Functional Specification

1. Product name: ByteMeCollege.com

2. Problem statement:

College students do not have enough information about the courses they take before enrolling. They need a platform to decide relevant courses to their academic interest from offered options. While choosing a university to attend, students don't have a one-stop-shop to compare similar programs offered by different academic institutions.

3. Product objectives

The main objective of ByteMeCollege.com is to provide user verified reviews of courses offered by different academic institutions (currently focusing on SJSU). We hope to help students by providing constructive information to help them choose the courses that are most relevant to their academic goals and interests. Students can also do real world comparisons of similar courses offered by various institutions to help them choose the right academic programs and most fitting career path for their educational goals.

4. Functional requirements

1. Administrators must enter departments, majors, courses.
2. Must students with SJSU email sign up for account.
3. Anybody must be able to read course reviews.
4. Registered members must be able to modify content.
5. Registered members must be able to leave a review.
6. Registered members shall request to add a course.
7. Registered members must be not able to modify review after 24 hours of posting.

5. Nonfunctional requirements

1. The server must be continuously running.
2. Website interface must be easy to use and navigate.
3. Reviews must be submitted by respective institution's students.

4. System shall send confirmation email to new users within 5 min of account creation.

6. Use cases

Use Case Description 1

Use Case name:	About us
Product name:	ByteMeCollege
Team:	ByteMeCollege
Date:	September 22, 2016

1. Goal

Provide user with ample amount of information about the website's purpose.

2. Summary

A user must quickly understand the purpose of the website and who the administrators are.

3. Actors

Actor 1: The user

Actor 2: Website's homepage

etc.

4. Preconditions

- The website server is running.
- The user has navigated to the website’s homepage.
- The user navigates to the “about us” section on the website’s homepage.

5. Trigger

The web application’s homepage controller detects the user’s HTTP request for access.

6. Primary Sequence

Step	Action
1	Homepage controller detects a user’s request for access.
2	Homepage controller calls the homepage view.
3	Homepage index.html.erb file is rendered to the users.
4.	User scrolls down to the “about us” section of the rendered view.

7. Primary Post Conditions

- User is informed of the website’s purpose.
- User is informed of the website’s services.
- User is informed of who are the administrators of the website.

8. Alternate Sequences

Alternate Trigger

The application’s homepage controller detects the user’s HTTP request for access but then terminates.

Step	Action

1	An alternate controller detects a user’s request for access.
2	The alternate controller calls its corresponding view.
3	The called view is rendered to the user.
Alternate Post Conditions	
· The user quickly aborts access to the website. · The user is ignorant of the website’s purpose, services, and administrators.	
Alternate Trigger	
The web application’s homepage controller detects the user’s HTTP request outside its domain.	
Step	**Action**
1	A controller detects a user’s HTTP request for access to an external site.
2	The controller is no longer supporting the user’s requests.
Alternate Post Conditions	
· The user is no longer connected to the web application. · The user is ignorant of the website’s purpose, services, and administrators.	

9. Nonfunctional Requirements

- The view pages must be rendered in less than 3 seconds.
- The web application shall run on Google Chrome, Internet Explorer, Microsoft Edge, Firefox, and Safari on any platform.
- The web application must be optimized for mobile devices.
- There must be a navigation bar at the top of every page view.

10. Glossary

Homepage controller = is routed as the root of the web application.

Alternate controller = any other controller other than the homepage controller (application, departments, or majors)

HTTP = Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems. It is the foundation of data communication for the World Wide Web. Hypertext is structured text that uses logical links (hyperlinks) between nodes containing text. HTTP is the protocol to exchange or transfer hypertext.

Use Case Description 2

Use Case name:	Request to add a class
Product name:	ByteMeCollege
Team:	ByteMeCollege
Date:	September 22, 2016

1. Goal

User requests to add a class that isn’t listed

2. Summary

A user who has logged in can send one or more requests to add a course to be reviewed that is not already listed.

3. Actors

Actor 1: User
Actor 2: System
etc.

4. Preconditions

·The server has been connected
·The user has navigated to the web page
·The user has logged in with their unique username and password
·The user has searched for a course and was unsuccessful in finding said course
etc.

