I. Introduction .................................................................................................................................................. 2

II. Learning Goals .......................................................................................................................................... 3

III. Approach ................................................................................................................................................ 4

IV. The Software: ......................................................................................................................................... 5

V. Milestones .................................................................................................................................................. 6

VI. The LifeCycle ......................................................................................................................................... 8

VII. Software Requirements Specification .................................................................................................. 10

1. The Purpose .............................................................................................................................................. 10

2. Similar Websites ....................................................................................................................................... 10

3. Definitions ................................................................................................................................................ 11

4. Use Cases ............................................................................................................................................... 12

5. System Sequence Diagram .................................................................................................................... 23

6. Revised Requirements ............................................................................................................................ 25

VIII. Analysis

1. Functional modeling ............................................................................................................................... 27

2. Noun Extraction ..................................................................................................................................... 30

3. Entity Modeling ....................................................................................................................................... 32
I. Introduction

In the Computer Science and Engineering department of California State University, Bakersfield, each student is required to design, document and implement a final project to show that they can apply the skills which they have developed through the courses of the program. Our team consisting of two students, Joel Medina and Simranjeet Singh have decided to take on a project which requires the use of tools which we are not very familiar with. Web design and the use of the tools to implement websites are not widely covered by the department. We have decided to take on a large project consisting of designing and implementing a fairly large website which makes the use of modern and emerging technologies. We agreed this project would prepare us appropriately for a real world scenario in which a small team is given a large project which the team members are not familiar with. Also the fact that not very much time (10 weeks) is given to implement the project gives the team members a high sense of urgency and necessary commitment which is often faced in the workplace.
II. Learning Goals

Learning the most during the development of this software is the ultimate goal which both developers have. There are very many different types of technologies being used to design and build high quality websites. It is our goal to make use of these emerging technologies in order to become familiar with them enough to utilize them effectively in future projects, whether it is in the workplace or in personal projects. Also great familiarity with the software development cycle is necessary in order for both team members to become highly effective software engineers.

After having completed the project, the developers will be highly familiar with:

- Use Case Diagrams
- System sequence Diagrams
- Business Modeling
- Component Design
- Entity Class Modeling
- Communication Modeling
- Interface Design
- Architectural Design
- Detailed Design

Also the developers will gain significant experience in the following technologies:

- AJAX (Asynchronous JavaScript And XML)
- PHP (PHP: Hypertext Preprocessor)
- MySQL
- JavaScript
- JSON
- JQuery
- HTML5
- And any other technology that is found applicable and worthy
III. Approach

We have agreed that by following the appropriate software engineering methodologies we would be able to gain the proper experience required to develop software. A strict set of software engineering principles will be applied throughout the development of this project. Some of the practices which will be used include: the use of use case diagrams, following a specific software lifecycle which best fits the development of the project, and designing the proper diagrams throughout the lifecycle (ER Database model, component design, interface design, flow charts and Architectural Design). It is important to follow the correct software development phases in order to track progress of the software product and easily target any problems by documenting every aspect of the software. This practice will also facilitate future changes to the software, code reuse and will also help avoid reinventing the wheel.
IV. The Software:

Our website will consist of the interaction between students and other members of the community with each other by asking questions relevant to common course-subjects taught in universities, and having them answered by another member of the community. The website will be a Web 2.0 type of website meaning that it is not static, and it will be primarily generated by the users of the website. The site will have sub directories for every course-subject of every discipline which will contain the questions and answers made by the users. Those sub directories will also have content relevant to the course-subject such as job opportunities, current news and books on the subject.

V. Milestones

Milestones historically are stone markers placed on roads to mark every mile traveled. In software development it is very important to create a set of Goals to be achieved by the team at certain points in time. Productivity is a very important factor in software development and human motivation is one of the strongest contributors to productivity. Every time that our team
achieves a milestone, the team will be filled with satisfaction and be extremely motivated to move on and tackle the next sub-project. The following list will show an overview of the goals which must be met at the according times.

