Team #3
Chris Colella
Team Member 2
Team Member 3
Team Member 4

campusCall
Software Requirements Specification
Version <1.0>

Revision History

Date	Version	Description	Author
10/21/2021	1.0	Added a dashboard feature for teachers	Chris
<date here>	<version here>		
<date here>	<version here>		
<date here>	<version here>		

Software Requirements Specification

1. Introduction

campusCall is a centralized web application where a student and teacher can access the information they need, in a centralized location.

1.1 Purpose

The purpose of this document is to showcase, analyze the ideas around campusCall, and how campusCall takes an innovative approach to addressing dated problems. We will evaluate flaws within current methods that campusCall aims to resolve. This document will include detailed information for who the end users and customers are, along with how it shapes their current work flows. The general overview of campusCall is to utilize technology to make lives easier and more efficient, providing our customers with a new, more convenient approach to consistent problems, and be a better alternative to the current market of school, web-based software.

1.2 Scope

Within our Scope:
As mentioned in our proposal paper the things that we can deliver within the time period agreed upon (4 months) are:

1) Web Pages required:

2) Login
   a) Username
      i) Email Address
   b) Password
   c) Register/Signup
      i) Form

3) Dashboard (Student)
   a) Schedule for the day/All current classes (Student) & (Teacher)
   b) Links to other features of the website, this is your homepage.
   c) Tasks, todo/feedback

4) Dashboard (Teachers)

   a) View current courses being taught

      i) Manage assignments/grades

      ii) Manage attendance/announcements

5) Calendar (API)
   a) Uses username/email address for calendar
   b) Updates using Gmail (this is a Gmail calendar)

6) Announcements
   a) Professor announcements
   b) IT/Maintenance announcements

7) Course Page
   a) Courses taken in the past/all courses

8) Communication page
   a) Messages from other students/project teams
      i) Share schedules
   b) View your team/current invites

9) Settings
   a) Notification alert customization
   b) Change password
   c) Add/change mode of contact

Outside of our Scope:

Some of the ideas we are hoping we can implement in the futures include:

1. Currently in regard to attendance the process is not completely automated. In the future we will be automating it through the usage of a random code generated for each class that only the students that are attending the lecture can view.
2. Within our scoop we will be able to create a messaging service between students, but it will lack video and voice call capabilities. In the future we are planning on adding that feature.
3. Adding the ability to create online tests or quizzes.
4. Allowing students to view recordings of lectures that are recorded through zoom or Panopto.

1.3 Definitions, Acronyms, and Abbreviations

LMS	Learning Management Software
SQL	Structured Query Language
PHP	Hypertext Preprocessor
HTML	Hypertext Markup Language
DBMS	Database Management System

2. Overall Description

Flaws with the current systems used in universities and schools.

From a student's perspective, we strive for ebb and flow. Canvas, in its current iteration, is clunky, lacks essential features, and doesn't seem to have the student and professor as the focal point. With campusCall, we tailor the features to make for a more user-friendly experience that benefits both students and professors.

Currently, most universities use a sign-in sheet that is handed in class for students to sign to mark absences and track attendance. This is both time-consuming and disruptive; the sheet being passed around shifts the student's attention to the attendance during lecture and the method, more often than not, requires students to move around to pass the paper.

Professors also use excel sheets to store the data which is time-consuming and there is a high margin of error since most professors teach multiple classes and doing it manually always has a risk.

Currently, the professor has to create a quiz in order to make a survey to ask for any feedback or any questions. The problem is that most students would consider the survey as a quiz and would not provide valid/full answers. Another problem is some students cannot be open and honest when giving feedback because they don't want to risk their relationship with the professor.

And group projects and teams require students to use external software that is not attached to the university. In its current state, Canvas does not allow teams to submit projects together. The typical and archaic method is to have one individual submit on behalf of the group. This creates disarray amongst students and professors. With campusCall, you are able to create teams that will allow for group submissions.

The solutions that campusCall provides.

Our software-to-be would be able to shift the responsibility from the professor to track absence to instead having the students accountable for attendance themselves. By allowing the students to enter in a specific, unique code, we're able to take away the time-consuming and disruptive process of passing around the attendance sheet.

If we use polls the professor can have instant anonymous feedback. The professor can track answers easier and allow students to make suggestions. Additionally, this will allow the professor to effectively collect ideas and suggestions, then discuss said submissions. By using the software as an extra set of ears, this gives an opportunity for the quiet students to speak their minds. campusCall is a one-stop-shop for all the students and professors alike.