5. Trigger

The user selects the “request to add course” option

6. Primary Sequence

Step	Action
1	The user clicks to request to add a course
2	The system checks whether or not the user has logged in
3	If the user is not logged in, the system displays message to log in
4	If the user is logged in, the system will check to see if course already exists

7. Primary Postconditions

·The user gets redirected to homepage
·

etc.

8. Alternate Sequences

Alternate Trigger

User submits a request for a course that does not exist

Step	Action
1	The user clicks to request to add a course
2	The system checks whether or not the user has logged in
3. *4.*	If the user is not logged in, the system displays message to log in If the user is logged in, the system will check to see if course already exists

Alternate Postconditions

·Admin returns a message stating that course does not exist
·User gets redirected to homepage

Alternate Trigger

User submits a request for a course that already exists

Step	Action
1	The user clicks to request to add a course
2	The system checks whether or not the user has logged in

3 *4*	If the user is not logged in, the system displays message to log in If the user is logged in, the system will check to see if course already exists

Alternate Postconditions

·Admin return a message stating course already exists
·User gets redirected to homepage

9. Nonfunctional Requirements

·Friendly user interface
·Email to be sent within 5 min to user that request has been completed

10. Glossary

User = person who wants to submit a request to add a course
System = ByteMeCollege.com: a college course reviewer web application
Admin = Team who created ByteMeCollege.com

Use Case Description 3

Use Case name:	User does a class search
Product name:	ByteMeCollege.com

Team:	ByteMeCollege
Date:	9/23/2016

1. Goal

To locate a course on the website to read reviews.

2. Summary

User attempts to locate a course he/she intends to get information in search bar and reads reviews.

3. Actors

Actor 1: User
Actor 2: ByteMeCollege.com server

4. Preconditions

- Computer has been started up
- Computer is connected to internet.
- User is on ByteMeCollege.com page in an internet browser

5. Trigger

User clicks on the search bar

6. Primary Sequence

Step	Action
1	types course code or course related keywords in search bar.
2	ser clicks on search button next to the search bar or hits enter. omputer communicates with server to search for results relevant to the query.
3	Computer shows relevant searches on the screen.
4	User clicks on the search result that matches with the course he intends to find information on.
5	Computer opens the course page and shows it on the screen.

7. Primary Postconditions

- User is able to read the reviews on the page
- User is able to rate reviews if he/she wishes.

8. Alternate Sequences

Alternate Trigger

Customer clicks on search button or hits enter.

Step	Action
1	Computer communicates with server to search for results relevant to the query but does not find a match
2	Computer shows “no match found” on screen asks user to modify query or submit a request to add desired course.

Alternate Postconditions

- User either modifies the query or submits a request to add course.

Alternate Trigger

Step	Action
1	
2	
etc.	

Alternate Postconditions

·
·
etc.

9. Nonfunctional Requirements

- System responds to each customer input within 3 seconds
- System displays content in English

10. Glossary

User: Person who is visiting ByteMeCollege.com site
Course: An educational class taught at the college or university

ByteMeCollege.com: A website that provides reviews on various college courses
System: Server on which ByteMeCollege.com is running on
Computer: A device that lets user interact with the system.

Use Case Description 4

Use Case name:	Adding course review
Product name:	ByteMeCollege
Team:	ByteMeCollege
Date:	September 23, 2016

1. Goal

Student add course review.

2. Summary

A logged in student is able to add only one review to multiple courses which are list on the system.

3. Actors

Actor 1: Students
Actor 2: ByteMeCollege website system

4. Preconditions

- ❏ A student must be logged in.
- ❏ The student has not added a review to a particular course.

5. Trigger

After the student finishes entering review, the student clicks “post” button.

6. Primary Sequence

Step	Action
1	The system lists previous review of the particular course.
2	If the student has not posted a review to the course, the system will display a posting text area for the student to write review and post it. It has a minimum number of character.
3	The student confirms and submits the review.
4	The system display all the reviews with the one just submitted.

7. Primary Postconditions

The student will be able to view the post, it also can be modify within 24 hours.

8. Alternate Sequences

Alternate Trigger

- ❏ The student clicks “post” button.

Step	Action
1	The student did not enter enough desired number of character

Alternate Postconditions

The system will not save the review that the student entered.

9. Nonfunctional Requirements

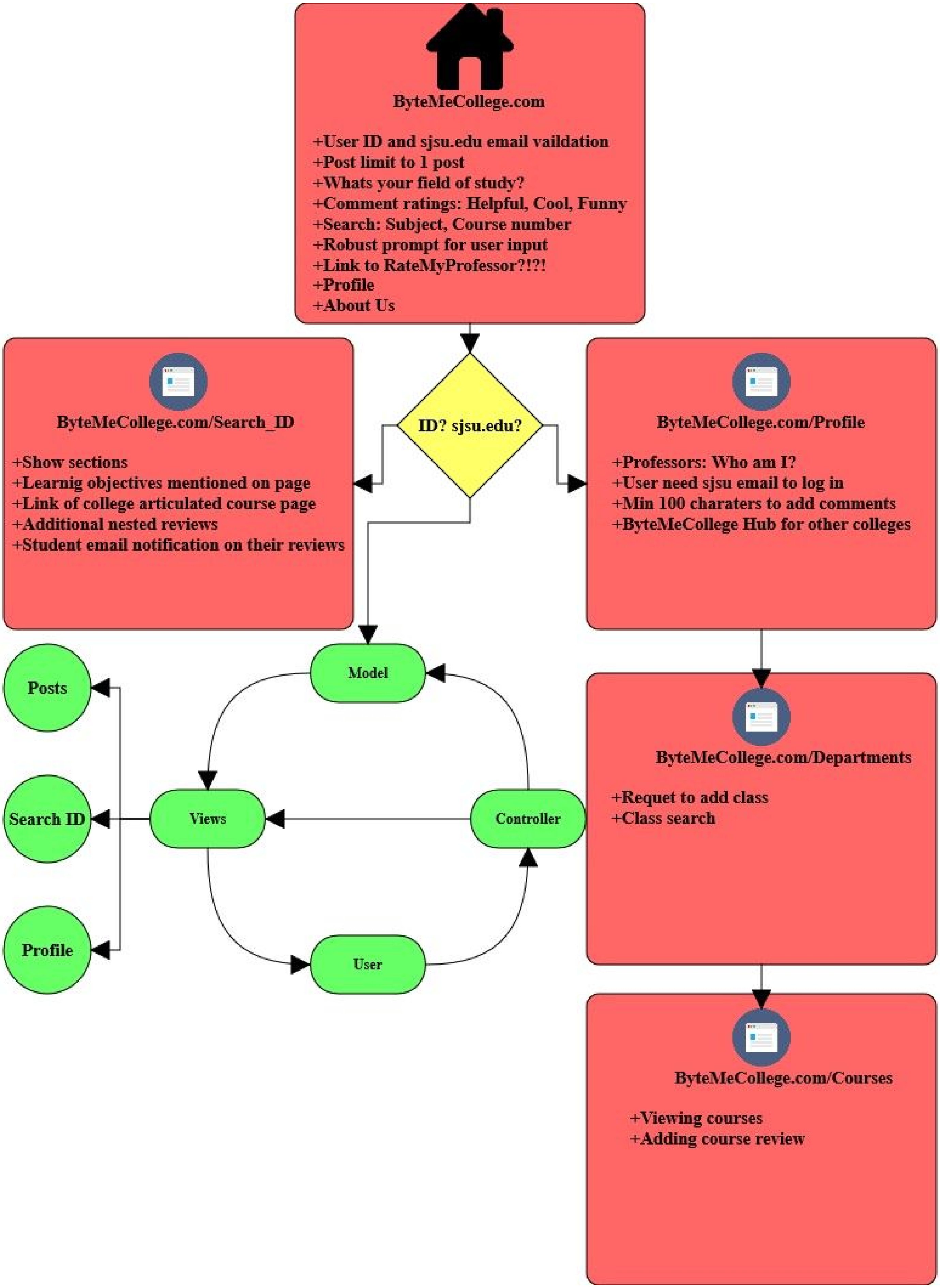
The system displays text in English.

❏ The system responds to the students immediately. May vary depends on the network.

