CEN 4010

Milestone 1 

Project Proposal and High-Level Description

**GROUP INFORMATION**

**Project name:**

Campus Snapshots

**Team Name:**

Group 3

**Team Members:**

Alina Tutuianu [atutuianu2017@fau.edu](mailto:atutuianu2017@fau.edu)

Christian Coronel [ccoronel2016@fau.edu](mailto:ccoronel2016@fau.edu)

Naiara Foster [nfoster2016@fau.edu](mailto:nfoster2016@fau.edu)

Prince Abraham [pabraham20152@fau.edu](mailto:pabraham20152@fau.edu)

**Trello account link:**

<https://trello.com/b/Z9kx4ifI/cen-4010-milestone-1>

**Course:**

CEN 4010 Principles of Software Engineering, Summer 2018

**Date**:

June 18, 2018

**Revision History:**

|  |  |
| --- | --- |
| Reason for Changing | Date |
|  |  |
|  |  |
|  |  |

1. **Executive summary**

This is the proposal for the development of the Campus Snapshots, a web system capable of displaying and alerting students’ real-time information about events happening on campus. In addition, the system will allow registered users to upload information to the website notifying issues or events happening in real time. An administrative team will be in charge of maintaining the webpage, reviewing reports and checking the uploaded news on the page.

The goal is to create a powerful and user-friendly tool for students, a one-stop experience, where they can find useful and reliable information about the school. In addition, Campus Snapshots will help to quickly and more efficiently solve problems by reporting in real time.

Campus Snapshots will be developed in the summer term, 2018, for the class Principles of Software Engineering by group 3. Its creation will significantly improve the way today’s students get information about events in their schools. Although the school offers an alert email system that notifies us of certain events and emergencies, Campus Snapshots will improve upon the concept and allow students to report events or issues themselves in a website environment, this feature will allow information to be exchanged quicker, making problem solving and information sharing significantly more dynamic.

The Campus Snapshots will be a website, so it won’t have a need for particular hardware or local software other than a web browser. The project will use a database to store events and user information. A user friendly website will be developed that showcases all the features available.

The project will be developed throughout the semester and it will be divided in teams and stages. There will be a web development team, a database management team and a testing and quality control team. In the first stage the group will discuss and decide the main features and visibilities for the website. Next, we will develop the website and implement the desired features, later, we will integrate with the database. The final stage will be to implement and test the website as users.

**2. Competitive analysis**

After deciding about the project’s theme and its main features, we analyzed other competitive products available today on the market. Following is a table comparing the key features of competitors versus our product. In comparing the products we focused mainly on the high-level functionalities.

**Campus Snapshots**   **Competitors**

|  |  |
| --- | --- |
| Report instantaneously issues | User reviews/feedback |
| Large Database | User friendly website |
| Search for School Activities | Easy issue upload/report |
| 24/7 Service/emergency | More variety of report tools |
| Directions on safety procedures | Data on previous reports |

There are many online school systems, where university campuses around the country can keep up to date and track on environment issues and school activities. However, some competitors do not offer a 24/7 events/issues happening in real time. Competitor’s products are mainly a news page, rather than a tool for detecting and solving problems on campus. Campus Snapshots will offer a better quality of service for students to feel safe, a good living and good study environment keeping the campus clean and safe.

**3. Data definition**

1. Campus Snapshots — A dynamic web system that publishes information about on campus events and problems, in addition to allowing students to contribute by submitting information about issues happening in real time.
2. User – A person (student, faculty, staff) that utilizes the web system to make posts about events and/or problems.
3. Admin – A team consisting of members from the university’s facility management, responsible to monitor the activity Campus Snapshots, and solve the problems that are reported.
4. Problem — Any event that can disrepute students’ everyday life (*Ex: A building build closed or school transportation system not working)*
5. Database — A collection of information that is organized so that it can be easily accessed, managed and updated.

**4. Overview, scenarios and use cases**

Campus Snapshots is a web system that provides real time information about the life on a university campus. The system allows users to share events and activities that take place on campus, but also to report problems that need to be fixed. The administrators of the university, called in the document admins, can use the system to monitor the well functioning of the campus, taking action when necessary to solve the reported issues.

