**Milestone 1**

**Project Proposal and High-Level Description**

**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)

**Course:**

CEN 4010 Principles of Software Engineering, Summer 2018

**Date**:

June 18, 2018

**Revision History:**

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

Executive summary

Competitive analysis

Data definition

User – student, faculty, staff

Admin – facility managers

Problem

Event

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. In 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

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.

High-level functional requirements

1. The application shall offer information about events and problems with text and photos
2. The users shall be able to add and read comments on existing posts
3. The users shall be able to upload pictures and create a report thread
4. Each reported instance should have a status report (reported, in process, resolved)

Non-functional requirements

1. The application should run on the latest versions of the following web browsers: ....
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 *(we have to decide for which functions)*
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

High-level system architecture

Team

Product Owner:

Scrum Master:

Project Team: