**Web Archive**

By: Nidhi Patel, Ahmed Alfaris, Zachary Wireman, Nathaniel Leonardo

Professor: Bruce Maxim

I. Table of Contents

**1. Introduction............................................................................................. 3**

1.1. Goals and objectives---------------------------------------------------------------- 3

1.2. Statement of scope------------------------------------------------------------------ 3

1.3. Software context---------------------------------------------------------------------- 3

1.4. Major constraints--------------------------------------------------------------------- 3

**2. Usage scenario....................................................................................... 4**

2.1. User profiles--------------------------------------------------------------------------- 4

2.2. User stories--------------------------------------------------------------------------- 4

2.3. Special usage considerations----------------------------------------------------- 4

**4. Functional Model and Description........................................................ 5**

4.1. Description for Function 1---------------------------------------------------------- 5

4.1.1.1 Use case name------------------------------------------------------------------ 5

4.1.2.1 Actors------------------------------------------------------------------------------- 5

4.1.3.1 Preconditions---------------------------------------------------------------------- 5

4.1.4.1 Triggers----------------------------------------------------------------------------- 5

4.1.5.1 Scenario Description------------------------------------------------------------ 5

4.1.6.1 Post Conditions------------------------------------------------------------------- 5

4.1.7.1 Exceptions------------------------------------------------------------------------- 5

4.3 Use Case Diagrams------------------------------------------------------------------ 6

**8. Appendices.............................................................................................. 7**

8.1. System traceability matrix---------------------------------------------------------- 7

1.0 Introduction

We are to develop a web archive that will allow visiting alumni / family and friends of alumni to access photos and student information of CECS graduates at the University of Michigan Dearborn. This web archive is to be accessed only through a kiosk located at the ELB building, which will allow users to search the database by name, major, degree level, graduation year, and graduation semester. Additionally to viewing such information, this application will also allow a visiting alumni to submit personal information to be vetted and included in the database upon approval from administration.

1.1 Goals and objectives

* The goal of the document is to provide a description of the functionality, necessary requirements, and performance of the web application we are creating.
* We want to increase usability of this website by developing a searchable database which have access to student information, and photos. Also provide access to users to add personal information.

1.2 Statement of scope

The user should be able to:

* Use the search feature to find student information easily
* View student information, such as
  + Name
  + Graduating year and semester
  + Major
  + Photo
  + Additional/post-graduation info
* Have access to add personal information and submit it for approval, than it will be added to the database
* Admin should be able to approve or decline the submitted request
* Admin can add new student information to the database

1.3 Software context

* This software will help people at U of M Dearborn search for and view graduated student information.
* In order to make the archive sustainable, we have to consider how we are going to make the website easier for the admin to add and edit student information.

1.4 Major constraints

* Time - We have to set out requirements, plan, develop, and test the archive all within 8 months. This is doable, but will be our main constraint for the project.

2.0 Usage scenario

2.1 User profiles

There will be two levels of users:

* Full Control (Administrator)
* Read/Search/Submit information (General Public)

2.2 User stories

**Full Control User:**

This is the system administrator level which will be able to change any application settings as well as update and maintain the database.

**Read/Search/Submit user information :**

This level of users will be able to search/read the database and cannot insert, delete or modify any information in the database, the only exception is this user can submit information to be added to the database upon approval from administration.

2.3 Special usage considerations

* Users must be able to view student information and request for changes when opening the kiosk
* Maintain consideration for users with color blindness in the UI design.
* Admins/users can edit the information and achievements.
* Make pop up notifications and other important elements large and clear to see for users  
  with visual impairments

**3.0 Data Model and Description**

This section describes information domain for the software

**3.1 Data Description**Data objects that will be managed/manipulated by the software are described in this section.

The main data that this archive will manage are the graduate profiles. We are defining a graduate profile as all of the information related to a single graduated student.

Additionally, there will be objects representing each user-submitted piece of information to be added to graduate profiles, as well as objects representing user reports. Both of these objects will be in queues for the administrators of the archive to go through and make decisions on.

**3.1.1 Data objects**Data objects and their major attributes are described.

The major attributes that each graduate profile will have are:

* Name
* Graduating year
* Major
* User-submitted information

The attributes for user submitted information will be:

* Graduate profile identifier
* Submitted content

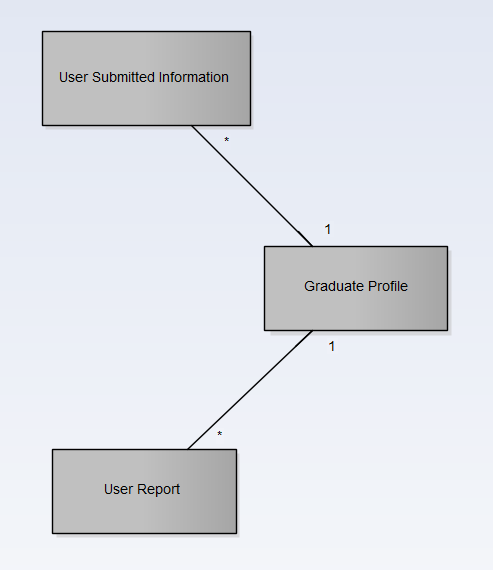
The attributes for user reports will be:

* Graduate profile identifier
* Report message

**3.1.2 Relationships**Relationships among data objects are described using an ERD- like form. No attempt is made to provide detail at this stage.

