Interaction Design- Lecture Information Management Android Application

Tao Liu
Computer Science Department
North Carolina State University
Raleigh, NC 27695, USA
tliu11@ncsu.edu

Abstract—This paper describes an Android interactive application that can help college students manage lecture information and recommend lectures based on their interests. It gives the detailed representation about the design of the interaction interface, following the design principles in Human-Computer Interaction. This application combines mobile platform with lecture information and recommendation from algorithm and social network, which can give college students a convenient and easy to get the lecture information they want.

Keywords—Interaction design, Android application, lecture information,recommendation

1 Introduction

It's an age of information explosion. Everyday we are exposed to so much information on the Internet, bulletin board, leaflet, etc. Sometimes it's really difficult for us to gather information, and more difficult is how to get the information that we are really interested in. For college students, lectures are important part of academic life by giving us a quick view about a topic. There are so many lectures on campus,and it's really inconvenient for us to search the Internet and bulletin board to find those you're interested in. You may get notification about some lectures from your department, but what about lectures from us department that you also interested in? We really need something to help us gathering all the information, and hopefully recommending us some good lectures based on our interests.

This paper presents an interaction interface design of Android application, following some design principle of Human-Computer Interaction, and more specifically, Human-Android Interaction. The long-term goal of this project is an Android system that can gather lecture information from college website and mailling list, display it, and recommend some lectures based on the keywords the users set and some behavior of the users, such as selecting "Like", "Block"or "Report Spam".

The remainder of this paper describes the modeling and implementation considerations relevant to the building of this Android applications, the interactive interface design principles, evaluation of the system, and some discussion about challenges and future work.

2 Related Work

Android platform is the most popular smartphone platform nowadays. Google makes Android system so great by making it an open source environment. This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. On the Android official website, there are many guidelines about Design, Develop, and Distribute of Android Apps.

Android Apps provide various interaction between users and apps by using the touchscreen, virtual keyboard, voice input, and many function widgets. You can design many interaction patterns using these features. For example, you can use listview to deal with the data,

use the navigation bar to navigate between different parts of the app, get notifications of updated information, personalize the interface and functions, use various screen interaction and interact with you friends.

There are many android applications aimed at information gathering and recommendation. For example, TED mobile application, which can display and recommend TED speech videos; Mtime Application, which can gather all the movie information, search the nearby cinema based on your location, and share the information with your friends. Here I'm going to talk about another two applications in more details.

The first one is an very interesting application named Food-bot(see figure 1). It's developed by a CMU student, and the idea of the application is "College students rejoice. The ultimate source of free food events is finally here." It's designed to help college students find free food events on campus. And it seeks to serve two groups: students looking for free food, and student groups looking for some free publicity for their events. So it's a win-win! The application gets free food events information in two ways:1.People post event information manually, and 2.Food-Bot processes emails, extracts free food event information, and posts it on the site. It uses probability theory to analyze if a certain event provides free food, and then use natural language processing to extract information such as data and places. But due to the automation, sometimes Food-Bot will post events that are not appropriate for the general public to attend. Users can point out if there is a certain event (or type of event) that shouldn't be here and/or help by deleting those events. As for the interface, first, you can choose different universities, and the next page you see is a calendar with all the food events. You can view the events, send the events to your reminder, add events by touching "Add Events" button and input date and place, you can also make some personalization settings,such as how long the cache will keep the events, how long before the event that the application will remind you. This application sparks my thoughts a lot by using artificial intelligence to find events. It allows users to create events while also take spam information into consideration. And the reminder function is also very helpful.

The second one is DoubanFM(figure 2), a personal online music radio service. It's highly acclaimed for the concise interface and sophisticated algorithm to recommend music. There are many music channels, for example, personal channel, rock music channel, country music channel. If you like a song, you can touch the red heart "like", and if you don't like it, you can select the "dustbin" or just go to next song. Each time DoubanFM's sophisticated algorithm learns a bit more of your music taste and plays music that suits you better. Recommendation is a important aspect of interaction: it allows users a unique opportunity to explore their tastes, and learn about new items. The system records your behavior and then predict what you like, thus can give us a great user experience. DoubanFM gives me the idea of recommendation by indicating what you like and don't like. It can also be used in my lecture information recommendation system, and besides, we can also use keywords setting to tell the system what we're interested in.

In the paper *Interaction Design for Recommender Systems*[4], it highlights the role of transparency (understanding of system logic), familiar recommendations, and information about recommended items in the users interaction with the system. Most recommender systems work by asking users to rate some sample items and collaborative filtering algorithms use this input to match the current user with others who share similar tastes. But beside the algorithms, interaction design is also a very important factor which leads to satisfaction with a recommender system. To design an effective interaction, one must consider two questions: (1) what user needs are satisfied by interacting with the system; and (2) what specific system features lead to satisfaction of those needs. This paper gives me another thought about recommender: the recommendation can not only be online recommendation using algorithms, but also social recommendation from friends. Since friends know what you are interested in, they can send you recommendation they think useful through social network.

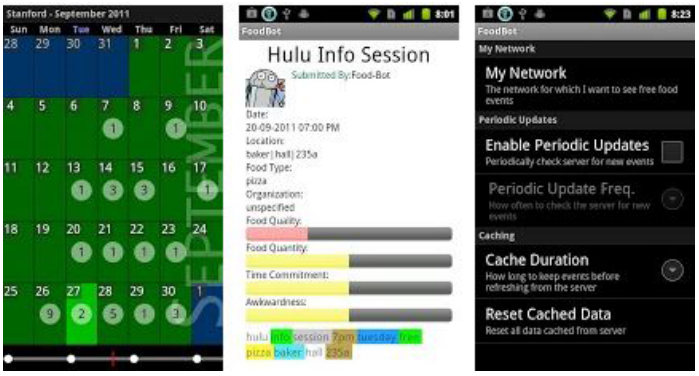

Fig. 1. Interface of Food-bot

Fig. 2. Interface of DoubanFM

3 Project Planning

3.1 Problem Statement

The goal for this application is that college students can have access to all the lecture information on campus and get some recommendation from the system. When designing the system and the interface, there are several problems needed to be solved:

How to get lecture information on campus?