Scenario for reporting a problem in Campus Snapshots

The user that wants to report the problem has successfully logged on to Campus Snapshots. The user has one or more digital photographs documenting the problem saved on a personal computer.

After the user chooses to report the problem, the system will ask the user to complete a report form. The user is asked to choose from a list of possible problems, to choose the location, and fill in the date of the observation. Based on the specifications given by the user, the system will search for similar reports. It will then generate a list of posts containing similar problems, and the user will have to check that he/she isn’t reporting the same problem as other users. If the problem is unreported, a new post is being created. The user is required to write a brief description and to upload photos. On completion of the post, the system automatically sends a notification to the admins, and generates an on-screen message to the user that the problem has been successfully reported to the administrators.

The user will get notifications whenever the status of the problem was changed.

Scenario for monitoring problems in Campus Snapshots

The admin is successfully logged on to Campus Snapshots. The admin can view the problems classified in the following categories: reported, in process, solved. If the admin chooses to see the reported problems, the system will display a list of the new posts. Each post has a status, which the admin can change. If the admin chooses to change the status of a problem to *in process*, the system will generate a form asking the admin the name and contact information of the person in charge of solving the issue. If the admin chooses to change the status to *solved*, the system automatically sends a notification to the user posting the problem, and to the other users that subscribed to the thread.

Use Cases

I, in the role of a user, need the functionality of creating a new post, to achieve the goal of communicating an event or a problem from the university campus.

I, in the role of a user, need to be able to upload photos, to achieve the goal of documenting a post.

I, in the role of a user, need the functionality of reading and adding comments on existing post, to achieve the goal of creating a thread.

I, in the role of an admin, need the functionality of notifications, to achieve the goal of being informed about the current problems reported.

I, in the role of a user, need the functionality of notifications, to achieve the goal of being informed about the status of a problem.

I, in the role of an admin, need the functionality of a problem status, to achieve the goal of updating the status of a reported problem.

**5. High-level functional requirements**

1. The application shall offer information about events and problems

This functionality allows users to make a post regarding an event or a problem. In the case of reporting a problem, the user shall fill in a form with details about the issue. The problem reported shall be kept in a database.

1. The users shall be able to upload pictures

The users can document their report with pictures.

1. The application shall offer a notification system

The notification system shall alert the admin in the case of a reported problem, and shall also notify the user if the status of a reported problem has changed.

1. The users shall be able to add and read comments on existing posts

Sharing opinions shall be facilitated with the comments functionality.

1. Each reported instance should have a status report (reported, in process, resolved)

The admin shall be able to monitor the reported problems, and to take action in solving them. This functionality will help to make the process of solving the problem more transparent.

**6. Non-functional requirements**

1. The application should run on the latest versions of the following web browsers: Chrome, Internet Explorer
2. Data shall be stored in the database on the server
3. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users
4. Security of the site shall require users to register and login in order to make a post
5. The language used shall be English
6. There should be no prior training required for using the application
7. Site shall be attractive and media rich in appearance
8. The site shall be easily searchable by the major search engines

**7. High-level system architecture**

Following is a list of the main software products, tools, languages and systems that we will use in the development of Campus Snapshots:

Back end: MySQL and PHP  
Front end: HTML5, CSS, JQuery , JavaScript  
Other: Internet explorer, Chrome, Mozilla, Microsoft edge, GitHub, Canvas, MEETS for WebEx, WhatsApp, Atom and Brackets.

**8. Team Members**

Alina Tutuianu (Frontend Developer)

Christian Coronel (Product Owner, Frontend Lead)

Naiara Foster (Scrum Master)

Prince Abraham (Backend Lead, Github Master)

**9. Checklist**

Team decided on basic means of communications - Done

Team found a time slot to meet outside of the class - Done

Front and back end team leads chosen - Done

Github master chosen - Done

Team ready and able to use the chosen back and front-end frameworks - Done

Skills of each team member defined and known to all - Done

Team lead ensured that all team members read the final M1 and agree/understand it before submission - Done