The objects in the added information and report queues will link to specific user profiles, with many user submissions able to link to one profile, but only one profile linked to each submission.

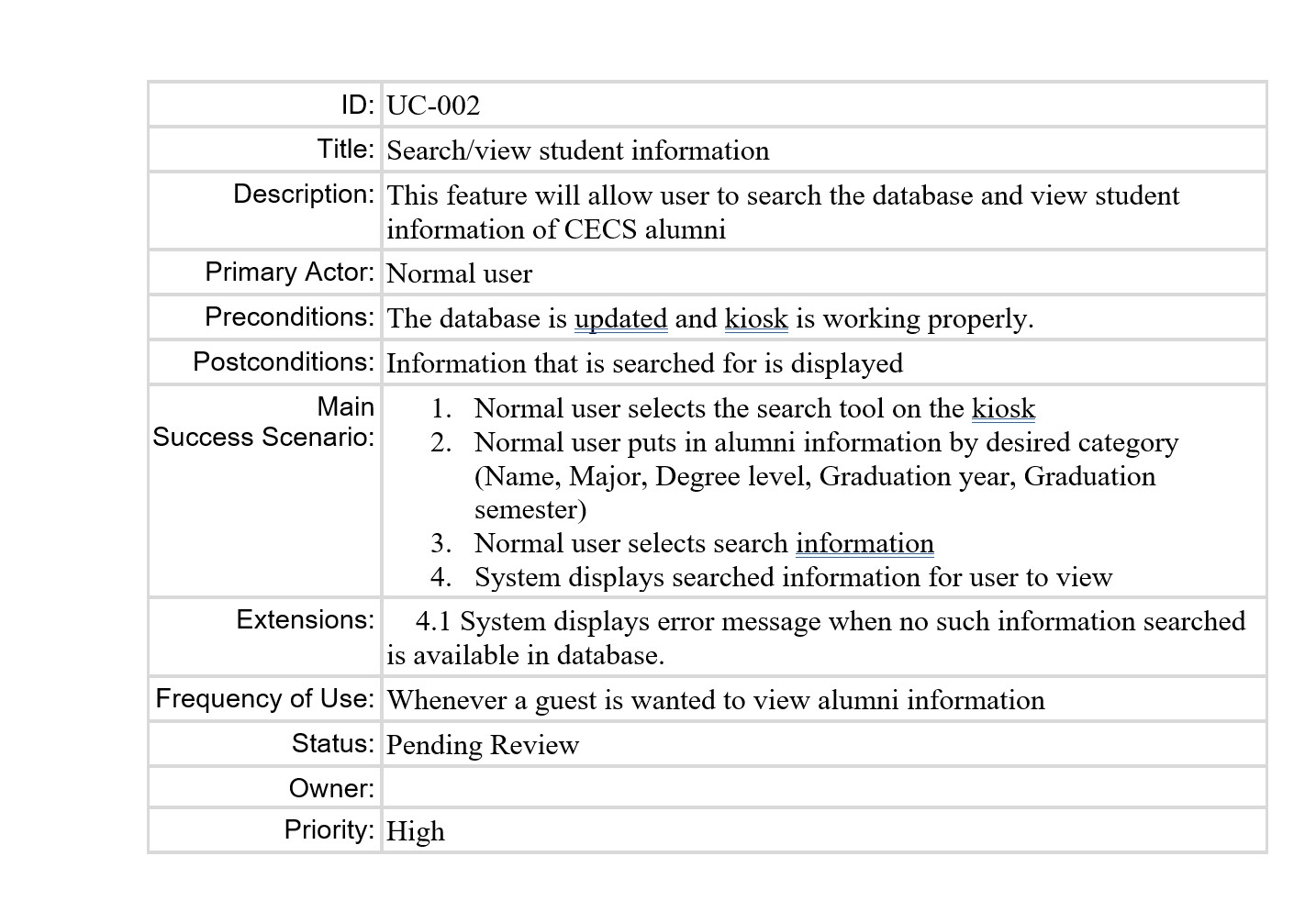
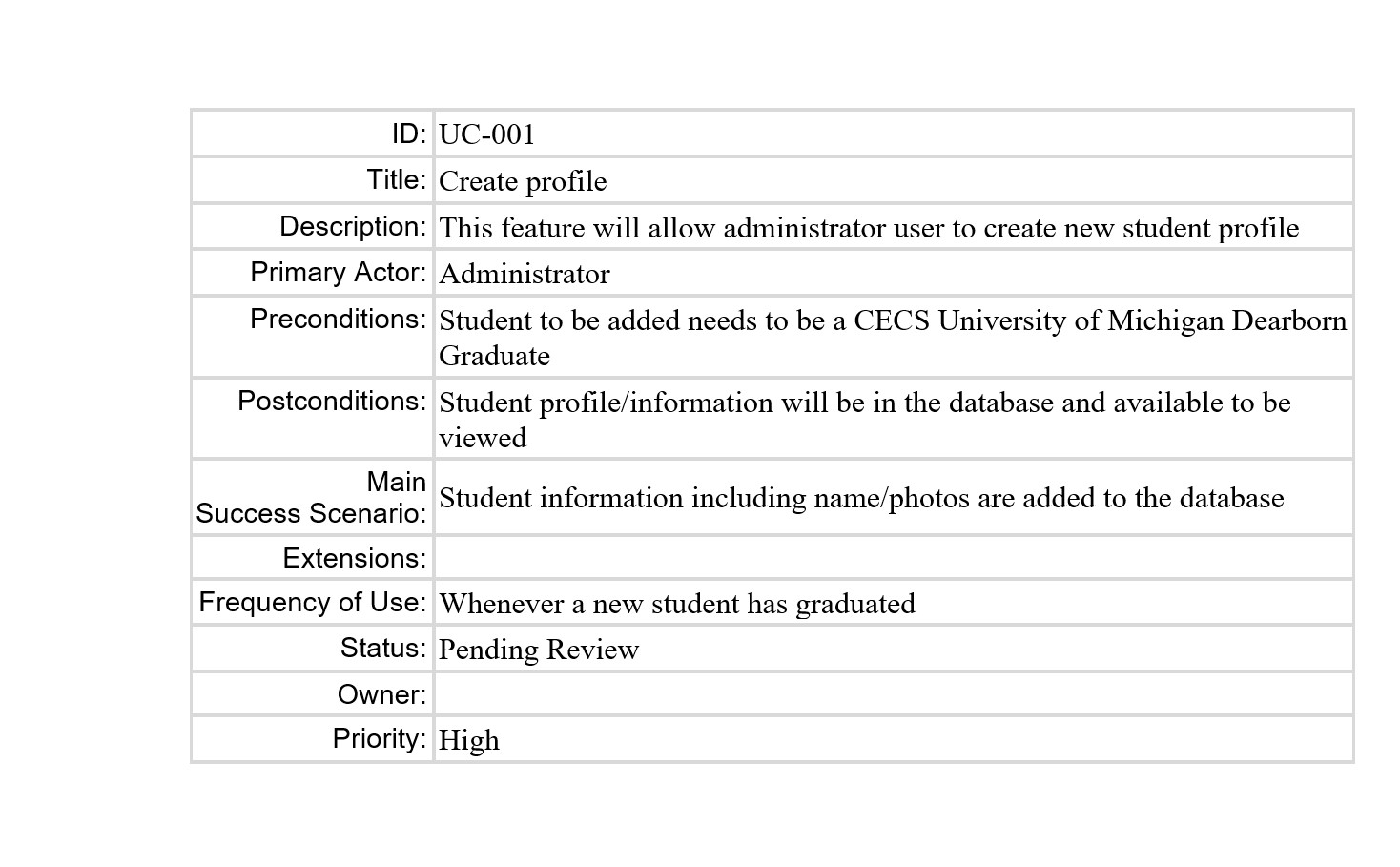
**3.1.3 Complete data model**An ERD for the software is developed

