Team Name: Teenage Mutant FullStack Turtles

College Application Tracker

By: James Castro, Joseph Spivack, Nathaniel Chan, Deanna Liu

CSE 416

Requirements

Table of Contents

Overview, Functional Requirements, and Other Requirements

c4me helps students decide where to apply for college. Its functionality is similar to several successful commercial software systems such as Naviance (licensed by many high schools) and commercial websites such as CollegeData.com. ***c4me*** is for students planning to get a Bachelor's degree.

Use Case Diagram

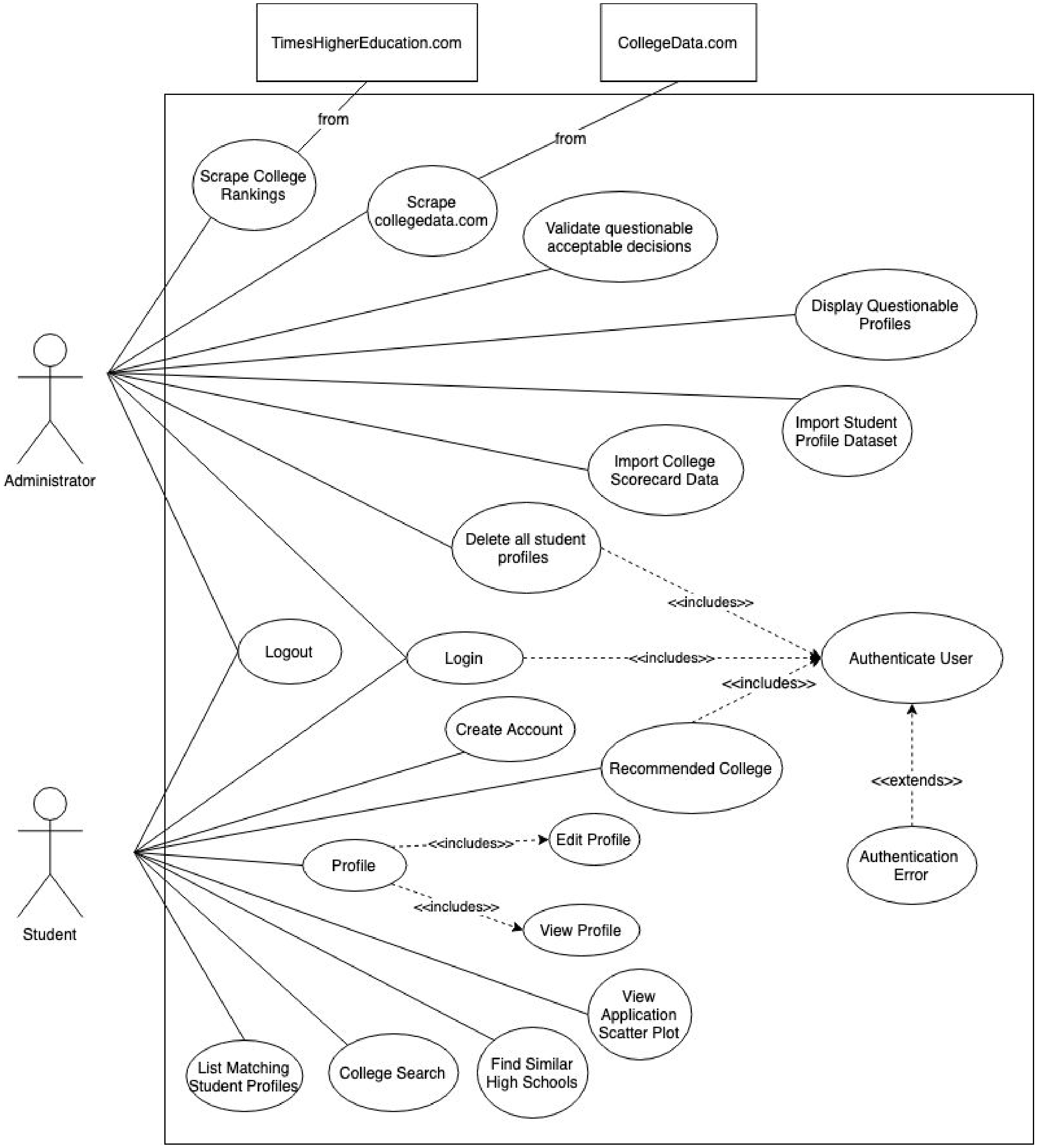

Figure 1: c4me use case diagrams with Student and Administrator

Use Cases

Create Account

Primary Actor: Student (no account)
Pre-Condition: Account associated with Student does not exist yet
Post-Condition: New Account created
Primary Flow:

1) Student clicks ‘create account’ button
2) System Prompts Student for credentials
3) Student enters Credentials
4) System validates Credentials and creates a (not yet validated) account
5) System sends email to users supplied email
6) User clicks validate in email
7) Includes Login
8) System Validates account

Alternative Flow:

Credentials invalid
4.1) System Fails to validate credentials
4.2) System Prompts user to re-enter credentials
Return to step 3

View Profile

Primary Actor: Student
Post-Condition: Student’s profile is displayed
Primary Flow:

1) Student clicks ‘Profile’ button
2) Includes Authenticate
3) Students profile is displayed

Edit Profile

Primary Actor: Student
Pre-Condition: User is logged in as a Student
Post-Condition: Student’s profile saves the changes made
Primary Flow:

1) Student click ‘Edit Profile’ button
2) Systems takes Student to an editable page of their Profile

Alternative Flow:

Search for colleges

Primary Actor: Student

Primary Flow:

1) User clicks search collage button
2) User Inputs Filters (Admission rate, cost, location, majors, name, ranking, size, test acceptance ranges)
3) User clicks search
4) System Validates Input
5) System returns list of colleges based on search requirements

Alternative flow:

Error validating input

4.1) The server redirects to the search page with the erroring fields removed. This should not happen as the javascript on the page will pre check the input. This is in case the user disables JS.

