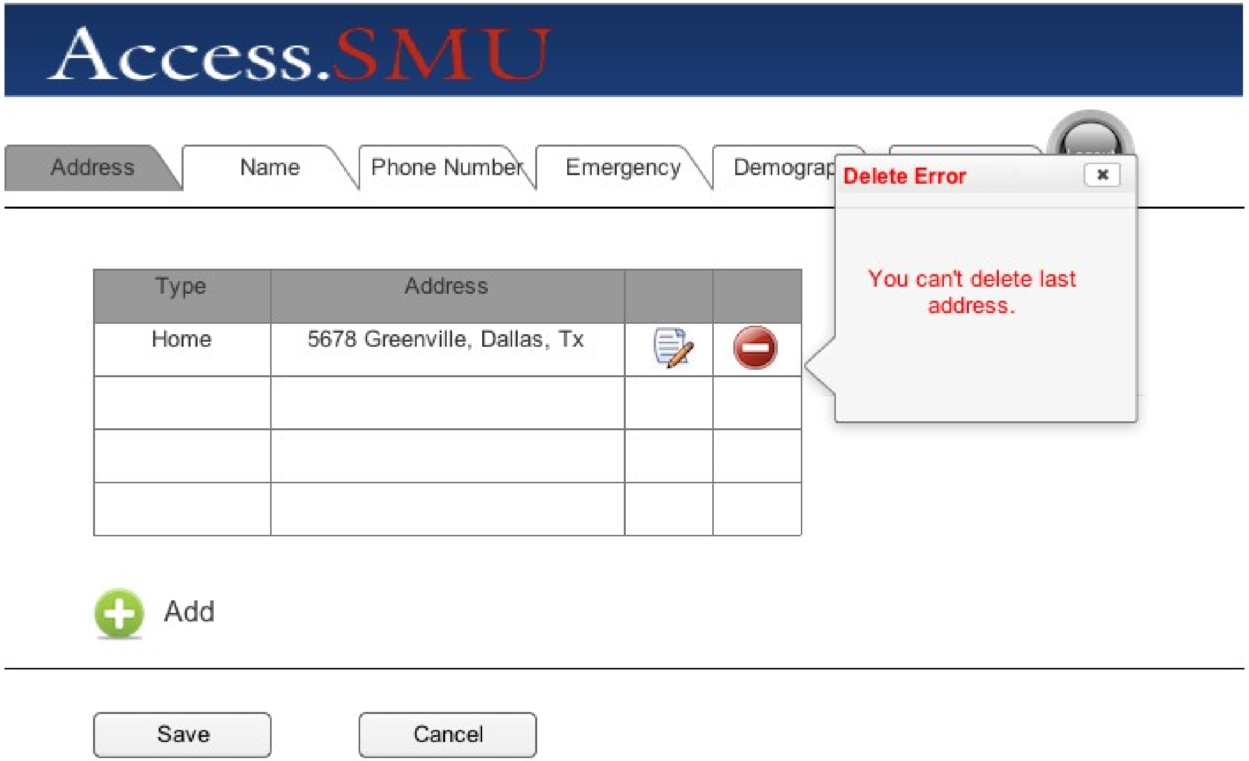
2. Error Message



Name Type	Name		
Primary	Mr. Luise	✎	⊖

Delete Error

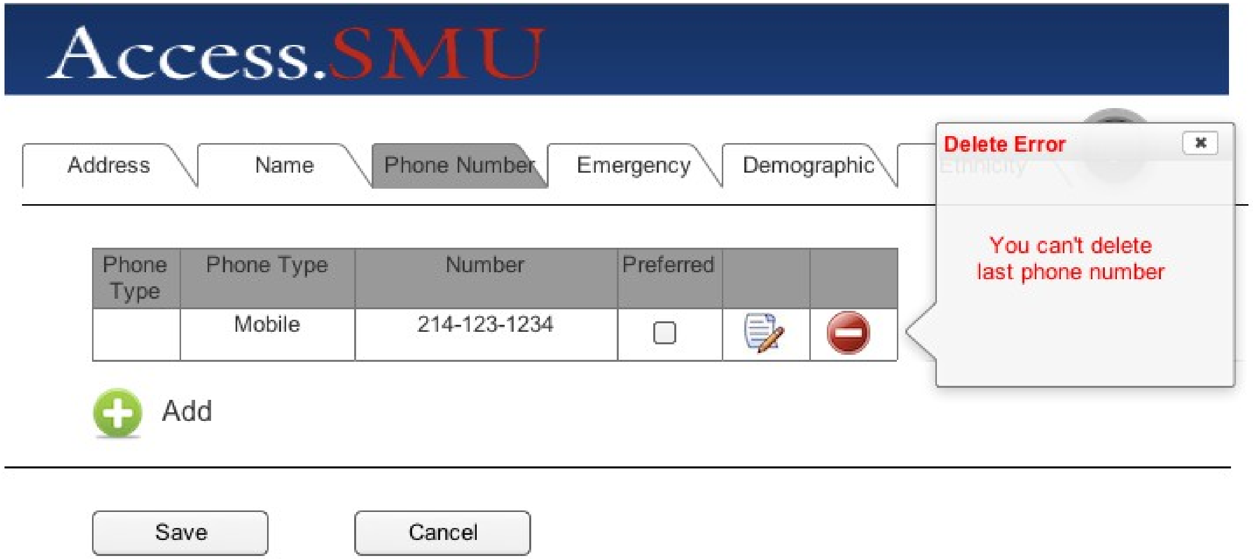
You can't delete
Last Name



Type	Address		
Home	5678 Greenville, Dallas, Tx	✎	⊖

Delete Error ×

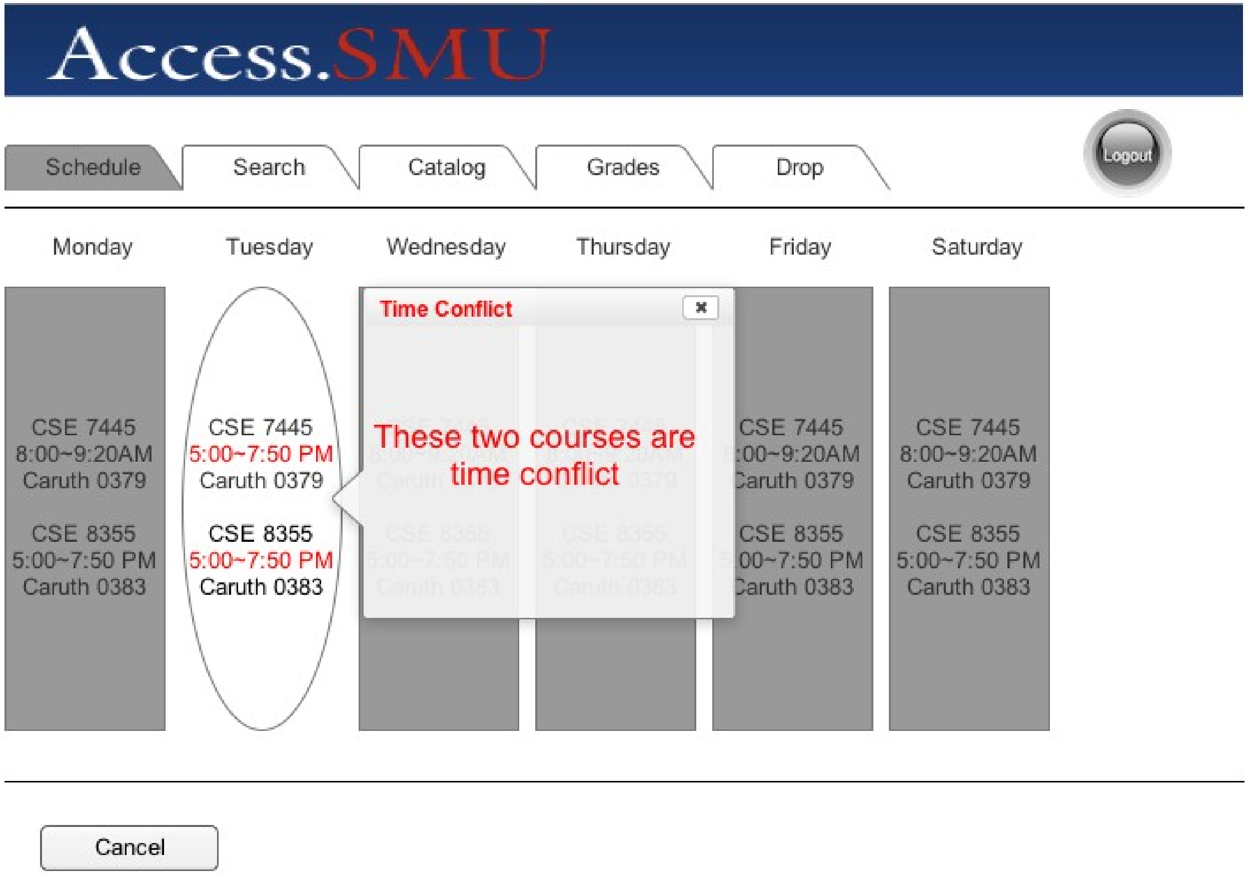
You can't delete last address.