3. Specific Requirements

3.1 Functionality

Introduction

This subsection contains the requirements for campusCall. These requirements are organized by the features discussed in our proposal document. Features from the proposal are then refined into use case diagrams and to sequence diagrams to best capture the functional requirements and improvements that campusCall will provide.

3.1.1 ***Feature 1: Login for Student/Teacher***

3.1.1.1 The system shall allow student users to login and view their customized campusCall page for viewing and other activities.

3.1.1.2 The system shall allow Teachers and Professors to login to view their campusCall page.

3.1.1.3 The system shall allow users to recover their passwords if they forgot them.

3.1.2 ***Feature 2: View Class Schedule***

3.1.2.1 The system shall allow Students to view and organize their class schedule in relation to their current semester.

3.1.2.2 The system shall allow Professors to view their daily schedule and update dates or cancel certain appointments.

3.1.2.3 The system shall allow Students to view their daily tasks/todo list.

3.1.3 ***Feature 3: Update/Assign Projects***

3.1.3.1 The system shall allow Professors/Teachers to assign homework to any class they teach.

3.1.3.2 The system shall allow Professors/Teachers to edit assignments currently assigned, or assignments assigned in the past.

3.1.4 ***Feature 4: Polling***

3.1.4.1 The system shall allow Professors/Teachers to send a poll out to the class.

3.1.4.2 The system shall report all feedback back to the professor.

3.1.4.3 The system shall allow Students to provide fill in and multiple choice feedback.

3.1.5 ***Feature 5: Add or remove students from roster***

3.1.5.1 The system shall allow a Professor/Teacher to add a new Student to their current roster.

3.1.5.2 The system shall allow a Professor/Teacher to remove Students to their current roster.

3.1.5.3 The system shall allow a Professor/Teach to take and download a classes attendance.

3.1.6 ***Feature 6: Announcements***

3.1.6.1 The system shall allow Professors/Teachers to send announcements to a class.

3.1.6.2 The system shall allow Students to view their announcements.

3.1.6.3 The system shall allow users to see announcements made by IT for maintenance hours.

3.1.7 ***Feature 7: Calendar***

3.1.7.1 The system shall allow Students to see when their classes are.

3.1.7.2 The system shall allow Students to see what days their assignments are due.

3.1.8 ***Feature 8: Communication Page.***

3.1.8.1 The system shall allow students to message other students in their team or class.

3.1.8.2 The system shall allow students to share their schedules with other students in their team or class.

3.1.8.3 The system shall allow students to view current teams they are in.

3.1.8.4 The system shall allow students to view current invites sent or received.

3.1.9 Feature 9: Settings

3.1.9.1 The system shall allow users to view their account information.

3.1.9.2 The system shall allow users to add a profile picture.

3.1.9.3 The system shall allow users to change their password.

3.1.10 Feature 10: Dashboard

3.1.10.1 The system shall display all classes being taken by a Student.

3.1.10.2 The system shall display tasks a Student has set as well as assignments due.

3.1.10.3 The system shall display all courses a Professor/Teacher is instructing.

3.2 Usability

3.2.1 Graphical User Interface

The user interface shall provide a clear, concise way for users to navigate the system.

3.2.2 Accessibility

The users will be able to change text size, allowing for access to visually impaired individuals.

French, Spanish, Mandarin, and Arabic users can customize web page language to fit their preference.

3.2.3 Privileges

The users will be limited to Students, Professors and Systems administrators.

The system shall provide access on and off campus for students, professors and Systems administrators.

3.3 Reliability & Availability

The system shall provide 24/7 access to students and professors.
The system shall maintain a 99.9% uptime guarantee.

3.4 Performance

The system shall provide live updates (2 seconds) when polling,

The system shall provide schedule and class information within 3 seconds

3.5 Security

3.5.1 Data Transfer

The system shall provide end-to-end encryption

The system shall utilize login authentication for its activities

The system shall use SQL as the database

The system shall use PHP and Javascript to interact with the database

3.6 Design Constraints

3.6.1 Development Tools

The system will be built using JavaScript, Java, and HTML/PHP

Relational DBMS

IntelliJ IDEA for Java development

Atom for JavaScript development

3.6.2 System Type

The system will be a web-based system