

EVOTING AS A TOOL FOR PRACTICAL MEETINGS

Pance Ribarski
Faculty of Computer Science
and Technology
Skopje, Macedonia

Jovan Bicevski
DMBUC Ilija Nikolovski Luj
Macedonia

ABSTRACT

This paper gives overview of requirements for information system for meetings management including evoting capabilities. This information system is supposed to release the heavy task of manually organizing meeting with many participants, manage the meeting agenda or perform various voting processes. The proposed system has added methodology for organizing Auditions with given list of participants and given list of possible voting points. This system is work in progress, showing that complex meetings can be easily arranged and simplified by a simple web application.

I. INTRODUCTION

Electronic voting is taking its place in the voting processes where traditional voting has been used by default. The low cost, easy maintenance, fast setup and fast response are the pros for electronic voting. The special hardware and software prerequisites are the cons that need to be inspected very closely.

There are many aspects in the implementation of electronic voting for meeting purposes. This paper identifies the needs for information system geared with electronic voting capabilities and the requirements of this information system. These requirements were gathered by doing a real evoting meeting software for "State Music Highschool Ilija Nikolovski Luj".

In Section II we have given general specifications about the information system. Here are mentioned the main purposes of such a system. Section III shortly explains the auditions that can be arranged with this web application. Section IV shows the types of users that can interact with the system. Section V presents the current implementation model of the meetings information system. Here are given the User and Administrator part and each is shortly explained. In Section VI we explain that this system is work in progress and that there is much more work to be done into security and cryptographic algorithms that will support the evoting processes.

II. GENERAL SPECIFICATIONS

The requirements given by the board of teachers for their meeting purposes were scraped from real meeting scenarios. Their routine was to gather on a meeting comprising hundred or more teachers. The attendees sign up on a meeting sheet upon arrival. Then the chairperson goes through the meeting agenda and calls for comments if any and finishes every topic with eventual voting. All the attendees can sign up for an agenda topic prior the topic is opened by the chairperson. The

voting process on the agenda topics can be various by nature. This requires usage of different types of choices:

- One choice of many
- K choices of many
- Open ballot choice

The meeting can be set with start date and time, and end date and time. It also can be manually started or ended by the system administrator. The voting processes also can be scheduled by start and end date and time or be manually started and ended.

III. AUDITIONS

The board of teachers needed another form of voting for audition purposes. The school has auditions for concerts and grades students from a given list with available grades. They specifically asked for five votes of five points, five votes of four points, etc. Then the teachers are given a list of students and a list of available votes. They match a vote with student until all votes are used or all students are exhausted. At the end of audition the concert list is created from descended sorting students by vote points. The count of students in the concert list is calculated by the sum of students' performance time length and the given concert time length.

IV. TYPES OF USERS

The system requires three kinds of roles for the users:

- Regular user
- Meeting administrator
- Super administrator

The super administrator can add users or edit their information. The super administrator can reset password for a user upon request. The meeting administrator can set meetings and edit their preferences. This role can manage voting processes for the created meetings and auditions as well. The regular users can sign up for meetings, sign up for agenda topics and vote on available voting processes and auditions.

V. IMPLEMENTATION

Our implementation for easy meeting organizations is a web application. It is done with Microsoft ASP.NET MVC 3.0 which is implementation to the MVC pattern to the classic ASP.NET. As data layer we are using Microsoft Entity Framework 2.0 which is Object Relational Mapping paradigm to databases. This choice of technologies made it easy to

develop a complex information system such as the one with our prerequisites.

The web application is divided in user and administrator part. The user part is presented with menu choices of Meetings, Voting processes and Auditions. In each of the mentioned sections the user is presented with a list of available items and actions with the listed items. The list of possible auditions that are visible for the user is given in Fig. 1.

р.б.	Наслов	Почеток	Место
1	Аудиција за ученички концерт	01-01-2012	школје
2	Новогодишен концерт на класата за виолин инструменти	01-01-2012	Хорска сала

Figure 1: Presented list of possible auditions and their actions.

In the presented list for meetings the user can join a meeting by clicking the action Join. After joining the audition the user is transferred to the voting processes available for the joined meetings. If a voting process is available the user can give their vote for the appointed agenda topic.

In the presented list for auditions the user sees the audition that he/she is entitled for. Not every user can vote for any audition. If the user chooses the Vote action for audition he/she is presented with the choice of students and vote points as in Fig. 2.

Избери	Поврати
Александар Јордановски - 4	Ангела Трајческа - 4
Сања Стојановска - 3	Теодора Трајковска - 5
	Дејан Николовски - 5
	Геле Гелев - 2

Figure 2: Making and audition vote

References to figures and tables should be done as Fig. 1 and Table 1, respectively.

The administrator part of the web application is the place for managing the information system. The Super administrator role can access the Users section where the possible actions are adding users, changing their information or resetting their password. The Meeting administrator is meant to create meetings, manage the meeting agenda and voting processes or create auditions.

The Meeting create or edit place contains the usual data about meeting like start date and time and end date and time, place, whether it is active and an IP subnet from which the users are allowed to join a meeting. This is necessary because the system is on the Internet and users can join meetings without being physically on the meeting place. From the list of meetings the Meeting administrator can manage the meeting agenda and the meeting voting processes.

Administrators can choose between K-of-N voting process and open-ballot voting process. In the K-of-N we can choose K to be 1 – thus getting the special case of 1-of-N voting process. In any voting process the Administrator can choose whether the voting is anonymous, meaning that the system will not record the link between the cast vote and voter.

The Audition create or edit place contains the usual data like place, start and end time, time length, list of auditioning students and list of eligible voters.

VI. REMAINING WORK

The implemented work is only reflection to the general specifications. The interface presented in this paper is an effort to comply with the requirements engineering in the information system planning process. Currently no cryptographic algorithms are implemented or even discussed to be used in the evoting processes. Implementation of state of the art evoting algorithms stays as remaining work for this information system. Despite the lack of any secure elements this system can be used for non-critical meetings and voting processes where the risk of possible attacks is minimum. This is supported by the fact that almost all voting processes are non-anonymous. This means that the users can easily verify if the system or someone else corrupted the voting processes.