Florida Atlantic University

CEN 4010 Principles of Software Engineering Fall 2019

Milestone 3 Project Proposal and High-Level Description

**Team Name**

Navigators

**Project Name**

Campus Snapshots

**Team Number**

16

**Students**

Alexandra Solares Ojopi

Dalton McClain

Emmanuel Sowunmi

Jim Brockley

Luis Gustavo Grubert Valensuela

**Documentation Date**

September 21, 2019

**History Table**

|  |  |
| --- | --- |
| **Date** | **Version** |
| 09/21/2019 | 1 |
| 11/03/2019 | 2 |

1. **Executive Summary**

Our application, *Campus Snapshots* will help provide Florida Atlantic University students with the best campus experience. Overflowing garbage cans, broken equipment, emergency situations and the like can detract from the quality of campus life. Through the application, students and staff will be able to report outages, maintenance issues, and incidents requiring police or security involvement. Doing so will make the user more of a part of the FAU community. Following a matrix barcode available throughout the campus, users be able to access the system. After registering, the user will be presented with a simple form to detail the item needing attention.

1. **Competitive Analysis**

|  |  |
| --- | --- |
| **Campus Snapshots** | **“Maintenance Direct” request** |
| User and mobile-friendly interface | Cumbersome web page |
| Easily accessed through barcodes placed around campus | Must navigate through massive FAU website |

The current system, “Maintenance Direct”, requires the user to pick from over forty categories of maintenance items. This is discouraging for a passer-by of an item needing attention to continue with the request. Our App will allow campus users to instantly report issues needing attention in a simple manner, requiring only a few details and photos if needed.

1. **Data Definition**
2. App: Our web system, “Campus Snapshots”
3. User: A registered client in the system that is able to submit a request.

A user can be categorized under the role of student, staff or

administrator.

1. User\_id: A unique identifier for each user
2. Ticket: A unique identifier for a submitted request
3. Post: A recorded campus issue
4. Post\_id: A unique identifier for each post
5. **Overview, scenarios, and use cases**

The Campus Snapshots app will be used by students and staff of Florida Atlantic University. Users of the app will be able to report outages, maintenance issues, incidents, and also activities that are happening on Campus. The user goals are to be able to post/report an issue or activity, comment on posted/published subjects, and also like or dislike a post. The following are the description of scenarios and use cases.

**4.1. Registration**

1. The first time the user enters the app, it will be prompted to register in the system.
2. After completing the registration, the system will provide a welcome message.
3. The user will then have access to the main page.

**4.2. Login**

1. If the user has already registered in the system, the app will prompt them to enter their login credentials.
2. After the user is logged in to the system, a welcome message will be provided.
3. The user will have access to the main page.

**4.3. Main Presentation**

1. The user will access the main page which will bring the most reported cases
2. The user can do the following tasks:
   1. Search for posts
   2. Comment on a specific post
   3. Update the status of a post
   4. Report/Post a new issue/activity
   5. Report an abusive post
   6. Report an emergency situation