Phone Type	Phone Type	Number	Preferred		
	Mobile	214-123-1234	□		

Delete Error

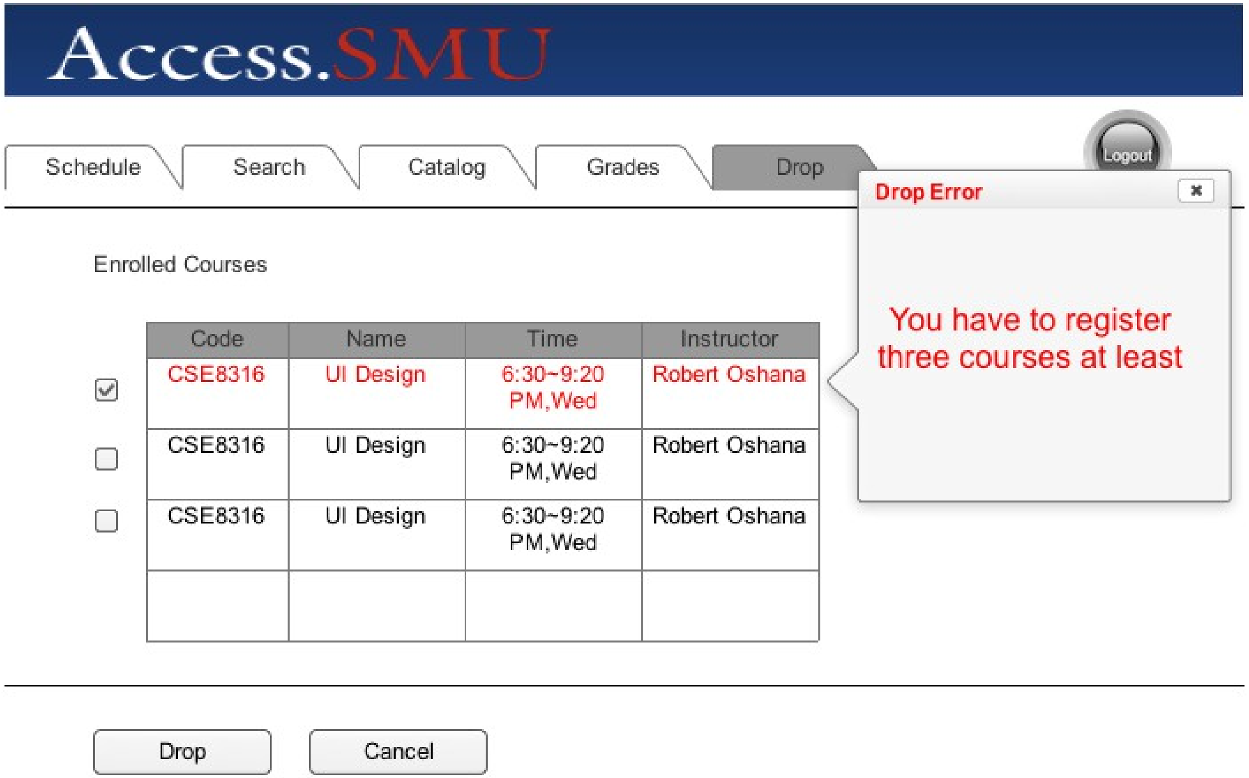
You can't delete
last phone number



Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 5:00~7:50 PM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383

Time Conflict ✖

These two courses are time conflict



Enrolled Courses

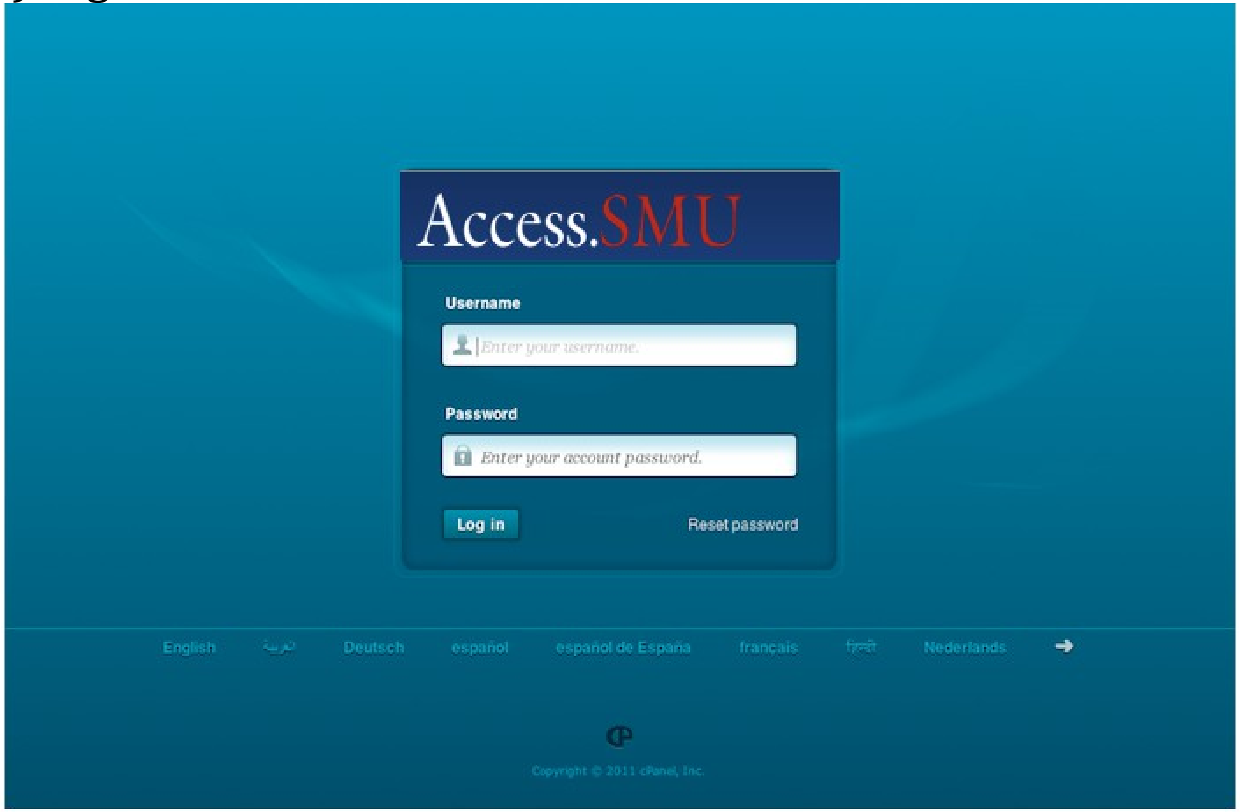
	Code	Name	Time	Instructor
☑	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana
□	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana
□	CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana

Drop Error ✖

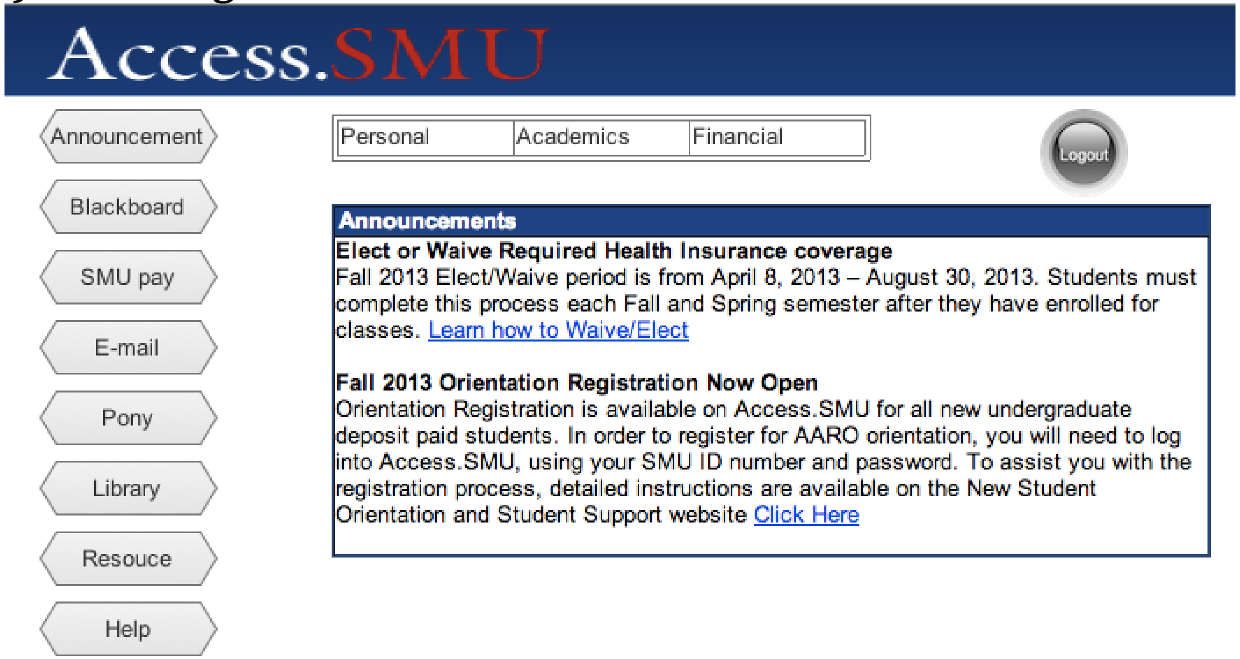
You have to register three courses at least