How to extract information from the online information source, for example, lecture title, data, places, abstract,etc?

How to judge if the information is spam or not?

How to display the information structurally and clearly?

How to make the interface self-describing?

How to prevent errors for users?

How to design the interface to make users faster with the application?

How to recommend lecture information for users?

How to encourage users tell the system their interests more actively in order to better predict users' interests?

How to know what things are most important to users?

How to make users feel at home and in control by allowing personalized settings?

How to design functionalities of the application to make life easier?

In the design and implement process, we need to keep these problems in mind and solve them. To judge if the problems are solved, we need to evaluate the system. In 1994, Nielsen claimed that there were four basic ways of evaluating user interfaces:

- Automatically
- Empirically
- Formally
- Informally

Usability Evaluation Methods can generally be described as either empirical or informal using Nielsens words. Figure 3 shows us some evaluation methods that we can use to evaluate our system.

3.2 Task Analysis

Task Analysis is the study of the way people perform their jobs. Aim is to determine:

- what they do
- what things they use
- what they must know

It gathers both declarative and procedural knowledge:

- Declarative: objects and relationships
- Procedural: task sequences, goals and subgoals
- Also dependencies and constraints

Empirical Methods	Informal Evaluations
User Walkthrough	Heuristic Evaluation
Focus groups	Expert reviews
Structured observations	Cognitive walkthroughs
Cooperative evaluations	Predictive modelling – GOMS
Activity logging	Guidelines review
Data logging	Consistency inspection
Observations	Critical event analysis
Questionnaires	Dialogue Error Analysis
Interviews	Usability testing
Controlled user tests	
Physiological data analysis	

Fig. 3. Evaluation Methods

This application is aimed at providing college students with lecture information and recommend lecture for them. I'm going to use task decomposition method to analyze the tasks. Here is the Hierarchy description of the task domain:

0. Get information about lectures
    1. get general information about lecture
        1.1 search lectures in search box
        1.2 set the date range of lectures you want to view
        1.3 select locations you want
        1.4 get more details about the lecture
            1.4.1 view detailed information about lecture,like time, date, place, speaker, abstract,link
            1.4.2 If you want to get reminded about the lecture, add it to reminder
            1.4.3 If you think a certain lecture is a spam, report it as spam
        1.5 view the lecture events in a calendar
    2. get recommendation from the system
        2.1 set keywords about topic that you're interested in
        2.2 set the date range
        2.3 select locations
        2.4 you can also search lectures in the search box
        2.5 indicate your attitude towards a certain recommended lecture
            2.5.1 if you like it, select "like"
            2.5.2 if you don't like it, select "block"
            2.5.3 if you want to share it with your friends, select "share"
        2.6 get more details about the lecture
            2.6.1 view detailed information about lecture,like time, date, place, speaker, abstract,link
            2.6.2 If you want to get reminded about the lecture, add it to reminder
            2.6.3 If you think a certain lecture is a spam, report it as spam
    3. get reminded of lectures from system
        3.1 edit a certain lecture reminder, or delete it
        3.2 add a new lecture to reminder
    4. get location information about lectures on map
        4.1 search a location you want to find
        4.2 touch a location and see lectures going on there

Plan 0: do 1, 2, 4 in any order,as needed when you want to look the reminder, do 3
Plan 1: do 1.1,1.2,1.3,1.4,1.5 in any order, as needed
Plan 1.4 do 1.4.1,1.4.2,1.4.3 in any order
plan 2: do 2.1,2.2,2.3,2.4,2.5,2.6 in any order
plan 2.5: do 2.5.1, 2.5.2,2.5.3 in any order
plan 2.6: do 2.6.1,2.6.2,2.6.3 in any order
plan 3: do 3.1,3.2 in any order
plan 4: do 4.1, 4.2 in any order

The target users of this application are college students who want to get information about lectures in a quick and convenient way. To better illustrate the task and usage of this application, I'm going to give a simple representative scenarios. Scenarios describe the problem situation using natural language understood by all stakeholders. They describe use in detail, but as a tentative, working representation. They are concrete descriptions but are also very flexible.

Representative Scenarios:

Tom is a graduate student in Computer Science Department. He likes listening to lectures to get a vision about advanced topics in computer science. He uses this "TouchLecture" Android application to get information about lectures. He wants to know lectures going on this week. He opens the application and the first page is general lecture information page. He selects department CSC, sets the time range as"in this week". Then he looks through the

lecture title displayed. The first one is about HCI, and he is interested in it, so he touches the title and a new page appears with more detailed information about the lecture. He thinks, that's really an interesting lecture, and I'm going to attend it. It's 3 days later, so he adds it into reminder in case he forgets it.

He also tries the recommendation function. Since he is very interested in HCI and AI, so he sets the keywords as "HCI" and "AI". He looks through the recommended items, and he finds a lecture really interesting, so he touches the quick action button and selects"like", and he also wants to share the information with his friend Leon who is also interested in HCI , so he selects "share" and share it to Leon.

Three days later, 30 minutes before the lecture, he gets notification that there is a lecture today. Then he goes to the lecture.

4 Interaction Interface Design

User interface is what users directly interact with. So interface design is a very important part of a successful application. As Paul Rand said: "To design is much more than simply to assemble, to order, or even to edit; it is to add value and meaning, to illuminate, to simplify, to clarify, to modify, to dignify, to dramatize, to persuade, and perhaps even to amuse." Design is a complex and meaning work. In this section I'm going to present the user interface design process of my Android application, following some design principles. I decompose the task into 4 tasks: Lecture (general lecture display), Recommendation, Reminder, Map. The following content is organized in the order of whole layout to detailed design.