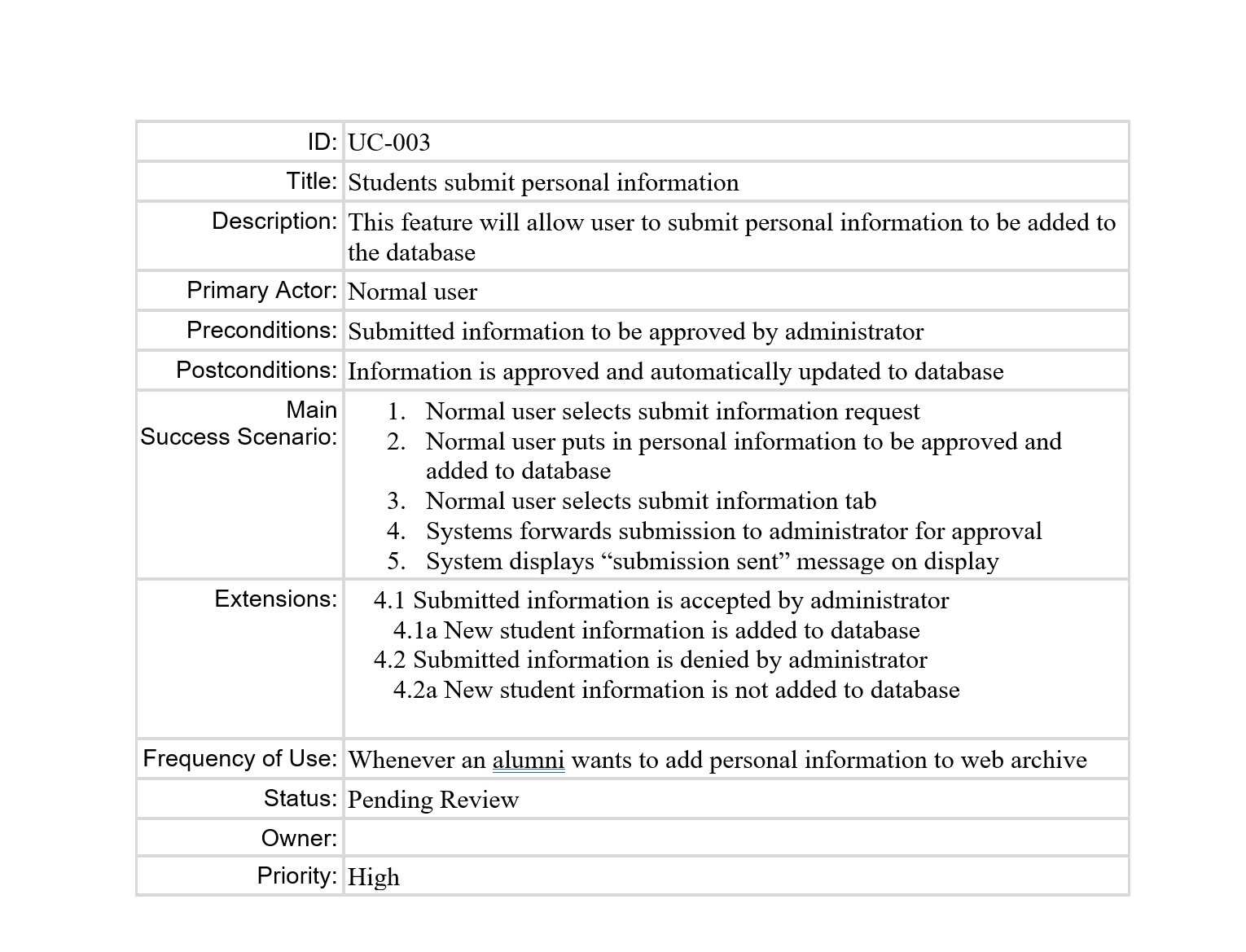


4.0 Functional Model and Description

4.1 Description for Function

Note: In this context a profile is the database entry for one graduated student.





**4.2 Software Interface Description**The software interface(s)to the outside world is(are) described.

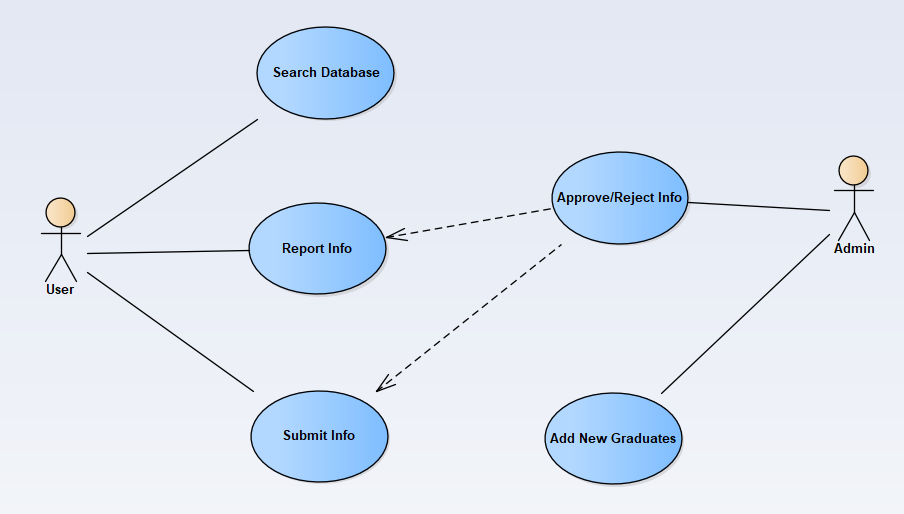
**User interface:**

The user interface will be accessed on a stationary kiosk located in the ELB building at the University of Michigan- Dearborn campus. Once the user starts a session the interface will display photos/information of the CECS graduate classes. Users will have the ability to navigate through a search tool to find specific information that the user would like to view. Users will also be able to submit a student information form for approval to be added to the database.