3. Prototype

1) Login Interface



2) Home Page



Announcements

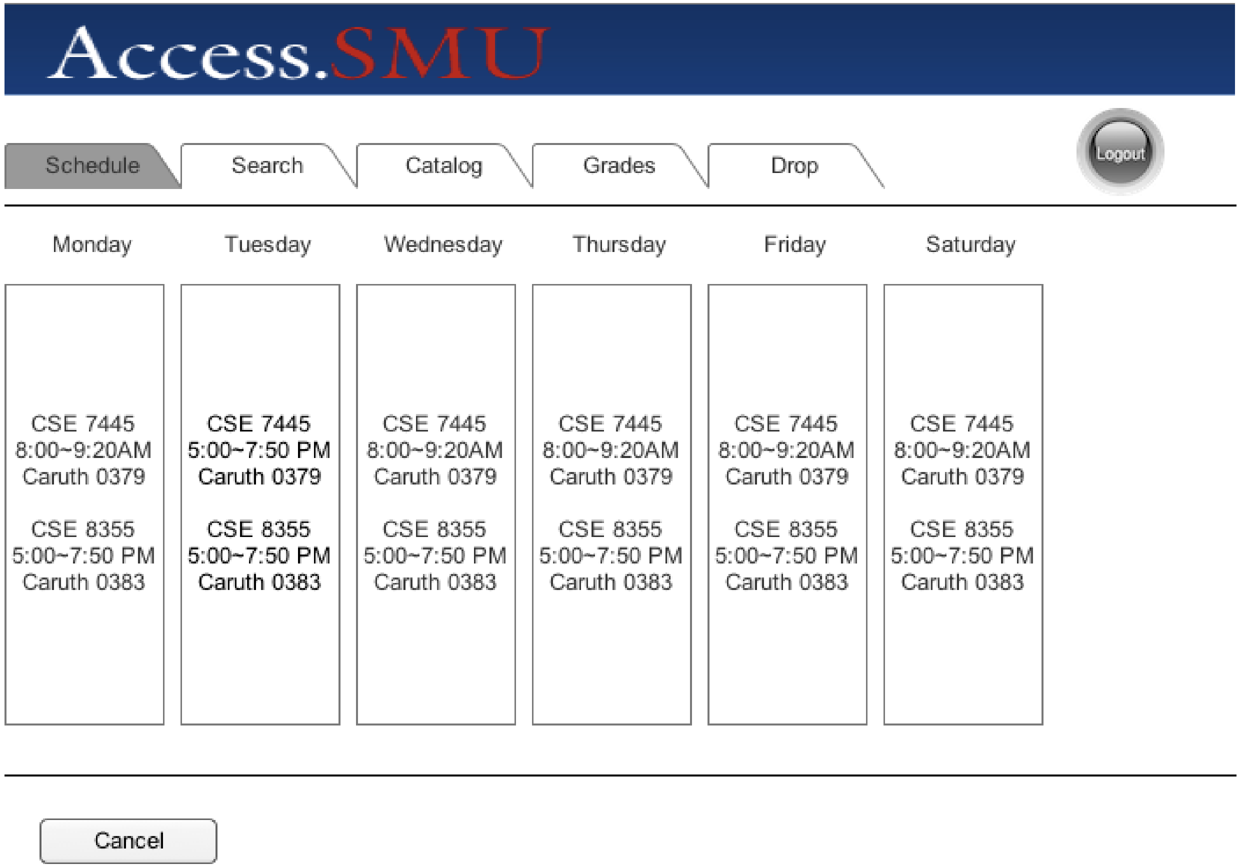
Elect or Waive Required Health Insurance coverage

Fall 2013 Elect/Waive period is from April 8, 2013 – August 30, 2013. Students must complete this process each Fall and Spring semester after they have enrolled for classes. Learn how to Waive/Elect

Fall 2013 Orientation Registration Now Open

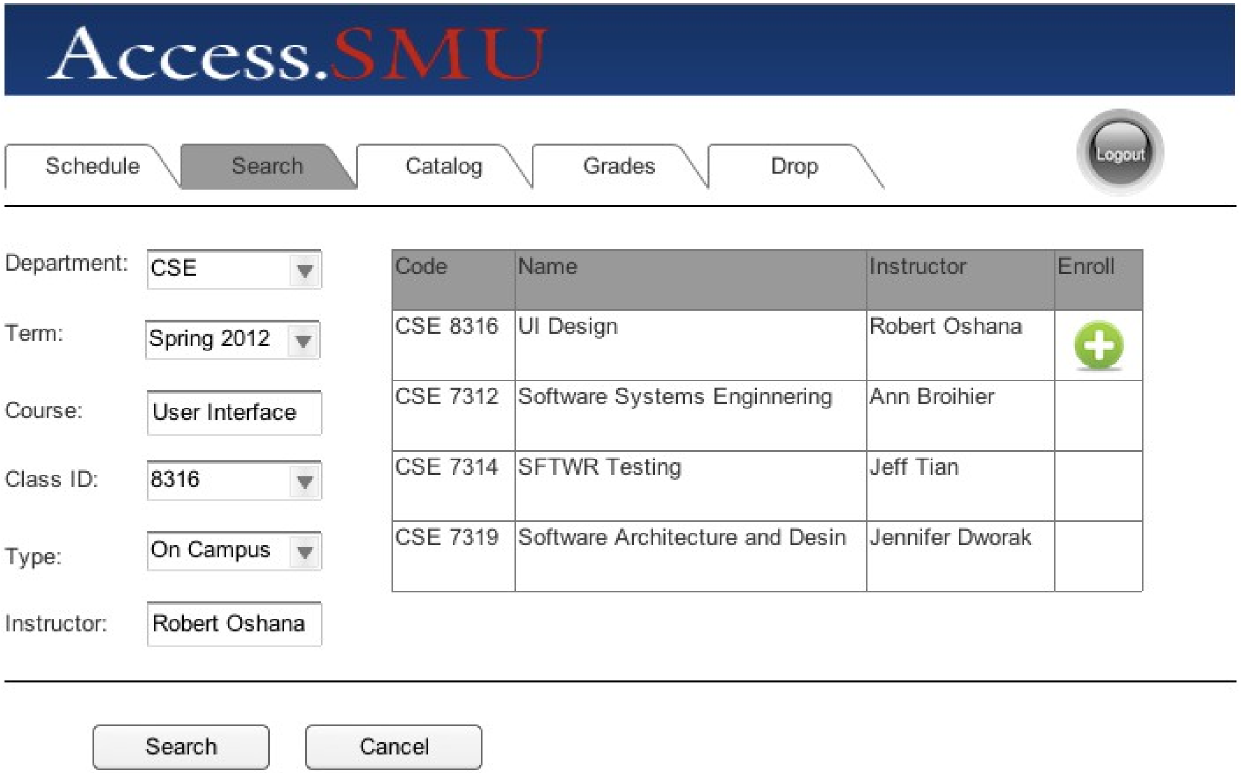
Orientation Registration is available on Access.SMU for all new undergraduate deposit paid students. In order to register for AARO orientation, you will need to log into Access.SMU, using your SMU ID number and password. To assist you with the registration process, detailed instructions are available on the New Student Orientation and Student Support website Click Here

3) Academic -Class Schedule



Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 5:00~7:50 PM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383	CSE 7445 8:00~9:20AM Caruth 0379 CSE 8355 5:00~7:50 PM Caruth 0383

