SOFTWARE REQUIREMENTS SPECIFICATION

POLLME

TEAM 2

Version	Written by	Revised by	Approved by	date
0.3	1. Amir Shobak 2. Hamza Shobak 3. Hamed Abdelrehim 4. Hassan Mohamed 5. Mirna Abdelaziz 6. Joseph Samir 7. Bishoy Abdel Sayed	Paula Ragheb	Joseph Samir	16/12/2018

TABLE OF CONTENTS

1. INTRODUCTION

1. PURPOSE

The purpose of this document is to present a detailed description of the Web Publishing System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

2. EXECUTIVE SUMMARY

PollMe is a website providing feedbacks to moderators or speakers by giving the audience or students a quick poll depending on the speaker's questions or opinions. PollMe can be linked to any official or virtual education system. PollMe should be considered as an official feedback that everyone can get information by it. Also, PollMe can provide the speaker with all the material they need by fast and simple ways.

3. DOCUMENT OVERVIEW

This document introduces PollMe web application. It is the software requirements specification document. It introduces general description, technical description, functionalities and it follows the standard IEEE of SRS issued in 1998.

4. GLOSSARY

Term	Definition
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
PollMe	Website name
Speaker	The admin of the website that makes the poll and receiving the results of the poll
Audience	Any person that fill in the poll and attend the lecture or the conference
Xref	A reference from one part of a document to another part containing related information.

5. REFERENCES

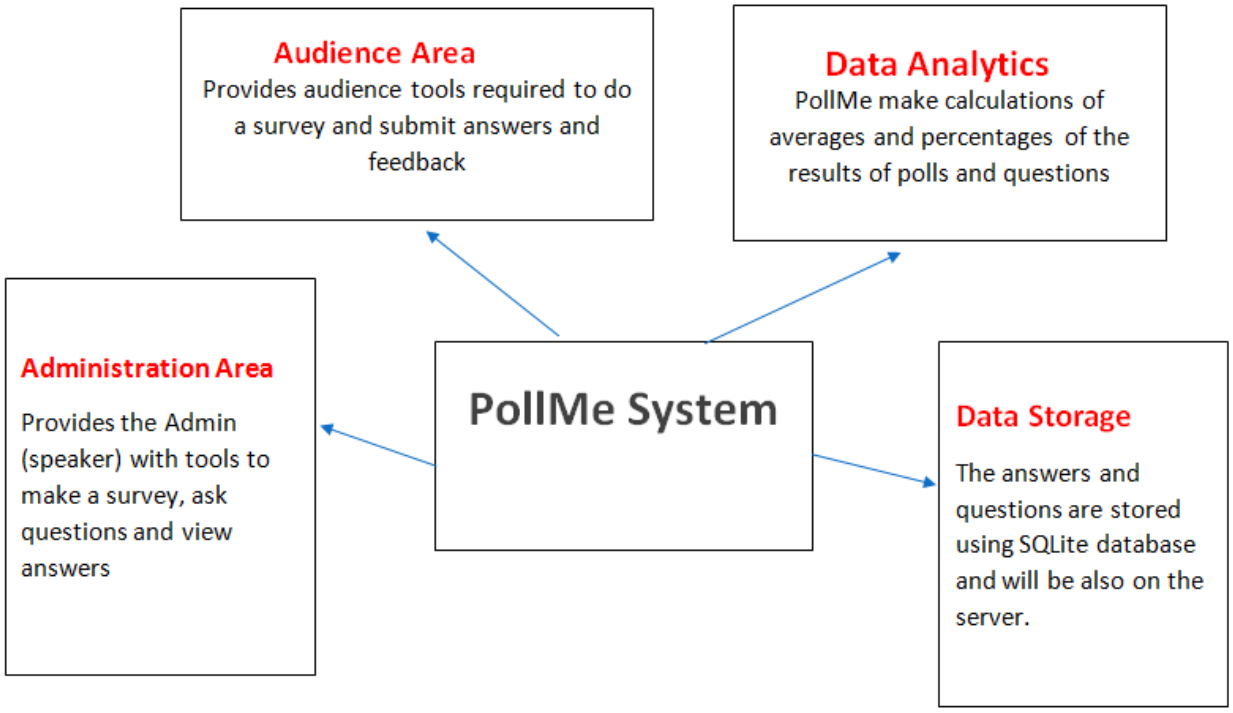
1. IEEE 930-1998 standard
2. http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_group2.pdf

2. OVERALL DESCRIPTION

1. SYSTEM ENVIRONMENT

PollMe System has many users who are the audience and one speaker who is also the admin of the system. Audience can take a poll through the local network. Any user (audience) communication with the system is through the home page. The speaker (admin) accesses the poll result through admin home page.

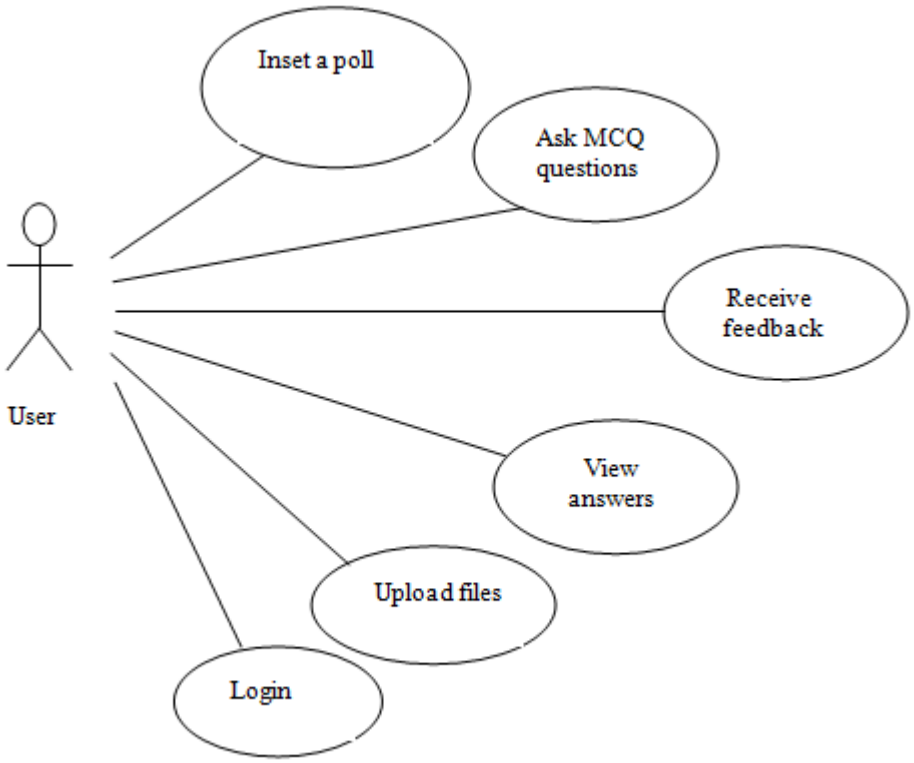
PollMe offers an easy way of interaction between a speaker and his audience. Through taking a poll, ask MCQ and see the percentage of the audience that answered right, get questions from the audience and see their questions on the speaker's (mobile / laptop). Also, it allows the speaker to get feedback about the topic introduced through a positive or negative poll, which he can see how many people in the audience understand the topic well. PollMe can be used in lectures, conferences, events and any other activities that need a feedback from the audience. The following context diagram shows the main parts of the system:



2. FUNCTIONAL REQUIREMENTS SPECIFICATION

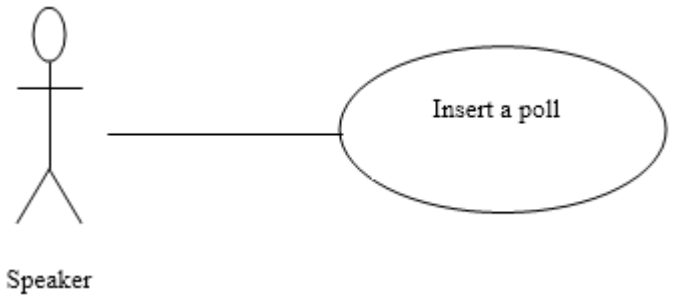
SPEAKER USE CASE

The speaker who is also the admin has the following sets of use cases



USE CASE: INSERT A POLL

- **Diagram**



- **Brief Description**

The Speaker submits a poll for the audience to take.

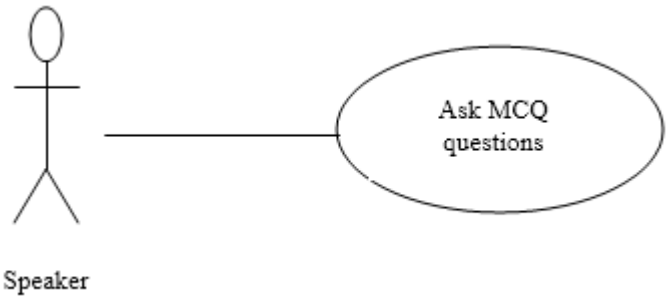
- **Initial Step-By-Step Description**

The speaker has to be logged in

1. The speaker inserts the poll and its answers.
2. The speaker uploads the poll.

USE CASE: ASK MCQ QUESTIONS

- **Diagram**



- **Brief Description**

The speaker uploads questions and answer choices.

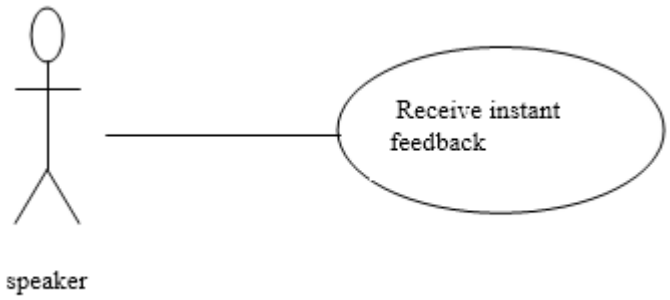
- **Initial Step-By-Step Description**

Before this use case can be initiated, the speaker is logged in and chose to submit an MCQ

1. The speaker selects the number of questions and the number of different choices for all the questions.
2. The speaker writes each question and its answers.
3. The speaker clicks submit.

USE CASE: RECEIVE FEEDBACK

- **Diagram**



- **Brief Description**

The speaker receives instant feedback that he specified its type in the admin settings.

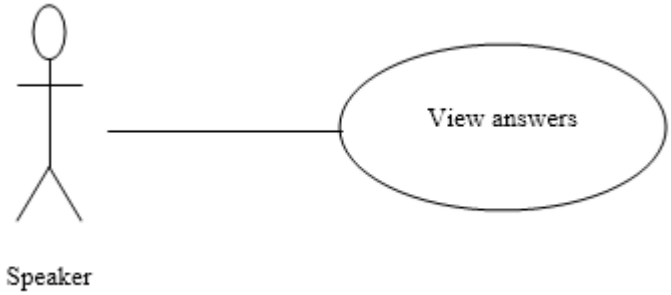
- **Initial Step-By-Step Description**

Before this use case can be initiated, the speaker has already specified the feedback type. And accessed the page of receive feedback.

1. The speaker views the percentage of up votes if this is the type of feedback he decided.
2. The speaker views the average score he gets from the audience if this is the type of feedback he decided.
3. The text messages submitted as feedback are also displayed on the same page.

USE CASE: VIEW ANSWERS

- **Diagram**



- **Brief Description**

The speaker views the percentages of the answers and the most and least chosen one.

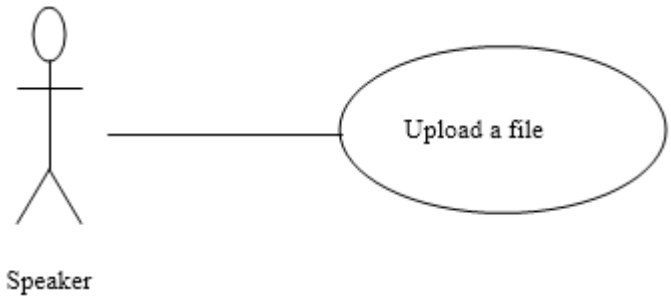
- **Initial Step-By-Step Description**

Before this use case can be initiated, the speaker has already accessed the page where he/she can submit view the answers..

1. The speaker goes to answers page.
2. The speaker views the number of “yes” and “no” answers of yes/no questions.
3. The speaker views the total number of audience who selected a certain answer for each question and the maximum and minimum selected choices.

USE CASE: UPLOAD A FILE

- **Diagram**



- **Brief Description**

The speaker uploads a file for the audience.

- **Initial Step-By-Step Description**

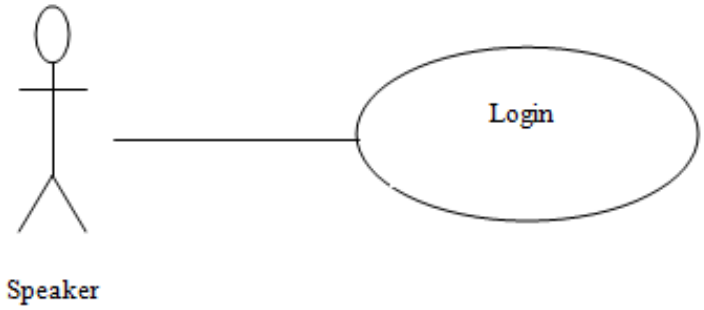
Before this use case can be initiated, the speaker has already accessed the page where he/she can upload a file.

1. The speaker clicks the upload button.
2. The file will start uploading.

The file will be available to be downloaded by the audience.

USE CASE: LOGIN

- **Diagram**



- **Brief Description**

The Speaker logins to the PollMe website to access the admin page.

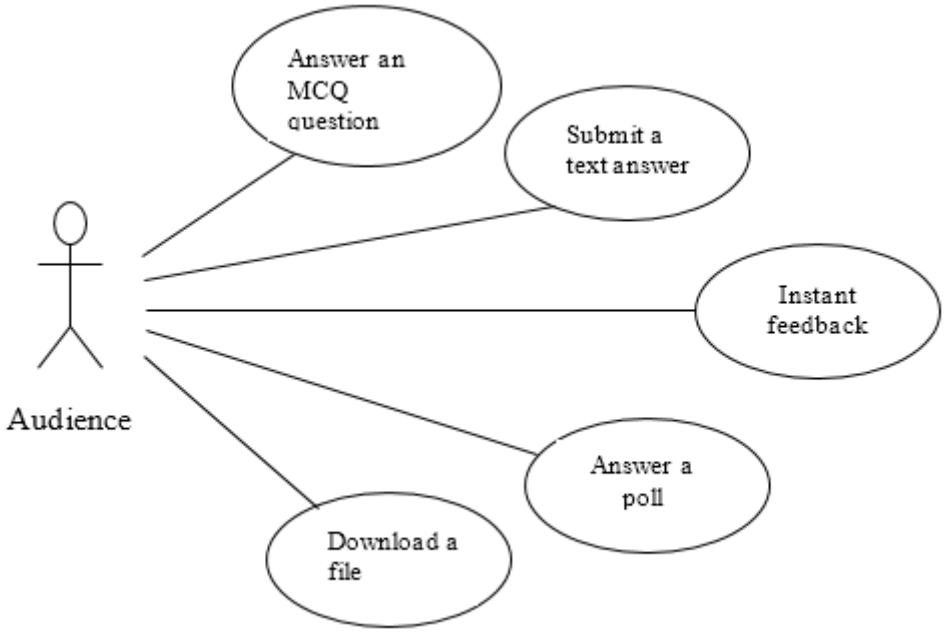
- **Initial Step-By-Step Description**

The Speaker has entered the login page.

1. The speaker gives his username and password.
2. The speaker is redirect to the admin page if the username and password were correct, or required to re-enter them if they were false.

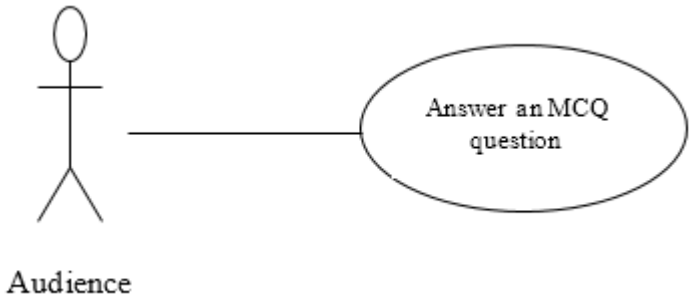
AUDIENCE USE CASE

The audience user has the following sets of use cases:



USE CASE: ANSWER AN MCQ QUESTION

3. Diagram



- **Brief Description**

The audience answers an MCQ question that the admin put.

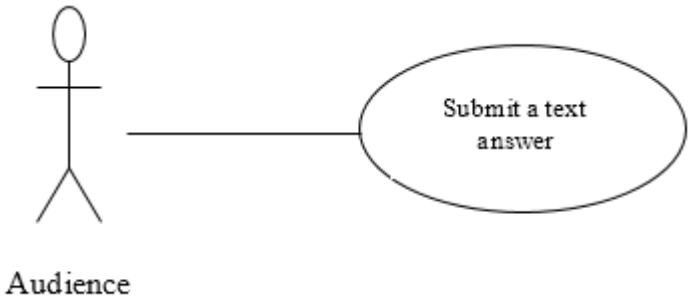
- **Initial Step-By-Step Description**

Before this use case can be initiated, the Admin has to start an MCQ question in the admin interface.

1. The audience user selects one of the choices presented to him/her.
2. The audience user clicks on submit.

USE CASE: SUBMIT A TEXT ANSWER

- **Diagram**



- **Brief Description**

The audience submits a text answer for the admin.

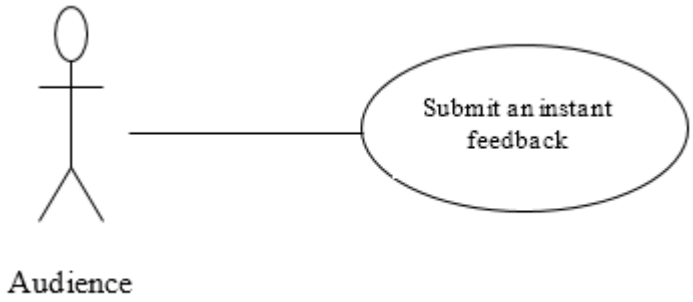
- **Initial Step-By-Step Description**

Before this use case can be initiated, the user who is anyone from the audience has already accessed the user page where he/she can enter a text answer.

1. The audience selects the text box where he/she can write the answer.
2. The audience types the answer.
3. The audience clicks submit.

USE CASE: INSTANT FEEDBACK

- **Diagram**



- **Brief Description**

The audience submits an instant feedback that the admin specified its type in the admin settings.

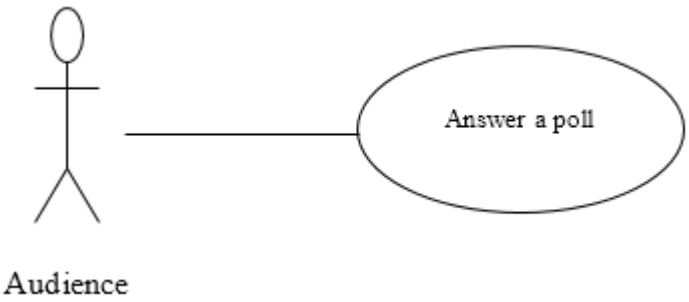
- **Initial Step-By-Step Description**

Before this use case can be initiated, the audience has already accessed the user page where he/she can submit an instant feedback, and the admin has already specified the feedback type.

1. The audience is presented by the type of feedback that the admin decided (up/down vote feedback, stars based feedback, text feedback).
2. The audience submits his/her feedback.

USE CASE: ANSWER A POLL

- **Diagram**



- **Brief Description**

The audience user submits an answer to a poll that the admin created.

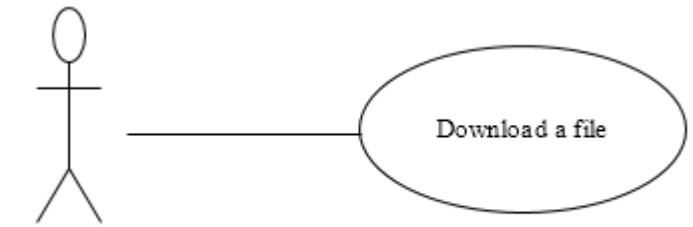
- **Initial Step-By-Step Description**

Before this use case can be initiated, the audience has already accessed the user page where he/she can submit an answer to a poll, and the admin had created a poll.

1. The audience is presented with the poll choices.
2. The audience chooses an answer.
3. The audience clicks submit

USE CASE: DOWNLOAD A FILE

- **Diagram**



- **Brief Description**

The audience downloads a file that the admin uploaded.

- **Initial Step-By-Step Description**

Before this use case can be initiated, the Audience has already accessed the user page where he/she can download a file, and the admin had uploaded a file.

1. The audience clicks the download button.
2. The file will start downloading

3. USER CHARACTERISTICS

THE SPEAKER DETAILED DESCRIPTION

The first type of user is the speaker is required to login on PollMe to make polls for his audience. He is also the admin. The speaker can be a teacher, lecturer or doing a presentation. He can submit the MCQs and see the total number of participants in his poll and the percentage of each answer choice on True or False questions. He can receive questions from his audience as texts during his lecture or conference, or receive answers to questions he just asked in the microphone without submitting it on the website.

THE AUDIENCE DETAILED DESCRIPTION

B) The second type of user is for those who take part of the poll with their opinions. They are the audience. The audience can be students in university or listeners in a conference. For simplicity of the system, no accounts are needed for them. They enter the home page, take polls introduced by the speaker, fill in their choice on MCQ questions provided by the speaker, ask questions for the speaker, and leave a feedback. They are not allowed to read other feedback messages from other users. Also, by giving a feedback on how much they understand the content of the topic. The audience also uses the program by writing the questions they want to ask or by asking for another explanation, breaks, or ask him to repeat something.

4. LIMITATIONS

This is a simple system for a university project and is very far from product quality. We only have 1 speaker who is also the admin. The system can't have different speakers as it is browsed locally from the admin PC. The router used also plays a big role on how many people in the audience can submit their answers at the same time.

For simplicity of the database, no accounts are required from the audience. In the home page they can submit their, questions, answers, feedback, and requests.

3. REQUIREMENTS SPECIFICATIONS

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

1. EXTERNAL INTERFACE REQUIREMENTS

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

USER INTERFACES

A first-time user of the application should see the log-in page when he/she opens the application. If the user has not registered, he/she should be able to do that on the log-in page. This user will be speaker (admin). He will log in and be redirected to his home page to submit polls, view answers of his questions and the feedback he received. He also has a settings page to control the number of choices and questions and the type of feedback he wants to receive.

If the user is not a speaker and just wants to give answers, he/she should be able to see the poll page directly when the application is opened.

HARDWARE INTERFACES

Since the web portal have any designated hardware, it does not have any direct hardware interfaces.

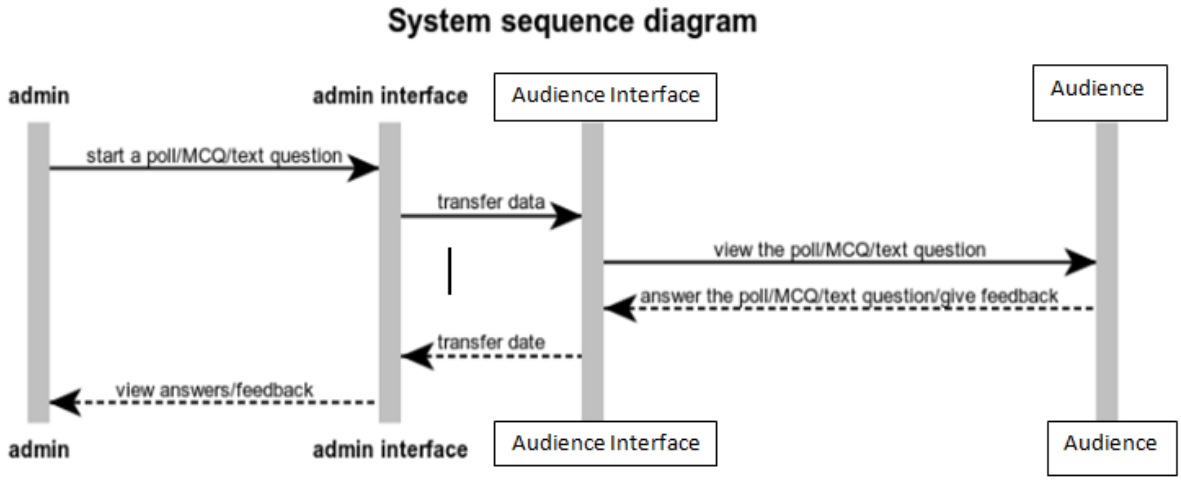
SOFTWARE INTERFACES

The communication between the database and the web portal consists of operation concerning both reading and modifying the data from the database. The main sources of data are:

1) Answers of the MCQ, polls and the yes/no questions. These answers are stored and they give the number of audience with their answer of each choice. Then, we can determine which choice is the maximum and which choice is the minimum. We display the results like statistics of people's choices for the speaker.

2) The feedback information which is about every topic introduced by the lecturer to the audience. This happens through an up-vote, which means that the audience

understands. And the down-vote, which means the audience doesn't understand. The up-vote and the down-vote appear as a percentage of votes of who understands and who doesn't. Another way of giving feedback is doing an evaluation from 1 to 10 on the topic's value and how much they understand it. This appears as an average that can be made into an equation of: the sum of all values divided by the number of feedbacks. Also the results appear for the speaker but not for the audience.



2. FUNCTIONAL REQUIREMENTS

AUDIENCE CASE

- **Answer MCQ Questions**

Use Case Name	Answer MCQ Questions
XRef	Section 2.2 Audience use case
Trigger	The audience has already accessed the user page where he/she can answer MCQ questions.
Pre condition	The speaker (admin) has to start an MCQ question in the admin interface
Path	1. The audience chooses the answer of each question. 2. They submit their answers
Post condition	The answers have been added to database
Exception Paths	Anyone of the audience may abandon the MCQ at any time.
Other	The speaker has deleted the MCQ questions or edit it

- **Submit a text answer**

Use Case Name	Submit a text answer
XRef	Section 2.2 Audience use case
Trigger	The audience as already accessed the audience page where he/she can enter a text answer.

Precondition	The admin has to ask question in the admin interface
Basic Path	1. The audience user selects the text box where he/she can write the answer. 2. The audience user types the answer. 3. The audience user clicks submit.
Post condition	The text is sent to the admin
Exception Paths	The attempt may be abandoned at any time.
Other	None

- **Instant feedback**

Use Case Name	Instant feedback
XRef	Section 2.2 Audience use case
Trigger	The audience user has already accessed the user page where he/she can submit an instant feedback, and the admin has already specified the feedback type.
Pre condition	The audience user is presented by the type of feedback that the admin decided (up/down vote feedback, stars feedback, text feedback).
Basic Path	The audience user submits his/her feedback.
Post condition	The speaker (admin) receiving the instant feedback.
Other	The admin could stop the feedback at any time.

- **Answer a poll**

Use Case Name	Answer a poll
XRef	Section 2.2 Audience use case
Trigger	The audience has already accessed the user page where he/she can submit an answer to a poll.
Pre condition	The speaker has already created a poll.
Basic Path	1. The audience is presented with the poll choices. 2. The audience chooses one. 3. The audience clicks submit.
Post condition	The speaker sees the results of the poll.
Exception Paths	The audience may abandon the operation at any time.

- **Download a file**

Use Case Name	Download a file.
XRef	Section 2.2 Audience use case
Trigger	The audience has already accessed the user page where he/she can download a file.
Pre condition	The Speaker has already uploaded a file.
Basic Path	The audience clicks the download button.
Post condition	The file will start downloading.
Exception Paths	The audience may abandon the operation at any time.

SPEAKER CASE

- **Ask MCQ Questions**

Use Case Name	Ask MCQ questions
XRef	Section 2.2 Speaker use case
Trigger	The speaker(admin) chooses MCQ page
Pre condition	The speaker has to start an MCQ question in the admin interface
Path	The speaker puts the questions which he wants. He submits his questions He determines the answers from the audience.
Post condition	The question have been added to database
Other	The speaker deletes the MCQ questions or edit it

- **Make a poll**

Use Case Name	Insert a poll
XRef	Section 2.2 Speaker use case
Trigger	The speaker chooses Poll page
Pre condition	The speaker has to start an Poll question in the admin

	interface
Path	The speaker put the choices which he wants. He submit his choices He determine the answers
Post condition	The choices have been added to database
Other	The speaker delete the POLL questions or edit it

- **Receive feedback**

Use Case Name	Receive feedback
XRef	Section 2.2 Speaker use case
Trigger	The admin choose FEEDBACK page
Pre condition	Feedback appears
Path	The admin choose FEEDBACK page He choose the feedback type: 1. Up / down vote 2. Scale from 1 to 10 3. Text feedback
Post condition	Percentage or average feedback appear

- **View answers**

Use Case Name	View answers
XRef	Section 2.2

Trigger	The speaker choose View answers page
Pre condition	The speaker has to start a POLL question or MCQ in the admin interface
Path	The speaker sees the answers
Post condition	The answers have been added to database
Other	The speaker can show all answers and know percentages of it.

- **Upload files**

Use Case Name	Upload files
XRef	Section 2.2 Speaker use case
Trigger	The speaker chooses Upload files page.
Precondition	The speaker has to choose upload files in the admin interface
Path	1. The speaker chooses a file. 2. The speaker uploads it. 3. The files appear on the page.
Post condition	The files have been added to database.
Other	The speaker can upload and delete files from the web.

- **Login**

Use Case Name	Login
XRef	Section 2.2 Speaker use case
Trigger	The speaker enters login page
Precondition	The speaker has to be the admin to give saved username and password
Path	1. The speaker enters the login page. 2. The speaker submits his username and password. 3. If the username and password were correct, the admin page appears, if not, he is required to re-enter his data.
Post condition	A new session is created.

3. NON-FUNCTIONAL REQUIREMENTS

SECURITY

We care about the privacy of the information that speaker has on his page, so we locked his data that he only can get accessed to it. Only the admin has a password to enter while the audience can submit their answers without entering usernames and passwords.

USABILITY

Our interface is easy to learn how to use and easy to remember how to use. Users should not be required to consult a manual each time the use PollMe. We want it used by anyone, be simple to use the first time around without instructions.

Efficiency of use: goals are easy to accomplish quickly and with few or no user errors.

Intuitiveness: the interface is easy to learn and navigate; buttons, headings, and help/error messages are simple to understand.

Low perceived workload: the interface appears easy to use, rather than intimidating, demanding and frustrating.

TECHNOLOGY

We will use the following languages:

HTML, CSS, JavaScript and SQL.

For hosting, we use Heroku associated with Github. For testing, we run the website on Xampp apache server.

DELIVERY:

Prototype: 16/12/2018.