**Week 1**

During this week the software concept will be defined and the team will agree on what the software will consist of including its purpose and how it will be used.

**Week 2**

This week will be highly important in that it will be the week dedicated to the Requirements Phase in which all of the required functionality of the software will be explained in detail.

**Week 3**

The Component and Architecture of the software will be designed during this time frame to effectively implement the modules in future phases.

**Week 4**

In week 4 of the software development the team will focus on coding the sub-projects. At the end of this week the core of the website must be done, including the database, major classes which will be used to query and return data and the major pages of the website.

**Week 5**

After the websites core functionality is up and running, it will be time to add some of the features which will facilitate navigation of the website. During week 5 team members will be working on separate sub projects which include the addition of the capability to choose a course-subject from the home page and the ability to create a new profile.

**Week 6**

At the end of this week the following components must be in place and functioning properly:

1. Users must be able to post questions
2. Users must be able to post answers
3. Users must be able to message each other.
4. Users must have an inbox style page in order to view their messages

**Week 7**
Goals to meet by end of week 7:

1. Users must be able to post comments on each other’s questions and answers.
2. User must be able to rate comments, answers and questions.
3. Users must be able to follow Course-Subjects, Questions and other users.
4. Automatic updating of sections in the course-subject pages must be finished for the following sections:
   i. Related Jobs
   ii. Helpful Videos
   iii. Helpful links
   iv. Related Books

**Week 8:**

1. Complete all asynchronous components of the website.
2. Test every Component of the website.

**Week 9:**

1. Continue testing including the use of outside users to test the software.

**Week 10:**

1. Launch website and begin marketing.

### VI. The Life Cycle

Deciding on the appropriate Software Engineering Life Cycle is a crucial component of the software engineering process. Based on the time given to finish the software project, and the structure which our team has, we must decide on a life cycle which makes these conditions work in our favor instead of against our team. After analyzing our conditions and taking a look at widely
used software life cycles, we concluded that it is crucial for our team of two developers to be able to work simultaneously on separate independent modules throughout development. We also agree that it is very important to have all of the major components of the software designed before any development begins. Due to these requirements the software life cycle which we chose is a modified version of the waterfall life cycle.

The main difference between the waterfall lifecycle and the modified life cycle which we will be using is that after the architectural design of the software, the team members will begin to work separate portions of the software which will be integrated. Each team member will be working on a separate subproject, and when one subproject is finished and integrated, development of the next subproject will begin. The same steps will be iterated for each subproject until the 9th week of development, where testing of the complete software system will take place. After the system is verified and fully functional, then the release will ultimately follow.

The following diagram shows the lifecycle which our team will follow during the development of the project:
VII. Software Requirements Specification
1. The Purpose

The purpose of our site is to facilitate the spread of knowledge throughout students, professionals and other members of the community who have limited access to other resources. When a student comes to a roadblock or is seeking an answer to something which cannot be found elsewhere, there will be a good possibility that another person has encountered that problem and has the answer.

2. Similar Websites

There are several websites which consist of question and answer posting by the community. Examples include: stackoverflow.com (Q & A for programmers), Yahoo Answers (general Q & A), and Anwers.com (also general). What will set our website apart from what is already out on the internet will be the ease of use and structure of the website. We would like to make use and take advantage of existing and emerging technologies which create a more pleasant experience for the user. Techniques like automatic loading without page reload and infinite scrolling make the user experience a much more enjoyable interaction with the website and make all of the aspects of the website easier to navigate.

3. Definitions
It is important to have complete understanding of the terms that will be used in the following sections; these terms refer to parts of the software which people may not be familiar with. Once having a clear understanding of the terms, the rest of the documentation will be able to be understood and followed in a smooth manner.

The Following terms will be used throughout the documentation:

- **Post (noun):** In the context of this software the word “post” can be used as a noun to refer to content which has been added by a user to the website. This includes: questions, answers, comments and related content. (Verb) the word can also be used as a verb to mean to add the content to the website.
- **Follow (verb):** In the context of this software, to “follow” means to subscribe to certain content or to a post and be updated with the activity which that content or post undergoes.
- **Related Content (noun):** In the context of this software “related content” is the content which is in some way related to the “course subject”. This content may include: books, videos, links and jobs.
- **Course Subject (noun):** In the context of this software, a “Course Subject” is a page which pertains to a specific sub-field of study. Here users can post questions pertaining to the subject. Ex: Computer Science > Machine Learning. “Machine Learning” is the course subject and has a page of its own.

4. Use Cases:
Use Cases are essential to the requirements phase because it gives a detailed look into how the “Actors” will be interacting with the software. Use Cases are used to represent a certain aspect of the software product which the Actor, in this case a User, takes part in through some action or input to the software. These diagrams give the developers a much more detailed look into what the software truly requires. Multiple iterations are taken during the development of these use cases and a much more thorough understanding of what needs to be built is obtained during this process.

The following diagrams are the use cases which the team agreed are necessary to have. Each diagram is followed by a description of the use case.

The Use Cases are:

1. Log In
2. Register
3. View Course Home
4. View Related Content
5. View Question
6. Post Question
7. Post Answer
8. Comment a Post
9. View Message Inbox
10. Read Message
11. Rate a Post Up/Down
12. Compose Message
13. Follow Content/Post
14. Log Out

1. Log In
Description: In this use case the user has decided to log in. At this moment the user does not necessarily need to be a registered user, however, they will be given the option to become one. The extended portion of this diagram represents a condition that can be met, but does not have to be. For example, if the user is already registered, then they do not need to register again.

2. Register

Description: In this case the user is not yet registered and decides that he or she would like to register to the website in order to have posting privileges. The “include” portion of this diagram represents a necessary condition to this use case: it is necessary for the user to confirm the registration through an email confirmation.

3. View Course Home
Description: When the user finds interest in a certain course subject they decide to click on the link and various options are then presented. In this case the user does not need to be registered for they are only navigating the website and not posting any content. The “extended” portion of this use case depicts the options which the user has once they are in the home page of the course subject. Some of the options are applicable only to registered users.

4. View Related Content
Description: Related content can be viewed by either a registered viewer or visitor as depicted by this diagram. This is an option which is given to a user once they are viewing the home page of a course subject.

5. View Question
Description: In this use case, the user has decided to view a question. Once navigating to the page where the question resides, the user will be given various options. These options include: Posting a comment, Posting an answer, Rating a post up/down, or following the question. The user does not need to be registered in this case and can view as many questions as they would like to.

6. Post Question
Description: This case involves the user posting a question to the website. In this case it will be necessary for the user to be registered as shown by the note in the diagram. At this point the software must verify that the user is in fact a registered user, and not just a visitor. The user then waits for their question to be answered.
Description: In this case after the user has viewed a question of another user, they decide to post an answer to that question, again in this case the user must be a registered user. The “extended” portion of this diagram represents the option of following the question if the user would like to. If the user decides to follow the question, they will be notified of any significant interaction of other users with the question.
Description: This use case depicts the moment in which the user decides that they can give some valuable feedback to the post of another user. A post can be a question, answer or another comment. A necessary condition to take this action is for the user to be registered.

9. View Message Inbox

Description: Viewing Messages in the Inbox is another necessary action for the user to be able to do. This action requires the user to be registered, and once they are presented with the inbox, they will have the options of either composing a message or reading a message.

10. Read Message
Description: This use case depicts the action of reading a message, which only a registered user can do. After reading the message, the user has the option to compose a reply message.

11. Rate Post Up/Down

Description: Rating a post is also a necessary action which only a registered user can do. The user can either rate a post up or down meaning they either like it or dislike it.

12. Compose Message
Description: This Use case depicts the event in which a user decides to compose a message, in order for the user to have this option, they must be registered. Once the registered user composes a message, they then have the option of either discarding the message or sending the message.

13. Follow Content/Post

Description: In this case the user decides that they would like to follow a post, meaning that they would like to be updated on the activity that is taken within or on that post or content. To have this capability the user must be registered.