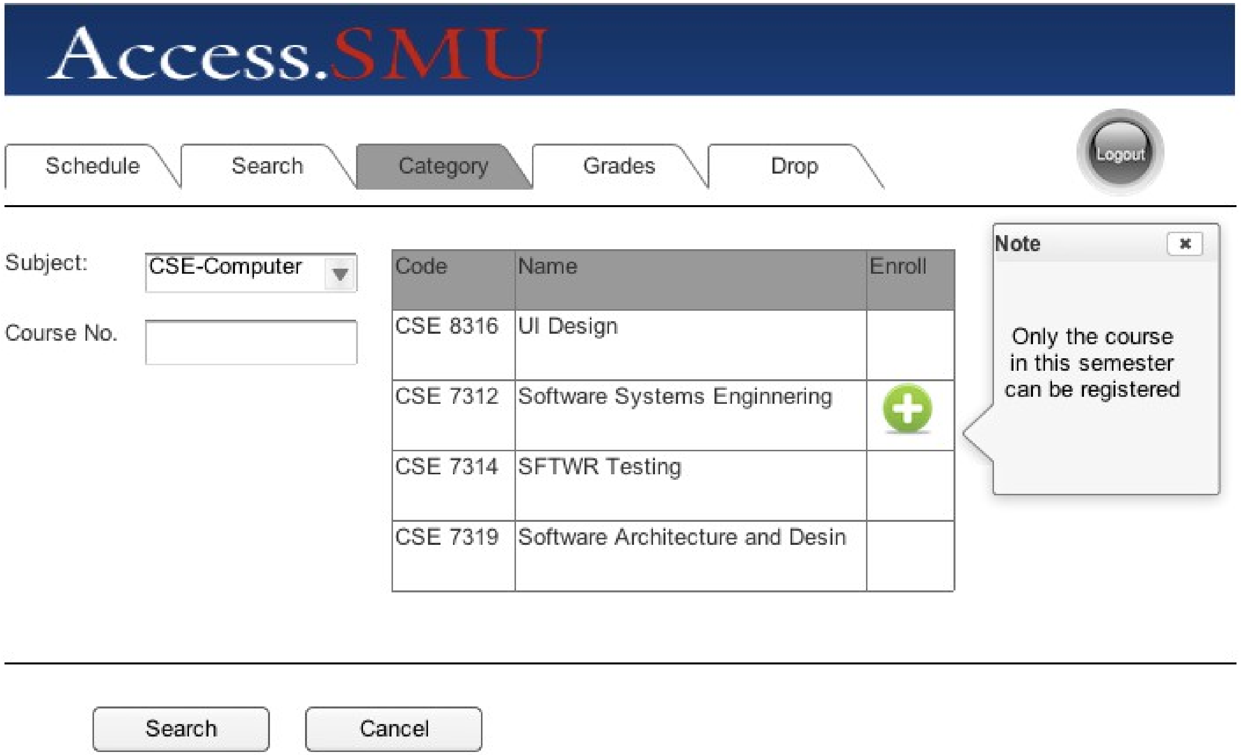
4) Academic-Class Search



Department:	CSE
Term:	Spring 2012
Course:	User Interface
Class ID:	8316
Type:	On Campus
Instructor:	Robert Oshana

Code	Name	Instructor	Enroll
CSE 8316	UI Design	Robert Oshana	⊕
CSE 7312	Software Systems Enginnering	Ann Broihier	
CSE 7314	SFTWR Testing	Jeff Tian	
CSE 7319	Software Architecture and Desin	Jennifer Dworak	

5) Academic-Class Catalog



Subject: CSE-Computer

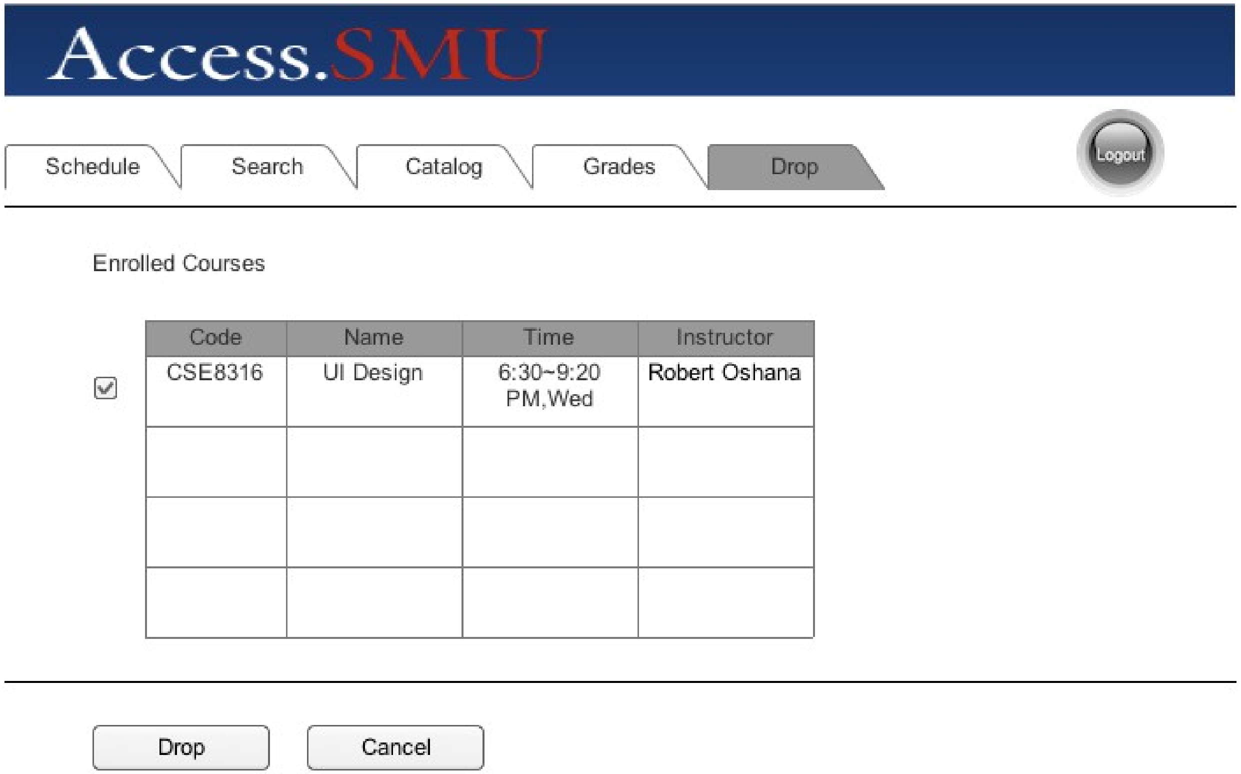
Course No.

Code	Name	Enroll
CSE 8316	UI Design	
CSE 7312	Software Systems Enginnering	⊕
CSE 7314	SFTWR Testing	
CSE 7319	Software Architecture and Desin	