10. Glossary

❏ Student = a student who already logged in to the system

❏ System = ByteMeCollege.com



ByteMeCollege.com

+User ID and sjsu.edu email vaildation
+Post limit to 1 post
+Whats your field of study?
+Comment ratings: Helpful, Cool, Funny
+Search: Subject, Course number
+Robust prompt for user input
+Link to RateMyProfessor?!?!
+Profile
+About Us

ByteMeCollege.com/Search_ID

+Show sections
+Learnig objectives mentioned on page
+Link of college articulated course page
+Additional nested reviews
+Student email notification on their reviews

ByteMeCollege.com/Profile

+Professors: Who am I?
+User need sjsu email to log in
+Min 100 charaters to add comments
+ByteMeCollege Hub for other colleges

ByteMeCollege.com/Departments

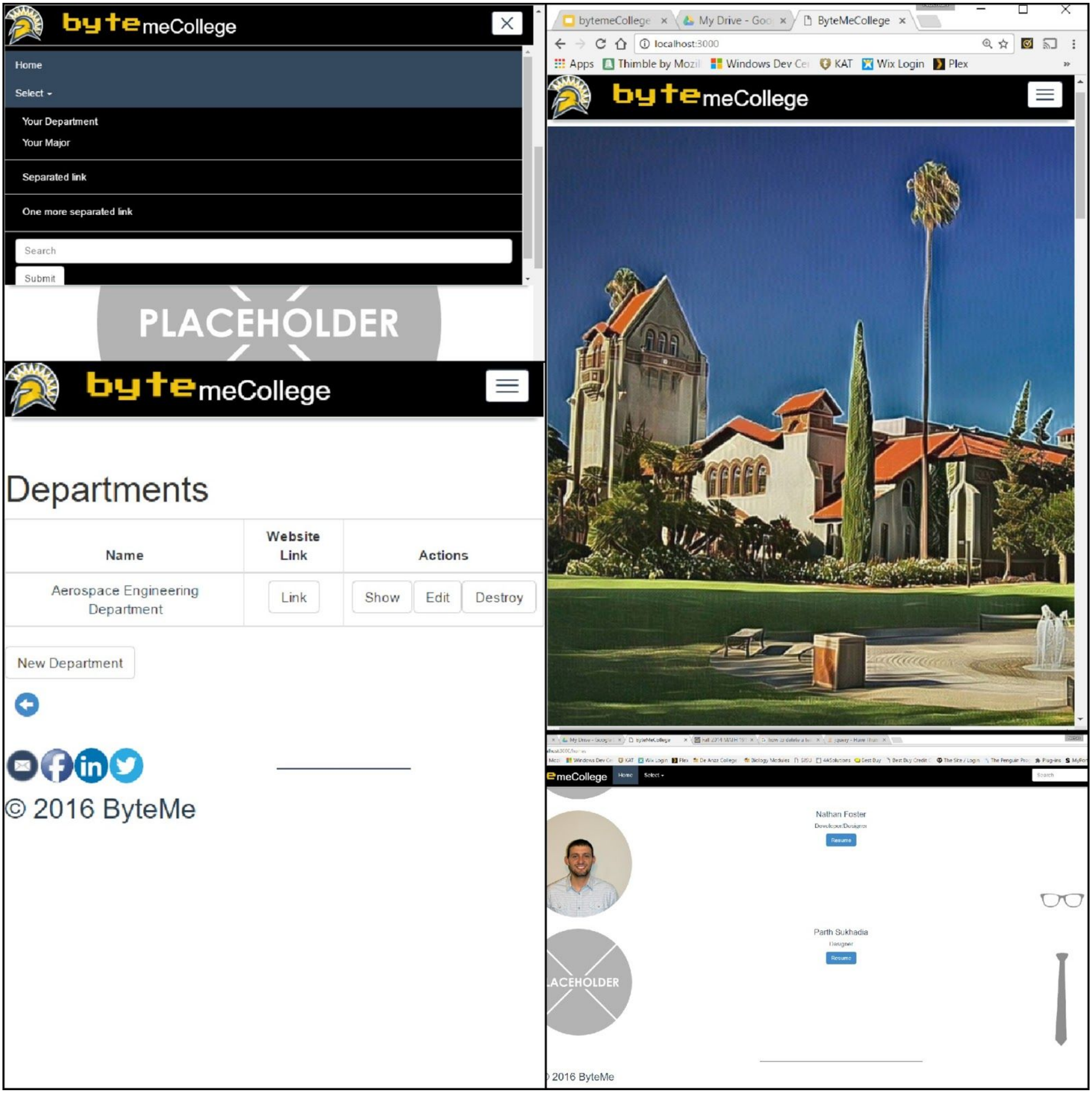
+Requet to add class
+Class search

ByteMeCollege.com/Courses

+Viewing courses
+Adding course review

Conceptual design

Screen Shots



Departments

Name	Website Link	Actions
Aerospace Engineering Department	Link	Show Edit Destroy