14. Log Out
Description: Here the user decided that they would like to finish the session on the website and log out. It is necessary for the user to first be registered in order to log out.

5. System Sequence Diagram
A System Sequence Diagram is a type of diagram which depicts in detail a specific scenario in which the actors interact with the system. This diagram will show how, as time passes, the actor makes decisions and the software system gives options based on the decisions made. This will help understand the process taken by the user and the website and how interaction will take place. This along with the use cases gives insight into what some of the components, namely the classes, will be.
Description: This System Sequence Diagram shows the process that is taken in order for a registered user to log in, choose a course subject, choose a question, and answer the question. During this process the software system must make several verifications to decide whether the user has these privileges.
6. Revised Requirements

Our site will have very many features which will be required to deliver a very pleasing experience to users and visitors of the website. It is essential to present to the user a website which can be navigated very fluidly by providing relevant content. As shown by the use cases and diagrams, the users must be able to do the following:

- Log in
- Register
- Post Questions
- Post Answers
- Post Comments on other Content
- View Questions
- View Course Home
- View Messages in Inbox
- Read Messages
- Rate Posts either up or down
- Choose Top Answers
- View Related Content
- Compose Messages
- Follow Content and Posts

In order to deliver these features in a manner which makes it much easier for the user to find them and make use of them, various technologies will be used. The primary programming languages and tools which will be used include: PHP, JavaScript, HTML5, MySQL, AJAX, JSON and perhaps some ActionScript. If tools or technologies which can be very useful are discovered during the development of the software then they will also be implemented.

Also techniques for bringing the relevant content from various parts of the internet to the Course Home will be explored. Perhaps design of algorithms may be necessary which will be explained in detail in the following phases. With a much better understanding of what the product consists of, the development team can move on to the next software development phase.
VIII. Analysis

The next phase necessary in the software development cycle is the analysis phase. In the analysis phase the primary goal is to extract the entity classes which will be used in the implementation of the software. The requirements phase has shed light on what will be needed of the software project, and in the analysis phase we will now determine the essential components of the software that need to be represented as classes in the code. However, we will not yet design those classes, only how each communicates with the other classes. Also a dynamic analysis needs to be done in this phase in order to fully understand the operations that will be performed by the system based on the user’s interaction. The dynamic analysis will be carried out by making use of a state chart to effectively depict all of the possible paths that a user can take when using the software system. As a follow up to the requirements phase, we will develop a functional model consisting of detailed scenarios.
1. Functional Modeling

   Functional modeling is a follow up section to the requirements phase. In this section we will examine a use case in great detail. Two scenarios will be portrayed in this segment of the analysis phase. One will be a normal scenario which will depict a scenario which will be common for a user to experience and the other will be an exception scenario which show a setting in which an uncommon occurrence will develop for the user of the software. The specific steps will be listed in order and a more extensive understating of the operations taking place in the software will be developed. After the scenarios are covered a method known as noun extraction will be used to extract the entities that will be represented as classes in the software.
The normal scenario which we will take a look at will be the following sequence of events:

- Logging In
- Viewing a Course Home
- Posting a question

This is perhaps the scenario which will play out most frequently due to purpose of the website and how users intend to make use of it. The following diagram shows the use cases involved in this scenario and is followed by the sequence of detailed operations that will take place both on the side of the system and the user.

1. The user arrives to our site
2. The user chooses to log in and clicks on the log in button.
3. The user is prompted for their email and password.
4. The user enters their email and then clicks on the password text box
5. The user enters their password in the password text box.
6. The user presses the log in button.
7. Verification is made of the user’s email and password by comparing with the database.
8. The user is brought back to the website homepage with a status of logged in.
9. The user clicks on the Computer Science Subject button.
10. A list of courses pertaining to computer science is presented to the user.
11. The user clicks on Database Systems.
12. The Database Systems Course Home is presented to the user.
13. The user clicks on the “Ask a question” button.
14. Verification is made to check if the user is a registered user.
15. The User is presented a text field to begin typing the question.
16. The User types: “How to connect to Oracle with C#?”
17. The user clicks the “Post Question Button”.
18. The Question becomes viewable in the Course Home page and is open to be answered.
The exception scenario which will be depicted will be a case in which the user attempts to log in but cannot remember their password. This is an exception scenario because it is not something that all users experience every time they visit the website. Because this scenario is a special case of the log in use case, only that use case will be needed.

