**Texas Tech University Computer Science Blog**

James Little

Jason Weber

Patrick Braud

Randall Harper

CS 4366 Senior Capstone Project

Texas Tech University

Fall 2015

****

1. **Introduction 3**
2. **Terminologies and Abbreviations 3**
   1. TTU 3
   2. CS 3
3. **Functionality 3**
4. **Functional Requirements 3**
   1. Users 3
   2. Front Page 3
   3. Post 3
   4. Search 3
5. **Role Assignments 4**

4.1 James 4

4.2 Jason 4

4.3 Patrick 4

4.4 Randall 4

1. **Use Case Model 4**
2. **Non-Functional Requirements 4**

6.1 Security 4

6.2 Performance 4

6.3 Reliability 4

6.4 Usability 4

6.5 Implementation Constraints 4

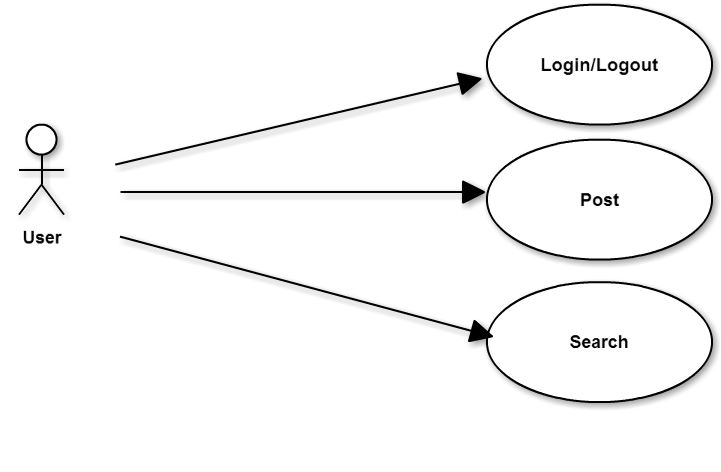
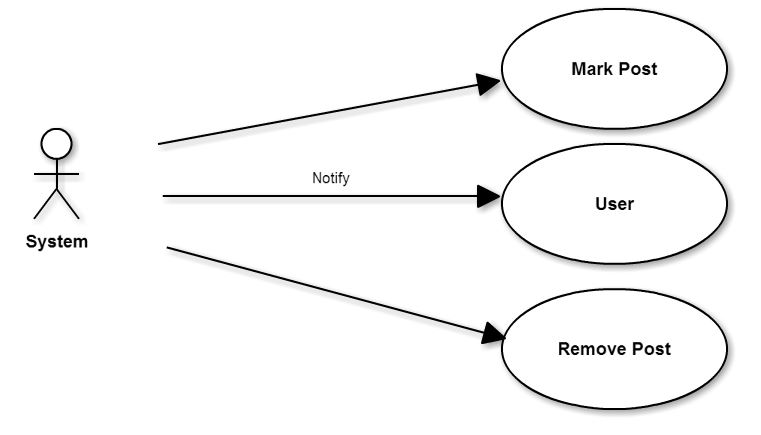
1. **Introduction**

TTU CS Blog will be a community hub where students can reach out to peers seeking advice ranging anywhere from class topics to research. Anyone can visit the forums, but only registered users may post a question to the community. Users will be able to search through the forums for specific topics.

1. **Terminologies and Abbreviations**
   1. TTU – Texas Tech University
   2. CS - Computer Science
   3. WAMP - A Windows Web development environment for Apache, MySQL, and PHP databases
   4. AWS – Amazon Web Services
   5. MySQL – an open source relational database management system.
   6. MAMP – A Macintosh Web development environment for Apache, MySQL, and PHP database
2. **Functionality**

3.1 Actors in the system are labeled as ‘registered user’, ‘guest’, ‘faculty’, or ‘administrator’. 3.2 Everyone can post in the forums except for guests. 3.3 Posts will fall under different topics ranging from programming languages to course discussions. 3.4 Everyone can search the forums. 3.5Posting questions will run off a points system. Users will need points to post a questions in the forums. Points will be awarded back to users for correct answers.

1. **Functional Requirements**
   1. They system shall have users.
      1. The system users will be registered, a guest, a moderator, or an administrator.
      2. The system users will have a photo associated with their accounts.
         1. The system user ‘guest’ will not have a photo.
   2. The system shall have a front page.
      1. The system front page will have blog links, a search bar, a blog archive, and posts.
   3. They system shall allow users to make posts.
      1. The system users will have different access to posting.
         1. The system user ‘guest’ will not have access to post.
      2. The system shall maintain a total number of topics and courses.
      3. The system users will have different topics and courses to post to.
      4. The system users will gain credit for correctly answering a posted question requested by another user.
      5. The system user ‘moderator’ will have to authority to close or move a post.
      6. The system shall automatically email the user who made a post when another user has attempted to answer.
      7. The system shall include a link directly to the user’s original post.
      8. The system shall mark each post as answered, unanswered, or recommended post.
      9. The system shall close a post if it remains unanswered after the time out period.
   4. The system shall allow users to search the blog.
      1. The system shall allow any user to search the front page of the blog.
      2. The system shall allow any user to search under any topic.

1. **Role Assignment**
   1. James: “Backend”
   2. Jason: Documentation
   3. Patrick: “Frontend”
   4. Randall: Testing
2. **Use Case Model**
3. **Non-Functional Requirements**
   1. Security: The system should have password encryption for its users.
   2. Performance:
      1. Response Time: The response time of the webpage will only be limited by the user’s internet connection. However, it will not take longer than 1 second to scroll up or down a webpage in the domain.
      2. Resource Usage: The webpage should not affect the system any differently than another similar webpage. Different web browsers may produce a different amounts of resources used.
   3. Reliability: The system should be accessible 99% percent of the time. The other one percent will be allotted for any server updates or maintenance.
   4. Usability: The webpage should be usable by anyone with basic computer skills.
   5. Implementation Constraints:
      1. Language: The system will be written in the PHP programming language.
      2. Operating Systems: The system will be available on Windows operating systems as well as Macintosh, since it is written in PHP.

All code and information is being saved and stored using Version Control / Switch Version Control called GItHUb. We are also implementing

* Bootstrap,
* HTML,
* PHP,
* Text Editor and
* IntelliJ

with each team member contributing to the project under different program environments.