New Department

© 2016 ByteMe

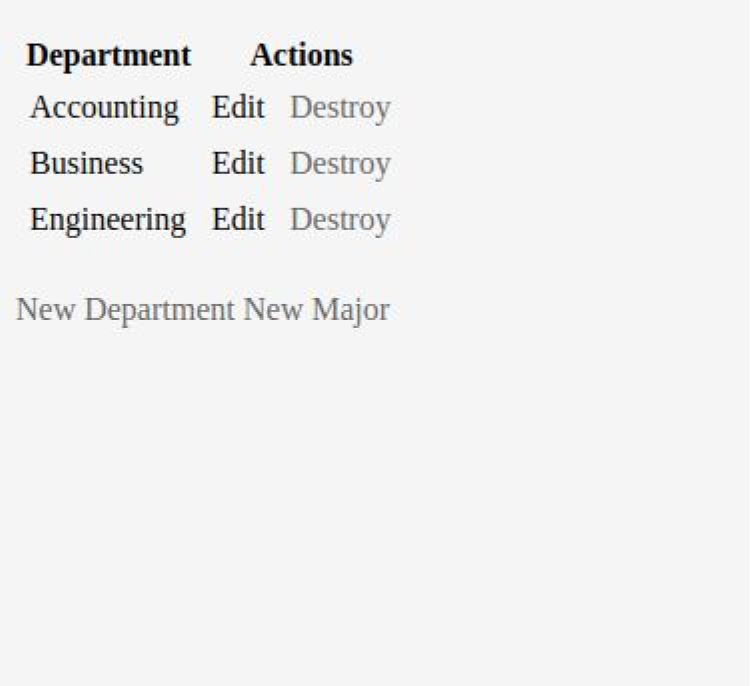
Nathan Foster

Developer/Designer

Parth Sukhadia

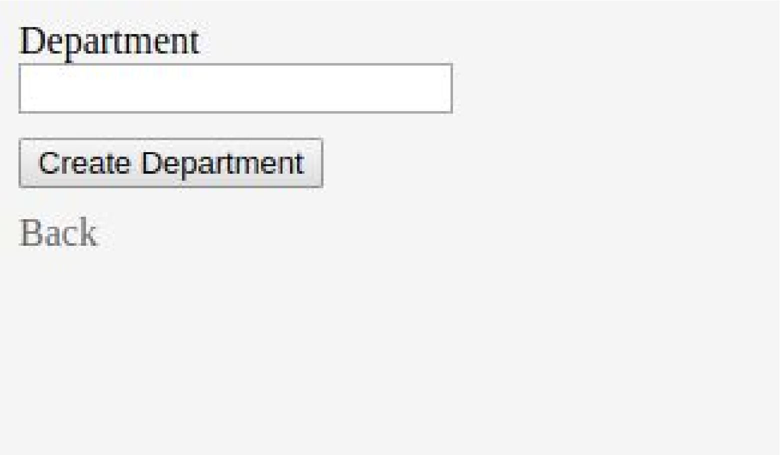
Designer

© 2016 ByteMe

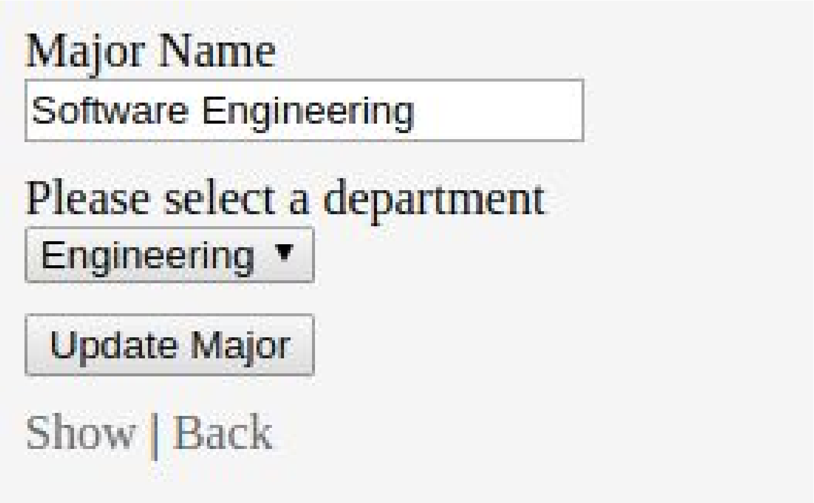


Department	Actions	
Accounting	Edit	Destroy
Business	Edit	Destroy
Engineering	Edit	Destroy

- Admin can simply add and modify new departments or majors.
- Once a department or major is created or modified it is stored into the database.



Department

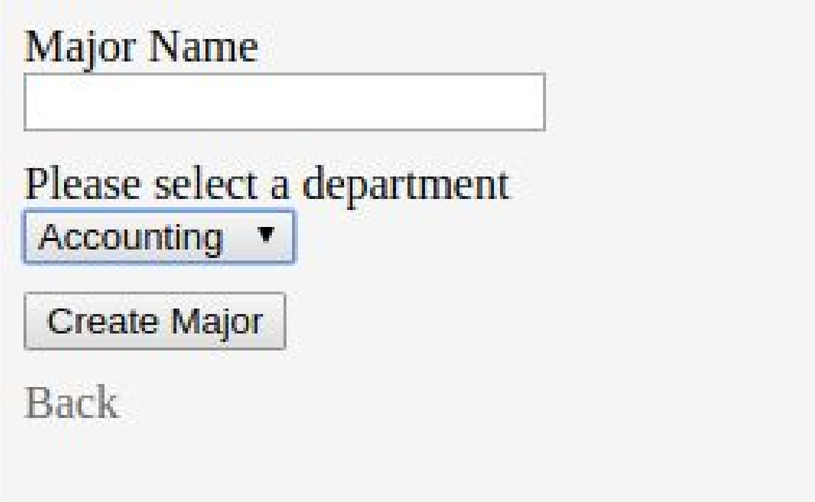


Major Name

Software Engineering

Please select a department

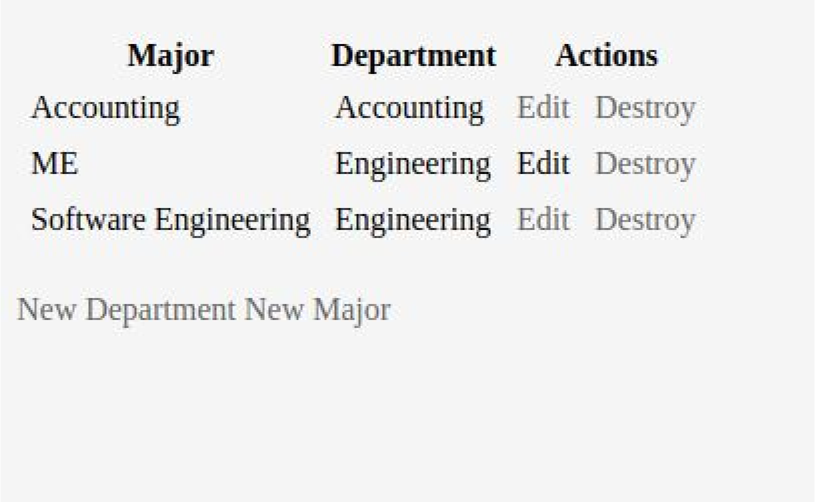
Engineering ▼



Major Name

Please select a department

Accounting ▼



Major	Department	Actions	
Accounting	Accounting	Edit	Destroy
ME	Engineering	Edit	Destroy
Software Engineering	Engineering	Edit	Destroy

What are we?

The main objective of ByteMeCollege.com is to provide user verified reviews of courses offered by different academic institutions (currently focusing on SJSU). We hope to help students by providing constructive information to help them choose the courses that are most relevant to their academic goals and interests. Students can also do real world comparisons of similar courses offered by various institutions to help them choose the right academic programs and most fitting career path for their educational goals.

Why this Project?