4.1 Themes

Themes are Android's mechanism for applying a consistent style to an app or activity. The style specifies the visual properties of the elements that make up your user interface, such as color, height, padding and font size. The system themes provide a solid foundation on top of which we can selectively implement our own visual stylings. In my application, the base color is blue,gray and white. As for the theme, I apply the theme provided by Android: "Theme.Wallpaper.NoTitleBar".

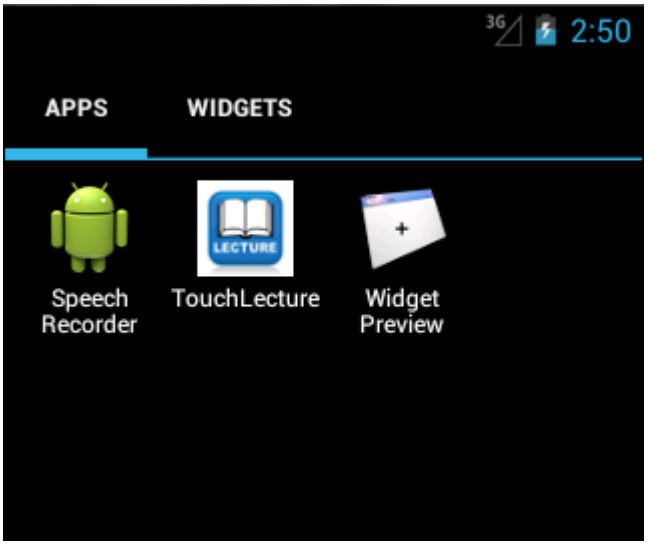

Fig. 4. Launcher Icon

4.2 Icon Design

Creating a unified look and feel throughout a user interface adds value to your product. Pictures are faster than words.They get people's attention and can be much more efficient than words. Well designed icon can make the interface more attracting and can create a polished and unified experience for the user.

4.2.1 Launcher Icon

A launcher icon is a graphic that represents your application. Launcher icons are used by Launcher applications and appear on the users Home screen. Launcher icons can also be used to represent shortcuts into your application. Application launcher icons have three primary goals:

•Promote the brand and tell the story of the app.

•Help users discover the app on Google Play.

•Function well in the Launcher.

Since my application is about lecture information, I name it "TouchLecture". Figure 4 shows the launcher icon of this application. The book in the Launcher icon represents knowledge. And the characters "Lecture "under the book indicates this is a application about lecture.

4.2.2 Tab Bar Icons

Tab icons are graphical elements used to represent individual tabs in a multi-tab interface.

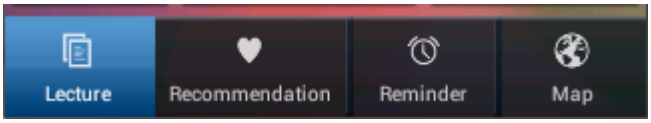

Fig. 5. Tab Bar Icons

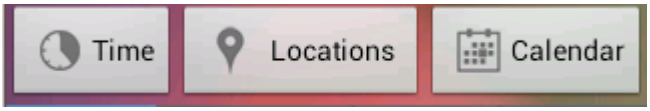

Fig. 6. Menu Icons 1- Time,Location, Calendar

Fig. 7. Menu Icons 2- Search

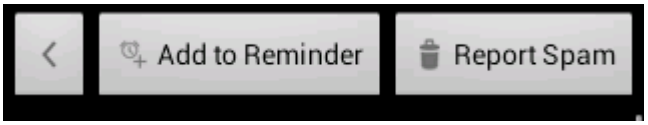

Fig. 8. Menu Icons 3- Back, Add to Reminder, Report Spam

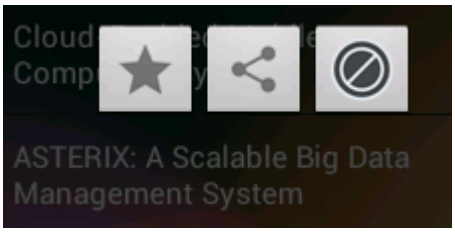

Fig. 9. Menu Icons 4-Like, Share, Block

Each tab icon has two states: unselected and selected. I discompose my application into 4 tabs: Lecture, Recommendation, Reminder and Map. Figure 5 shows the Tab Bar icon of this application. One of the principles of interface design is that it should be self-describing, that is, the design should convey what it does. So the Tab Icons should describe their functions explicitly.

4.2.3 Menu Icons

Menu icons are graphical elements placed in the options menu shown to users when they press the Menu button. They are drawn in a flat-front perspective and in greyscale. Elements in a menu icon must not be visualized in 3D or perspective. The design of menu icon should follow one of Shneiderman's Eight Golden Rules of Interface Design: strive for consistency. That is, if users see a menu icon, he /she will know what it means because the same icon serves the same function in other application. Google designs many icons that developers can use to keep consistency of applications. Figure 6-9 shows the menu icons of this application.

4.3 Lecture Page

The Lecture Page is the main page of this application. It's the first page that users see,so it need to serve some general use. In this application, the first page displays the general lecture information. There are some widgets for users to get the information they want: a search box, a button to set time range, a button to select locations, and a button to view in calendar. And in order for students in different department to get lecture information in their department quickly, I add a spinner so users can select their department.Visual hierarchy is very important in interface design. First, Widgets that users access most should be in high hierarchy. From the survey on my friends, I think the search box should be in highest hierarchy, so I put it in the top of the page. The list which displays the lecture is the main part of this page, it should be in the center part of the layout. As for some set buttons, it's reasonable to put them at the bottom, and since their functions are related, they should be grouped together. Figure 10 shows the whole layout of this page.

4.3.1 Department Selection Spinner

A spinner provides a convenient way to make selections. I collect the department name of the entire school and put it in the selection list. The default selection is all department, and if you need to specify department, just select it. Figure 11 shows the spinner.