**4.4. Reports**

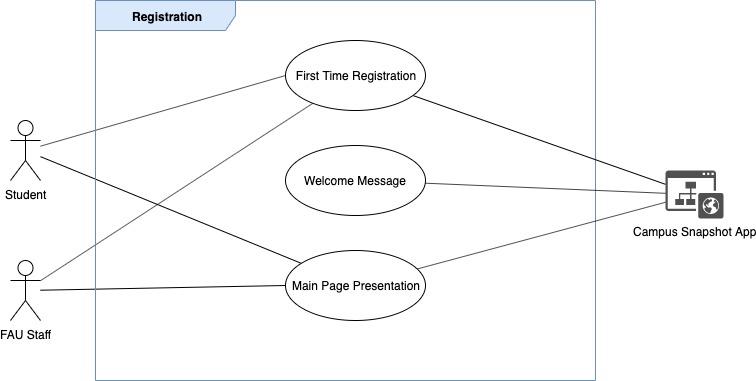
1. Presentation of Trending Unsolved Posts
2. Presentation of Trending Solved Posts
3. Presentation of users' posts statistics
4. Number of active users in the system (Administrator Only)
5. Number of inactive users in the system (Administrator Only)
6. Active posts that deserve immediate response (Administrator Only)
7. **Functional Requirements**
8. **Registration**
   1. User will provide name, email address, and a password in order to be able to access to the system. If the password is forgotten, a method of recovery will be provided.
9. **Login**
   1. Area with the following fields (Returning Users):
      1. Email address
      2. Password
      3. Forget Password Recovery
10. **Main Presentation**
11. This section will contain reported cases or posts, comments on a post, and the ability to update the status of a post.
12. **Reports**
    1. Solved and unsolved posts will be available to administrators of the system. A user may look up their own posts here.Solved and unsolved posts
13. **Non-Functional Requirements**
    1. Privacy Disclaimer available for all new users
    2. The downtime of the system is not so important, but the users should be presented a message telling that the system is down when they try to register/login to the system.
14. **System Architecture** **and Database organization**
    1. The system architecture will consist of an HTML front-end user form, PHP server-side code and MySQL database (LAMP).
    2. Bootstrap Framework will be used so the webpage is responsible for both computers and mobile devices.
    3. For coding, each developer is free to choose an IDE. As a suggestion, the free Visual Studio Code is recommended.
    4. Database Organization:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table** | **Field Name** | **Description** | **Data Type** | **Length** |
| USERS |  |  |  |  |
|  | user\_id | A unique identifier for each user | int | 10 |
|  | role | User’s role  (student, staff or admin) | varchar | 32 |
|  | name | Name of the user | varchar | 32 |
|  | username | Unique name used to identify each user in App | varchar | 32 |
|  | email | User’s email address | varchar | 32 |
|  | password | String of characters used to verify user’s identity | varchar | 64 |
|  | created\_at | Date user was created | date |  |
|  | updated\_at | Date user updated information | date |  |
| POSTS |  |  |  |  |
|  | post\_id | A unique identifier for each post | int | 10 |
|  | date\_posted | Date post was created | date |  |
|  | subject | Post’s subject | varchar | 100 |
|  | reporter | Name of post’s reporter | varchar | 32 |
|  | days\_active | Number of days posts has been active | int | 10 |
|  | views | Number of post’s views | int | 10 |
|  | status | Post’s current status (open or closed) | varchar | 32 |
|  | post\_comment | Post’s comments | varchar | 100 |

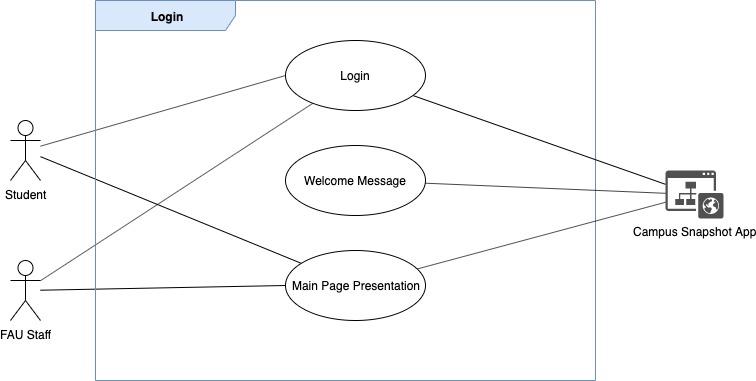
* 1. Search/filter architecture and implementation:
  2. Media storage:
  3. APIs:
  4. Search results will be displayed based on status with snapshots with “open” status being listed first.