**Administrator Interface:**

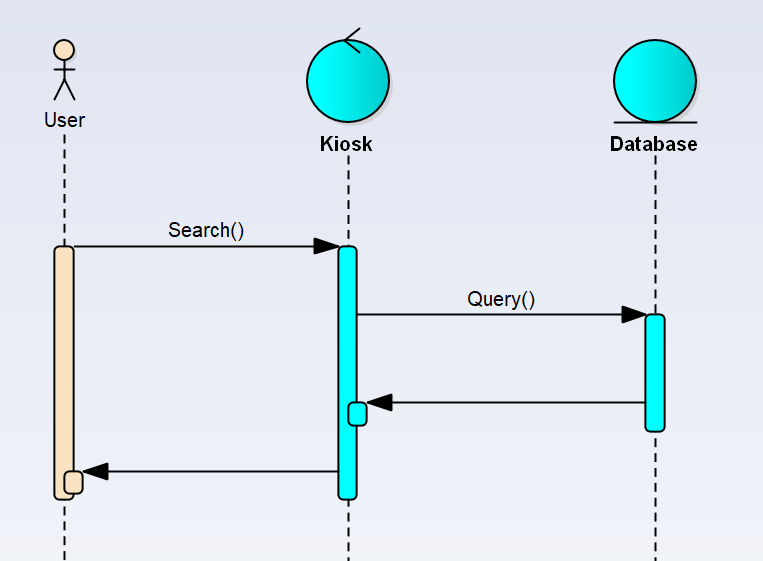
The administrator interface will be accessed on a personal computer by the administrator. The interface will provide the administrator with the ability to edit the database as desired and also approve student information submission forms and will update the information to the database upon approval.

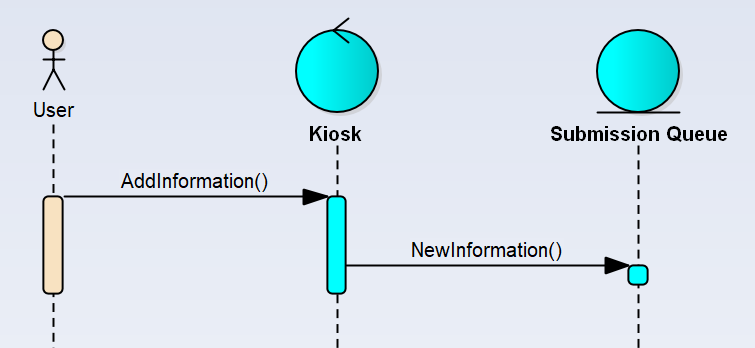
4.3 **Use Case Diagrams**

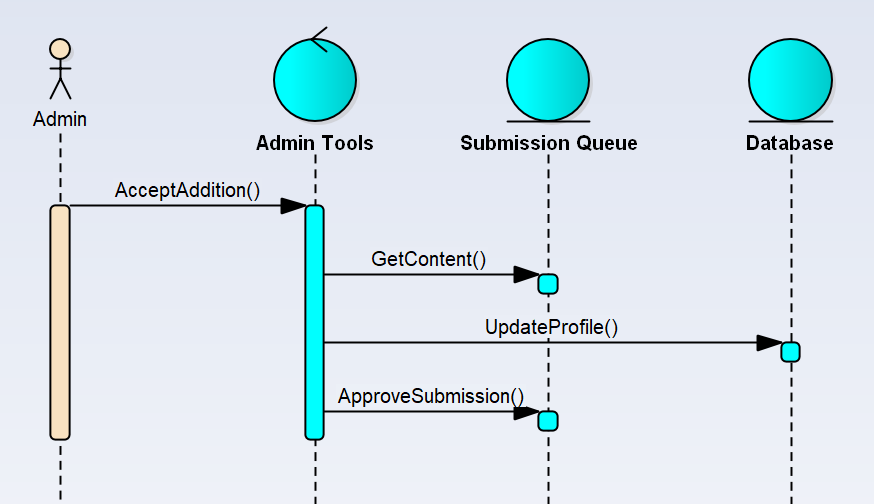
****

4.4 Sequence Diagrams

Used to model the class interactions needed for the use cases.







4.5 Communication Diagrams

Used to model the message passing structure of the system functions.

5.0 Behavioral Model and Description

A description of the behavior of the software is presented.