College students do not have enough information about the courses they take before enrolling. They need a platform to decide relevant courses to their academic interest from offered options. While choosing a university to attend, students don’t have a one-stop-shop to compare similar programs offered by different academic institutions.

How is it different than RateMyProfessor?

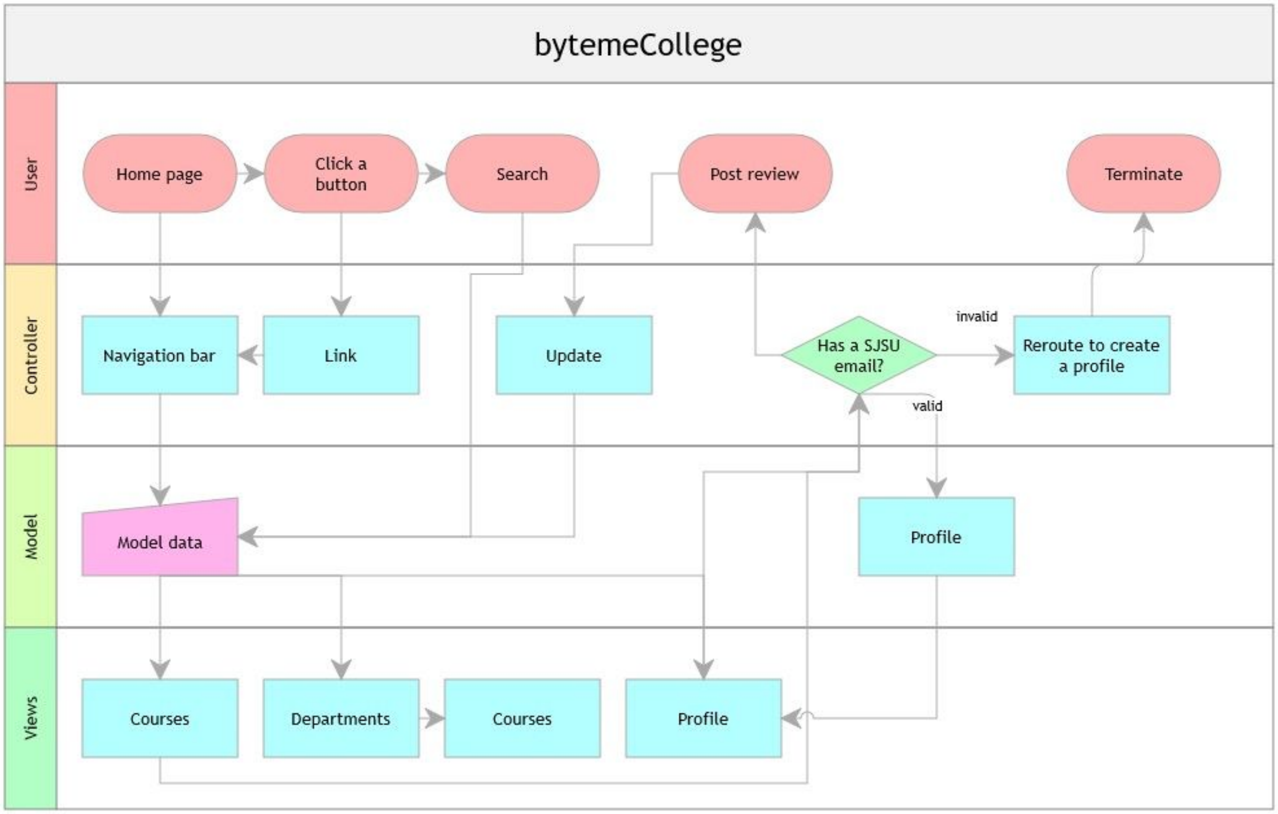
- Classes are rated instead of professors
- Users are encouraged to provide constructive reviews of the course content, class structure

and what they got out of it instead of how hard the professor is.

- Only students with registered university email addresses can create accounts and leave reviews.

- User can compare courses between colleges while making decision on what college to go to.

Diagram


bytemeCollege
User
Controller
Model
Views
Home page
Click a button
Search
Post review
Terminate
Navigation bar
Link
Update
Has a SJSU email?
invalid
valid
Reroute to create a profile
Model data
Profile
Courses
Departments
Courses
Profile