1. The user arrives to our site
2. The user chooses to log in and clicks on the log in button.
3. The user is prompted for their email and password.
4. The user enters their email and then clicks on the password text box
5. The user enters their password in the password text box.
6. The user presses the log in button.
7. Verification is made of the user’s email and password by comparing with the database.
8. The database returns an error and the user is notified that a wrong password was inputted.
9. The user is prompted for their email and password again.
10. The user enters their email and then clicks on the password text box
11. The user enters their password in the password text box.
12. The user presses the log in button.
13. Verification is made of the user’s email and password by comparing with the database.
14. The database again returns an error and the user is notified that a wrong password was inputted.
15. The user realizes that they have forgotten their password.
16. The user presses the “Forgot Password” option next to the log in button.
17. An Email is sent to the user.
18. The user is notified that an email containing a new password has been sent to them.
19. The user clicks on the link in the email.
20. The user arrives to our site
21. The user chooses to log in and clicks on the log in button.
22. The user is prompted for their email and password.
23. The user enters their email and then clicks on the password text box
24. The user enters their password in the password text box.
25. The user presses the log in button.
26. Verification is made of the user’s email and password by comparing with the database.
27. The user is presented to the home page of the website.
2. Noun Extraction

In this section of the analysis phase a technique known as noun extraction will be used to determine the entities in the software which will be represented as classes in the code. The technique requires that a detailed description of the software be done in the form of a paragraph. The nouns in that paragraph will then be found and listed. With a broad understanding of what the software will consist of; due to the requirements phase, it can now be easily explained in plain English. After the nouns are extracted and listed, an ideal list of entities will be compiled and will ultimately become classes. The following paragraph is a detailed description of the project.
Our site is a website which will facilitate the interaction of students and professionals from different parts of the world. The site consists of a collection of pages which hold questions pertaining to a specific course subject in academic majors. Visitors who are not registered can read the questions and answers pertaining to the course subjects in the pages. Users who are registered have the ability to ask and answer questions. Each registered user will have a profile which will hold their basic info and information on their activities like questions asked and reputation. A reputation can be built by answering the questions of others and having those answers chosen as the best answer. Also, answers as well as questions can be rated either up or down; and based on these ratings; the owners of the question or answer will be rewarded with respect points. Comments will also be allowed to be posted on different types of posts such as on answers and questions. In addition to the exchange of answers and questions, the users will also have the capability of sending private messages to each other. One component of the website which will not be generated by the users will be the related content which will reside in the pages of the course subjects, this related content section will contain any books, jobs, news, videos and other material which is directly relevant to the course subject. This content will be extracted from different websites on the internet and presented in an orderly fashion. In order to keep track of content which a user finds interest in, registered users will also have the ability to follow content and be sent notifications when that content changes.

The entities which have been extracted from the descriptive paragraph are the following:

<table>
<thead>
<tr>
<th>Questions</th>
<th>course subject</th>
<th>majors</th>
<th>answers</th>
<th>reputation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comments</td>
<td>private messages</td>
<td>posts</td>
<td>Users</td>
<td></td>
</tr>
</tbody>
</table>
3. Entity Class Modeling

Now that the major classes which will be used in the code have been determined we can diagram their interconnection. The following diagram shows the classes as they are related to each other.
The classes which have been derived in this portion of the analysis make up the classes which will be main classes which the user will be interacting with, however other necessary classes which will control other aspect of the website will be necessary. In designing a website it is common to use the architectural design known as Model View Controller, or MVC, this model will be discussed further in the design phase and other classes which must be created but do not represent any part of the website or the user will also be introduced.