5.1 Description for software behavior  
A detailed description of major events and states is presented in this section.

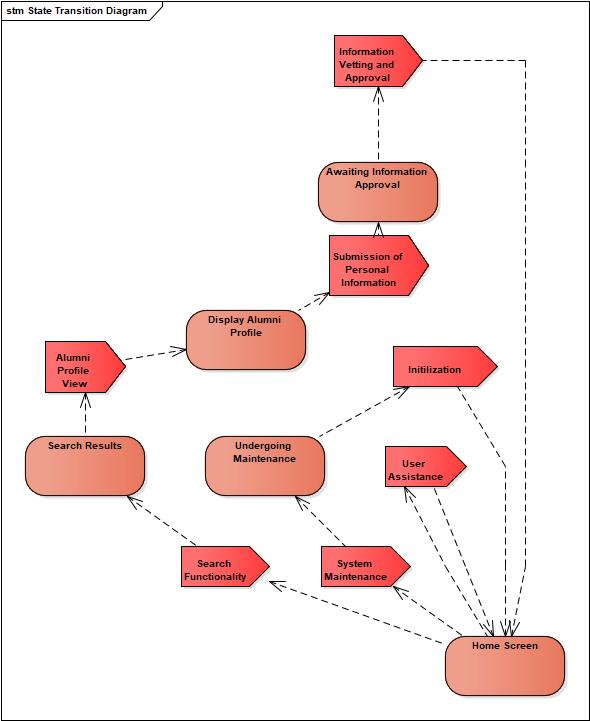
5.1.1 Events  
A listing of events (control, items) that will cause behavioral change within the system is presented.

1. Initialization - Software initializes and loads database. Sets up components necessary for functionality.
2. Search Functionality - Users can search the database using various criteria, including name, major, degree level, graduation year, and graduation semester. The software allows users to enter these parameters to find specific alumni or a group of alumni that meet the specified criteria
3. Alumni Profile View - Upon selecting an alumni record from the search results, the software displays a detailed profile of the selected individual.
4. Submission of Personal Information - Software provides a submission feature for visiting alumni to enter information they desire to be displayed on their profile. The entered data is submitted to the admin for review and approval.
5. Information Vetting and Approval - The admin reviews the submitted information to ensure authenticity and appropriateness. Upon approval, the submitted data is validated and incorporated into the database.
6. User Assistance - Software may include features to assist users, such as a help section or on-screen instructions, to guide users in using the archive effectively.
7. System Maintenance - Maintenance activities may include data backups, software updates and security checks.