Recommended Colleges

Primary Actor: Student

Primary Flow:

1) Student clicks 'Recommend Colleges' button
2) Student inputs a recommendation score
3) System computes recommendations based other information
4) System displays result for Student in a prompt

Alternative Flow:

Student is not logged in

1.1) System prompts User to enter their login credentials
1.2) User enters login credentials
1.3) System validates login credentials
1.4) Student is now logged into their respective account

Alternative Flow:

Student has no search classes

3.1) System exits computation
3.2) System prompts Student that there are no search results

Find Similar High Schools

Primary Actor: Student

Primary Flow:

1) Student clicks the 'Find Similar High Schools' button
2) System prompts student to specify a High School
3) System validates High School input
4) System returns a list of high schools similar to that specified in 2) sorted in descending order by similarity

Alternative Flow:

Invalid High School specified

3.1) System prompts student to re input High School

List Matching Student Profiles

Primary Actor: Student

Pre-Condition: Student is on the page of a college

Primary Flow:

1) User clicks “View Matching Students” on college page
2) System displays student profiles as a list

Alternative Flow:

No students

1.1) System prompts User that no students who applied to the college

View Application Scatter Plot

Primary Actor: Student

Pre-Condition: Student is on the page of a college

Primary Flow:

1) User clicks “View Scatterplot”
2) System displays student profiles as scatterplot

Alternative Flow:

No students

1.1) System prompts User that no students who applied to the college

Login

Primary Actor: User

Pre-Condition: User has validated Account (Student or Admin)

Post-Condition: User is logged in

Primary Flow:

1) User inputs credentials
2) Server Validates Them

Alternative Flow:

Invalid login credentials or profile not found

2.1) User is prompted to re-enter credentials (possibly after a timeout based on the number of failed logins)

Logout

Primary Actor: User

Pre-Condition: User is logged in

Post-Condition: User is logged out

Primary Flow:

1) User Clicks Logout
2) Server disassociates the SID with that account

Scrape college rankings

Primary Actor: Administrators
Secondary Actor: TimesHigherEducation.com
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1) Includes Authenticate
2) Administrator clicks option to scrape data from website.
3) Database is populated with newly scraped data.

Alternate Flow:

3.1) The script does not run due to some connection error.

Post-Condition: Database is updated entirely, or not at all (Atomicity)

Import College Scorecard data file

Primary Actor: Administrators
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1)Includes Authenticate
2)Admin selects option to import scorecard data from guo
3)The file containing college scorecard data is parsed and the database is updated.

Alternative Flow:
3.1) The file is improperly formatted, and no changes are made.
Post-condition: The database is updated entirely or not at all

Scrape CollegeData.com

Primary Actor: Administrators
Secondary Actor: CollegeData.com
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1) Includes Authenticate
2) Administrator clicks option to scrape data from website.
3) Database is populated with newly scraped data.

Alternate Flow:

3.1) The script does not run due to some connection error.

Post-Condition: Database is updated entirely, or not at all (Atomicity)

Delete all student profiles

Primary Actor: Administrators
Pre-Condition: User is logged in as an Administrator
Post-Condition: All Student Profiles are deleted
Primary Flow:

1) Includes Authenticate
2) System prompts “Are you sure you want to do this?”
3) User clicks “Yes”
4) Prompt closes
5) Page refreshes and Database is updated with all student profiles deleted

Alternative Flow:

No students exists
2.1) Error prompts telling User “No students to delete”
2.2) Prompt closes with no changes made

Import student profile dataset

Primary Actor: Administrators
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1)Includes Authenticate
2) The administrator selects import student profile dataset option from GUI.
3) The File Containing student data is parsed and the database is updated.

Alternative Flow:

3.1) The file is improperly formatted, and no changes are made to the database.

Post-condition: The database is updated entirely or not at all

Display questionable profiles

Primary Actor: Administrators
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1) Includes Authenticate
2) Administrator Clicks View Questionable Decisions
3) System gets questionable Applications based on certain criteria and displays them

Validate questionable acceptance decisions

Primary Actor: Administrators
Pre-Condition: User is logged in as an Administrator
Primary Flow:

1) Extends Display questionable profiles
2) Administrator clicks Validate on application
3) System marks that application as no longer questionable

Authenticate

Primary Actor: Student or Admin
Primary Flow:

1) User Browser sends cookie with session ID to server

2) Server determines what account it’s associated with

Alternative flow:

User Not logged in:

2.1) Include login

2.2) associate the SID with newly logged in account.