Course Title and term: CEN 4010 Principles of Software Engineering, Summer 2018

Milestone 1 Project Proposal and High-level description

Group 2 Campus Snapshot

Team 2

Cameron Sarni: [csarni2015@fau.edu](mailto:csarni2015@fau.edu)

Moise Medina: moisemedina2014@fau.edu

Date 6/18/18

|  |  |
| --- | --- |
| History Table | |
| 6/18/18 | Document Created |
| 7/23/28 | Milestone 3 additions |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1.Executive Summary:

The inception of “Campus Snapshot” (which we call CSnap) came out of discontent at the current system in place for reporting complaints and an even greater lack of gratitude given when these complaints are solved. With CSnap reporting a complaint is made as simple as taking a picture and tagging a location with a short description. The primary target of this product is the education sector focusing on college and high schools where students and faculty have smartphones and laptops to report incidents. With the target market in mind the functionality of Csnap has been designed with fluidity in mind keeping the access of its features simple and easily accessible while still performing the tasks which are required of it. Tasks such as taking pictures to upload to Csnap or sharing your location with the party in charge of the complaint have become simple thanks to the coding of our team. We are always innovating and updating the application to better fit the needs of our current and future users. Even right now features such as a maintenance tracker and an ability to communicate straight from Csnap are being developed for our users. With the rate that it is expanding it is no wonder why we are so popular with our users.

2.Competitive Analysis:

|  |  |
| --- | --- |
| Campus Snapshot vs Competitor | |
| GPS location | User input description of location |
| Optimized Accessibility | Limited Accessibility |
| Optimized for quick response | Archaic response process |
| Able to Track progress of request | No tracking of request |
| Customized to meet specific user demand | Generic layout for all users |

3.Data Definition:

CSnap: Shorthand for Campus Snapshot

modern devices: Refers to smartphones, laptops, smartwatch, google glass and all devices which can connect to the internet

streamlined: Made as fluid as possible with the least amount of barriers

CRS: Complaint Reporting System

Complaint: Refers to any issue or incident which is reported by users of CSnap

4.Overview, Scenarios and use cases:

CSnap uses the mobility and accessibility of modern devices to create a streamlined CRS which can be used by a variety of users to report complaints in their area to the proper officials. The CRS then notifies the designated officials who oversee the area where the complaint exists.

Example 1: Beth the Freshman

Beth is a college freshman who has just moved into her dormitory and realizes that the shower drainage system is not working properly resulting in it filling with water. She then takes out her smartphone and opens her favorite version of the CSnap application, she is then prompted to either report a complaint or check on an existing complaint as she is already logged into the application. She selects “submit new complaint” and CSnap accesses the camera of her device allowing her to take a picture of the clogged drain. Since her dorm room was already set as her location she just adds a short description and submits to complaint. A short while after she submits the maintenance arrives to fix the clogged drain, the maintenance then verifies he is at the right location and proceeds to fix the drain after proper authorization from Beth.

Example 2: Tom the Maintenance

Tom works as maintenance personnel on a college campus. He is an end user of Csnap utilizing it to maintain the campus buildings and surrounding grounds. One day he gets a notification from the Csnap desktop application that a freshman college grounds resident name Beth has filed a complaint in his area. From the complaint form he sees that the drain in her shower is clogged leading to flooding of the shower area. Immediately he sets the progress as ‘viewed’ also allowing Beth to contact him by supplying his email and phone number. He then(using the maps provided by Csnap) navigates to Beths’ dorm room and knocks on the door allowing the resident to answer. He then confirms that it is Beth and the complaint is valid, this leads to him again adjusting the status to ‘in progress’ and he works to correct the complaint. After he is done with the complaint he allows Beth to view that the shower is working and select that he has completed the work.

5.Initial list of high-level functional requirement:

* Google location Services: Used to track the location of Users and where the complaint occurred. Will be used for easier submission of complaints and a streamlined process
* Camera: Use of the users’ camera to document the complaint and verify its existence
* Tracking Progression: Updates the user as the complaint is being resolved
* Live agent dialogue: Used to confirm and determine severity of complaints

6.List of non-function requirements:

* Storage: Size of application in comparison to space available on device; Storage of complaint photos submitted by users for future usage
* Usability: CSnap is engineered for simplicity on all devices as to accommodate all types of users
* Accessibility: There currently exists a CSnap software for every device on the market which is capable of internet connection and is equipped with a camera.
* Security: CSnap is up to date on all security standards and requirements given by the governing body where the User base exists. But as a company we attempt to exceed those requirements, as we value and understand our users’ security is of upmost importance.

7.High-level system architecture:

* Supported on all browsers
* Using PHP, HTML, JavaScript, CSS, AJAX, SQL
* PhpMyAdmin
* Hosted by FAU LAMP servers
* Google API
* Adobe Photoshop & Illustrator

8.Team:

Cameron Sarni: Co-Owner

Moise Medina: Co-Owner

9.Checklist:

1. Team decided on basic means of communications: DONE
2. Team found a time slot to meet outside of the class: DONE
3. Front and back end team leads chosen: DONE
4. GitHub master chosen: DONE
5. Team ready and able to use the chosen back and front-end frameworks: ON TRACK
6. Skills of each team member defined and known to all: ON TRACK
7. Team lead ensured that all team members read the final M1 and agree/understand it before submission: DONE
8. Vertical Demo: DONE