5.1.2 States  
A listing of states (modes of behavior) that will result as a consequence of events is presented.

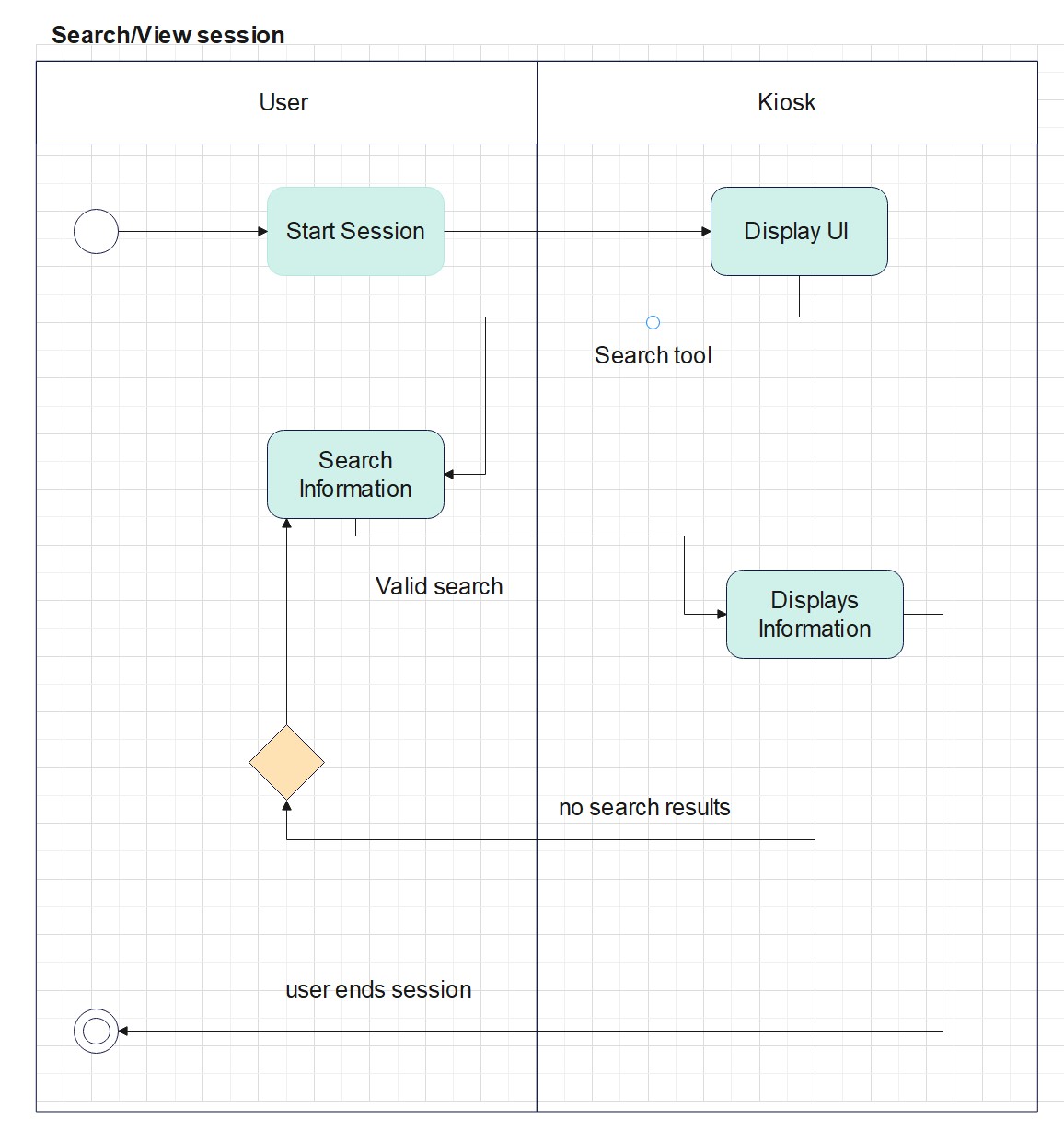
1. Home Screen - Software presents a user-friendly home screen on the kiosk, providing options for searching and browsing the database. Users can input their search criteria or explore the archive in a general manner.
2. Search Results - Software retrieves the relevant alumni records based on the search criteria and presents them to the user. Results may include names, photos, majors, degree levels, and other pertinent information
3. Display Alumni Profile - Software displays a detailed profile which may include academic achievements, career highlights and additional information, subject to approval.
4. Awaiting Information Approval - Information that requires approval before the profile it is associated with requires vetting from administration before the new information is displayed.
5. Undergoing Maintenance - The software may periodically undergo maintenance to ensure the database remains up to date, secure, and well-functioning.

5.2 State Transition Diagrams  
Depict the overall behavior of the system.

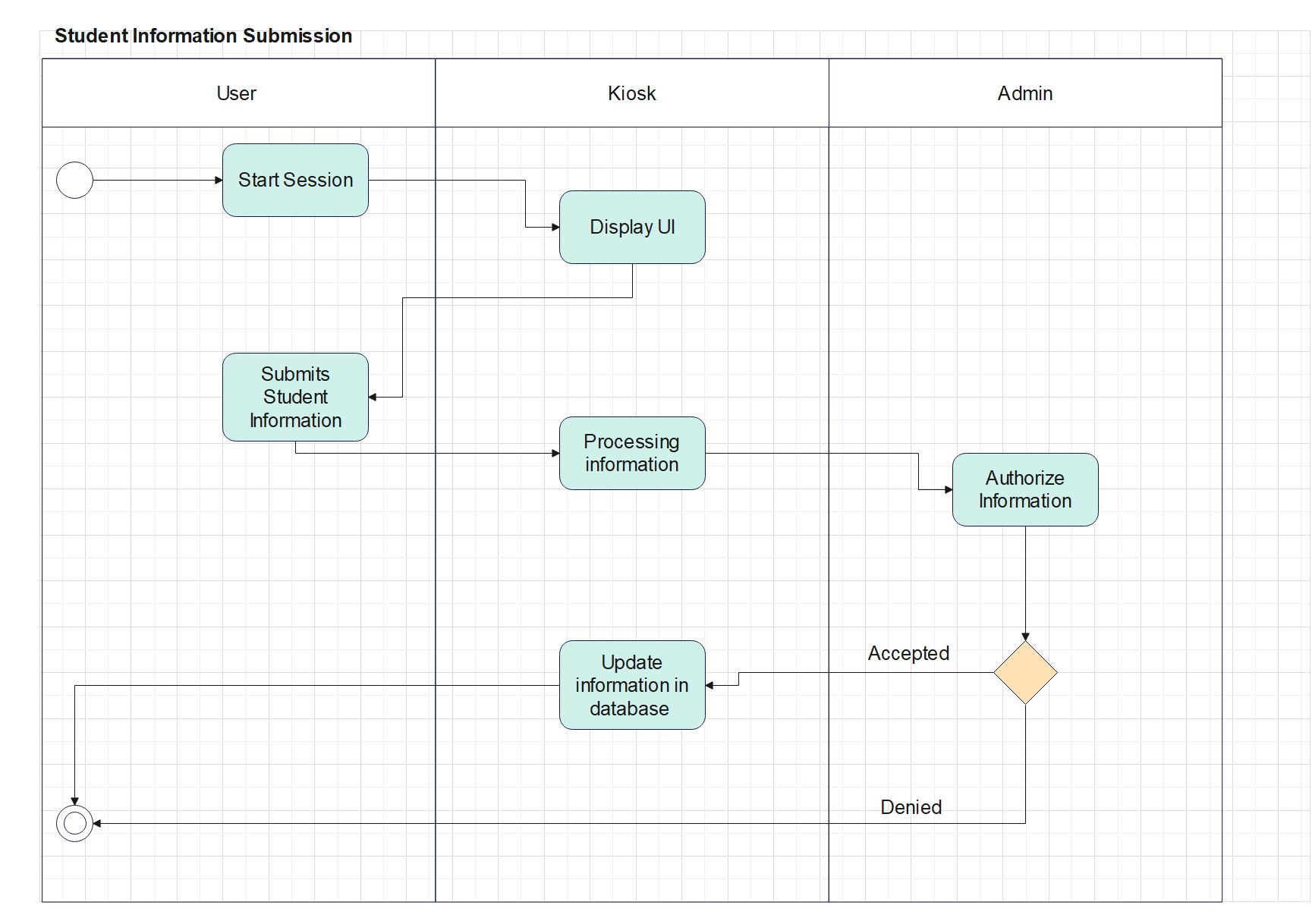


5.3 Activity Diagram   
Depict the manner in which control is managed by the software.

Search/View Use Case:



Student Information Submission Use case:



6.0 Restrictions, Limitations, and Constraints

Special issues which impact the specification, design, or implementation of the software are noted here.

**Time:** In order to build a useful site for users, our team must work around its schedules and pick up new skills more quickly than normal because database designs are relatively new to it. Next semester, we want to try to finish this project completely, with a focus on testing, coding, and UI design.

**Funding**: For specific tools to create and test the functionality of our software, funding may be required.

7.0 Validation Criteria

We plan to use a functional testing methodology. Based on the use cases we have so far, we are planning to apply system testing.

7.1 Classes of tests

| Test # | Test Name | Result |
| --- | --- | --- |
| 1 | Create Profile | Test if admin is allow to create new student profile |
| 2 | Search/View student info | Test if user are able to view and search student information |
| 3 | Add student personal info | Test if user are able to access editing pages |
| 4 | Submit student personal info | Test if users are able to submit approval requests to the administrator. |

7.2 Expected software response

The estimated software response is determined by the number of features we can integrate in the site in the allotted period. The website is designed to help users to get student information.

| Test # | Test Name | Description |
| --- | --- | --- |
| 1 | Create Profile | Admin can create a profile. |
| 2 | Search/View student info | User is able search student information and view them |
| 3 | Add student personal info | Student can add any personal information to their profile |
| 4 | Submit student personal info | Student needs to submit edited information to administrator for approval |

7.3 Performance bounds

* Errors shouldn't occur when storing user data.
* Errors shouldn’t occur when the user sends an approval request and searches student information.
* Down time should not exceed multiple hours

8.0 Appendices

8.1 System traceability matrix

| Key | ID | Test cases | Description | Request | Priority | Status |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | UC - 001 | Search Feature | Users should be able to search by name, class,Major, Degree Level, Graduation year, and Graduation semester | Admin/Users | 5 |  |
| 2 | UC - 002 | Create Profile | Admin is allow to create new student profile | Admin | 5 |  |
| 3 | UC - 003 | Submit Info | User allow to makes changes upon admin approval | User | 5 |  |

**8.2 Product Strategies**A description of relevant product strategy is presented here.

In our ideal system, users would be able to upload personal information and view and search student data. The users will be able to add information and submit it for approval. A flexible system that enables the user to smoothly follow.

**8.3 Analysis metrics to be used**A description of all analysis metrics to be used during the analysis activity..

Our user stories will be used as analytical metrics. All processes will be timed, and we'll keep track of them to make sure our web application's operation time complies with the essential principles of the metrics.

**8.4 Supplementary information (as required)**

* Use case template and an example. Product Management Insights. (n.d.). http://pmblog.accompa.com/2009/10/08/use-case-template-example-requirements-management-basics/
* R.S. Pressman And Associates, Inc. www.rspa.com/docs/index.html.

Document template from: [APM Documents](http://www.rspa.com/docs/index.html)