Recording Class Attendance in a Student Registration System

Sanhu Li, Mao Zheng
Department of Computer Science
University of Wisconsin-La Crosse
La Crosse WI, 54601
mzheng@uwlax.edu

Hao Fan
School of Information Management
Wuhan University
Wuhan, Hubei, 430079
hfan@whu.edu.cn

Abstract

Recording classroom attendance by hand for large classes can be a time consuming task. We have developed a mobile application and registration system that allows collection of attendance information. The information is collected with Android devices and transferred and stored in a web-based student registration system. Both the Android application and the registration system are part of this project. The registration system serves as a test-bed for the recorded data and can be used to analyze the data.

There are two main challenges when recording class attendance. First, it is important to be certain the student physically comes to the classroom. Second, the application needs to verify the recorded data is for the correct student. The current solution provided in this project is to let the instructor generate a random QR code during class time, in the classroom. The student will use his/her mobile device to scan the QR code and submit the scanned data along with his/her student ID to the registration system during the recording time period. Only students who are enrolled in the class are able to submit the scanned data and this data can only be recorded once during a valid time period. The instructor can view the recorded class attendance information and each student can verify that his/her attendance was recorded correctly.

There are two major platforms in the mobile device community: iOS and Android. This project chose Android development mainly for the reason of its openness. In addition, all the tools in the Android development are free and no special hardware is required.

The web-based student registration system uses Bootstrap framework as the theme. The front-end are static html files using AngularJs together with jQuery and Ajax to provide the dynamic pages. In the back end, the system uses JSP, Spring-mvc framework and MySQL database. Connections are managed by the Hibernate.
1 Introduction

Recording classroom attendance by hand for large classes can be a time consuming task. We have developed a mobile application and registration system that allows collection of attendance information. The information is collected with Android devices and transferred and stored in a web-based student registration system. Both the Android application and the registration system are part of this project. The registration system serves as a test-bed for the recorded data and can be used to analyze the data.

There are two main challenges when recording class attendance. First, it is important to be certain the student physically comes to the classroom. Second, the application needs to verify the recorded data is for the correct student. The current solution provided in this project is to let the instructor generate a random QR code during class time in the classroom. The student will use his/her mobile device to scan the QR code and submit the scanned data along with his/her student ID to the registration system during the recording time period. Only students who are enrolled in the class are able to submit the scanned data and this data can only be recorded once during the valid time period. The instructor can view the recorded class attendance information and each student can verify that his/her attendance was recorded correctly.

2 The Web-based Student Registration System

A big component of this project is the web-based student registration system. It serves as the test bed: 1) To verify student’s class registration records: only students which are registered for the current course are able to submit their class attendance. 2) To ensure the recording time for the class attendance is valid. For example: the course CS741 in Spring 2015 is offered Monday Wednesday and Friday from 11:00 am to noon. If the class attendance recording time is on Tuesday at 11:00 am, the system should report an error message.

Most of the validations discussed above are addressed in the database design. Below is the E-R diagram for the student registration system. The system uses MySQL for the database. The student_record table is used to keep all the class attendance records for a student. The student ID needs to be existed in the student_course table that is used to store the student’s course registrations. Similarly, every class attendance record has a date and time that can be used to check if it is within the valid course time.

The student registration system has three types of users: Student, Instructor and Registrar. The use case diagram is shown in Figure 2.
Figure 1 E-R Diagram for the Student Registration System

Figure 2 Use Case Diagram for the Student Registration System
3 Generating QR Code

The system uses the Z-xing API [3] to generate the QR code based on the record ID. However the record ID will be encrypted using AES first. The key which is used for encryption and decryption by AES is a random generated string.

The Source code for generating the QR code is shown below.

```html
<img src="https://api.qrserver.com/v1/create-qr-code/?&data=<%=org.lsh.helper.cipher.AESUtils.encrypt(record.getRid() + "", Constants.KEY)%>" alt="qr code" />
```

Figure 3 Source Code Segment for Generating the QR Code.

The client side of this project is a mobile application developed in the Android environment. There are two major platforms in the mobile device community: iOS and Android. This project chose Android development mainly for the reason of its openness. In addition, all the tools in the Android development are free and no special hardware is required.

The student can use his/her mobile device to scan the QR code and submit the scanned the image along with his/her ID and the stored key in the app, to the server. The server will check if the student ID, record ID, recording time and class time are valid and consistent.

Figure 4 is the screen shot of the student scanning the QR code using his/her mobile device. Figure 5 is the test data of the recorded class attendance along with the date and time.

<table>
<thead>
<tr>
<th>Record Id</th>
<th>Record Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>12/03/2014</td>
</tr>
<tr>
<td>6</td>
<td>12/03/2014</td>
</tr>
<tr>
<td>7</td>
<td>12/03/2014</td>
</tr>
<tr>
<td>10</td>
<td>02/02/2015</td>
</tr>
<tr>
<td>11</td>
<td>02/02/2015</td>
</tr>
<tr>
<td>12</td>
<td>02/02/2015</td>
</tr>
<tr>
<td>13</td>
<td>02/15/2015</td>
</tr>
<tr>
<td>16</td>
<td>03/18/2015 03:29:31</td>
</tr>
</tbody>
</table>

Figure 4 Scanning the QR code

Figure 5 Recorded Class Attendance
4 Spring MVC Framework

This web-based student registration system uses a Spring-mvc framework and the MySQL database on the back end. Connections are managed by the Hibernate.

The Interceptor class in Spring-mvc framework helps the validation. As we described in the previous section, there are three types of users in the system. Each has his/her own access. Instead of having to validate the user’s role on every page, we can define the access based on the user’s role in the Interceptor class. The code segment is shown below:

![Code Segment](image)

Figure 6 The Interceptor Class

We are using AngularJs in the static html files, together with jQuery and Ajax to provide the dynamic pages on the front-end. It supports a clear and better Model-View-Controller(MVC) structure. If the data is changed, the view will be automatically changed. The controller only needs to manipulate the data triggered by the user event. Below is the code segment example for writing the sidebar using angular.js. It will be directed into individual static html files that use Ajax to communicate with the server to get the data.
The future work of the current project is aiming to research some indoor wifi-position methods. This is a better way to ensure the student physically comes to the classroom. We need to add the functions of submitting current wifi environment from the client to the server and checking if it is from one of the classroom’s wifi hotspots.

References:


