EzSeminar – A mobile application for convenient seminar management

Tian Lan
Texas A&M University
welkinlan@gmail.com

Yang Zhou
Texas A&M University
zhouyangjulia@gmail.com

Liming Ye
Texas A&M University
liming.ray.ye@gmail.com

Xiaohui Shen
Texas A&M University
sxhbeckham@gmail.com

Junseok Lee
Texas A&M University
junseogi44@gmail.com

ABSTRACT

In this report, we introduce a convenient light-weight framework for users to manage seminars easily. This framework is adapted from the classical server-client framework, in which we introduce an Android mobile app as the light-weight client and develop the Apache web server as the database repository and data processing center. We conducted a pilot user study on the students in TAMU CSE department, and the results showed that our system was easy and convenient to use.

Author Keywords

System design, Computer and new media, Android, Client-Server infrastructure

1. INTRODUCTION

Students or professors get too many emails about the seminars, and are easy to forget the information for the coming one. For example, In Texas A&M University, each graduate student must attend the course of Graduate Seminar before graduation, which means the student needs to attend at least 12 seminars. In order to receive the credit for this course, students also need to turn in at least 6 paper reviews before the corresponding seminar begins, and turn in seminar reports at the end of each seminar. To handle all these related things, one should be managed to make a good schedule for the seminars. However, except for the fixed schedule of seminars in the semester, there are also new scheduled seminars, which will be notified by the professors via email. Students are encouraged to attend as many seminars as possible.

But with the all the related or unrelated emails flooding into your mailbox every day, it is so hard for a student or a professor to filter related emails, choose the seminars he is interested in, make schedule for each seminar, and prepare for the coming seminars, etc. So a mobile application to handle all these seminar related schedules is highly needed. Since almost everyone uses and checks mobile phones every day, such an app is really convenient for either a student or a professor to handle seminar schedules if he is interested in attending seminars or has chosen the course of Graduate Seminar during a busy semester.

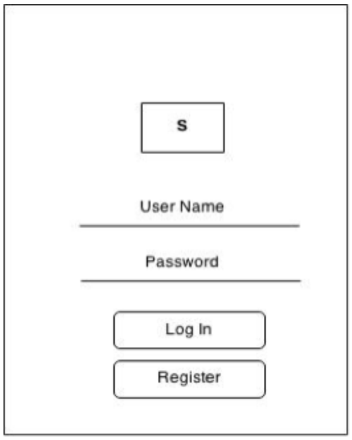
2. SYSTEM DESIGN

Our system is designed to employ the following client-server infrastructure to incorporate social networking capability.

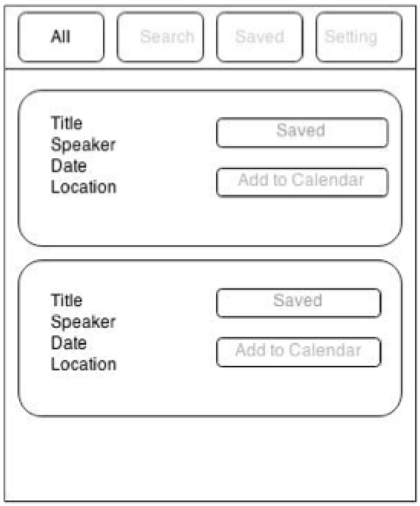
The client acts as a functional unit for our users to browse up-to-date seminars which are pulled down from the server. The user is able to save a seminar, search seminars and add a certain seminar to the Google calendar. Moreover, the available seminars are sorted by recommendation order, which are sorted by the popularity of the saved events. The client side is built in Android mobile system.

The server is the computation center which handles all the complex and time consuming tasks, such as all-time email extraction and natural language processing of the emails. The server also acts as a data center, which saves all the user and seminar information. With all the information, the server would be running a simple recommendation algorithm which recommends seminars to users based on the popularity of the seminar.

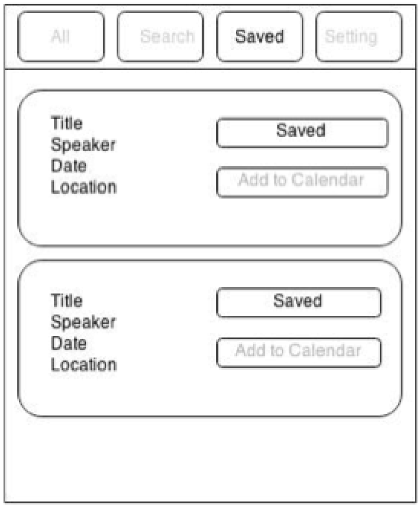
With our design, the client never suffers from limitation of computation and storage capability of an Android mobile device. The server take over all tough work and send the “processed” information to the client.



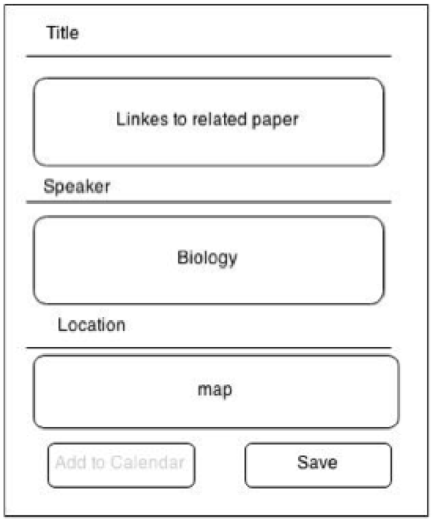
(a)



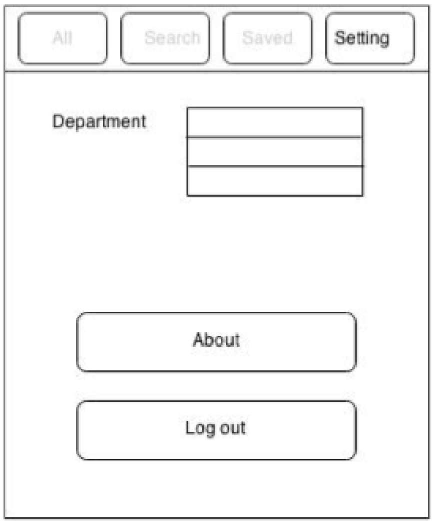
(b)



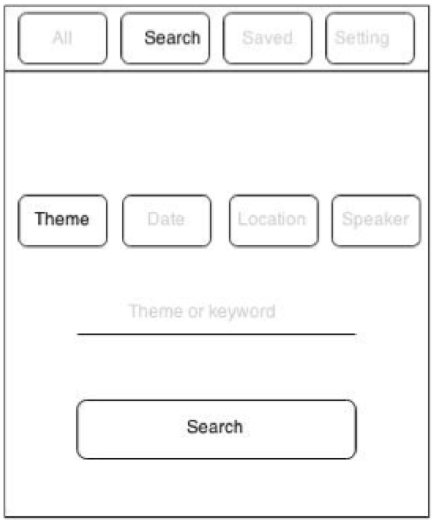
(e)



(c)



(f)



(d)

Figure 1 User interface design (a). login and register page. (b). available seminars page. (c). seminar detail page. (d). search seminar page. (e). saved seminars page. (f). settings page

2.1 User Interface

Login and register page

User needs to register for our EzSeminar application, and uses the account to log in. Every account is bounded with the user’s settings, saved seminar, thus enabling the server to provide recommendation seminars. See Figure 1(a).

Available seminars page

All seminars page displays all the upcoming seminars with brief descriptions, including title, speaker, date, and location. Moreover, user’s actions towards the seminar, which includes whether the user has saved the seminar and whether the seminar has been added to the calendar, will also be displayed. The user can also scrolled down the screen to see more seminars. Please see Figure 1(b).

Seminar detail page

The user can see several brief descriptions for the seminars. If the user double clicks one seminar, more details will be provided in the details page. As shown in Figure 1(c), under the title, we will provide links to related paper. Under speaker, the biology of the speaker will be displayed. A map will also be shown for the location of the seminar.

The users can add the seminar schedule automatically into the Google Calendar built-in for every Android device. As shown in Figure 2. Google Calendar API supports the function and the users can verify the seminar information using Google Calendar.

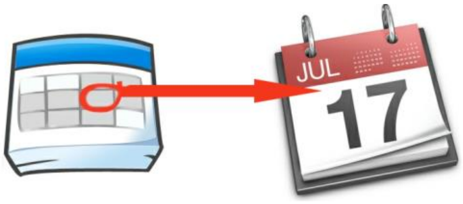


Figure 2 Google calendar integration

Search seminar page

In the search page, the user can search for seminar according to the theme, date, location, and the speaker. The user only needs to type in keywords and click the search button, the seminars in the relevance order will be displayed. See Figure 1(d).

Saved seminars page

As shown in Figure 1(e), in the saved page, the seminars that the user saved will be stored here. The user can save the seminars he is interested in this page in order to schedule for the upcoming seminars to avoid missing an interesting seminar in a high-paced life.

Settings page

There is also a page for the setting, so the user can choose the seminars of which department he wants to know about. Now the application will only include seminars for the department of computer science, but in the future more departments will be extended. See Figure 1(f).

2.2 Server Infrastructure:

We establish a server to act as computation center to deal with the complex work. The benefit of employing a server as a hub for collecting users' information is that this server could act as a social network engine, which will recommend popular seminars to the users. In the future we could extend the server to enable more social events for the users.

Core functions:

- All-time email extraction and analysis: The server will run a infinite loop which keeps updating emails, running NLP for the seminar contents and scrawling the related articles. All the extracted information will be saved into the database.
- User and seminar management: The server stores all the information of user (Login and Register actions) and seminar in the database and the admin is able to manage the user and seminar information tables.
- Data storage and communication: The server will provide XAMPP framework for communication. XAMPP is a free and open source cross-platform web server solution stack package, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming languages [1].
- Social service: To promote collaboration and networking, we plan to provide seminar recommendation, which will recommend the seminar with the most number of "save" from all the users. Other social features are still in discussion.

System Environment

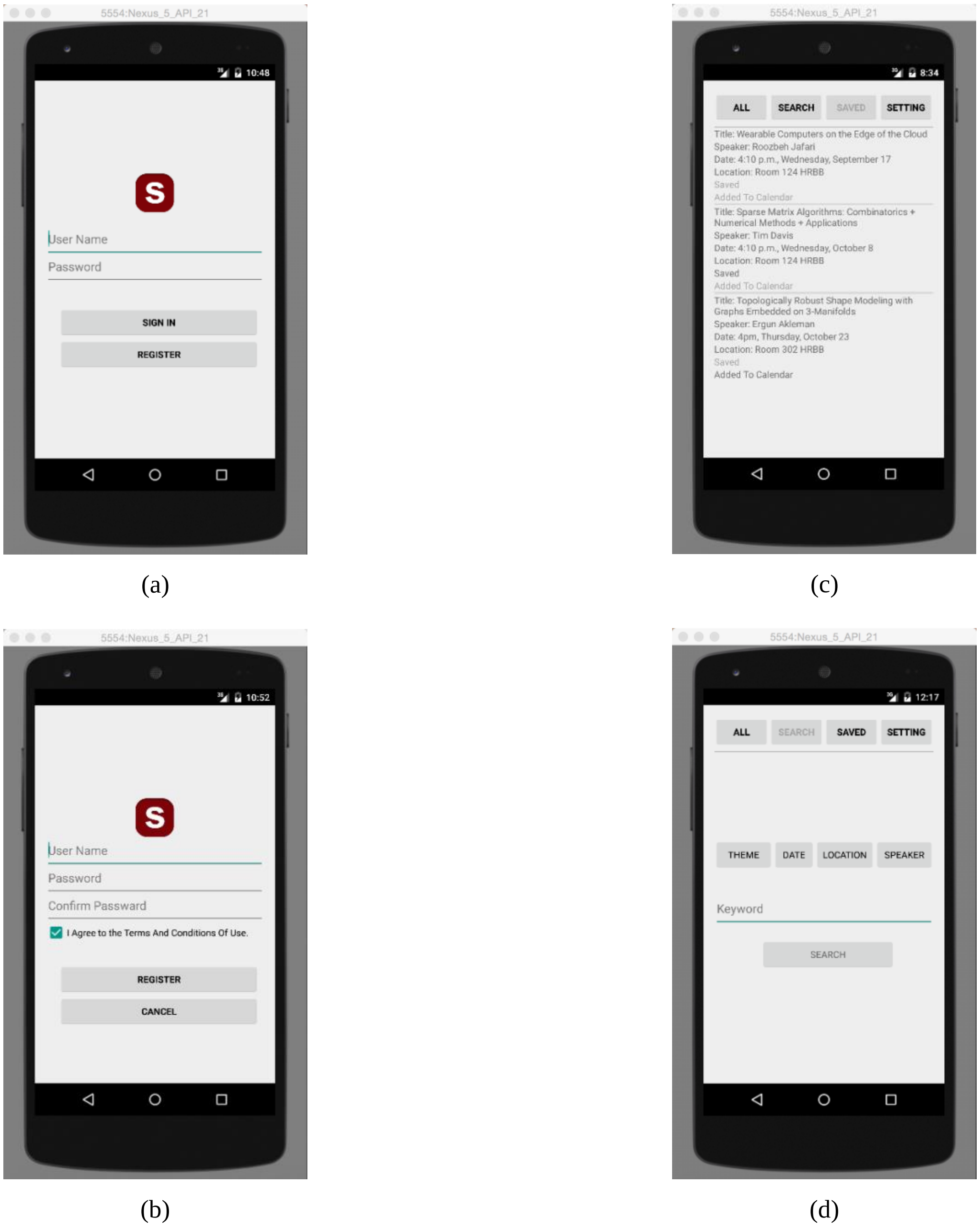
- Environment: Ubuntu 14.04 Apache server.
- Server framework: XAMPP (Apache + MySQL + PHP + Perl/Python)
- Database: MySQL Workbench
- Core languages: PHP for service, PYTHON for NLP analysis, SQL for DB operations

2.3 Client-server communication:

For the communication between the server and the Android client, we plan to establish reliable and readable connection through HTTP service. With our server taking over, the native client does not need to run all the complex work locally, which will make our native app energy-economic. The only information that the Android app needs from the server is the metadata (JSON/XML), which will be converted and displayed via Android wonderful UI framework.

3. SYSTEM IMPLEMENTATION

We uses Android Studio IDE to design and implement our client app. XAMPP framework is adopted to develop the server with the usage of Apache, MySQL and PHP. Python is used for related article crawling and NLP.



(a)

Title: Wearable Computers on the Edge of the Cloud
Speaker: Roozbeh Jafari
Date: 4:10 p.m., Wednesday, September 17
Location: Room 124 HRBB
Saved
Added To Calendar

Title: Sparse Matrix Algorithms: Combinatorics + Numerical Methods + Applications
Speaker: Tim Davis
Date: 4:10 p.m., Wednesday, October 8
Location: Room 124 HRBB
Saved
Added To Calendar

Title: Topologically Robust Shape Modeling with Graphs Embedded on 3-Manifolds
Speaker: Ergun Akleman
Date: 4pm, Thursday, October 23
Location: Room 302 HRBB
Saved
Added To Calendar

(c)

(b)

(d)

Figure 3 User face GUI implementation (a). login page. (b). register page. (c). available seminars page. (d). seminar detail page. (e). saved seminars page. (f). search seminars page. (g)(h). settings page

3.1 User Interface:

For the user interface, we use Android Studio to implement the UI. We use the color of maroon as the theme color since maroon stands for Texas A&M University.

Following the designing scratches, we have implemented the user interface for our EzSeminar application. We have provided the realization pictures for the application. We also designed a logo for it, using the color of maroon, and the capital letter S stands for Seminars, which is simple and clear.

Login and register page

Figure 3(a)(b) shows the login page and the register page. Every user needs to register for an account when using the application. And used the registered account to login. The server has stored related information for each user, including which seminars the user has saved and added to calendar.

Available seminars page
Figure 3(c) shows the all seminars page and the saved seminars page. We use a list structure to display all the seminars. Every item includes a brief description for an upcoming seminar. We will show the title, speaker, date, location, and the status of whether it has been saved or added to calendar.

Seminar detail page and Saved seminars page
Figure 3(d) shows the detail page. From the all seminars page or saved seminars page, if the user click one seminar, it will display the details of the chosen seminar. This part will provide the user with links to related paper, and the biology of the speaker. For the location, we use Google Map to display the location. And for the Date, we provide a button to enable the user to add the event to Google Calendar. A button of save will also be provided.

Search seminar page
Figure 3(f) shows the search page. The user can input keywords as well as choosing which attribute to search about. There are four options: Theme, Date, Location, and Speaker.

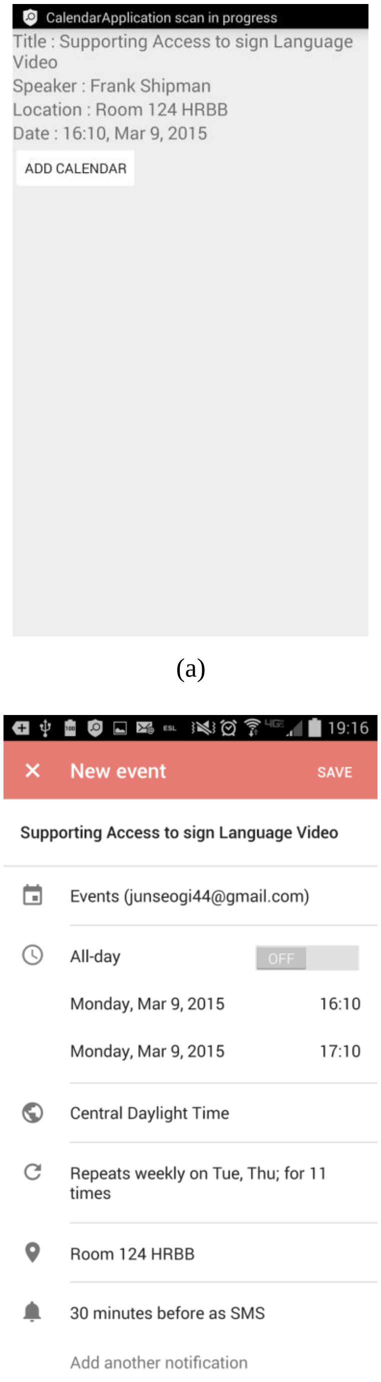
Settings page
Figure 3(g)(h) shows the setting page. In this page, the user can log out to switch to a different account. The user can choose different departments. Depending on the departments, our system will display related seminar events.

Integration with Google calendar
We also implemented a prototype of add calendar application. As shown in If you click the "ADD CALENDAR" button, you can show the new event page with the information of the seminar filled in accordingly. See Figure 4.

3.2 Server Development

Server Construction
We used XAMPP for the server constitution. Apache2, MySQL and PHP5 are installed on the server.

- The server URL is seminar.welkinlan.com



CalendarApplication scan in progress

Title : Supporting Access to sign Language Video
Speaker : Frank Shipman
Location : Room 124 HRBB
Date : 16:10, Mar 9, 2015

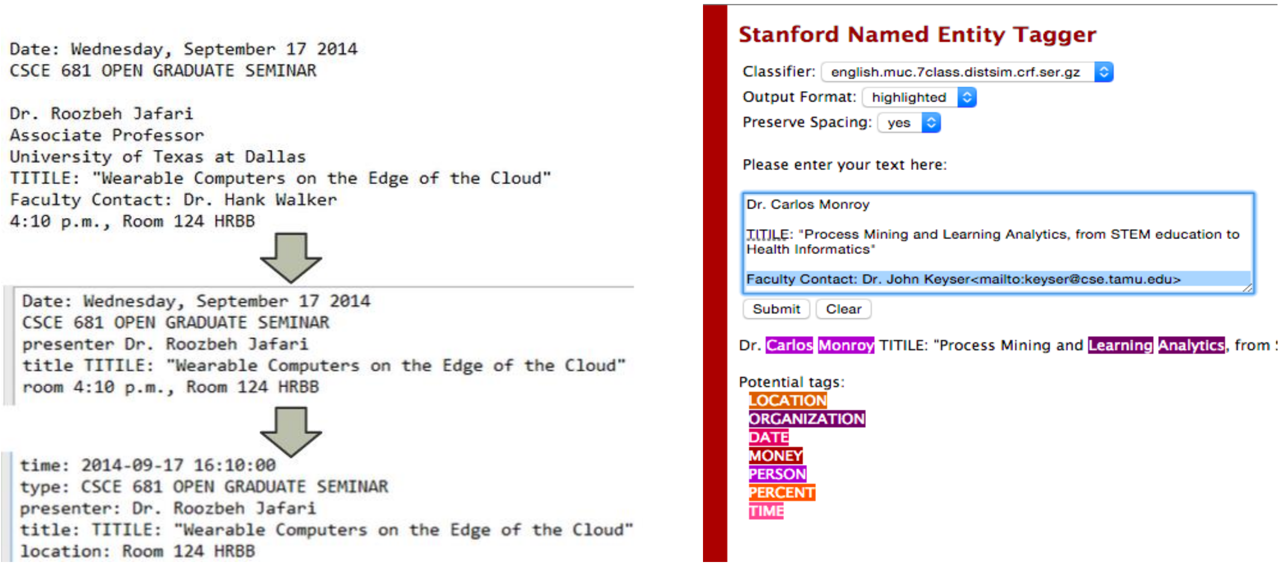
(a)

Supporting Access to sign Language Video
Events (junseogi44@gmail.com)
All-day OFF
Monday, Mar 9, 2015 16:10
Monday, Mar 9, 2015 17:10
Central Daylight Time
Repeats weekly on Tue, Thu; for 11 times
Room 124 HRBB
30 minutes before as SMS
Add another notification

(b)

Figure 4 Integration with Google calendar

- The database management is on http://welkinlan.com/phpmyadmin. For limited privilege, you could use username "*test*" and password "*test*" for inspection.



```
Date: Wednesday, September 17 2014
CSCE 681 OPEN GRADUATE SEMINAR

Dr. Roozbeh Jafari
Associate Professor
University of Texas at Dallas
TITILE: "Wearable Computers on the Edge of the Cloud"
Faculty Contact: Dr. Hank Walker
4:10 p.m., Room 124 HRBB
```

```
Date: Wednesday, September 17 2014
CSCE 681 OPEN GRADUATE SEMINAR
presenter Dr. Roozbeh Jafari
title TITILE: "Wearable Computers on the Edge of the Cloud"
room 4:10 p.m., Room 124 HRBB
```

```
time: 2014-09-17 16:10:00
type: CSCE 681 OPEN GRADUATE SEMINAR
presenter: Dr. Roozbeh Jafari
title: TITILE: "Wearable Computers on the Edge of the Cloud"
location: Room 124 HRBB
```

Stanford Named Entity Tagger

Classifier: english.muc.7class.distsim.crf.ser.gz
Output Format: highlighted
Preserve Spacing: yes

Please enter your text here:

Dr. Carlos Monroy

TITILE: "Process Mining and Learning Analytics, from STEM education to Health Informatics"

Faculty Contact: Dr. John Keyser<mailto:keyser@cse.tamu.edu>

Dr. Carlos Monroy TITILE: "Process Mining and Learning Analytics, from

Potential tags:
LOCATION
ORGANIZATION
DATE
MONEY
PERSON
PERCENT
TIME

Figure 4 Example of word analysis and Stanford NLP analysis

Server Controller

The server provides several controllers to deal with multiple requests from the client side. These controllers are:

- Login and Register controller: These controllers are used to validate user login info or create new user info. They deal with MySQL database
- Push controller: This controller is used to push updated seminar information when the client requesting for available seminars. This controller also takes consideration of the settings on the client.
- Save controller: This controller is used to handle the save seminar event from the client to update the database accordingly. With this controller, the server is able to run a recommendation algorithm to recommend seminars with highest popularity.

For the best performance of the integration of PHP and MySQL database and the speed of database operations, we used the MySQLi Extension (MySQL Improved), which is a relational database driver used in the PHP programming language to provide an interface with MySQL databases [2].

Word Analysis

The server runs a time thread to extract seminar information from the emails sent from cs-grad@listserv.tamu.edu using the subject "CSE Events of the Week". These emails will provide a list of events that department of computer science and engineering will hold in the following week. Seminars are also included in these events. We used IMAP4 protocol client imaplib implemented in Python to extract emails and do analysis on the emails in gmail. Figure 4 is an example of the code snippet (upper section) and the results (lower section).

The imaplib to extract raw email content from the interface. And we extract the real content from the raw email content using some start symbol and end symbol as the delimiter. Since we get the real content of the email, we can do word analysis since then. The email is in a format that events in the week are separated by dates and events on the same day are listed under the date. And we use this format to extract the events for each date and store the information in the map data structure in the memory. The first line of each event is the name of the event. If the first line does not contain the key word "seminar", we will simply ignore it because it is a non-related event.

We have formatted information so far, but we need to generate a better formatted data beyond this. To achieve that goal we use natural language to help us.

Natural Language Processing

We select effective information from the emails and label them using natural language processing tool called Stanford Named Entity Recognizer. Here is an online demo to present how we gonna use it. We input the information about one email and the recognizer will label the information and return the information back to us so that we can insert them into the database in a structural format. Then we insert the seminar information into the database.

From the word analysis segment, we get a simply formatted data. Then, we send each line of the event through our named entity recognition system. If a line contains a name, then it must be the professor's name. If a line contains a key word "title", then it is the title of the presentation. If a line contains a time, then it is a time for the seminar. Using this technique, we generate data field for each seminar and we store them in a text file as a copy. And at the same time, we store them in our relational database to be the resource of our application.

3.3 Communication with server

For the communication between the server and the Android client, we use HttpClient to establish reliable and readable connection through HTTP service at client side. It uses Get and Post methods to request information from server, which includes login and register requests and seminar update request. All the requests are based on URL with different parameters. We implemented the communication via HttpClient. The client can send requests for login and register. Based on different response received by client, it will be converted and displayed via Android wonderful UI framework. When the android client requests information about seminars from server, server will reply all the information in JSON format. The client will parse the JSON object and bind these information to UI elements. There are two more communications between server and client, the client may update its local status of seminars(seminar can be saved or added to google calendar), these changes need to be updated in server for synchronization among multi-clients and UI refreshing.

4. SYSTEM EVALUATION

We conducted a pilot study aimed at evaluating the usability of our system. The pilot study included 6 graduate students in TAMU CSE department (N=6; 4 male; 2 female; ages 24-27 years). After a simple introduction of the Android app, the participants were allowed to register and login into our app. Then each participant was given 5 minutes of free time to play with the app. At the end of the study, we provided a short survey for each participants to describe their experience. When asked about *Do you like the app interface?*, all of the participants said they liked the interface, most of them said the interface was neat and easy to use. When asked *Do you think this app is useful?*, all of the participants answered "yes" and they all thought it convenient for seminar management. When asked *What do you like most about our app?*, 4 of 6 participants said they liked the "seminar recommendation" which was very social-like; 3 participants liked the "add to calender" function as a reminder.

When asked about *Where do you think we could improve our app?*, a participant had the idea to build social networking in our app where users could share their ideas or questions on the seminar; another participant said that we could recommend seminars based on user's preferences, for e.g., user's research interests.

5. WORK ALLOCATION

Our work is distributed well into two major groups: server group and client group.

In client group, Yang worked on UI design and implementation and Xiaohui worked on communication with the server.

In server group, Tian worked on server construction and development, Liming worked on word analysis and Junseok worked on related articles. All team members are self-motivated and eager to learn. We collaborated well with each other and everyone was very helpful.

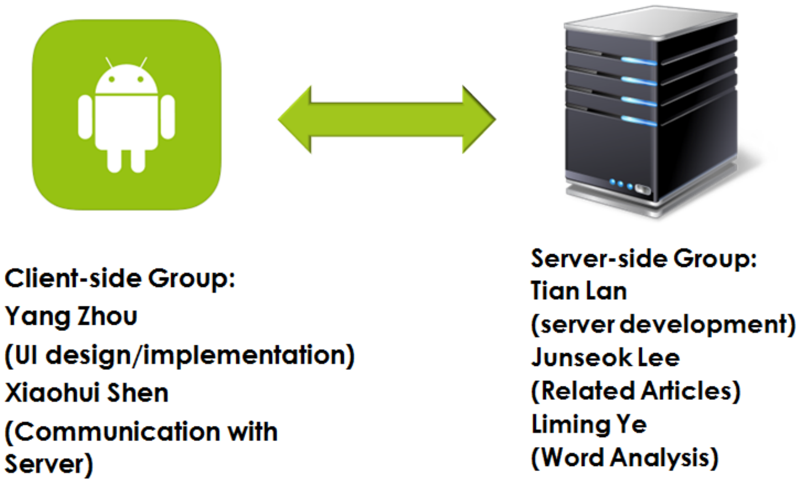


Figure 6 Work allocation

6. FUTURE WORK

Based on our user study, in the future we would like to add social networking functions on our client to enable more user interaction. The app could also be customized according to individual needs to provide better seminar recommendations.

And we could extend the app onto other mobile platforms, such as iOS and Windows Phone.

7. CONCLUSION

We developed a convenient light-weight framework for users to manage seminars easily. We adopted server-client framework, in which we introduce an Android mobile app as the light-weight client and develop the Apache web server as the database repository and data processing center. The pilot user study on the students in TAMU CSE department showed that our system was easy and convenient to use.

ACKNOWLEDGMENTS

We thank Dr Shipman and all the students in the Computer and New Media course to help us improve our design and implementation.

REFERENCES

1. XAMPP. http://en.wikipedia.org/wiki/XAMPP

2. Google Calendar API Guides, Google Developers: https://developers.google.com/google-apps/calendar/

3.MySQLi. http://en.wikipedia.org/wiki/MySQL

4.Android Studio. http://developer.android.com/tools/studio/index.html

5.Bray T. The JavaScript Object Notation (JSON) Data Interchange Format[J]. 2014.

6.Tjong Kim Sang, Erik F., and Fien De Meulder. "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition." In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4, pp. 142-147. Association for Computational Linguistics, 2003.

7.Named Entity Recognizer by Standford University. http://nlp.stanford.edu/software/CRF-NER.shtml

8.Rhodes, Brandon, and John Goerzen. "IMAP." Foundations of Python Network Programming. Apress, 2010. 243-262.

9.imaplib documentations. https://docs.python.org/2/library/imaplib.html