Note

Only the course in this semester can be registered

6) Academic-Class Drop

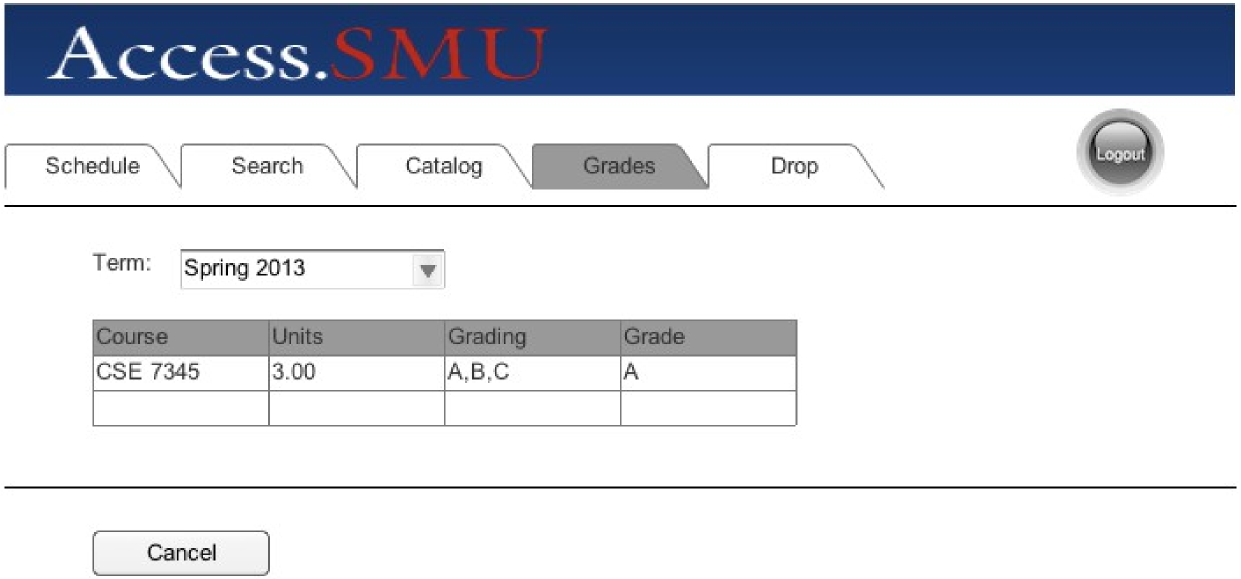


Enrolled Courses

☑

Code	Name	Time	Instructor
CSE8316	UI Design	6:30~9:20 PM,Wed	Robert Oshana

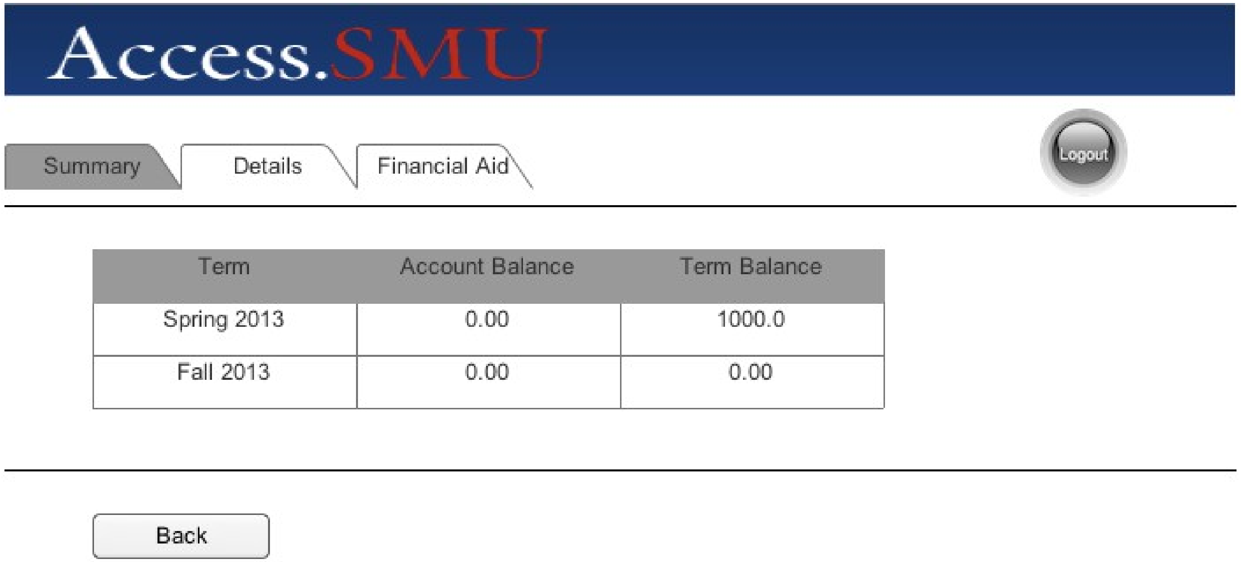
7) Academic-Grades



Term: Spring 2013

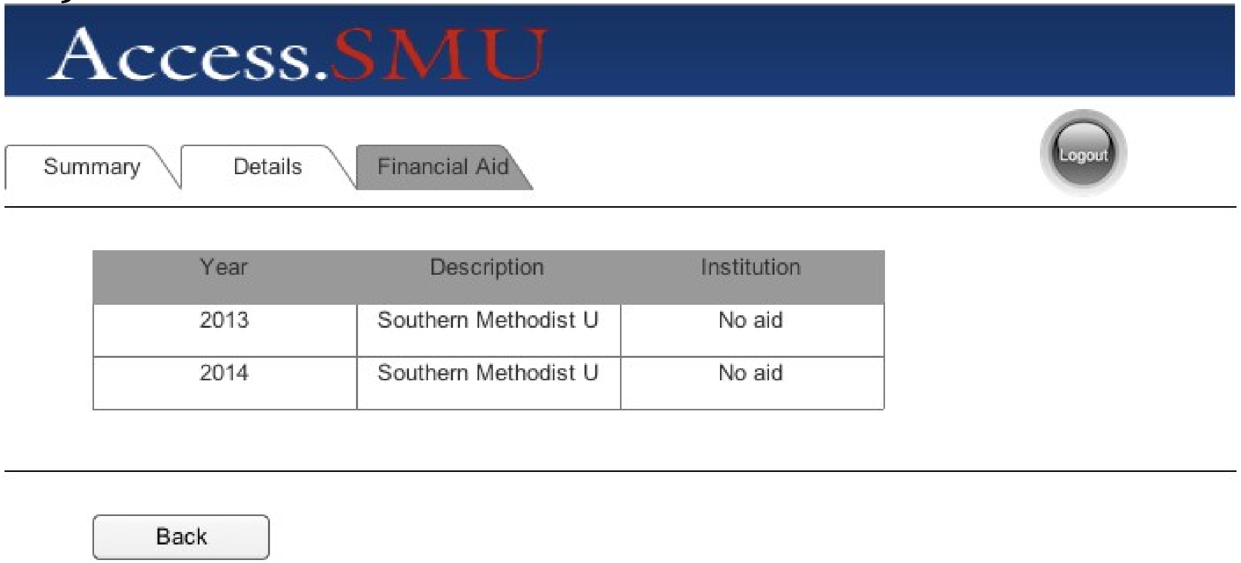
Course	Units	Grading	Grade
CSE 7345	3.00	A,B,C	A

8) Financial Information-Account Summary



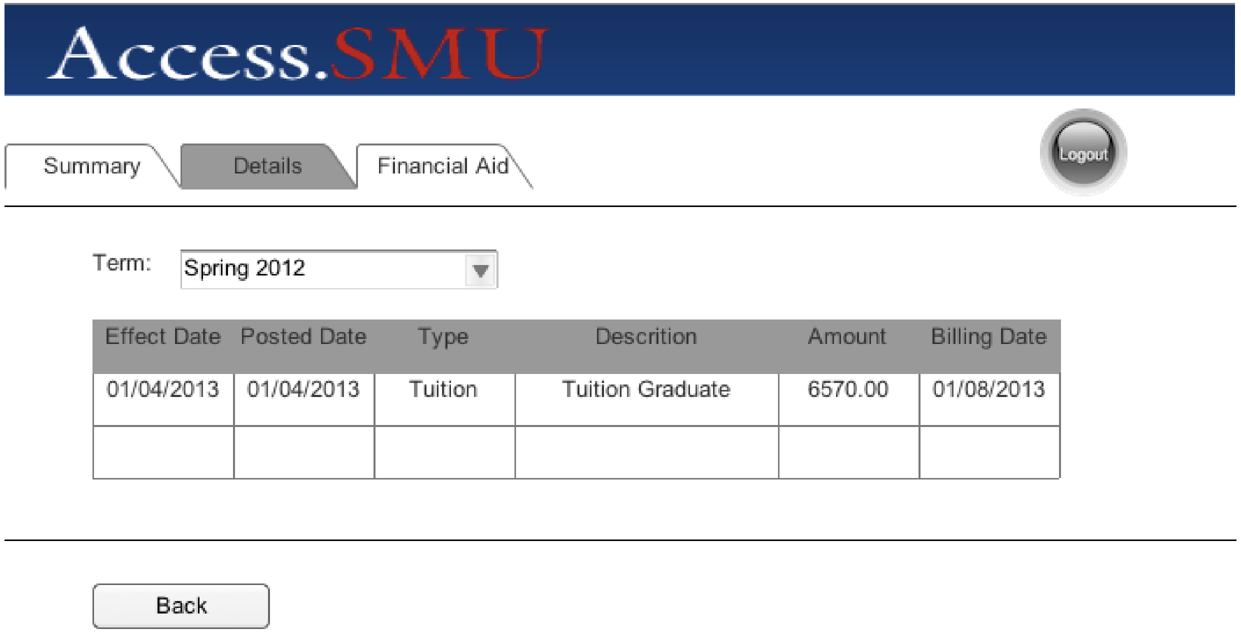
Term	Account Balance	Term Balance
Spring 2013	0.00	1000.0
Fall 2013	0.00	0.00

9) Financial Information-Financial Aid



Year	Description	Institution
2013	Southern Methodist U	No aid
2014	Southern Methodist U	No aid

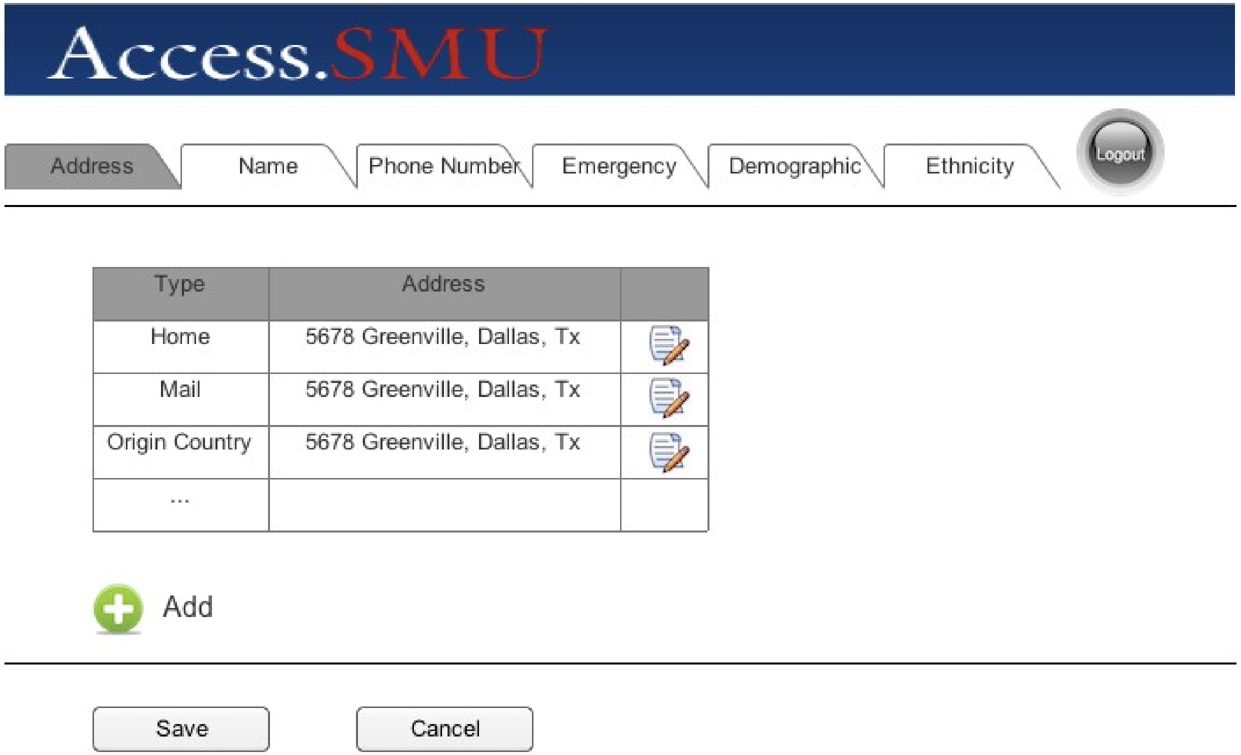
10) Financial Information-Transaction Detail



Term: Spring 2012

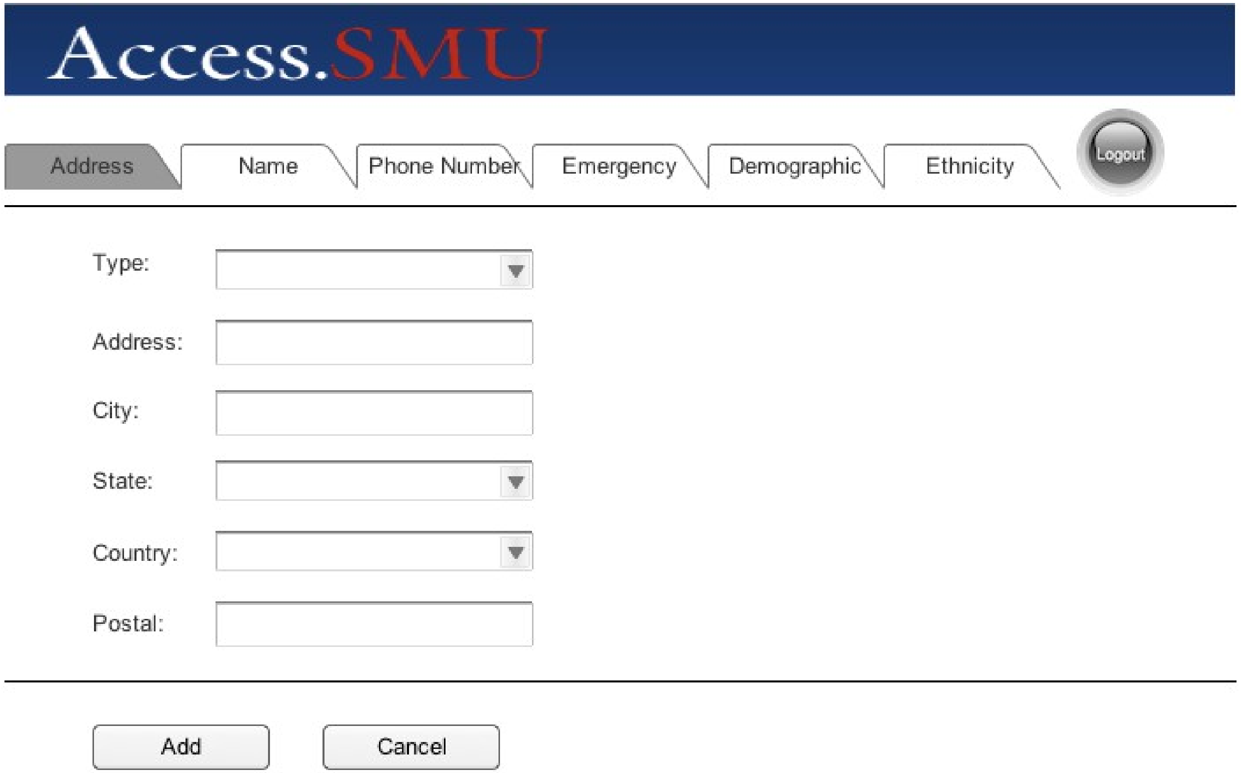
Effect Date	Posted Date	Type	Descrition	Amount	Billing Date
01/04/2013	01/04/2013	Tuition	Tuition Graduate	6570.00	01/08/2013

11） Personal Information-Address

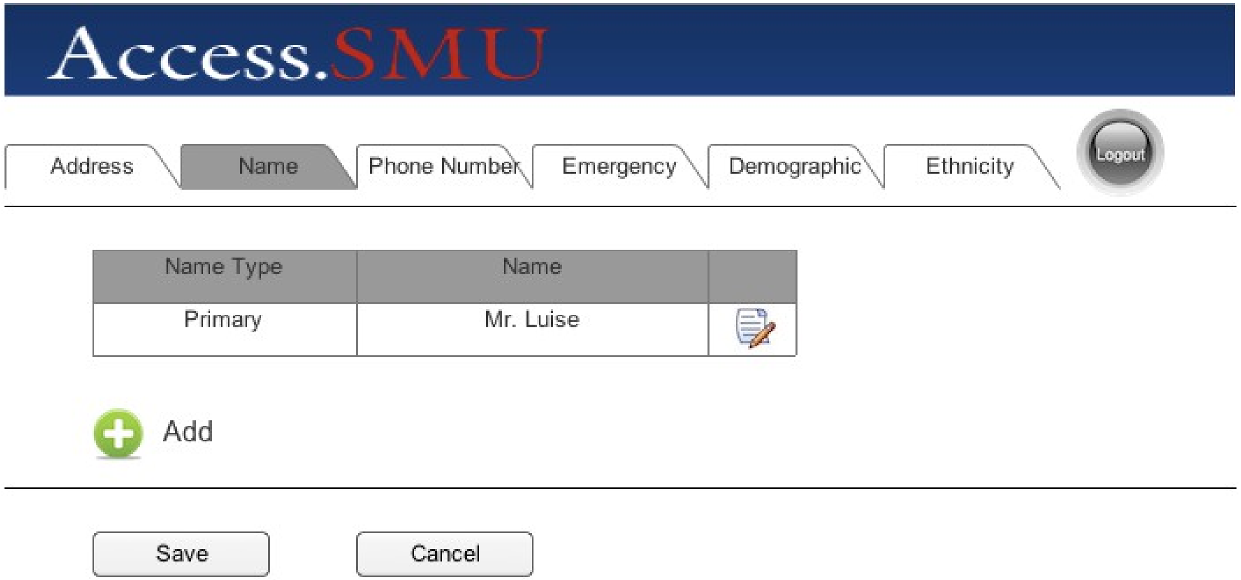


Type	Address	
Home	5678 Greenville, Dallas, Tx	✎
Mail	5678 Greenville, Dallas, Tx	✎
Origin Country	5678 Greenville, Dallas, Tx	✎
...		