1. **UML Diagrams**

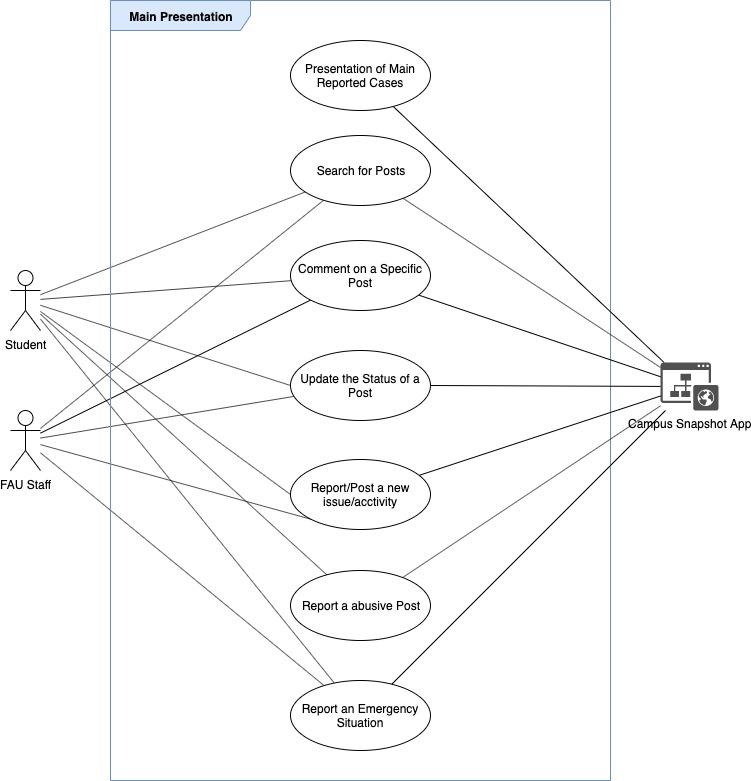
**8.1. Registration**



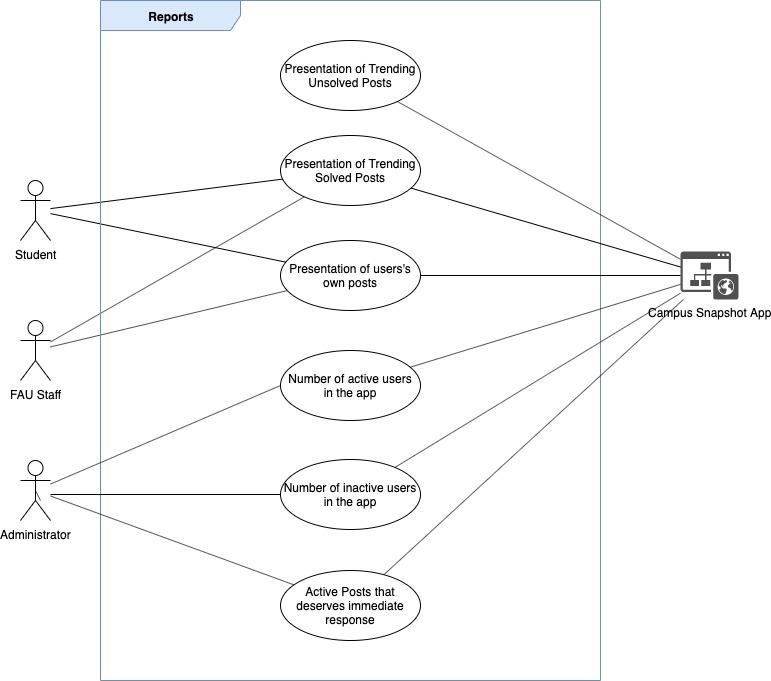
**8.2. Login**



**8.3. Main Presentation**



**8.4. Reports**



1. **Key Risks**
2. Adding unnecessary features and/or not including important features
3. Incomplete specifications and/or conflicting requirements
4. Rushed design process
5. **Team**

|  |  |
| --- | --- |
| **Name** | **Role** |
| Alexandra Solares Ojopi | Product Owner |
| Dalton McClain | Developer |
| Emmanuel Sowunmi | Developer |
| Jim Brockley | Developer |
| Luis Gustavo Grubert Valensuela | Scrum Master |