

Student Attendance System Case Description

Please Note: This description addresses all of the known requirements for the Student Attendance System case. The extent to which you should consider all of these requirements or just a subset of them depends on the assignment that you are doing while using this case. Please refer to the instructions for your assignment to determine the exact scope of requirements that you should address in your work.

Introduction

As an instructor at an iSchool in the Midwestern United States, Ken Tucker is required by his department to keep track of student attendance in all of his undergraduate courses. Presently, he has an approach to collecting attendance data that is very manual. He finds this system frustrating, and he believes that it sets a bad example for his students regarding the proper approach to data collection and processing. He wants to develop an automated system that is accurate and is easy to use for him and his students. While he expects that the system might eventually be expanded to be used by other instructors, he wants to develop this first version based on his requirements alone.

Current System

As stated above, the current system is mostly manual. Ken downloads spreadsheets of student information from the learning management system (Canvas) for each class. He then edits the spreadsheets to turn them into class sign-in sheets. Ken prints a new sign-in sheet for each class. Students sign the sheet as they arrive at class.

Ken keeps the completed sign-in sheets for past classes in a set of folders – one folder for each course that he is teaching. When he has time, he processes each of the sign-in sheets for a particular course against a master spreadsheet. Ken creates the master spreadsheets from the same student information spreadsheets that he originally downloaded from Canvas. The master spreadsheet has the same rows of students as the sign-in sheet. The master spreadsheet has a column for each class meeting. When doing processing of sign-in sheets, Ken types a value of 1 in the column to signify attendance at a class meeting. There is a final column that totals the attendance for each student. He can compare these total attendance figures to see which students have a pattern of missing classes.

The current system has many drawbacks. Students who arrive late at class often forget to sign-in. Occasionally, Ken will arrive at class having forgotten to bring the sign-in sheet. This requires a round-trip to his office to pick up a sheet. At the beginning of the semester, when students are adding and dropping classes, Ken needs to recreate the sign-in sheets at least once to include these changes in the sign-in sheets. Because of these add/drop changes, he often delays the creation of the master spreadsheets until after the add/drop period has passed. This delays the startup of processing the attendance sheets by several weeks.

New System Ideas

Ken wants the new system to be Web-based. It will have a student-facing portion and an instructor-facing portion.

The student-facing portion will allow the student to login when they arrive at class and record their attendance. Since Ken teaches classes in computer lab classrooms, he expects students to be able to login directly from their seat in the class. This will solve the problem of students arriving late and not getting a chance to sign the sheet on the instructor's desk. The system will need to be able to verify that the student really is in the class and not signing in from the local coffee shop. Ken has two ideas under consideration for the check of actual attendance. The first is limiting the ability of students to login from machines that do not have an IP address in the classroom. The other idea is to display a password on the white board in front of the class. While neither of these are foolproof, they are at least resistant to misuse. A related issue is that students should not be able to register their attendance outside of the proper class hours. This window might run from fifteen minutes before class to fifteen minutes after class. Another related issue is that students who are registered for more than one class that is administered under this system might need to choose which class they are attending from a list.

The student-facing portion of the system should also allow students to get reports of their attendance from the system. For each course, students should be able to generate a list of class meetings that have already occurred, an indication of whether they were present or absent, and totals for presence and absence.

The instructor-facing portion of the system should allow the instructor to set up record-keeping for each course and it should allow the instructor to get attendance reports for each course.

While setting up record-keeping for a new course, the instructor might need to:

- Setup a new course.
- Setup a new section number for a course.
- Setup an expected list of class meetings.
- Setup a list of students registered for the course.

Each of the capabilities above should have the typical Add/Change/Delete functions.

Special capabilities should be created to allow the instructor to make changes to the list of students registered for the course in some kind of batch mode. Ideally, this would allow the instructor to download a spreadsheet containing student names from Canvas and feed that spreadsheet to the Attendance Tracking System. The system would then automatically select adds, changes, and deletes of students to make the student list in the Attendance Tracking System match the student list in Canvas for that course.

Ken anticipates that some students will still approach him after class with the sad story that they forgot to register their attendance. Since they will no longer be able to login to the system and record their attendance, Ken would like a capability for the instructor to do this for students on an exception basis.

The instructor-facing attendance reporting features of the new system are as follows: A report should be available for each course that shows attendance and absence figures for each student for each class meeting. The report should also show to-date totals for attendance/absence in that course.

Scope-Related Issues

Ken would like to address most of the requirements described above in the first release of the system. One feature that he is willing to postpone is the feature that automatically coordinates lists of registered students between the new system and Canvas.

Known Functional Requirements

1. Student login and attendance reporting in classroom
2. Student generates a report of their own attendance / absence
3. Instructor setup of course
4. Instructor setup of section
5. Instructor setup of expected class meetings
6. Instructor setup of registered students
7. Instructor records student attendance on an exception basis
8. Instructor generates report of attendance / absence for entire class

Known Non-Functional Requirements

1. Support up to 5 simultaneous instructor users.
2. Support up to 70 simultaneous student users.
3. Provide users with 2-second response time or less.
4. System outages should be resolved within 15 minutes.
5. Data should be backed-up frequently to avoid data loss.