12 ）Personal Information-Address Add

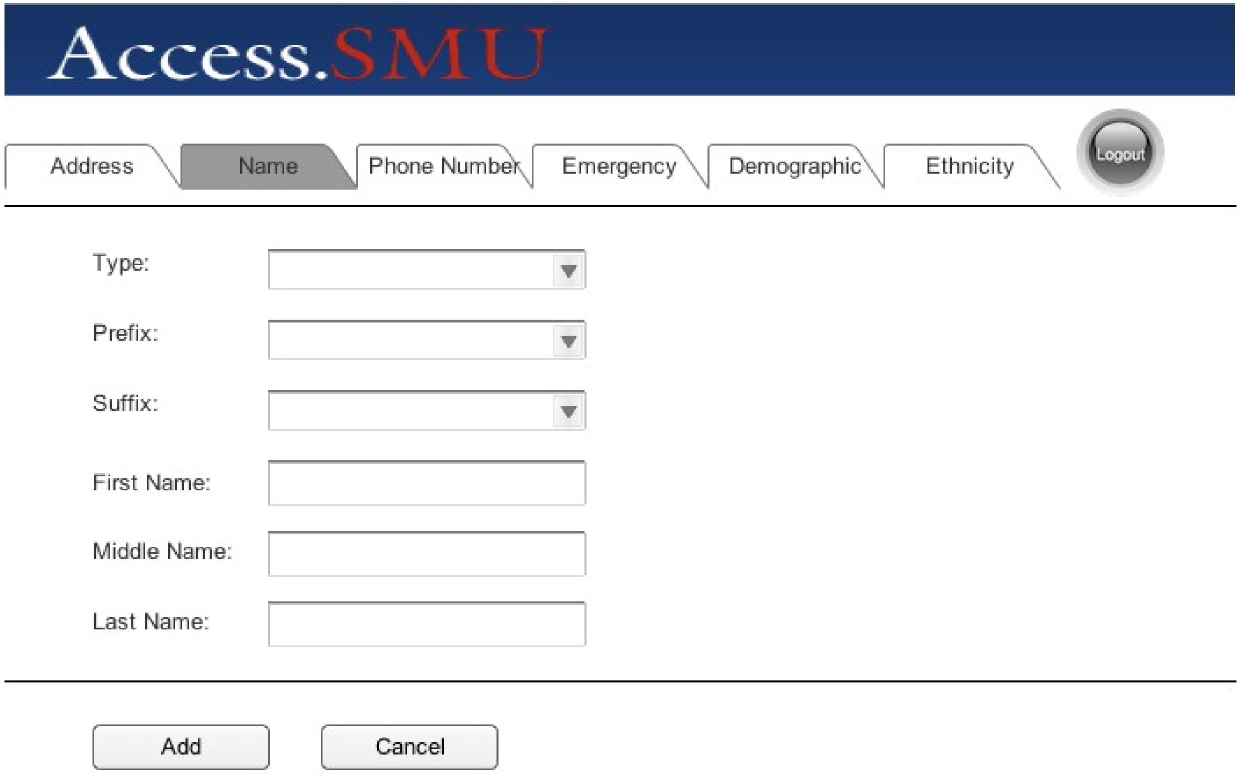


13) Personal Information-Name

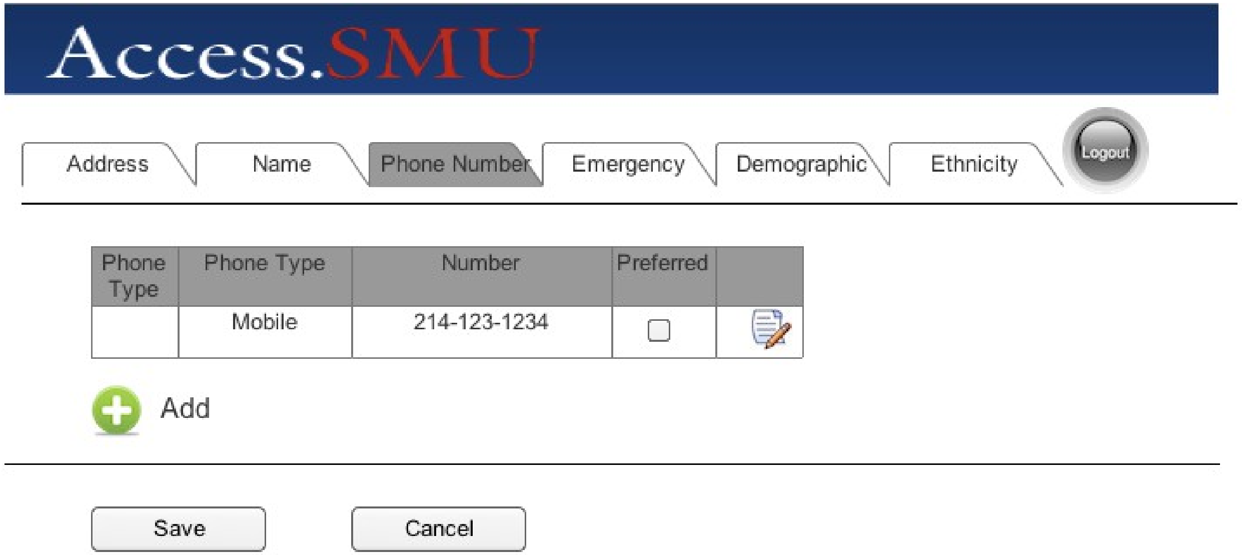


Name Type	Name	
Primary	Mr. Luise	✎

14) Personal Information-Name Add

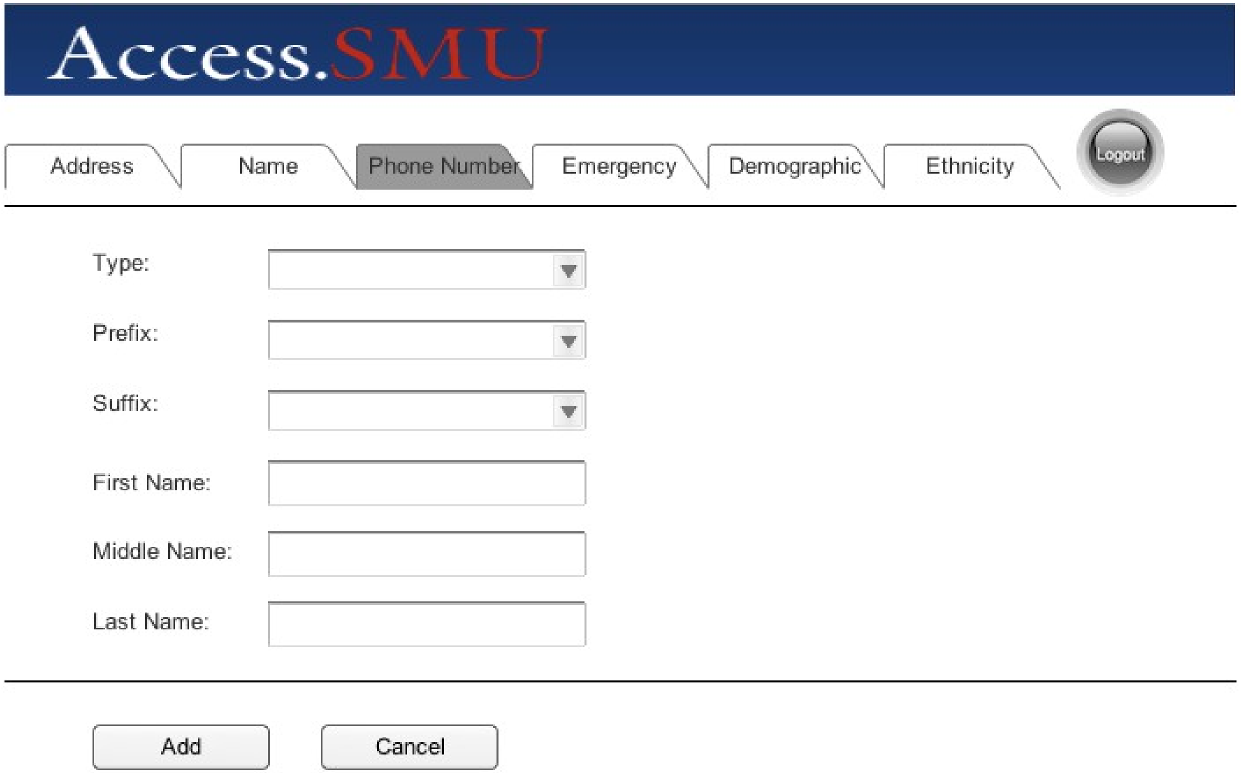


15□ Personal Information-Phone Number

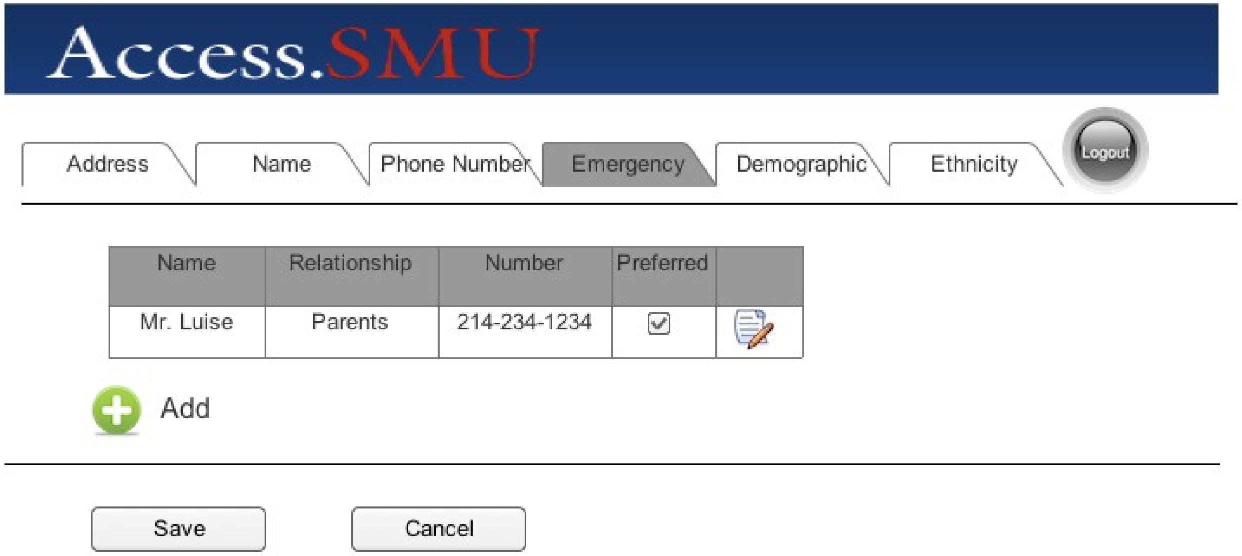


Phone Type	Phone Type	Number	Preferred	
	Mobile	214-123-1234	□	✎

16、 Personal Information-Phone Number Add

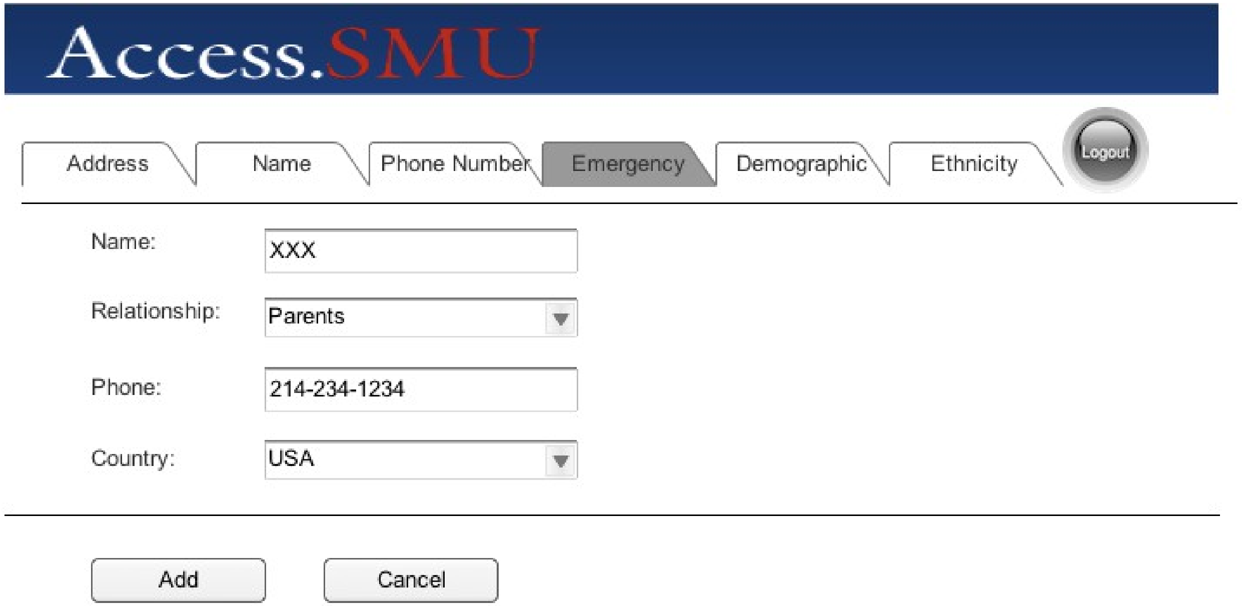


17) Personal Information-Emergency



Name	Relationship	Number	Preferred	
Mr. Luise	Parents	214-234-1234	☑	✎

18）Personal Information-Emergency Add



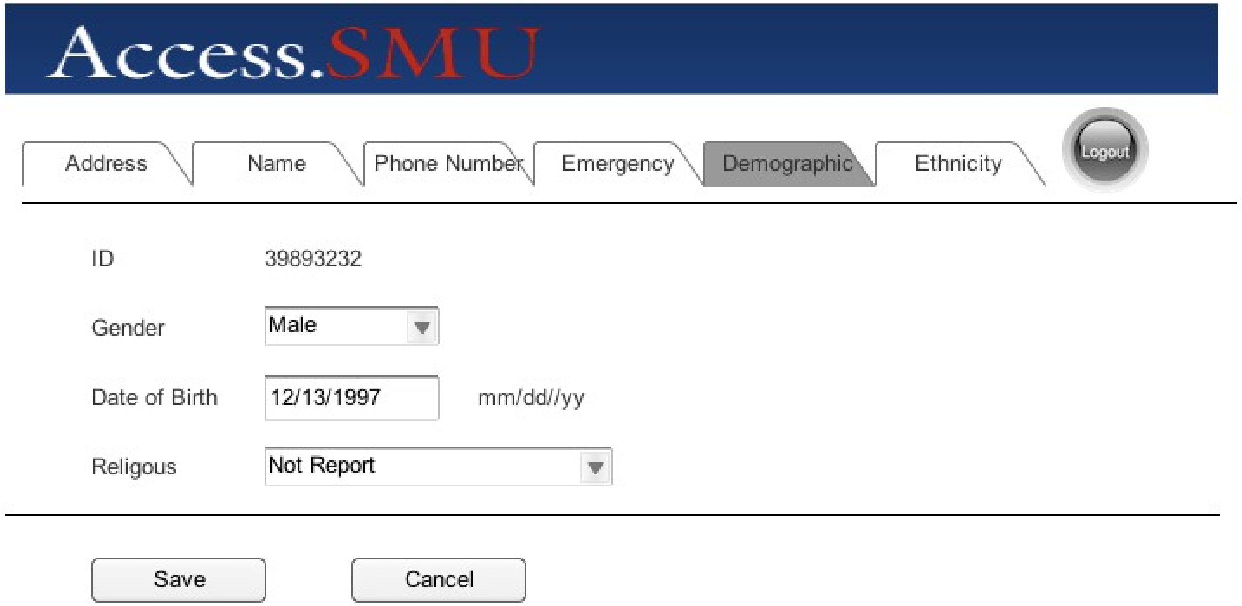
Name: XXX

Relationship: Parents

Phone: 214-234-1234

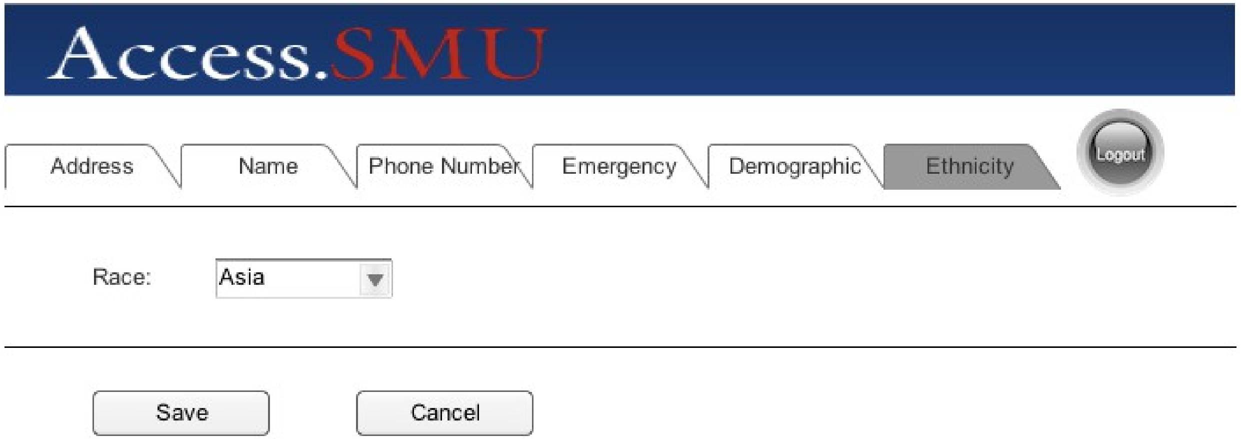
Country: USA

19) Personal Information-Demographic



ID	39893232	
Gender	Male	
Date of Birth	12/13/1997	mm/dd//yy
Religous	Not Report	

20）Personal Information-Ethnicity



Race: Asia