4.3.2 Lecture Listview

In interface design, it's important to pay attention on how to order and group, and how to structure data. Listview provides a good way to organize the lecture information. There are

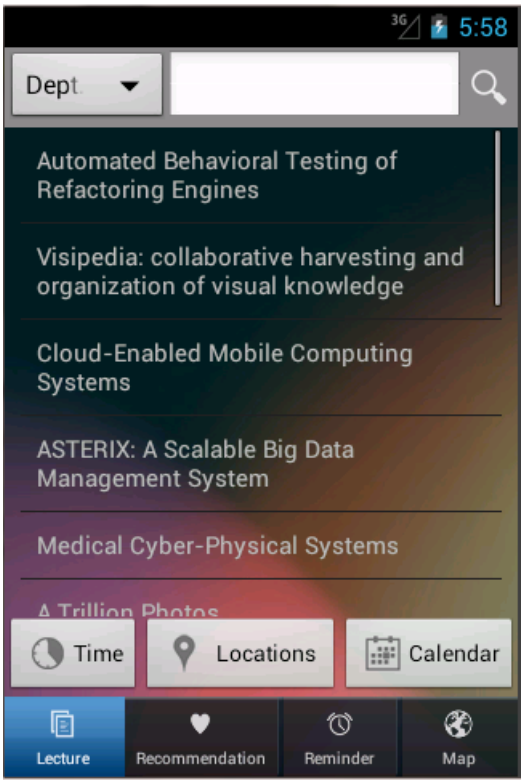

Fig. 10. Lecture page-whole layout

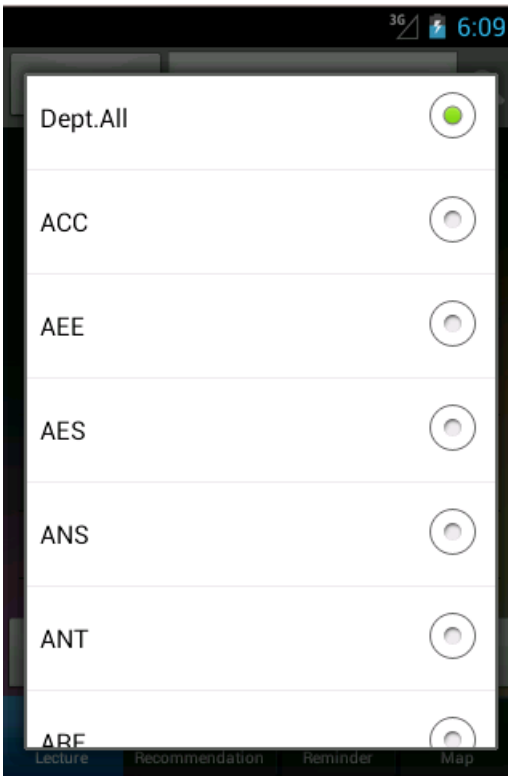

Fig. 11. Lecture page-Department Selection Spinner

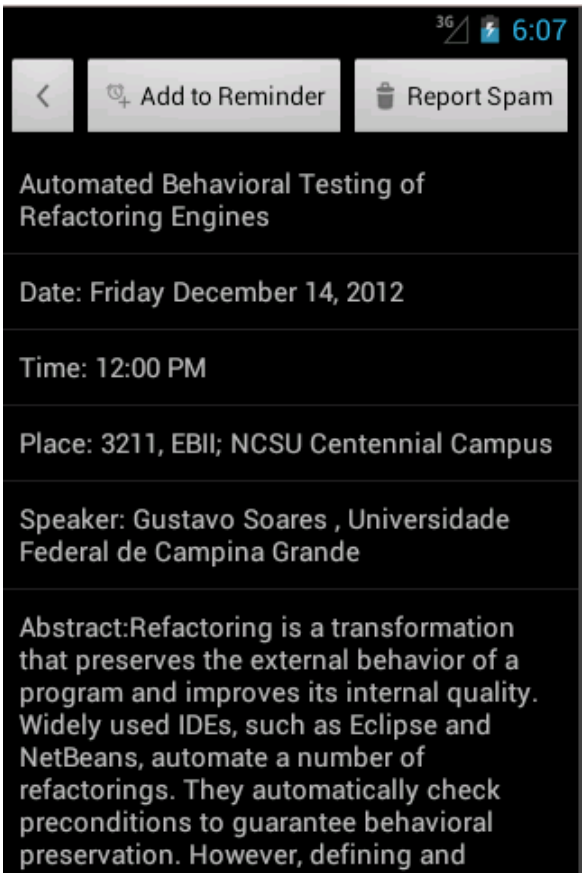

Fig. 12. Lecture page-Detailed lecture information

so many lectures, we can order them in time order in default, and use the scroll control to view it.

It's also important to "only show what I need when I need it". People get overwhelmed when they see too much at once. So we can hide options that aren't essential at the moment, and teach people as they go. So I only put the title of the lecture in the list, if you want to see the detailed information of the lecture, just touch the title. If the users find that the content of this lecture appeals to them, they can add it to reminder. If the users find it's just some spam information, they can report spam to help the system filter such information out. Figure 12 shows the detailed information page.

4.3.3 Set time range

If the users want to view lecture in a certain time range, they can set it here. At first, my design is what shows in figure 13. But since the timepicker and datepicker widgets are too

Fig. 13. Lecture page-Set time range-original

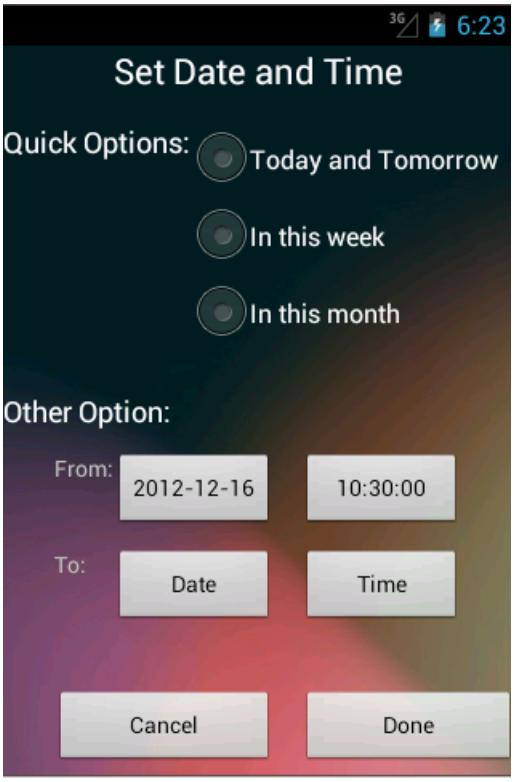

Fig. 14. Lecture page-Set time range-new

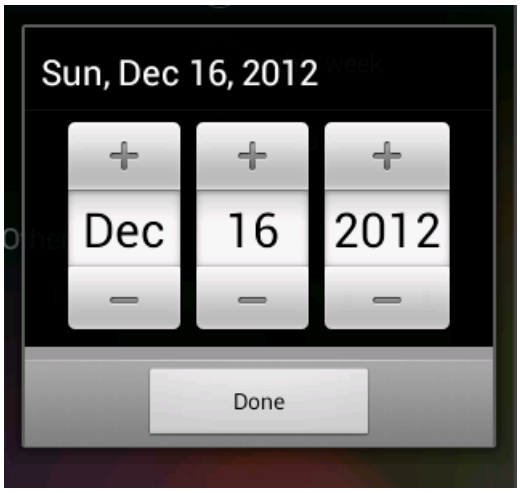

Fig. 15. Lecture page-Set date

large, and we can't set the size in Android, the page seems very crowded. In order to make it simple and brief, and also make important things fast, I add quick options, so that users can quickly select some range. And the "from" and "to" options is also designed to be more simple and clear. Following Shneiderman's rule " Offer informative feedback.", if users set time and date, it will appear on the button. As for the layout, we need to apply Proximity Principle Objects that are close to each other will be seen as belonging together. So the radiobuttons of quick options should be close. The 4 buttons of "other options " is another group, so they should be grouped together. "Cancel" and "Done" buttons sever the whole page, we can put them at the bottom, and they should be wider, as suggested in Fitts' Law. Figure 14 shows the newly designed page.

4.3.4 Set Locations

If users want to attend lectures at a certain location, they can set it in "Location" button. Here I use a dialog to display the location options.A dialog is a small window that prompts the user to make a decision or enter additional information. It provides a quick way of interaction instead of building a new activity, which would slower the reaction of the system.Figure 16 shows the set locations dialog.

4.3.5 Calendar View

A calendar makes it more clear to view what's going on in a week or month. Here the calendar

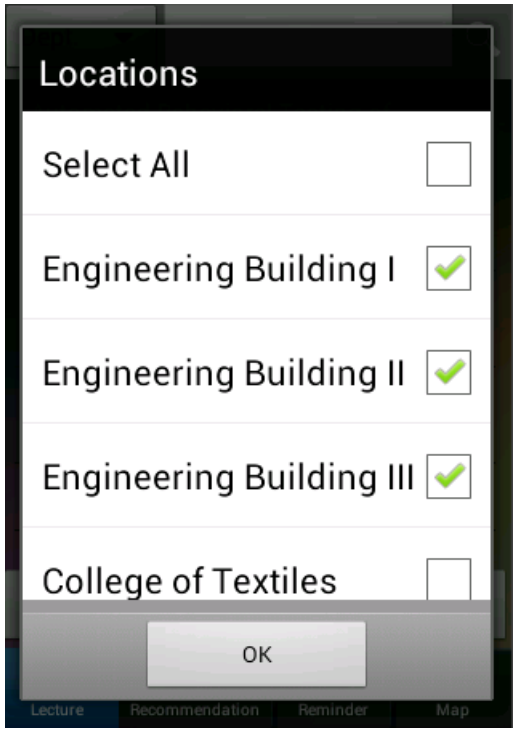

Fig. 16. Lecture page-Set locations

Fig. 17. Lecture page-Calendar view

is just a mock-up interface. I haven't added events into it. Figure 17 shows the calendar view.

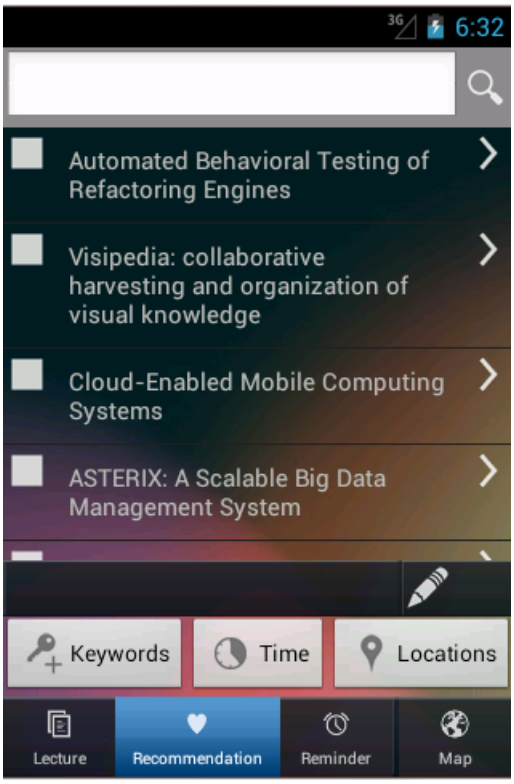

Fig. 18. Recommendation page

4.4 Recommendation

Recommender can act as personalized decision guides for users, aiding them in decision making about matters related to personal taste. An effective recommender system inspires trust in the system; has system logic that is at least somewhat transparent; points users towards new, not-yet -experienced items; provides details about recommended items, including pictures and community ratings; and finally, provides ways to refine recommendations by including or excluding particular genres. Users expressed willingness to provide more input to the system in return for more effective recommendations. A well designed interface can make users tell the system their interests more actively, and so the system can better predict their tastes. This page has similar layout as Lecture page. Figure 18 shows the page layout of Recommendation page.

4.4.1 Add keywords

Users can add keywords so system can know their interests and thus give recommendation based on that. After users input keywords, the

keywords will display on the screen. My idea is that users can add many keywords, and users can directly manipulate on the displayed keywords, for example, if users add many keywords, but just want to view on topic, they can click the keywords to select and highlight it, they can also delete any keywords.

4.4.2 Quick action

I decided to add quick action pattern after I experienced some Android application. Users can indicate their attitude towards a certain recommended item by touching the button in the item .They can select ”like” if they like the it, ”block” if they don’t like it. And they can also share it with their friends. They can also get notification if they receive recommendation from their friends. Social recommendation is an efficient and important way to get recommendation, as friends know you better. Figure 9 shows the quick action options.

4.5 Reminder

Reminder is a useful function to remind users before the lecture. Users can edit or add reminder items through ”Edit” and ”New” buttons. I realized only title here is not enough, there should be a quick way for users to know time and location in the reminder page. I add sub-items to display time,date and location, so it will be more structural. And to highlight the title, I use different colors and fonts for items(light blue,bold) and sub-items(white,italic). Figure 19 shows the page layout of Reminder page.

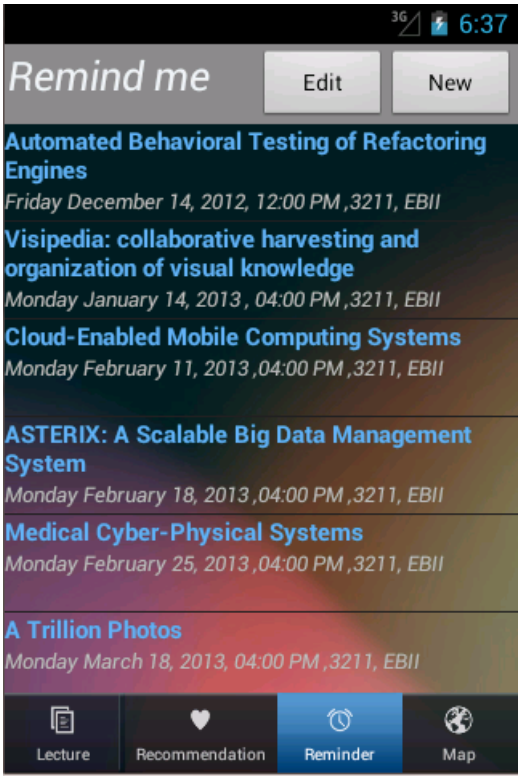

Fig. 19. Reminder page

4.6 Map

A map can help users find the location. It can also show lectures going to given in a certain building.Here I use Google map but haven’t added any function based on it.Figure 20 shows the page layout of Map page.

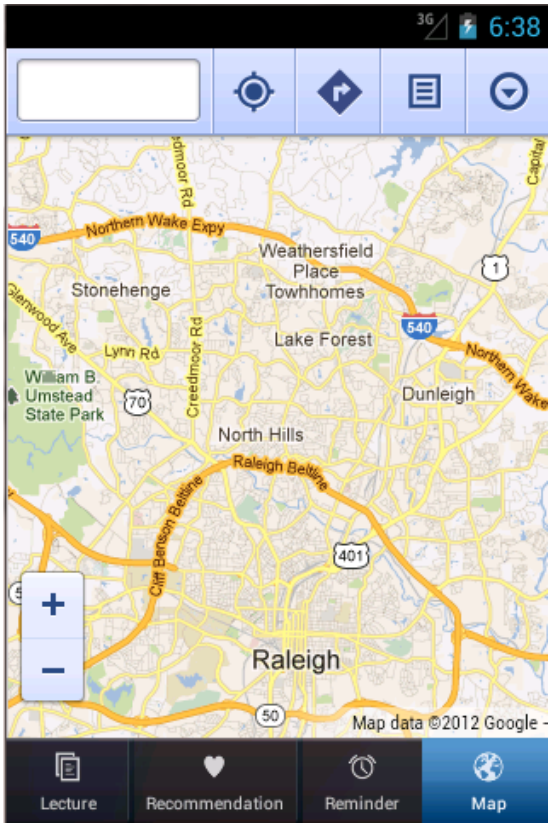

Fig. 20. Map page

5 Implementation Support

This application is built on Android platform. Android is an operating system based on Linux with a Java programming interface. It’s installed on thousands of devices from a wide

range of manufacturers. Android exposes your application to all sorts of hardware that you'll find in modern mobile devices digital compasses, video cameras, GPS, orientation sensors, and more. In this project, I use Android 4.1.2 version(API 16). It uses Intel x86 Atom System Image, which makes it much faster to run the Android Emulator. The Android Software Development Kit (Android SDK) provides all necessary tools to develop Android applications. This includes a compiler, debugger and a device emulator, as well as its own virtual machine to run Android programs. To run this application, first you need to set up the environment[9]. After that, you can run the application on Android Emulator or directly connecting it to your Android smartphone.

6 Final System

The final system is an Android application, but just a mock-up project. You can click all the widgets but most of their functions haven't been realized. You can see the final system in the video.

Here is some examples of walking through of some functionalities of the system.

1.**Search for lectures about HCI in this week at Centennial Campus**

First, start the application by touching the Launcher icon. Then input "HCI" in the search box, click "Time" button and go into "Set time range" page, here you just need to select "In a week" in "Quick options" . Go back and click "Locations" and select location"Centennial Campus". View through the search result (only titles here). If you find a topic interesting, you can click it and to view more details about it. You can also add it to reminder so you won't miss it.

2.**Get recommendation about HCI**

Goes to the Recommendation page, add keywords"HCI", here you can also specify time range and locations. Look through the recommended items, you can "like" it or "block" it or "share" it with your friends.

7 Evaluation

Evaluation is an integral part of the development process and can take the form of an informal walkthrough or a more structured heuristic evaluation. We need to evaluate the usability of an interface so wen can know the effectiveness, efficiency, and satisfaction with which specified users can achieve specified goals in a particular environment, and thus to improvement our interface.

Fig. 23. Action Bar

7.1 Empirical Evaluation

Here I use Heuristic evaluation to evaluate my interface. Heuristic evaluation is a form of usability inspection where usability specialists judge whether each element of a user interface follows a list of established usability heuristics.

The result of Heuristic evaluation is shown in Figure 22(at the end of this paper)

Identified problems:

1. The Department selection spinner is too long and the name of department should better be full name, not just abbreviation.(I don't know where to find the full name here. Just found the abbreviation on courses.ncsu.edu)

2. There should be more personal settings. (I'm planing to add an action bar on the top of each page(which shows in figure 23). There are personal settings and Help documentation when selecting "more"(the last icon). But I have some difficulty realizing it.)

3. There is no error messages or error prevention.

4. There is no "undo " function (also planning to realize in the action bar)

5. There is no help and documentation.

7.2 Analytical Evaluation

GOMS Model:

Define_model: "Search Lecture"
    Starting_goal is Do Search
Visual_object: Search_box
    Content is "Search".
Visual_object:Set_time_range
    Content is "Time".
Visual_object:Quick_options_In_this_week
    Content is "In this week".

Visual_object:Time_done
Content is ”Done”.
Visual_object: Set_location
Content is ”location”.
Visual_object: Location_options
Content is ”Engineering Building II”.
Visual_object: Location_done
Content is ”Done”.
LTM_item: Text_to_type
Name is Text_to_type.
Text is ”HCI”.

Method_for_goal:Do Search
Step a1. Accomplish_goal: Start_Launch_application. // 1054 ms
Step a2. Accomplish_goal: Type in_search_box
Step a3.Accomplish_goal: Set Time_range
Step a4. Accomplish_goal: Set Locations

Method_for_goal: Type in_search_box
Step b1. Look_for_object_whose Content is ”Search” // 1200 ms
and_store_under $< target >$.
Step b2. Point_to $< target >$ // 1100 ms
Step b3. Home_to keyboard. // 400 ms
Step b4. Recall_LTM_item_whose Name is Text_to_type // 1200 ms
and_store_under $< text >$.
Step b5. Type_in Text of $< text >$. // 718s

Method_for_goal:Set Time_range
Step c1. Look_for_object_whose Content is ”Time” // 1200 ms
and_store_under $< target >$.
Step c2. Point_to $< target >$. // 1100 ms
Step c3.Touch_button // 200 ms
Step c4. Look_for_object_whose Content is ”In this week” // 1200 ms
and_store_under $< target >$.
Step c5. Point_to $< target >$. // 327 ms
Step c6. Touch_button. // 200 ms
Step c7. Look_for_object_whose Content is ”Done” // 1200 ms
and_store_under $< target >$.
Step c8. Point_to $< target >$. // 327 ms
Step c9. Touch_button. // 200 ms
Step c10. Delete $< target >$; RGA. // 50 ms

Method_for_goal:Set Locations
Step d1. Look_for_object_whose Content is ”Locations” // 1200 ms
and_store_under $< target >$.
Step d2. Point_to $< target >$. // 1100 ms
Step d3.Touch_button // 200 ms
Step d4. Look_for_object_whose Content is ”Engineering Building II” // 1200 ms
and_store_under $< target >$.
Step d5. Point_to $< target >$. // 327 ms
Step d6. Touch_button. // 200 ms
Step d7. Look_for_object_whose Content is ”Done” // 1200 ms
and_store_under $< target >$.
Step d8. Point_to $< target >$. // 327 ms
Step d9. Touch_button. // 200 ms
Step d10. Delete $< target >$; RGA. // 50 ms

The recorded model is in file”recoder.txt”. From which we can see that the time for system response is long(since the emulator is a bit slow). And with the quick option in time range setting, it’s fast to set time range. And it’s slow to select location, since the scrolling takes some time.

8 Discussion

It’s my first time to write Java and Android code, so it really took me lots of time to set the environment, to learn Java language, and to learn the architecture of Android. And the design is also not an easy thing, you need change it now and then if you come up with some new idea or find some usability flaw. In this project, I implemented the entire interface(except the action bar). But as for the functionalities, I only realized time range_setting options, and switching between page and buttons, adding keywords. I also added Google Map in it, but haven’t added some functionalities such as show lectures going on in a certain building. There is still lots of work to do, such as getting lecture information from the Internet, deciding which items are actually lecture event, extracting lecture details from the items, and ordering the lectures according to time and location, adding lecture to calendar view, blocking spam information, and remind the users about upcoming events.

9 Conclusion

Interaction interface design on smartphone is an important research area of HCI as the popularity of smartphone. Users can have a more natural control of the interface with touchscreen. This paper described the interaction interface design and evaluation of an Android application "TouchLecture", and suggested the combination of mobile application with information collection and recommendation, and its incorporation with social recommendation. There is still lots of work to do to implement the system in the future.

References

[1] Open Valley course : Developing Android Applications, http://masl.cis.gvsu.edu/2010/10/12/android-dev-tutorial-now-available-on-the-gvsu-itunes-u-portal/

[2] Android Design, http://developer.android.com/design/index.html

[3] Food-bot: http://food-bot.com/home

[4] Swearingen, K., and Sinha, R. "Interaction Design for Recommender Systems," paper presented at the International Conference on Designing Interactive Systems (DIS 2002), London, June 25-28, 2002.

[5] Resnick, P, and Varian, H.R. Recommender Systems.

[6] Sinha, R. and Swearingen, K. Comparing Recommendations made by Online Systems and Friends. Proceedings of the DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries, 2001.
1997 Commun. ACM 40, 3 (56-58).

[7] Abe Crystal, Beth Ellington, Task analysis and human-computer interaction approaches, techniques, and levels of analysis,Proceedings of the Tenth Americas Conference on Information Systems, New York, New York, August 2004

[8] Draper, S.W. (1993) "The notion of task in HCI" pp.207-208 Interchi'93 Adjunct proceedings (eds.) S.Ashlund, K.Mullet, A.Henderson, E.Hollnagel, T.White (ACM)

[9] Android environment setting: http://developer.android.com/sdk/installing/index.html

[10] Nielsen's 10 heuristics for interface design:http://www.useit.com/papers/heuristic/heuristic_list.html

[11] Shneiderman's "Eight Golden Rules of Interface Design": http://faculty.washington.edu/jtenenbg/courses/360/f04/sessions/schneidermanGoldenRules.html

[12] Usability Technique: Heuristic Evaluation - A System Checklist: http://www.stcsig.org/usability/topics/articles/he-checklist.html

1. Visibility of System Status				
#	Review Checklist	Evaluator1	Evaluator2	Comments
1.1	Does every display begin with a title or header that describes screen contents?	Excellent	Excellent	
1.2	Is there a consistent icon design scheme and stylistic treatment across the system?	Excellent	Excellent	
1.3	Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?	Poor	Good	1.The square button in Recommendation doesn't appear in the same place. 2.Good
1.4	Is there some form of system feedback for every operator action?	Good	Good	1.Clicking on the button do have color change, but it still not as good as other Android application. 2. Some functionalities are not realized.
1.5	Is the current status of an icon clearly indicated?	Good	Excellent	1.but the bottom buttons change to blue, while the other buttons change to yellow.
1.6	Is there visual feedback in menus or dialog boxes about which choices are selectable?	Excellent	Excellent	
1.7	Is there visual feedback in menus or dialog boxes about which choice the cursor is on now?	Poor	Good	1. didn't see changes
1.8	If multiple options can be selected in a menu or dialog box, is there visual feedback about which options are already selected?	Excellent	Excellent	1. The check box looks great.
1.9	Are response times appropriate to the user's cognitive processing?	Good	Good	1. It's a little slow, maybe because it's on Emulator
1.10	If users must navigate between multiple screens, does the system use context labels, menu maps, and place markers as navigational aids?	Excellent	Excellent	1. The buttons on bottom look great.
2. Match Between System and the Real World				
2.1	Are icons concrete and familiar?	Excellent	Excellent	
2.2	Are menu choices ordered in the most logical way, given the user, the item names, and the task variables?	Excellent	Excellent	
2.3	Are menu titles parallel grammatically?	Excellent	Excellent	
2.4	Do related and interdependent fields appear on the same screen?	Good	Excellent	1.seems calendar is a little unrelated with time and location
2.5	Do menu choices fit logically into categories that have readily understood meanings?	Good	Good	2. The department name can be full name, not just abbreviation
3. User Control and Freedom				
3.1	Is there an "undo" function at the level of a single action, a data entry, and a complete group of actions?	Poor	Good	1.There are some cancel buttons, but no undo function.
3.2	Can users cancel out of operations in progress?	Poor	Good	1.not really
3.3	Are character edits allowed in data entry fields?	Good	Good	
3.4	If menu lists are long (more than seven items), can users select an item either by moving the cursor or by typing a mnemonic code?	Good	Poor	1.The department is too long, can select it by cursor but can't by typing a mnemonic code. 2. The department menu is too long.
3.5	Are menus broad (many items on a menu) rather than deep (many menu levels)?	Excellent	Excellent	
3.6	Can users easily reverse their actions?	Excellent	Excellent	
3.7	Can users set their own system, session, file, and screen defaults?	Poor	Poor	1.There is no personal settings.
4. Consistency and Standards				
4.1	Have industry or company formatting standards been followed consistently in all screens within a system?	Excellent	Excellent	1.It's pretty like an Android application
4.2	Has a heavy use of all uppercase letters on a screen been avoided?	Excellent	Excellent	
4.3	Are icons labeled?	Excellent	Excellent	
4.4	Are vertical and horizontal scrolling possible	Excellent	Excellent	1.horizontal scrolling is good, but no vertical scrolling,

5. Help Users Recognize, Diagnose, and Recover From Errors				
5.1	Is sound used to signal an error?	Poor	Poor	1. There is no sound here
5.2	Do prompts imply that the user is in control?	Excellent	Excellent	
5.3	Is there any error message?	Poor	Poor	1.There is no error message.
6. Error Prevention				
6.1	Does the system prevent users from making errors whenever possible?	Poor	Poor	1.not really 2.haven't seen that
6.2	Does the system warn users if they are about to make a potentially serious error?	Poor	Poor	
6.3	Do fields in data entry screens and dialog boxes contain default values when appropriate?	Good	Good	
7. Recognition Rather Than Recall				
7.1	Are prompts, cues, and messages placed where the eye is likely to be looking on the screen?	Excellent	Excellent	
7.2	Does the system gray out or delete labels of currently inactive soft function keys?	Poor	Poor	
7.3	Are borders used to identify meaningful groups?	Good	Excellent	
7.4	Is the first word of each menu choice the most important?	Excellent	Excellent	
8. Fexibility and Minimalist Design				
8.1	Does the system allow novices to use a keyword grammar and experts to use a positional grammar?	Good	Good	
8.2	Does the system provide function keys for high-frequency commands?	Good	Good	
8.3	If the system uses a type-ahead strategy, do the menu items have mnemonic codes?	Good	Good	
9. Aesthetic and Minimalist Design				
9.1	Are all icons in a set visually and conceptually distinct?	Excellent	Excellent	
9.2	Have large objects, bold lines, and simple areas been used to distinguish icons?	Excellent	Excellent	
9.3	Does each icon stand out from its background?	Excellent	Excellent	
9.4	Are menu titles brief, yet long enough to communicate?	Excellent	Excellent	
10. Help and Documentation				
10.1	Is there any help and document?	Poor	Poor	1.There is no help and documentation here. 2. None

Fig. 22. Heuristic evaluation