RWTH AACHEN UNIVERSITY

universität bonn

NoteCarrier

A Nomadic Application for Bi-Directional Class-Room Communication.

Thesis at the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Ahsan Nazir Sheikh

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Michael Meier

Registration date: Feb 13th, 2012
Submission date: Sep 2nd, 2012

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Bonn, September 2012
Ahsan Nazir Sheikh

Contents

List of Figures

List of Tables

Abstract

Student participation in the class plays a vital role in any lecture. Without student involvement the lecture loses its meaning and importance. A main reason for reduced participation in classes is the communication gap between students and lecturers. Many students feel hesitant when communicating directly in a lecture and prefer not to participate. This can negatively impact their studies.

In this thesis a proposed application NoteCarrier provides a communication channel between the lecturer and the student. This not only increases student interaction but also helps lecturers improve their techniques and deliver successful lectures by using smart devices.

The application evolved into its current state with the continuous user's feedback. The initial concept of NoteCarrier application was presented to the students in form of storyboards, to lighten the need of software in real scenarios. After the storyboard's evaluation, paper prototype was created to illustrate the possible features and interaction techniques to its potential users. These prototypes were assessed with group of students to extract the right balance between the user needs and features provided by the application.

Based on evaluation result, application's software prototype was implemented for Mac OS and Apple IOS, providing features for student's active participation and support for other features like taking notes, downloading slides, and synchronized off-line slide view. Software prototype is evaluated in actual class session at University of Bonn. Questionnaire and post interviews were conducted at the end of the session to gather user's feedback.

NoteCarrier is composed of three sub applications, a cocoa based application server which acts as an interface between client devices and Keynote. Two IOS based applications to facilitate students and assist lecturers during class. Students and lecturers exhibited their high interest in application current features and they also endorse that integration of such system will definitely have positive impact on their learning process.

Acknowledgements

I am very thankful to my supervisors Prof. Dr. Jan Borchers and Prof. Dr. Michael Meier for their stint, copious help and supervision during the development phases of the final Master dissertation and directing me toward the completion of a high quality thesis. I would like to thank Jonathan Diehl for giving precious suggestions, and support throughout my Master research thesis.

I am very grateful to Mr. Daniel Speicher for his time and help in the user study conducted during his lecture and also many thanks to those students who participated.

Gratitude is also due to many reviewers especially Misbah Ishaq and Charles Green for their devoted time and for giving their valuable helpful suggestions and feedback in completing this dissertation.

Last but not least, I would like to thank my parents for their full support and encouragement throughout my life.

Conventions

Throughout this thesis we use the following conventions.

Important terms are written in *emphasized typeset* when they appear in the text for first time.

Source code and implementation symbols are written in typewriter-style text.

```
myClass
```

In the thesis, *Lecturers* are also referred as *Presenters* and in same manner *Students* may also referred as *Audience*

The whole thesis is written in American English.

Chapter 1

Introduction

" The grace to be a beginner is always the best prayer for an artist. The beginner's humility and openness lead to exploration. Exploration leads to accomplishment. All of it begins at the beginning, with the first small and scary step."

— Julia Cameron

The current era of life is equipped with different smart devices like smart phones, laptops, navigation systems and digital diaries etc. With extensive use of such devices, these devices not only influence our life style but also change the traditional way of communication as well.

Presentation, a technique to share information

Group discussions, meetings and classrooms are the most general examples of such environments where people need to share thoughts in different forms. One of the most commonly used ways to share your ideas is by presentation. The presentation might be in an office environment where the presenter has to present a topic to his colleagues or higher management or any other situation where one person has to share his idea with a number of people.

Classrooms as an example for group discussions

The classroom a typical example of information sharing among a group of people. This information might be in the form of a presentation, presented by the lecturer or questions and comments raised by the students. Although the traditional concept of a classroom has not changed for many years. Delivery of a lecture, writing down notes and asking questions are the main activities involved in a lecture.

Communication gap in current classroom setups

In classrooms, most of the time information flows in one direction i.e. from a lecturer to the students. New technologies, such as laptops, projectors and other smart devices somehow affect the old way of delivering lectures but do not ease the communication from the students perspective. Black boards and white boards are replaced by projector screens and electronic boards but all these new ways ease only one way of communication from lecturer to students and ignored the communication problems from students to lecturers.

Asking questions is another important activity of the classroom environment. Students have the right to ask questions for better understanding and to make things more clear. Usually teachers encourage students to ask question during lectures, but sometimes the students feel uncomfortable in raising a question during a lecture. It might be due to the personal human nature or physiological phenomena called ”Crowd Fear”. In this way most of the students hesitate to ask questions during a lecture. In other cases, some students avoid raising their hand during class as they feel shy or do not want to interrupt the whole class with their personal questions. This makes them unable to fully understand the lecture and causes them to lose the whole concept of class study. All these situations result in a big communication gap between the teacher and the student, which influences the whole learning process.

Modern technologies and trends contribute very little to the activity of writing down the notes. Some students use a printed copy of the slides instead of plain paper. In this way a student can keep a link between his own notes and topics. But still it is too much hassle for students to arrange printed slides before the lecture. Either slides are not available before the lecture or the students are unable to take time to make a printout prior the lecture.

How available systems addressed these issues

Different approaches are introduced with the help of new technology to tackle these common problems. The Ubiquitous Presenter provides an option to students to have a synchronized view of the slide on their tablets or web based clients. This application also supports a mechanism for result submission from student devices. The application provides a way for communication from the student's end but to use the Ubiquitous Presenter one has to use its own tool for presentation which requires some special plug ins for exporting current presentations to ubiquitous compatible formats which might be an overhead for the lecturers and those people who habitually use standard presentation tools.

The Lecturer's Assistant is another project for viewing and note sharing. The system allows the students to use their tablets to point out or draw on the slide which is viewable to the lecturer and the projector display. The application requires a very customized environmental settings for practical use of the system which is again an overhead for the lecturer.

The ActiveClass focuses on some of the goals which we set for our project. One of the main goals is to allow students to post questions related to slides during lecture. The lecturer can view these questions after switching to the ActiveClass application, In contrast to ActiveClass, Our application sends push a notification on his device message which notifies the lecturer if any student has posted a question. It also takes away the hassle of periodic lookup on the questions screen.The software also links each question with a specific slide number which makes more sense to the professor to address the right slide for that specific question.

What different NoteCarrier will make in presence of other applications

Most of these systems require self created tools for presentation or need very customized hardware and software settings for their application. To overcome all these existing issues we purposed a solution *"NoteCarrier"*, a classroom application for students and lecturers to intensify the concept of the whole lecture process. Instead of creating new presentation tools, expensive hardware or engaging the lecturer in a new set of software, the main focus of this system was to come up with a solution which is compatible with current presentation tools and silently integrate itself the inside current system.

NoteCarrier works as a background application on the lecturer's Mac book, which enables it to integrate with current presentation tools without the lecturer's explicit interaction. The application detects the current running presentation on the lecture machine and configures itself accordingly. The application also converts the lecturer mobile phone into very handy presentation remote control with a bundle of new features. In this way, we also take away the overhead of using additional remote control and make it easy for the lecturer to interact with less devices during the lecture.

NoteCarrier provides a bidirectional communication channel

Apart from the lecturer application, NoteCarrier also encourages students to actively participate in the lecture. Besides having synchronized and off-line slide view on their device, the application provides an opportunity to post questions anonymously for those students who are hesitant to ask during the lecture. NoteCarrie provides a vary stable platform for students and lecturers to improve overall lecture performance. Michelle Wilkerson and Simon

The initial prototype is developed for Mac OS [1] where as the presenter slides are prepared in Keynote [2] Application. A further enhancement of the project may enable it to handle other presentation tools e.g. PDF documents.

Main components of NoteCarrier application

The Application consists of four small development blocks. The first part of the development is a Mac OS based server application, which receives requests from students and lecturers on smart devices and responds accordingly. The second part of the NoteCarrier applications is an Iphone/Ipod (Partially supported for Ipad) based application which will be deployed on student smart devices and serve as a client for MAC OS server. The third component is also an IOS based application for lecturer's smart devices which provides a set of features for lecturers. The last component of this project consist of AppleScripts, these scripts serve as a communication channel between the Keynote application and our software.

[1] http://www.apple.com/osx/
[2] http://www.apple.com/iwork/keynote/

1.1 Research Goals

NoteCarrier provides a support to use Keynote as presentation tool

One of the primary focus of this thesis is to propose a solution which can be easily integrated into current systems. Plug and play will be one of the desirable attributes for this system, so one can embed the application into a current scenario effortlessly. Instead of introducing a new set of tools and software for presentation, the goal will be to develop a solution which can be used with existing presentation tools. Lecturers can use their current set of slides without the need to invest additional time to recreate these slides to make them compatible with this new approach.

Another major goal for the application is to aid the students in class participation and to ease the students activities in current classroom environment. The intention behind the thesis is to investigate current styles and traditions involved in the classroom environment, analyze those investigation results and propose some adoptive solutions. The outcome will be a small demo application that can be used on smart devices inside the classroom.

My research contributions

Another part of this research is to enhance the communication between the lecturer and the student. What are the hurdles that exist in current mechanisms? How they affects on the learning process? Student shyness, hesitation to ask questions and fear of being wrong are major causes that keep a student from raising questions in the class. We will also analyze these new communication techniques to judge how they improves the learning process.

We will try to find the answers to the following questions

1. Design
   (a) How the design of NoteCarrier application make it easy it is to integrate in the current environment?
   (b) Is the application providing a realistic solution for current classroom environment with available hardware and software resources ?
   (c) Is the system is easy to adopt for new users ?

2. For Lecturer
   (a) In which situations does the the system assist the lecturers?
   (b) How system helps lecturers to improve their lecture delivery process?
3. For Students
   (a) In which scenarios application is beneficial for students?
   (b) How system encourages those students who are hesitant in class participation?
   (c) In which situations does the system help student in understanding lecturer better?

We will try to find answers of these questions by testing it in real classroom environments.

The initial demonstration might not be a complete solution with all of the above mentioned attributes. But the idea will be implemented as an initial prototype to support the concept of the system. A spiral approach can be used at later stages to enhance the software.

The contribution this paper will make in this specific research area is to create a concept which minimizes the communication gap between lecturer and student, it also encourages the students to actively participate in class. Additionally we will evaluate this concept with the help of a software prototype with an iterative development process to extract an appropriate system design which is compatible with current systems and easy to adopt.

1.2 Chapter Overview

Chapter No. 2 : In Chapter 2—“Related work”, I discussed different research projects that are relevant to my work in following thesis. I also mentioned some relevant IOS based applications that are currently available for download and the functionalities currently available under these applications, At the start of the chapter I also discussed some key terms which are used in the same chapter like nomadic operations, Augmented Work Environment and Ubiquitous Computing. The chapter also contains a comparative analysis between all previous work and my current research.

Chapter No. 3 : Chapter is about 3—"Requirements", In this chapter, I summarize the hardware and software requirements of the system. This chapter also describes the software and hardware which is currently supported by the system.

Chapter No. 4 : 4—"Storyboards" is the third chapter of this paper which describes the first prototype of the application in the form of storyboards. In this chapter, the concept of the application is described in the form of different practical scenarios. For better understanding these scenarios are also prepared in the form of storyboards and presented to potential users for evaluation purposes. These storyboards cover both situations, one in a classroom scenario without the application and then I present the same situation with the presence of my software. This helps to understand the main idea behind this project.

Chapter No. 5 : In chapter number five, 5—"Paper Prototype", I design the paper prototype for the application according to requirements of the system. To evaluate the usability of the system, the paper prototype is presented to a target audience to analyze the user impression about the application. This chapter also include the results of the evaluation session and a list of suggestions proposed by the users.

Chapter No. 6 : In chapter 6—"Software Prototype" I propose the first software prototype of the application. This chapter covers the whole design and implementation details of the application. This chapter also includes the overall architecture of the system and also explains implementation of different components used in this application.

Chapter No. 7 : After software implementation of the system, I evaluate this small prototype in a small user study, In Chapter 7—"Evaluation" I explain the whole evaluation process in detail. The application is presented in an actual classroom environment for user evaluation, questionnaire and interviews are also conducted at the end of the session to collect user feedback. The results and feedback of this session are also described at the end of this chapter.

Chapter No. 8 : The last chapter of this paper is 8—"Summary and future work", in this chapter I conclude my whole research work in a short summary. I also suggest a set of features and research topics in future work section. I also propose suggestions to improve the current design of the system.

Chapter 2

Related work

> *" Life can only be understood backwards, but it must be lived forwards."*
>
> *— Soren Kierkegaard*

Numbers of research based projects and commercial software are available for presentation purposes. A large number of these projects or software only facilitate the presenter during presentation. Major parts of these projects act as a simple remote for presentation tools and provide slide switching functionality. Only a few projects will allow the the presenters to use more advanced features such as cursor pointer, pen controls by using Iphone/Ipad devices.

The significance of information sharing in today's era

In the same way, a simple search on iTunes will return hundreds of applications for file sharing between IOS devices. With increasing numbers of IOS devices, data sharing and data management becomes one of the most hot research areas for new or upcoming applications. ICloud [1] a service provided by apple, helps to share data between different devices and also provides advanced features like streaming audio or video data from one device to another according to user demand, it's one of the biggest examples of such research projects.

But most of these research based projects and commer-

[1] https://www.icloud.com/

cial software exclusively benefit the presenter during the presentation without considering the audience which are greater in number. A complete package is required which can serve the presenter and the audience together by providing an opportunity to the presenter to entertain the audience during presentation. For this purpose, a special tool is required which can benefit both the audience and the presenter. Such tool should consists of several options e.g. file sharing, notes writing etc. These options should be available in a single tool which can be utilized by audience and presenter at the same time.

Some of the applications that share similar concept are explained in the upcoming chapter. Before going into details, a short introduction is given about the concepts used in these applications.

2.1 Nomadic Applications

The current era is an era of technology. We are surrounded by the number of devices in our daily life. With more investment in this area, new devices are introduced into the market to fulfill our needs and requirements. These new devices might be enhanced versions of some old solutions or a new solution for some uncovered problem. Today, multiple devices are used by a single person in his/her daily routine. As the use of devices is increasing day by day, there is a need for proper infrastructure which helps to manage and share data among these devices. It is not an easy goal to achieve with heterogeneous Operation systems and hardware structures.

Nomadic application, a new concept of sharing information

Nomadic application is a concept which takes the communication one step ahead than typical device data sharing approach. Instead of only sharing files over the network, application itself can be shared with other devices over the network in standardized formats e.g. in the form of objects. This new concept will makes the applications more generic so they can be used on heterogeneous devices. Achieving the ability to make applications independent of device infrastructure, will take the us closer to ultimate goal of Ubiquitous computing environment in which single application can be integrated on different devices. New challenges need to be addressed during the development of applications which make these applications smart enough to execute according to available device.

2.2 Augmented Work Environment

An Augmented Work Environment is designed to build a specialized space equipped with special devices such as smart phones, laptops, networking devices and projectors. This augmented environment provides an infrastructure for the user to interact with another group of people for meetings, presentations and other group activities. The desired goal of an augmented environment is to make communication and information sharing easy and effortless. With the help of specialized devices and settings, the design of such an environment should be reliable and efficient enough to make it easy to use.

2.3 Ubiquitous Computing

Ubiquitous computing (abbreviated as *"Ubicomp"*) is an approach used in HCI (Human Computer Interaction) techniques to serve the user in silent way.

The main goal of this approach is to build an infrastructure with the help of software and hardware in a way that user does not feel presences of devices in their surroundings. The goal can be achieved by embedding tiny chips inside these devices. An infinite network of these interconnected devices will form smart network which has a capability to serve the human needs without the need of external interactions.

Weiser defines the Ubiquitous computing as physiological phenomena called "Compiling", which states that when humans achieved a certain level of awareness for something in which brain can process it with it's prior knowledge and somehow it becomes invisible to them. For instance so many people understand the meaning of some traffic signal board even without fully concentrating on what is written on it. Its due to their prior knowledge about that board stored in their mind and they even did not realize that they had a look on a board while driving in that street.

The idea behind this methodology is, to come up with a solution in which different software and hardware offer their services to assist the users in their routine work. In other words to give an impact of "Magic" in real world, where the users have a feeling that work is done by some invisible force with some magic. This way of computing eliminates the current way of showing warnings and alert messages where devices does not know how to avoid unfavorable situations and need human intention to proceed further.

Some pioneers of this field define the ubiquitous computing as

> *Ubiquitous computing is the method of enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user* – Mark Weiser 1993

Being invisible is the most desired characteristic of ubiquitous computing, the hardware should be intelligent enough to communicate with other available resources and do not require manual human interaction to perform their activities.

> *Ubiquitous computing, or calm technology, is a paradigm shift where technology becomes virtually invisible in our lives.* – Marcia Riley

Ubiquitous computing actually does not minimize the use of hardware or hide them from human eye rather its a concept to utilize the hardware and software in so natural way that humans does not feels their existence in their surroundings.This vision still needs lot of research to become a true reality, as currently the vision is only achieved at very small level but have enough potential to be expanded on large scale. The current implementation of Ubiquitous computing is limited within boundaries to single room or building, further research and work is needed to take the concept beyond such limits.

2.4 Smart Space

JENG explained that Smart space is another name for Augmented Work Environment. With the technology evolution, new hardware is faster, cheaper and smaller in size as compared to previous ones. This efficient change in hardware provides a great opportunity to build a smart space. Smart space provides an infrastructure for ubiquitous computing, where devices interact with each other, having a common interface, over high speed wireless networks. The main purpose of this space is support for ubiquitous computing, which provides platforms of hardware and software to ease our daily life.

2.5 Air Projector

Air Projector[2] is a small application developed by qrayon. The project allows the presenters to present any PDF documents and images from IOS device to any browser based device via wireless network. The application does not require any additional software installation on the client machine except a web browser.

Air projector allows presenter to project files over wireless network

This application allows the presenter to projects any PDF or image file from an IOS device to multiple client devices. The only requirement for presenting via air projector is that all client devices should be connected to the same wireless

[2] http://itunes.apple.com/us/app/air-projector-free/id411380678?mt=8

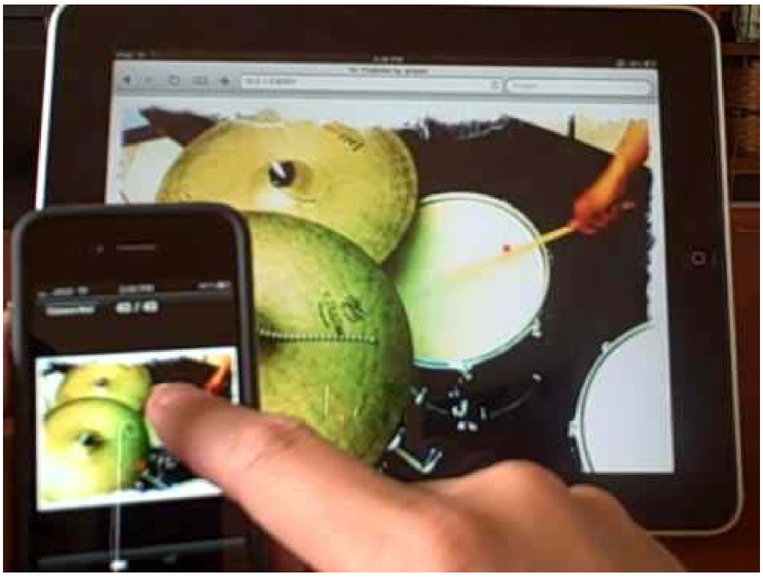


Figure 2.1: Air Projector : Using your finger to act as laser pointer (Source Project website)

network. Air projector will project a document on specific URL and port within that network. The document can be viewed by any standard browser by visiting the provided URL in client computer. The presenter can also use an IOS device as a laser pointer by holding his finger on specific point on the device. A small red pointer will be visible at that specific place on all client browsers. (see Figure 2.1)

The application have the same concept of sharing presentations over IOS devices as I suggested for my prototype. But application provides very limited feature as compared to NoteCarrier. Despite of view sharing NoteCarrier provide number of features to facilitate the lecturers and the students e.g. posting question, taking notes etc.

2.6 Idea Flight

Idea flight [3] is an application available for Ipad users. The application is developed on the idea of view sharing. The

[3] http://itunes.apple.com/us/app/idea-flight/id427659445?mt=8

application helps to share the idea with group of people over Ipads. Each Ipad will have the same view as on the presenters view (shown in the Figure 2.2), it helps to keep the audience on same page during presentation.

The application can be used in two different modes called Pilot and Passenger.

Application offers two different operational modes

Figure 2.2: Idea Flight : An application for ipads, for group meeting (Source Project website)

The presenter will run the application in *Pilot mode* to control the whole presentation. *Passenger mode* of the application is a read only view that will be used by the audience of the presentation. All passenger modes are synchronized with the main pilot mode, any change made on the pilot machine will be viewable on the passenger devices.

The only available control for the presenter is to switch between slides. This application provides a note taking facility in passenger mode. The application allows audience to save their notes related specific slide. These notes are stored inside the application in text format.

Except for this synchronized view application also provides an emphUnlock option, which allows the passengers to move back and forth freely.

The idea flight shares several goals with my prototype. As in Idea flight, NoteCarrier also provides two different operating modes, *Lecturer Mode* and *Student Mode* which acts similarly to *Pilot mode* and *Passenger mode* respectively. The application also assists the audience of the presentation to take and save their notes on the slide. In my prototype, the application also support the same idea of notes taking and allows students to write down their notes on the slides using their smart device. But in contrast to NoteCarrier, idea flight does not provide any mechanism to share the audience feedback with the presenter.

2.7 Ubiquitous Presenter

Tablets and browser based clients can communicate with Ubiquitous Presenter server

Ubiquitous Presenter [4] by Michelle Wilkerson and Simon is a project, especially build for classroom environment. It was first deployed in 2002[2]. The idea behind this project is to enhance the communication channel between lecturer and students. Ubiquitous presenter combines the electronic slides with an optional usage of electronic ink via tablet PCs. The ink feature can be used by lecturer or student for notes taking or writing down questions. In a later stage of development, to support non-tablet clients, the project provides web interface for PCs or other browser enabled devices (shown in Figure 2.3). Currently Ubiquitous presenter does not provide an ink feature for non-Tablet clients. The project provides synchronously and asynchronously views to all clients.

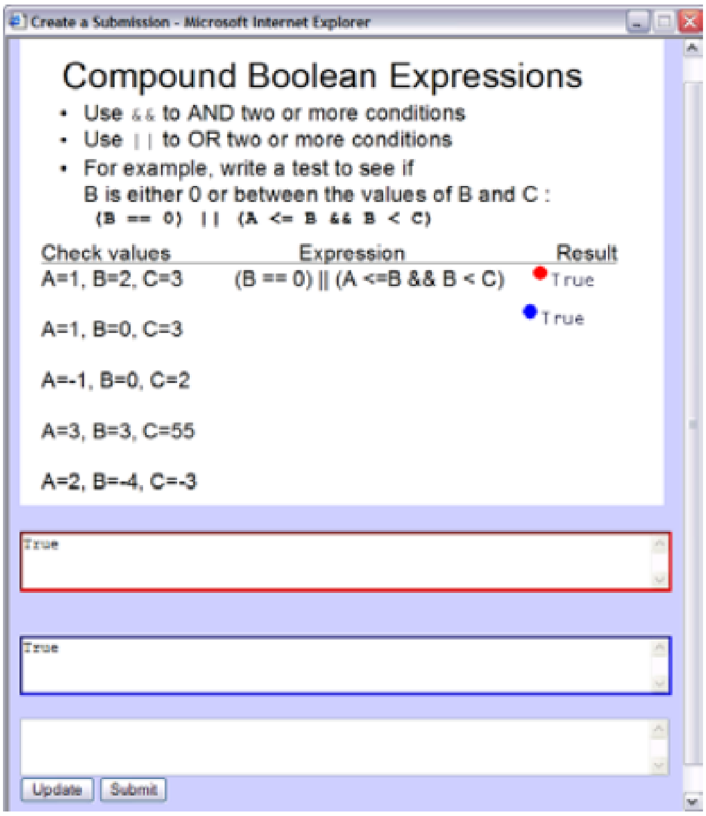
Digital ink is used to provide feedback in tablet based clients

The project allows tablet users to take notes using the electronic ink feature, these notes might be a solution for a problem presented in the slide or might be valuable comments that need to be shared with rest of the group. To endorse this idea, the software provides a two way sharing mechanisms between lecturer and students with the help of tablet devices. Students with tablets can submit their solution directly to the lecturer using ubiquitous presenter.

Ubiquitous presenter allows the audience to have two different modes to view slides. The students can either synchronize their view with the lecturer view or browse slides by their own. This feature helps the students learn better

[4] http://up.ucsd.edu/

when they have active control over slides and have an option to consult previous slides.



Compound Boolean Expressions

- Use && to AND two or more conditions
- Use || to OR two or more conditions
- For example, write a test to see if B is either 0 or between the values of B and C :

```
(B == 0) || (A <= B && B < C)
```

Check values	Expression	Result
A=1, B=2, C=3	(B == 0) || (A <=B && B < C)	● True
A=1, B=0, C=3		● True
A=-1, B=0, C=2		
A=3, B=3, C=55		
A=2, B=-4, C=-3		

True

True

Figure 2.3: Ubiquitous Presenter User Web Submission Screen taken from Michelle Wilkerson and Simon

Information is broad casted at the start of session

One of the limitations in early developments of ubiquitous presenter is related to data sharing between server and client devices. Currently the lecturer has to broadcast the slides at the start of the lecture, which means the later joiners of the class do not have any option to get the slides at a later stage. The other limitation with this process is with lecturer comments made during the lecture and aired to client devices. As the comments are transferred to clients immediately, so the late joiners will miss all those comments which are published before their joining.

Only focusing a solution for tablets will not be a realistic and cost efficient approach. To address this problem, the application provides a browser based view for non-tablet clients.

Ubiquitous presenter also supports web based architecture

Web based client server architecture (shown in Figure 2.4) is introduced to lessen these limitations. This new architecture not only supports multiple clients using web based technologies but also provides a storage feature to over come the limitations attached with broadcasting slides and airing notes in previous versions of this project. To facilitate the submission process from these users, ubiquitous presenter provides text based submission for non tablet clients.

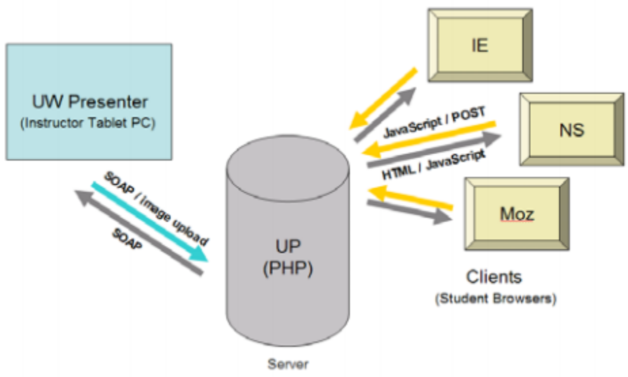


Figure 2.4: Ubiquitous Presenter: Presenter architecture to support web based clients taken from Michelle Wilkerson and Simon

Current network design of web based clients broadcast slides and ink through network polling mechanisms. Therefore current browser versions have to compete with other browser based clients for data access. This polling technique may lead to minor delays in fetching slides for non tablet users.

With the idea of a web based solution the project is developed and launched in 2005 (Ref. Michelle Wilkerson and Simon). Following are four major points which are focused for the design of the project.

1. Providing slides, in a browser supporting format to all web based clients.

2. Offering a synchronized view of slides with the ability to broadcast lecture ink.

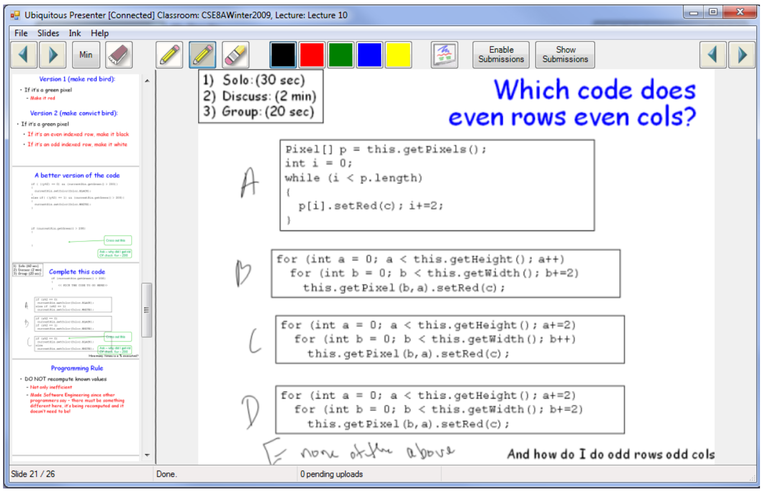
3. Giving an opportunity to web based clients to submit solutions via web browsers.

4. Allowing tablet based clients to use presenter in the conventional way.

All these goals are successfully achieved in the later development of the project. Which results in a more cost effective solution as compared to the previous tablet based approach.

The new design allows the lecturer to broadcast all slides at the start. These slides will be stored in the web server storage and can be accessed at later stage via web browsers as supported by transitional web based solutions. The lecturer can select ”Synchronize to Web” option to begin a lecture, which will mark that presentation as live. All synchronized web clients will have the same view as lecturer’s view. The lecturer may broadcast the comments and questions to all client’s views by using the ink feature in ubiquitous project. The application also provides an option to the

Web server support allows clients to fetch data on later stage



1) Solo: (30 sec)
2) Discuss: (2 min)
3) Group: (20 sec)

Which code does even rows even cols?

```
Pixel[] p = this.getPixels();
int i = 0;
while (i < p.length)
{
  p[i].setRed(c); i+=2;
}
```

```
for (int a = 0; a < this.getHeight(); a++)
  for (int b = 0; b < this.getWidth(); b+=2)
    this.getPixel(b,a).setRed(c);
```

```
for (int a = 0; a < this.getHeight(); a+=2)
  for (int b = 0; b < this.getWidth(); b++)
    this.getPixel(b,a).setRed(c);
```

```
for (int a = 0; a < this.getHeight(); a+=2)
  for (int b = 0; b < this.getWidth(); b+=2)
    this.getPixel(b,a).setRed(c);
```

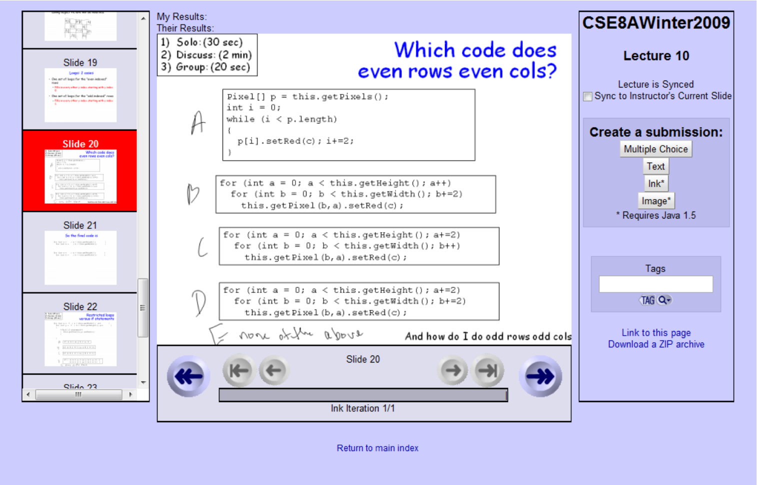
E none of the above

And how do I do odd rows odd cols

Figure 2.5: Ubiquitous Presenter: Lecturer View taken from Michelle Wilkerson and Simon

lecturer to enable and check student's submissions in the lecturer's view (Shown in Figure 2.5)

The web based client interface provides an easy to use method for the audience as well. Clients can not only view the slides in synchronized way (Shown in Figure 2.6) but also have an off-line (asynchronous) mode.



1) Solo: (30 sec)
2) Discuss: (2 min)
3) Group: (20 sec)

Which code does even rows even cols?

A

```
Pixel[] p = this.getPixels();
int i = 0;
while (i < p.length)
{
  p[i].setRed(c); i+=2;
}
```

B

```
for (int a = 0; a < this.getHeight(); a++)
  for (int b = 0; b < this.getWidth(); b+=2)
    this.getPixel(b,a).setRed(c);
```

C

```
for (int a = 0; a < this.getHeight(); a+=2)
  for (int b = 0; b < this.getWidth(); b++)
    this.getPixel(b,a).setRed(c);
```

D

```
for (int a = 0; a < this.getHeight(); a+=2)
  for (int b = 0; b < this.getWidth(); b+=2)
    this.getPixel(b,a).setRed(c);
```

E none of the above

And how do I do odd rows odd cols

Figure 2.6: Ubiquitous Presenter: Synchronized view for web clients taken from Michelle Wilkerson and Simon

Synchronized and asynchronous slide view

The students can switch between these two views according to their needs. Students may have the same view as the lecturer by choosing the option synchronized or shift to asynchronous view to look back other slides. With the involvement of a web server, lecturer comments are stored on the server storage. This new ability allows students to retrieve those comments from server which are made at the time when they were unsynchronized with the server.

Different new features are introduced to the system with the passage of time. The latest version of project (3.3.1) was released in June 2006. The project is available as a free download for academic use(Ref. Michelle Wilkerson and Simon).

2.8 AirTransNote

AirTransNote by Motoki Miura and Tanaka provides an augmented environment for classrooms. The application provides a base structure for notes sharing between the lecturer and students. The core concept behind the application is to enhance the communication and collaboration in the classroom in the same traditional way. AirTransNote does not influence the current way of note taking rather merge the solution in the same manner. This technique makes the application more adoptive and easy to use for the students. AirTransNote enhance the communication by sharing the student notes with the lecturer in real time.

Electronic pen and PDA's are used to store the students notes

With the help of a PDA and electronic pen, AirTransNote will send all the notes from regular paper to the server (See in Figure 2.7) . To accomplish this goal the normal pen is replaced by an electronic pen. This electronic pen is a combination of normal pen, set of sensors and IrDA relay. The electronic pen with the help of other sensors which is attached with the paper, saves all information about the strokes made by the student while writing down the notes. All the information collected by these sensors are stored on a student's PDA device. The compound structure of these sensors, PDA and electronic pen collects and transmits the information to the server.

To use AirTransNote, the student has to press a hot key on the PDA device which will activate the PDA information collection module. The tip of the electronic pen generates ultra sonic waves that are analyzed by the sensors for recognizing the pen location and its movements. All this collected information is then transmitted to the server through the PDA device. The newer version of AirTransNotes, uses one single electronic pen which also acts as a transmitter. These new devices are developed by NTT Comware Corp [Source][5]

Notes can be shared with lecturer over wireless network

The second major part of this application is a sever or manager. All information collected from students notes are first transfered to their PDA's. These PDAs then transmit that

[5] http://up.ucsd.edu/

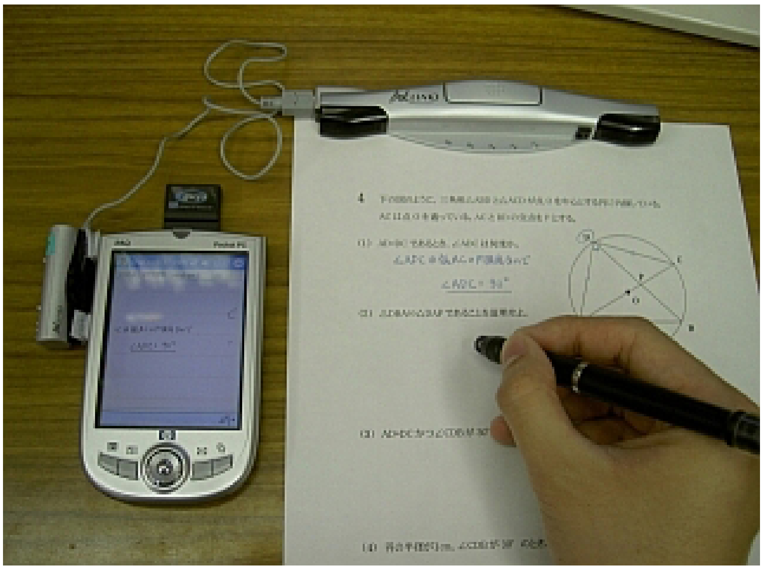


Figure 2.7: AirTransNote: An old transmitter system, with electrnoic pen, PDA and sensor taken from Motoki Miura and Tanaka

information to main server. Although all information is immediately sent to the server, but it still saves a copy on PDA device as a backup.

The AirTransNote server provides a real time view of the student notes on a browser view. This view helps teacher to keep track of student progress with time stamp information.(Shown in Figure 2.8)

AirTransNote browser gives an option to the teacher to select any student note and put it on a scanned empty answer sheet. This will gives a same look as a the student has on his printed paper sheet. The browser also allows the teacher to adjust zoom and focus attributes for better visualization and understanding. For better look, the browser performs all transactions like zoom or switching with smooth animations. AirTransNote can also be used for sharing one student's notes with others. The teacher can pick any solution, and explain it to others or make corrections in it if required.

AirTransNote saves all information about the strokes made during notes writing. By using time stamp information the

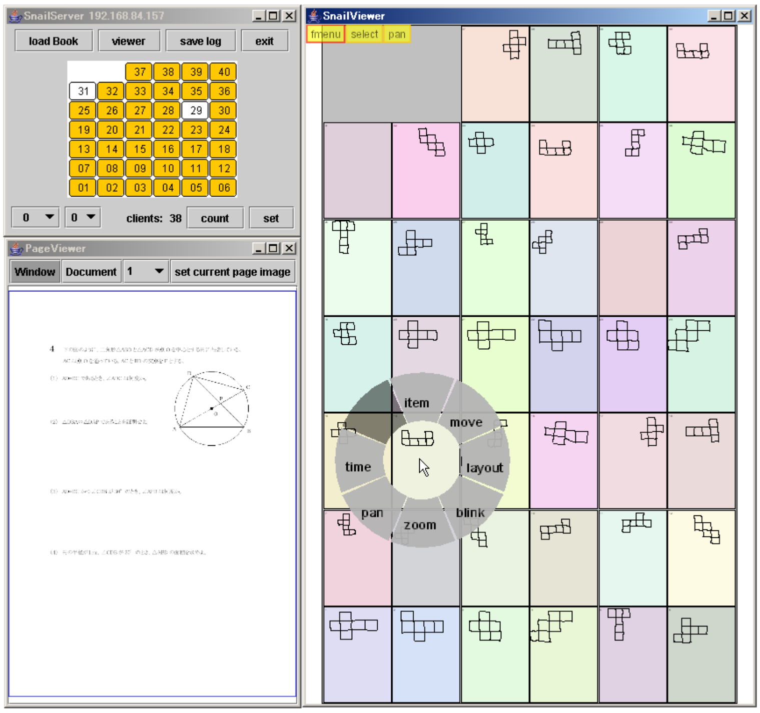


Figure 2.8: AirTransNote: Server view of AirTransNote taken from Motoki Miura and Tanaka

application can also reproduce the sequence of notes. This might be helpful for the teacher to understand the whole progress of the student and areas where the students have problems. The replay function will reproduce the solution step by step which can play a vital role for the teacher to discuss the solution in more detail.

Another most exciting feature of AirTransNote is the recognition algorithm. This algorithm provides support for automatic result recognition. This module will be a suitable solution to formulate the results of multiple choice questionnaires.

Recondition algorithm to recognize correct answers on multiple choice questions

AirTransNote provides a complete solution for the classroom environment. It not only helps teachers to keep a close look on student progress but also provides a better feedback mechanism for students as well. Teachers can give better explanations on student's created solution rather just presenting their own solutions.

The AirTransNote have many similarities with the NoteCarrier application. But it requires very specialized hardware e.g. digital pen and movement tracking sensors which are not as common as smart phones. One of the big advantages NoteCarrier has over other solutions that it provides its services using those tools and technologies which are already in practice rather limiting itself to only highly equipped environments. Another main difference in both application is for posting feedback and questions. The concept behind AirTransNote is to propose a solution which allows the lecturer to view the student's solution directly on his machine but NoteCarrier's idea for feedback mainly focus those students who are hesitant to participate in the class and encourages them by giving them an option to post comments anonymously.

2.9 The Lecturer's Assistant

The Lecturer's Assistant by Buckalew and Porter is developed with the idea of enhancing the communication inside the classroom. The system is proposed to provide a platform to support two way communication between students and the lecturer. The system has two different operating modes, it provides an option to the lecturer to edit and create slides and a set of features to control the flow of the presentation. The second mode of the application is implemented to enable the students to submit their feed back during the lecture.

Since the system is developed to facilitate both the lecturers and students, to accomplish this goal the application assists the lecturer by providing a set of features to prepare and present the lecture. The lecturer can not only create and edit slides by using this tool but also write a script to manage the sequence of slides during presentation. Displaying multiple slides at a time, drawing on slides and options to save edited slides during the lecture are major features of the project.

Student can post their feedback using tablet devices

In order to improve two way communication, the application also allows students to participate in the class using

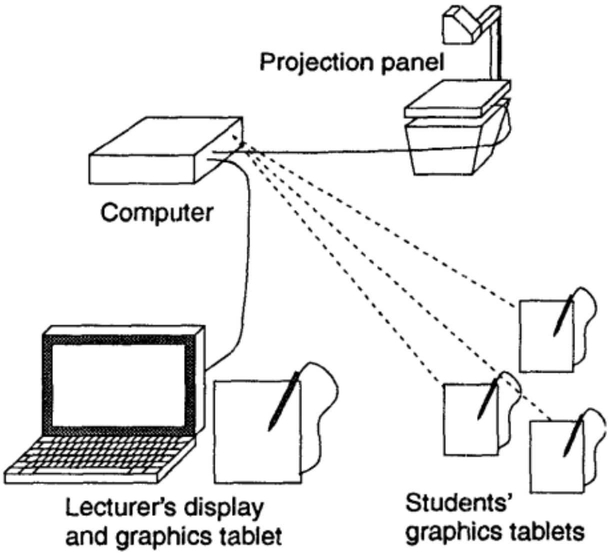


Figure 2.9: The Lecturer's Assistant: Image shows a System Design of Project Taken From Buckalew and Porter

their tablet devices. Students can also contribute to the lecture by posting their feedback and suggestions during lecture. Using their tablets, students can point to a specific point on the slide or draw or write on the slide to convey their thoughts to the lecturer.

Student tablets and computer systems are connected with each other on a wireless connection (System deign of the application is shown in Figure 2.9). To record the student's pen moment, infrared technology is used to transmit data between the tablet and main the computer. With each pen movement the infrared light beam is transmitted to the computer system which calculates its position on the slide. To draw on the slide, the student has to touch the pen tip to the surface of the tablet, now every movement made by the pen is transmitted to the computer which will calculate the pen position and draw a trail on the slide accordingly.

The lecturer has a complete view of the slides on the monitor screen, the lecturer can start the lecture and move the slide back and forth by using the control panel (See in Fig-

Application comes with a build in presentation tool ure 2.10). The lecturer can also use his tablet and pen device to draw on the slides to explain the its contents. Another feature supported by the application is showing multiple slides at a same time, this would help the lecturer to have a over view of upcoming slides on the same window.

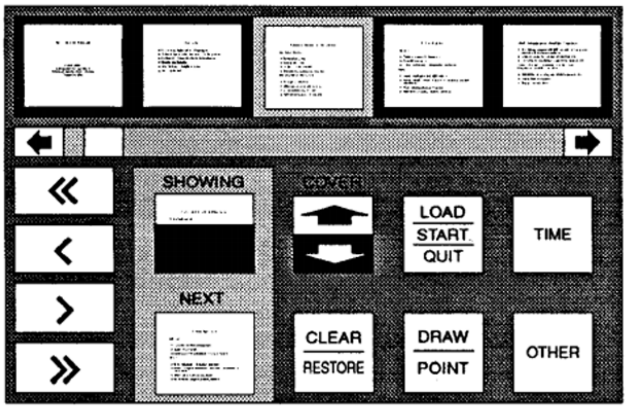


Figure 2.10: The Lecturer's Assistant: Lecturer Control view over the monitor screen (Ref: Buckalew and Porter)

The concept behind *The Lecturer's Assistant* is quite similar to the *NoteCarrier* application. The concept of promoting two way communication is shared in both of the applications. In contrast to *NoteCarrier*, *The Lecturer's Assistant* provides a complete presentation environment to the lecturer, in which they can present, edit ,add and save presentations. The project also assists in controlling the flow of the slides by writing specific scripts. Since specialized presentations tools like Keynotes, Microsoft PowerPoint already facilitate the lecturers in these activities, so In *NoteCarrier* it is more important to reuse these applications instead of creating a replica of available software.

Application provides some same feature but with different ideology

From a student's perspective, *NoteCarrier* also allows the students to post their feedback to the lecture, but the main theme for providing this facility to students is totally different in both applications, In *The Lecturer's Assistant* students can draw on slides using their tablets and pen devices from their seat, it helps them to point out the specific point over the slide or share their point of view without going to the lecturer's place. But In *NoteCarrier* the idea in introducing such a feature is to support those students who are shy about raising their voice in front of other class mates. This can not be achieved by adopting *The Lecturer's Assistant* approach.

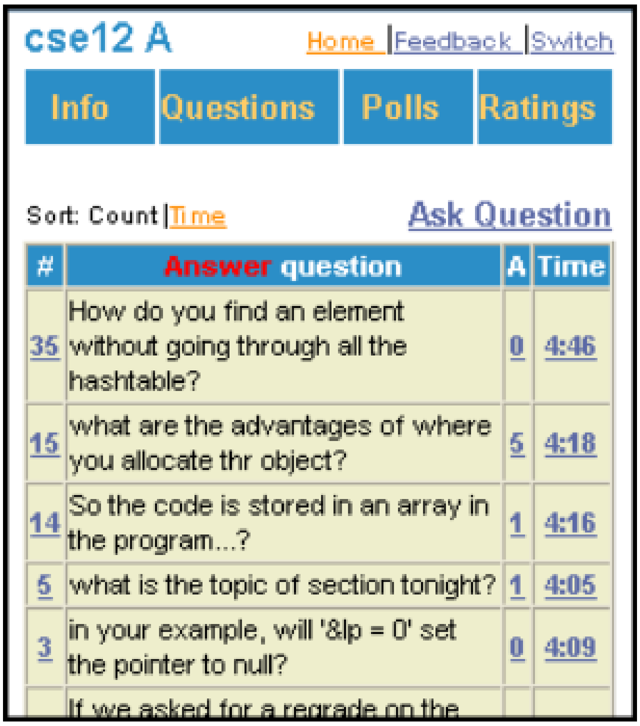
2.10 Active Class

Active class by William G. Griswold and Truong is a project to support student activities inside the classroom. The project is developed to assist the student in asking questions, voting on other students questions and giving feedback over the lecture. In the application by using personal wireless devices, students can post questions to the lecturer or provide their feedback related to the presentation. Students not only post their own questions but also have an option to vote on other student's questions in case they also want to ask the same question. This will also help lecturer to sort the question by their votes and address those question first that are asked by many students (Shown in Figure. 2.11)

Active class is a project which focuses the psychological issue of human nature. With the increase of class strength many students feel hesitant to ask questions during a lecture. William G. Griswold and Truong address number of issues that might be reason for such student behavior. The project defines a way to encourage these students to participate in the lecture by posting question anonymously. Using *Active class* a student can post questions without showing their identity via their wireless mobile device. These questions are forwarded to the lecturer anonymously, so students who are afraid to raise their voice during lecture can now participate more actively in the lecture.

Active class and NoteCarrier tackled the same issue with different user interaction techniques

The intention behind *Active class* is very similar to the student mode in *NoteCarrier*. Supporting students for active participation is quite identical feature in both project. But the same issue is handled in a very different way in *NoteCarrier*. *Active class* only supports the questioning mecha-



#	Answer question	A	Time
35	How do you find an element without going through all the hashtable?	0	4:46
15	what are the advantages of where you allocate thr object?	5	4:18
14	So the code is stored in an array in the program...?	1	4:16
5	what is the topic of section tonight?	1	4:05
3	in your example, will '&lp = 0' set the pointer to null?	0	4:09
	If we asked for a regrade on the		

Figure 2.11: Active class : A sorted list of posted questions taken from William G. Griswold and Truong

nism through wireless devices, this means the lecturer has to interact with separate devices to control the flow of the presentation where as in my prototype the lecturer can not only use the mobile device as the presentation remote but also have a look over a slide on the mobile device. Linking questions with specific slide also make it easy for lecturer to understand the question and its context in more better way. Lecturer's interaction with question list is also implemented in different style in both project, In *Active class* lecturer need to consult the questions list again and again to check the new questions, In contrast to this approach *NoteCarrier*, facilitate the lecture with push notification technique in which as soon as the new question is posted lecturer get an updated count on his mobile device.

Posting question is one of the feature supported by *NoteCarrier*, furthermore my software prototype provide number of other features that are not available in *Active class*. Writing notes, saving modified slides and slide browsing are the list of features additionally supported by *NoteCarrier*

2.11 Summary

In this chapter, I presented number of project which somehow share the same ideology with *NoteCarrier*, but still there is enough room available for improvements. In the presence of these applications, today's classroom environment has not extensively changed in many years due to the specialized and expensive hardware involved in these systems. Another reason that might be an issue in adopting such systems is the complex software systems and interaction techniques used in these systems. These factors make the application hard to use and make it difficult for lecturers and students to interact with these applications.

Combining different needed features in one application and improving the interaction techniques in a newly defined project will be the basic idea for my project. Instead of introducing new hardware systems and implementing complex new software, I would like to come up with a solution which is easy to use with an available set of hardware and utilize those software systems which are already in exercise.

Chapter 3

Requirements

> *" The reason most people never reach their goals is that they don't define them, learn about them, or even seriously consider them as believable or achievable. Winners can tell you where they are going, what they plan to do along the way, and who will."*
>
> *— Denis Waitley*

Before starting work on any prototype of the project, the requirements must be clear for that project. It helps to stay focused on the goal. It also makes it easy to define a set of steps to achieve that goal. The following chapter will cover the requirements for the initial demo project. This section also describes the assumptions and goals we set for this project.

The requirements are gathered and listed by doing the proper research study in the specific area, observations are made in a real classroom environment, collecting ideas from other students and brainstorming with friends and colleagues. In the following sections, only those requirements will be considered which are selected for the project and highly rated as compared to other requirements.

To support the idea of this project, a small prototype is developed for specific hardware and software. To keep the project more adaptive, we tried to keep the use of special hardware as minimal as possible. It would help us to integrate our system more quickly and easily into the current environment without integration of specialized hardware. The following section also describes all the needed software, hardware, settings and configuration involved in the system.

3.1 Hardware Requirements

NoteCarrier server application is implemented using Cocoa programing

Although the goal for this project is to support all existing tools and technologies used in presentations. But the initial demonstration of the application works with specific hardware. This limited hardware support constraint is due to the iterative development approach we adopt for development. In iterative development, a software prototype is developed at the first stage to analyze and test the project concept. Once the software proves that it has the ability to achieve it predefined target, the development team tries to make it more mature by adding new features and to support other available hardware.

The current development is a client server based application. The server is developed for Mac machines[1] where as the client side application provides full support for iPhone [2] / iPod[3] and partially support by iPad [4] .

3.2 Software Requirements

IOS 5.0 or above is required for client IOS devices

Limited hardware also puts a constraint on software that we can use for the initial application. To enable support for mac machines with Mac OS [5] ,the server is developed in Cocoa [6] , a programing language used to develop applica-

[1] http://www.apple.com/mac/

[2] http://www.apple.com/iphone/

[3] http://www.apple.com/ipod/

[4] http://www.apple.com/ipad/

[5] http://www.apple.com/osx/

[6] https://developer.apple.com/technologies/mac/cocoa.html

tions for mac machine. Apple smart devices are available with different versions of operating systems known as IOS [7] . The current client side application initially supports all IOS with version 5.0 or above.

Apple keynote [8] is used as a presentation tool to prepare slides for presentation/lectures. Although the current system does not provide any support for Microsoft Technologies[9] but test shows that Microsoft Power Point [10] presentation can also be imported in keynotes to use NoteCarrier application.

3.3 Application Requirements

In the following sections, we will discuss all functional and non functional requirements we want to achieve in the current version and upcoming versions of the NoteCarrier application. Since project development is following the iterative approach, this requirements list acts as an open list of requirements. By open list, we mean the requirements list can be changed by adding new tasks, removing any old requirement or changing any requirements according to user needs.

3.3.1 Non Functional Requirements

Non-Functional requirements can be defined as

A non-functional requirement is a statement of how a system must behave, it is a constraint upon the systems behavior. Source[11]

Non-Functional requirements help us while designing the project to make the application more user friendly and easy

[7] https://developer.apple.com/technologies/ios/
[8] http://www.apple.com/iwork/keynote/
[9] http://www.microsoft.com/en-us/default.aspx
[10] http://office.microsoft.com/en-us/powerpoint/
[11] http://lessons-from-history.com/node/83

to use. "Showing notification message after saving data" or "Ask confirmation before deleting data" are some common examples of non functional requirements.

Adaptability

No need to recreate the presentations, NoteCarrier supports standard presentation formats

One of the main goals of this project is to make it compatible with current presentation tools. Although very few such applications are available in the market, most of them requires to use their application as the presentation tool as well. These applications are not compatible with existing presentation tools. Either one has to import existing slides in that software or create a new one using a specific application. The target we want to achieve with NoteCarrier is to use it with any existing presentation tool without importing or converting existing presentations. This feature will make application easier to use for the lecturers and minimize the time to adopt this new solution for their presentations. This will make application more adoptive for the presenter as they can continue to work with those tools which they were using before.

Plug And Play

Easy to integrate with current system

Making the application compatible with the existing presentation tools also makes it more plug and play than other available applications. The only requirement for the presenter to use Note carrier, is to run the server application as a background process. The presenter will have full control over the application to switch it off at any stage of presentation. Since presenters are already using their old presentation tools for slides, no additional time will be required to reset to the old way. NoteCarrier also makes it easier for those lecturers who want to use their old presentation slides with the new system. Most lecturers use the same content many times in future classes.

3.3.2 Functional Requirements

Functional requirements are those features and functions that software addresses. These requirements are the main feature of the software. A formal definition for functional requirements can be explained as

A requirement that specifies a function that a system or component must be able to perform." Source[12]

"Saving Data" or "Printing a document on print button" are a few examples of functional requirements, In the following section we will describe a very high level of functional requirements of NoteCarrier. The complete list of functional requirements will be discussed in upcoming chapters.

NoteCarrier provides two different modes for the student and the lecture. In the following section we will explain these modes in more detail.

Lecturer Mode

NoteCarrier provides a complete secure view for the presenter. The presenter can only access this view after providing valid credentials. This mode will allow the presenter to use a smart device to control the flow of presentation. The big advantage of using NoteCarrier as a presentation remote over other traditional remotes is a live view of a slide on a smart device. The presenter can see the same slide on the smart device as on the screen and get a quick view of the content of the current slide. The NoteCarrier presenter view also notifies the presenter in case any question is raised by the student. This notification might be in form the of a question or a slide that is marked as unclear by the student. The presenter can view questions related to the current slide or a complete list of questions can be viewed under the lecture summary section.

[12] http://lessons-from-history.com/node/83

Student Mode

Since one of the goals for NoteCarrier to support the student/audience of the lecture, a set of features are available under the student mode to assist the students during the lecture. To avoid any unauthorized access to lecture material students can only access the view after providing valid login information. NoteCarrier provides a facility to students to view the slides on their device either in synchronized or off-line mode. A high level application overview is shown in Figure 3.1). Students can also post questions related to the slides or mark them unclear if the content of the slide is not understandable. We will discuss all these functional requirement in more detail in coming chapters.

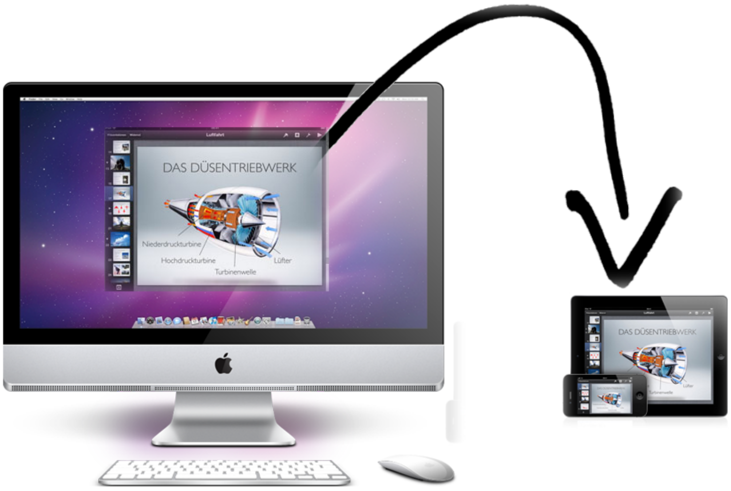


Figure 3.1: NoteCarrier : high level application overview

3.4 Iterative User-Centered Design

The software design patterns help us manage the project in a systematic way. To come up with a complete perfect solution in the very first attempt is nearly impossible to achieve. In contrast to the water fall development model in which all the development is completed before the test phase, the iterative approach is much more in practice these days. Instead of developing the whole project in one attempt, the whole development is broken down into different small projects. Each sub part of the project is treated as a complete project and follows a complete development life cycle. As a result each sub project passes through the analysis phase at the end of the iteration. This mid development analysis helps us to develop a more refined solution as compared to previous approaches.

Design, Implement and Analysis in each iteration

Nielsen [1993] describes the *Iterative User interface Design* as an iterative approach to overcome the drawbacks of the water fall development model. The DIA (Design, Implementation and Analysis) shown in Figure approach divides the development of each small chunk of the project into three phases i.e. design, implementation and analysis (Shown in Figure 3.2) . This division helps developers to get familiar with the issues at very early stage of the development. In each design phase, development team finalize the plan for the next iteration by incorporating the findings of previous analysis phase. The next phase is implementation in which development of the project carried out on the basis of prior design phase. The last phase of each cycle is analysis phase, in which project is evaluated using different techniques. User studies, surveys , post interviews and other methods can be used to record user's experience. These evaluation techniques are very useful to figure out the possible issues and problems that users faced during evaluation process. All these findings act as a feedback for next upcoming design phase, in which developer proposed a new design to overcome these known issues and reevaluate them in next analysis phase. This iterative approach provides a continuous feedback to the development team and make them able to address these issues and problems during the development process.

DIA approach is used to implement NoteCarrier

The DIA approach has been used in the following thesis as a software design pattern. Single feature is introduced with different solutions during development. These different solutions are presented to users during the analysis phase. The results of this phase help us to identify the best suited solution according to user ratings. Surveys and personal interviews are conducted during the development to maintain the quality of the project. Software is also presented

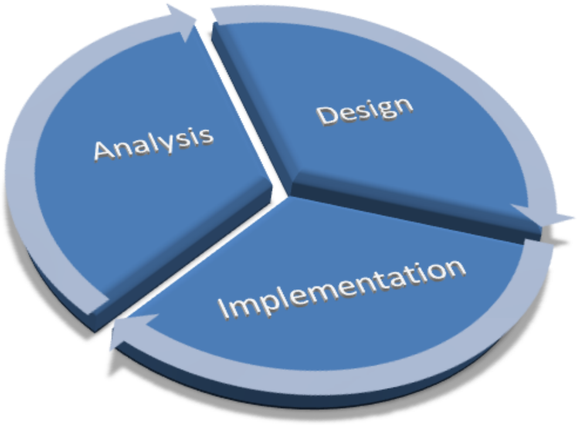


Figure 3.2: DIA Cycle : Design , Implement and Analysis

into a real class room environment to get the student's impression about the project.

Chapter 4

Storyboards

> *" Design is a funny word. Some people think design means how it looks. But of course, if you dig deeper, it's really how it works."*
>
> *— Steve Jobs*

storyboards an attractive way to illustrate your ideas

The storyboard is an attractive technique to describe the idea behind the system. It is a simple technique to picture the design of the system by simple sketches and drawings. This technique is widely used in different industries to depict the early design of a project.

storyboards as first prototype

In the field of software development, storyboards are useful tools to analyze systems before starting the coding phase. Application functionalities are described and evaluated using hand made designs which help to describe the concept behind that application. It not only saves time by not doing actual implementation at an early stage but is also useful to collect information about system usability.

Storyboards are useful to evaluate the application at a very early stage, It provides a very high level overview for the potential users of the system. It allows us to gather more feedback from the user by applying quick changes to system design by new sketching.

In the following chapter, we will look at the first design of the application on storyboards. The purpose of these storyboards is to demonstrate the idea of the project to target audiences and collect some early feedback about the project. It would also be useful in the design process for the better design of the system.

4.1 Application Scenario's

In order to gain the user point of view at an early stage of the application design, different storyboards were created according to features offered by the application. These storyboards covered different scenarios and situations in the real classroom environment. The storyboards were divided into two categories, one type of storyboards covered different features offered in the student mode where as the other type of storyboards were made to analyze the lecturer mode. Each situation was sketched with and without the presence of the proposed application.

4.1.1 Student Mode

storyboards were created to elaborate student mode of application

In the following section, we describe the storyboards for the student section. The storyboards were created to show both situations (See Figures 4.1 and 4.2), once when students are attending class without the NoteCarrier application and the other when they are equipped with this application.

Posting Questions or marking slides Unclear Without NoteCarrier

Figure 4.1 illustrate a situation in which John is one of shy students in a class and feels hesitant to ask questions during the lecture. John thinks that raising his voice may show his lack of knowledge about the topic, which he definitely doesn't want to show to the lecturer and the other students. Due to his shyness and lack of confidence, in many situations John decides not to ask a question and ignores those things which are unclear to him.

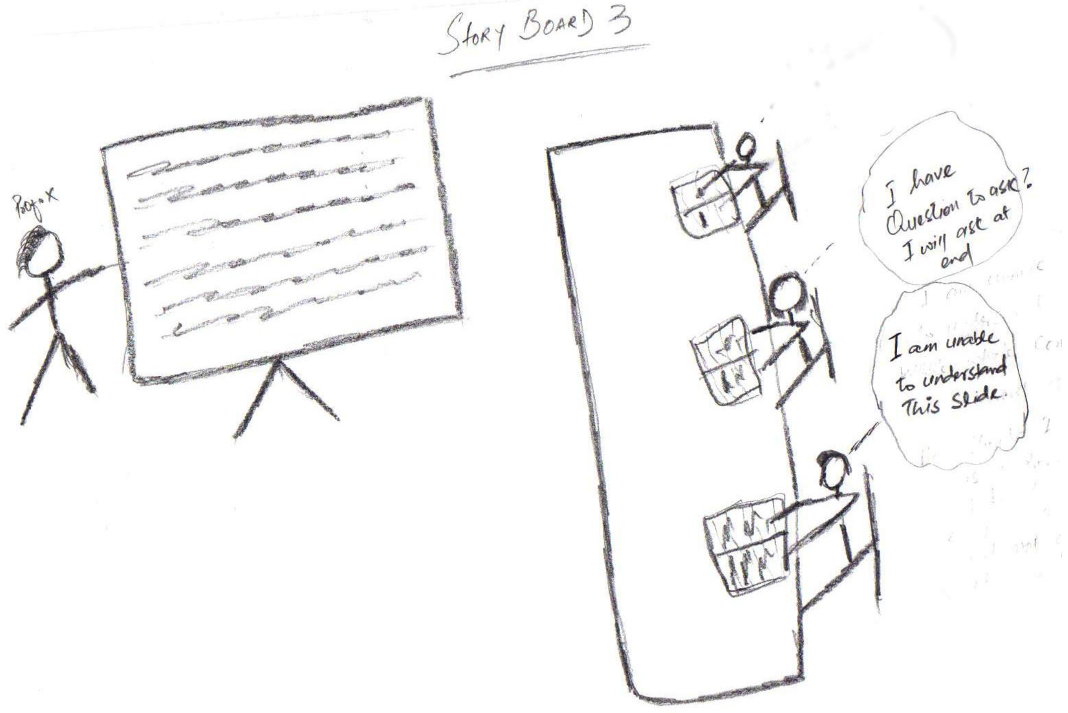

Story Board 3
Prof. X
I have Question to ask? I will ask at end
I am unable to understand This Slide.

Figure 4.1: Storyboard : The storyboards illustrate a situation of classroom where students are unable to understand the presented contents. Students are hesitant to ask question during lecture and no one is confident enough to raise his hand to point out that slide material is not clear to them

Today John has to attend a lecture on scrum and kanban software development models. Its one of his favorites topics and he wants to attend this lecture with his full concentration. He enters the class and chooses one of the last benches in the class as he feels too shy to sit in the front rows and does not want to have so much eye contact with the professor during the lecture.

Prof. S enters the class and prepares himself for today's lecture. He always delivers a lecture with the help of lecture slides so students can also have a view of the slide contents on a projector screen. Prof. S starts his lecture by explaining the content of the lecture.

At the very starting point John does not properly understand the concept of sprint involved in these development models. Due to his shyness John is hesitant to interrupt the class for his question. He thinks that may be on the other slides Prof. S will re explain the same concept so he will be able to catch up with rest of the class at that point. To escape from this situation he decides not to ask the professor to explain the concept of sprint which is unclear to him.

Due to lecture the sequence, all of the next slides are some how relevant to that concept which John unable to understand. Now it is becoming more difficult for him to understand the rest of lecture since he missed the main point at beginning of the lecture. Now John is totally lost and unable to understand anything in the lecture. John is feeling very frustrated because it is hard for him to understand anything taught by Prof.S

Posting Questions or marking slides Unclear With NoteCarrier

John enters the class with his iPhone device and as per his routine selects one of the last bench of the class to sit. John downloads all the slides from the lecturer′s computer to his smart device using the NoteCarrier application. During the lecture John is unable to understand the concept of sprint on the very first slide of the lecture. Being a shy student John is hesitant to ask questions of the professor but he does not want to move further without understanding the sprint concept. With his Iphone device showing the same slide on its screen John decided to ask question using the NoteCarrier application. He selects the post a question option from the menu and writes ″What is Sprint, Can you explain it ?″ and posts that question using NoteCarrier. Some other students were also unable to understand the whole content of the slide, they also want the lecturer to repeat that slide so they choose the ″mark unclear″ option from the main menu of the application to notify the lecturer that the current slide is not clear to them.

As soon as John posted the question, Prof.S receives a notification that 1 student posted a question related to the current slide. Prof. S also notices that number of other students are unable to understand the slide. To make the slide clear,

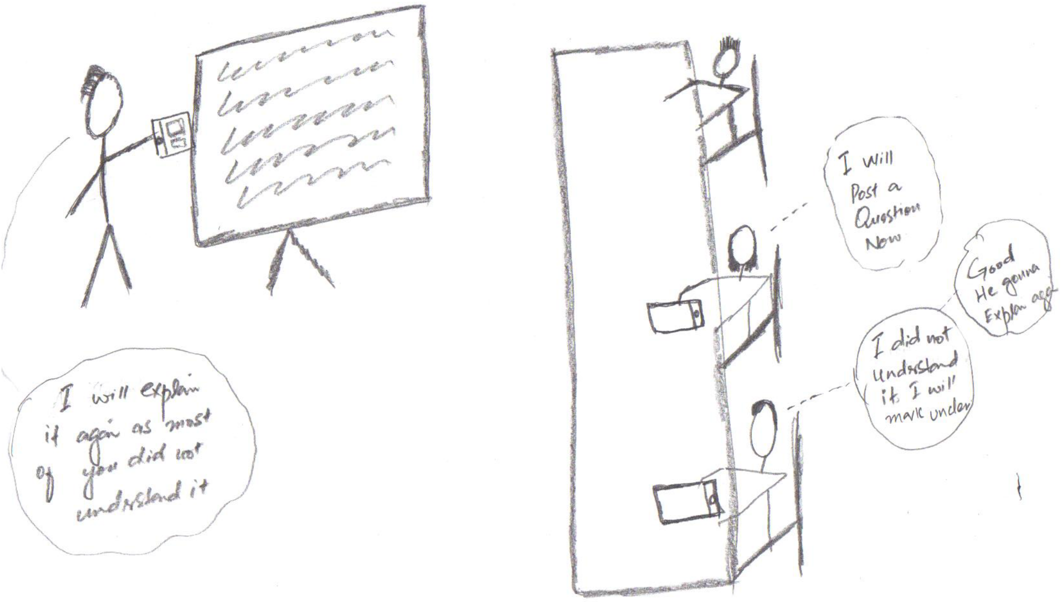

I will explain it again as most of you did not understand it
I will Post a Question Now
Good He gonna Explain aga
I did not understand it, I will mark uncle

Figure 4.2: Storyboard : The storyboards illustrate a situation of the classroom where students are unable to understand the presented contents. Student choose to post questions use NoteCarrier application. As soon as the student post the questions lecturer got an notification on his device and decided to answer the questions and repeat the slides to clear the content.

Prof. S checks the question and says,"One of you asks me to define the sprint concept, since its one of the main concepts for the whole lecture. So before moving forward i would like to rephrase it for those who were unable to understand it for the first time."

John is very pleased as Prof. S repeats the idea behind those concepts that he was unable to understand the first time. Now he is confident that he can fully understand the lecture and will post more questions if anything is unclear to him at any stage. Since NotCarrier posts questions anonymously, John does not feel any hesitation in posting more questions.

John is very comfortable with today's lecture as it was one of his favorite topics and he was able to resolve the confusion he had about these topics in his mind before the lecture.

Slide-view and taking Notes

Alex is a student of Informatics at the university. He is one of those students who attend every class with all his concentration and tries to gain as much knowledge as he can during the lecture. Since most of lecturers use electronic slides for the lectures, so its Alex's routine to bring a set of printed lecture slides before the class starts. He thinks in this way he can concentrate more on the lecturer's talk and also be able to save a lot of time by not coping the slide's content on his note book. By following this approach he is only required to just write down his own notes on the specific printed slide. Alex feels more comfortable with this way of note taking mechanism, since it also improves his readability while reviewing the lecture at a later stage.

Without NoteCarrier

Today Alex was a little late to reach the university. He could not manage to take a print out for the today's lecture before class started. During the lecture, he is really confused while taking notes since he has to first copy the contents of the slides and then writes down his own notes. By doing this he is totally unable to concentrate on the lecturer's talk and misses so many important points in the lecture. After half of the lecture he decides to write down only his own notes so he will rewrite them on the lecture printouts after the class. This will help him to pay more attention on the lecturer's talk.

After the class, Alex tries to copy the notes from his notebook to the lecture slides, but he is unable to create any link between his notes and the lecture slides, all of his notes are not making any sense to him at this point. Although he attended the lecture but he is unable to gain anything from it.

With NoteCarrier

Alex enters in the class with his iPad device, he forgot to bring the print notes for todays lecture. He opens the NoteCarrier application on his iPad, connects his device to the class network and downloads all the slides on his device using the application. He selects the off-line view mode of the lecture where he can see the complete set of slides. Now its easy for him to take notes directly on the slides without copying the same slide content. With NoteCarrier he is not worried about missing the lecture since he saved plenty of time by not writing all slides by hand. He saved all his notes on his device, so he can review them afterwards.

4.1.2 Lecturer Mode

The lecturer mode of the application was also presented to the audience with the help of storyboards. A set of storyboards are created to illustrate both the situations to users (See Figures 4.3 and 4.4)

A set of storyboards were drawn for lecturer mode

Prof.X is a lecturer at the university. He is delivering a lecture in the university for the last 4 years on a subject of his expertise. Since every semester he delivers the same lectures so he has prepared a set of slides for his lectures in his mac machine using Keynotes. He has used the same slides for the last four years and is very used to the routine and software he has used during the lectures. Prof. X feels a huge communication gap between the students and himself, he wants to minimize this gap by motivating students toward questions and discussions. With less participation from students, Prof. X thinks that students are not getting the full benefits from his lectures.

Prof. X has already searched some software solutions which encourage students to ask questions during lectures, but most of the software requires the use of specific tools for presentation and only supports specific slide formats. Since Prof. X prepared a large number of slides over the last four years, he does not want to recreate a large number of slides from scratch. Also he is using the Keynote application for a very long time and is already availing the Slide-Notes and other options for his lecture so it is hard for him to change the presentation tool at this stage. He is unable to find any suitable solution for his specific scenario.

Delivering lecture without NoteCarrier

Today Prof. X is going to deliver a very important lecture of the semester and wants every student to understand it properly. Since today's lecture is one of the most important lectures, Prof. X wants to encourage students to ask questions during the lecture. Prof.X asks everybody at the end of each slide "Is it clear to all of you?" or "Do you have any questions related to this slide? " . But unfortunately he is not getting an honest response from students. Prof. X knows that it is one of the most complicated topics of his lecture so the students must have some questions but they are hesitant to ask questions. He feels himself helpless in this situation where he is unable to use an application which might help students to communicate but requires a lot of extra work on his part which he cannot spare at this moment.

Delivering lecture with NoteCarrier

Prof.X is very happy since he is going to introduce a new classroom system to his students. He will deliver todays lecture by using the NoteCarrier application. Since NoteCarrier does provide its functionalities with existing presentation tools, Prof.X is relieved that there is no need to learn a new system or have to recreate his lecture slides. Additionally NoteCarrier allows Prof. X to use his smart phone as an advanced smart remote for his presentation. He is satisfied with NoteCarrier since it is compatible with the slides and softwares he was using for the last four years.

During the lecture Prof. X can have a slide view on his device so he can view these slides on his device without looking back to the projector screen. This helps him to focus more on the students during the lecture. Since students

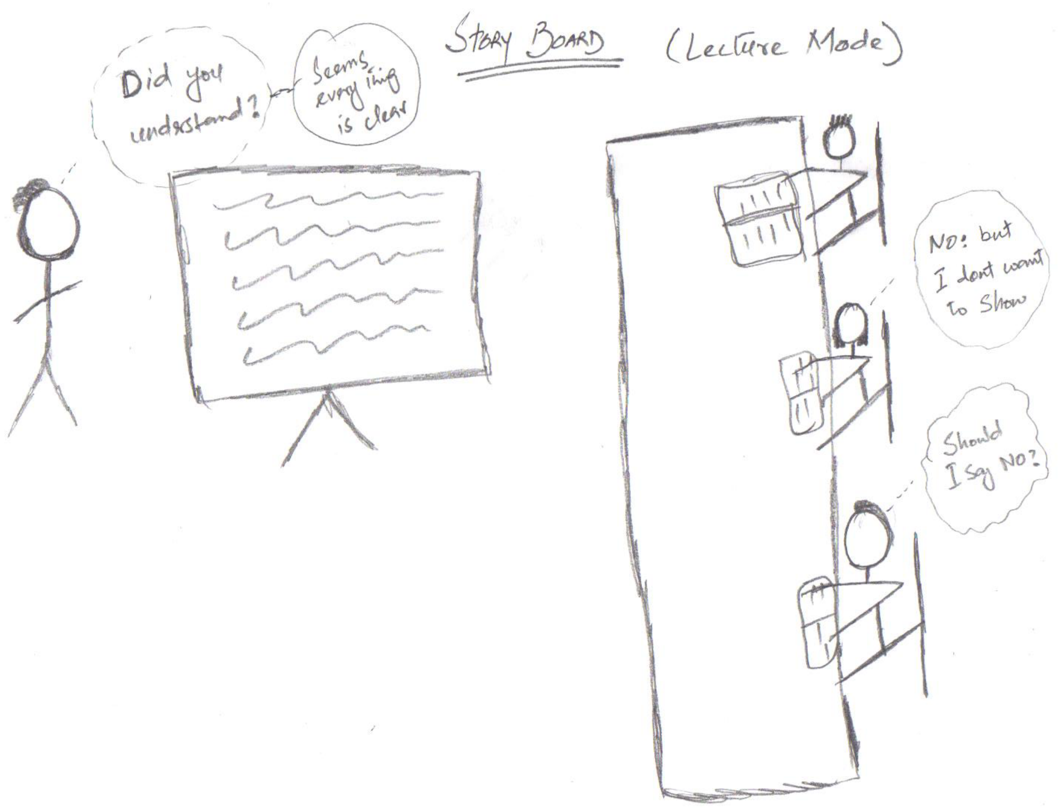

Story Board (Lecture Mode)
Did you understand?
Seems, every thing is clear
No: but I dont want to Show
Should I say No?

Figure 4.3: Storyboard : In this storyboard depicts a situation where lecturer presents a very important lecture to class and want student should participate in the class but unfortunately students are hesitating to participate in the lecture and lecturer not getting any response from student side

can post questions using this application, Prof. X notices that students are more active in participating in the lecture by posting their questions. Prof. X receives a visual notification on his device, that one of the students is unable to understand the current topic. Being a loyal lecturer, Prof. X wants every student in his class to fully understand the topic. So Prof.X tries to present the topic in different ways, to simplify it for the student.

A continuous feedback on each slide also assists Prof. X to understand the difficulty level of each slide and to judge his way of delivering the lecture. This would also help Prof. X evaluate himself so next time he would be able to deliver the lecture in a better way.

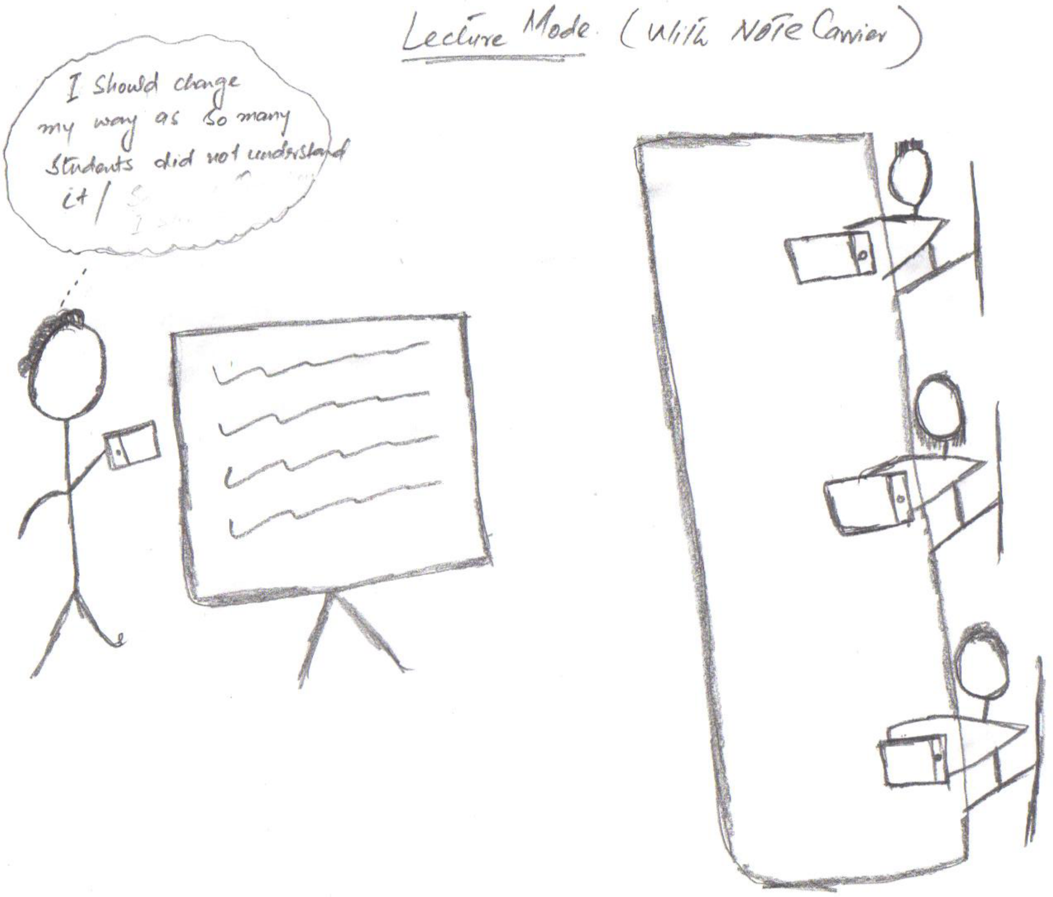

Lecture Mode (With NOTE Carrier)
I should change my way as so many students did not understand it/

Figure 4.4: Storyboard : Lecturer delivering a lecture and noticed that many student mark the slide unclear and posting questions. He noticed that specific part of lecture is bit complex for students so he decided to put extra time on that topic during this and upcoming lectures.

4.2 Evaluation

These storyboards are presented to a group of students for evaluation. The group consists of six students and one lecturer. Among six students four male students and two female students participated in this evaluation process. All of these are students of information technology at the University of Bonn. four out of six students are in their fourth semester of Master studies, the remaining two students were in their second semester of studies. One of Pakistani lecturers also reviewed to evaluate the storyboards to evaluate the lecturer mode of the application.

4.2.1 Setup

Different situations are presented with the help of storyboards

Four different storyboards were presented to students, each storyboard covered a specific feature of the NoteCarrier application. All these storyboards were hand drawn and presented on a paper sheet to each student along with a short questionnaire. The questionnaire consists of very simple multiple choice questions and one open ended question where the student can write comments about the overall idea of the application. Students were asked different generalized questions related to these scenarios. Is the storyboard clear to you ? Did you ever face the same situation during lectures ? Do you support the solution presented in the storyboard? Can it be helpful for you to handle the same situation ?

In the same way a questionnaire was prepared for the lecturer system as well. In that questionnaire we asked about the presentation tool and the difficulty involved if they have to change it on a later stage. Other questions cover the scenarios presented in the lecturer's storyboard.

4.2.2 Results

Six students participated in evaluation process

Gathered results show positive feedback from the students who faced such situations during lectures. Three out of six students responded that they also feel hesitant in asking questions during a lecture and the proposed solution will be a suitable approach to overcome this problem.

A large number of student. five out six admitted that it is easy for them to ask one or two questions related to slide but if they don't understand the whole slide they will not tell the lectures that the topic is unclear to them or will not demand that it be repeated. In this case they admired the NoteCarrier way of marking a slide unclear, as it will also encourage students to start a discussion on the topic if more students marked the same slide as unclear.

Questionnaires are used to collect users feedback

The student feedback shows positive interest in the application as well. They liked the overall idea and want to see its implementation in the real environment. The comment section of the questionnaire showed a great interest by students involved in this application. The overall feedback also showed the need for such system in the real environment and approves the concept of the NoteCarrier application. The application will be a suitable solution not only for the students but also for the lecturers.

The lecturers also showed a great interest in the NoteCarrier application. They also mentioned the same issue in which available applications requires to convert existing slides into specific format and force them to use their own created presentation tool. By adopting NoteCarrier approach they will be free to use the slides they already have and it would be easy for them to continue with previous presentation tools.

Chapter 5

Paper Prototype

> *" Its not good enough to just keep producing technology with no notion of whether its going to be useful. You have to create stuff that people really want, rather than create stuff just because you can."*
>
> — *Genevieve Bell*

Evaluate your idea before actual implementation

Usability is one of the important factors involved in the success of any project. If a project does not have enough potential to be used in the real environment, then it would be just a waste of time and resources, and lead towards failure. Brooke defines the term usability in terms of its particular context, which states that a product or project should be tested within the specific context for which it is being developed. It is necessary to not only define the project measurement but also the context of the project. We already defined a comprehensive context for NoteCarrier in chapter 3—"Requirements" and 4—"Storyboards" . The need is to define a way to test the usability of a project at a very early stage to minimize the risk of failure.

A time and cost efficient approach assess the design of application

The paper prototype is a cost efficient solution to judge the usability of a project. It is another way of expressing your idea to potential users before the actual implementation phase. This technique is used to test the usability of the application with hand drawn UI designs on paper. In the paper prototype, project UI are drawn on simple papers and presented to potential users of the project. A number of techniques can be used to mock the user interaction with the system. A very simple paper prototype may be a simple UI design with no user interaction, where as in other prototypes different ways of user interaction can be introduced to capture user behavior. This interaction might be a simple flip of paper, or an action controlled by humans.

The paper prototype provides a quick and dirty picture of your project to users. Since users are also aware of its simplicity, they feel more open in suggesting different solutions. The development team can also respond quickly to those suggestions by redrawing or altering the prototype according to the user's requirements. All these practices make this simple approach very effective in analyzing the project during its initial stage.

User interfaces are drawn on the paper

For NoteCarrier, a paper prototype approach was used to justify its usability. Different screens were drawn on simple paper sheets and presented to users for simulation. Since major development of the NoteCarrier application consists of nomadic operations, it was very hard to simulate the whole process via a paper prototype. Various techniques were used to simulate the nomadic operation for the NoteCarrier prototyping.These approaches are discussed in the following section in more detail.

5.1 Simulation Techniques

Most of features offered by the application are based on nomadic operations, in which information flows from one device to other. Since it is impossible to mirror this effect in a paper prototype in the same manner so different approaches were used to mock this behavior.

5.1.1 Sticky Notes

Sticky notes are used to simulate application nomadic operations

In the application, posting a question and marking slides as unclear are two main features in which information needs to be forwarded from the student device to the lecturer device. To mock these functionalities, sticky notes were used to pass information from the students to the lecturer (see Figure 5.1). Questions were written on the sticky notes with its relevant slide number, these sticky notes were then passed directly to the lecturer to report the question being asked by the student.

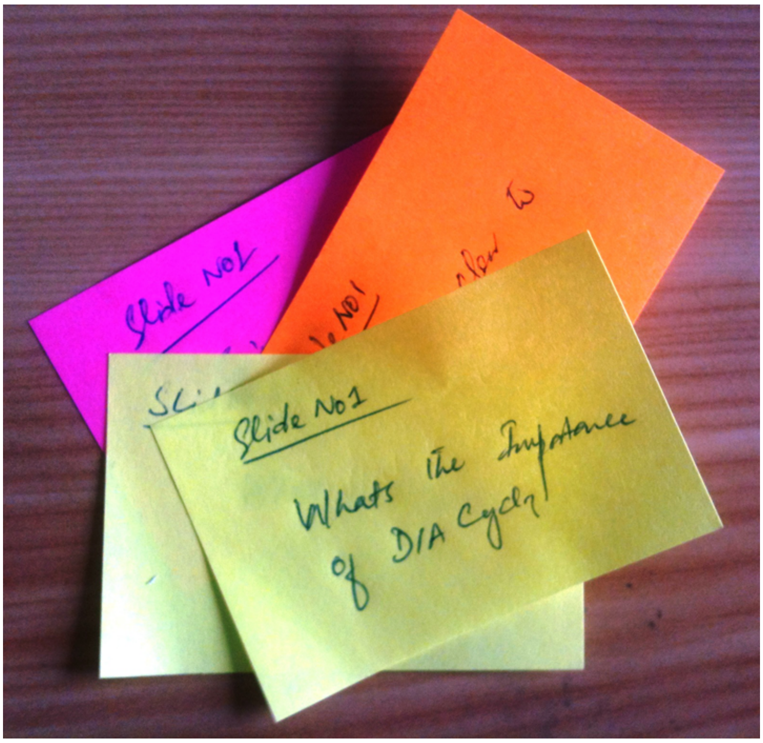


Figure 5.1: Paper Prototype : Posting questions with the help of sticky notes

In the same way, marking a slide unclear functionality is being simulated by sticky notes. The number of sticky notes posted on a single slide with the mark "Unclear" shows the count of students unable to understand that slide.

5.1.2 Beforehand Information

Printed slides are used to simulate the on device slide view

To simulate the behavior of a slides' view on the personal smart device, slides were drawn on the paper before handing them to the students along with hand written slides, printed slides were also used for paper prototyping. For printing a specific print scale was used to fit the slide inside the hand drawn UIs (Shown in the Figure 5.2).

Taking notes on printed slides is a very common technique and already in practice by many students, so it was an easy option to adopt for students. This also helped students to realize the need for the application in their routine activities.

5.2 Paper Prototype For User Interfaces

A number of UIs were drawn on papers to cover different features of the application. As per application architecture the whole screens were divided into two categories. The first set of prototypes were built to simulate the lecturer mode of the application where as the other set of prototypes were created for the student mode.

Although NoteCarrier consists of a cocoa based application server and a number of apple scripts which are running on the server machine, these components are developed to work as a background process for the purpose of handling all of the requests issued by IOS devices. The main purpose of these components is to provide network support for information fetching from keynote applications. Since these software blocks do not have any direct interaction with users, all such low level detail is also kept hidden in paper prototyping.

5.2.1 Prototype for Student Mode

Different UIs are drawn to cover different section of the application

A series of paper prototypes were drawn to explain the concept of the student mode of the application. Different UI were built to express the different screens and features offered by the NoteCarrier. As shown in Figure 5.3 , a set of hand drawn UI's, lecture slides and sticky notes were provided to the students to simulate the behavior of the ap-

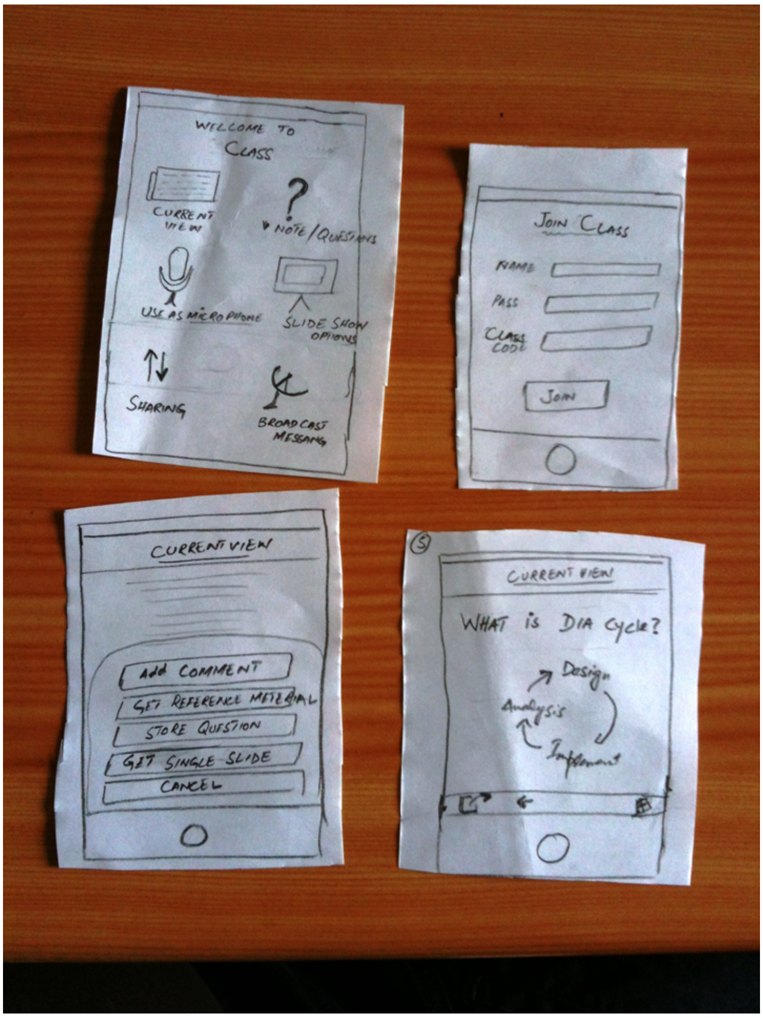


Figure 5.2: Paper Prototype :Paper prototypes represents different sections of the NoteCarrier application for the student mode, Menu screen, Slideview screen, Action toolbar and join class screen

plication. Following is a list of screens developed for prototyping of the student mode.

- Menu screen : Showing the main menu of the application in student mode. Each different section of the application can be accessed using this main menu screen.
- SlideView Screen: This section of the application

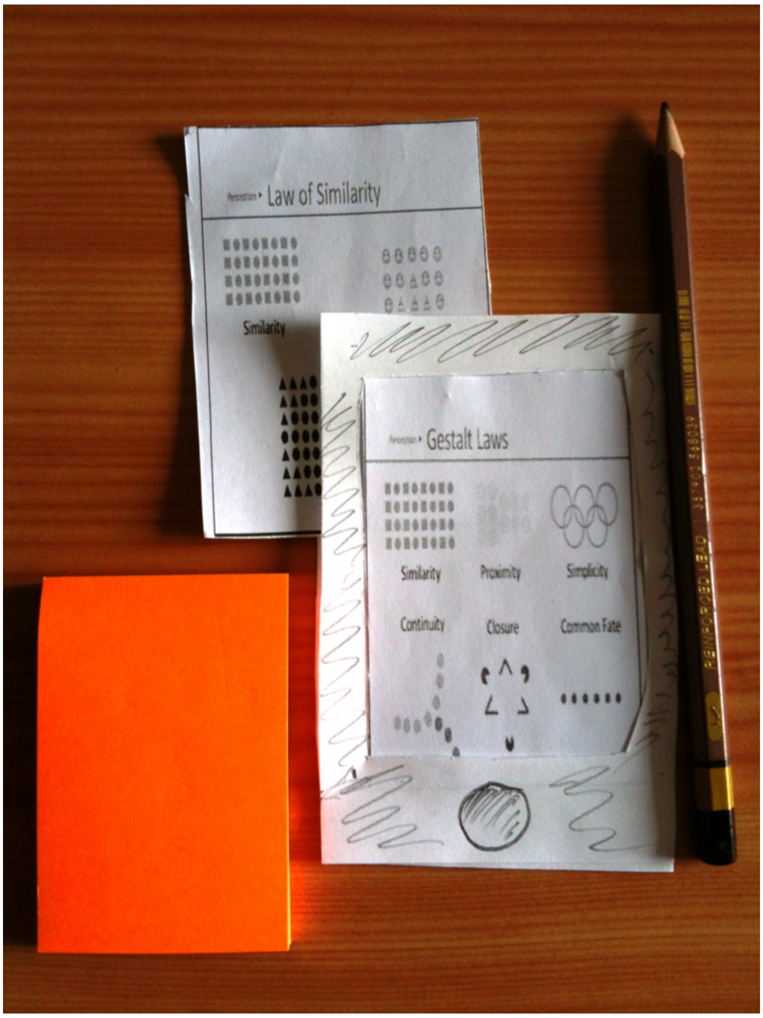


Figure 5.3: Paper Prototype : Paper prototype for student mode, printed slides were used to give the impression of off-line slide view mode of the application

shows the slides on your own device, for this paper prototype we mocked the off-line view mode with the help of printed slides.

- Onslide Menu : It is a replica of the SlideView screen but with an extra menu screen at the top of the slide view. The user can select different actions from this list of actions according to their need. User can post questions or mark slide unclear by using this menu list.

- Post Question : To simulate the post question screen, sticky notes were provided to the users. The user can write a question on these sticky notes with the slide number or write unclear with the reference of a current slide. These sticky notes were then forwarded to the lecturer to give an impression of an on screen question notification.
- Login Screen : In the paper prototype, a login screen was also drawn to convey the idea of authorized access on the content of the lecture, but it was not used during the test rather just used as a symbolic representation of the login module of the application.

5.2.2 Prototype for Lecturer Mode

Since the lecturer mode of the application is more dependent on the students questions and feedback, in this case the paper prototype for the lecturer was much more simpler than the student mode. To make the lecturer aware of questions and feedback posted by students, sticky notes were used as a communication channel between the student and the lecturer.

- SlideView Screen : Although NoteCarrier provides a complete feature set for the lecturer, which includes a device slide view, questions notification, Unclear slide notification and lecture summary sections. Since most of the features are related to notifications which are fired by the students, in the paper prototype lecturer mode did not have too much UI to design in the lecturer mod

5.3 User Testing

Prototype is evaluated with students to simulate the scenarios

It was a difficult job to test the paper prototype in the actual classroom environment due to the time constraint and tight schedule of lectures. Presenting and conducting the test definitely requires a sufficient amount of time from the allocated time for the lecture and does not seems a suitable option for lecturers who already have a strict lecture plan. To avoid this situation, the NoteCarrier application prototype test is conducted with the help of students in the same lecture manner to simulate the classroom environment.

5.3.1 Participants

Mocked the classroom environment with small group of people

Six people participated in this small activity in which four were males and two were females. One of the male participant acted as a lecturer and the rest of the group acted as students of the class. Instead of using a projector screen, a LCD display was used to display the slides on a larger display. Students were provided with a paper prototype with printed slides. Sticky notes were also distributed among students for writing down the questions or putting a mark "Unclear" .

5.3.2 Task

Information shared using sticky notes

To make the test approximate an actual case scenario, a small number of the lecture slides were used from the subject *"User Centered Software Design"* which was taught at the University of Bonn. The whole test was conducted in a small room equipped with the necessary hardware and software components. The size of the room also played a vital role in passing sticky notes, which were used to to provide data sharing support over the human network. I was responsible for forwarding student written sticky notes to the lecturer in appropriate slide order.

A fifteen minute session was conducted in which approximately ten slides of the lecture were covered. Students noted the questions related to different slides and handed them to me. At the end of each slide I forwarded all the questions to the lecturer which were related to that slide so that the lecturer could review them before moving to the next slide.

5.4 Findings

The whole purpose of this prototype test was to get a first impression of the application and to validate its usability on a small scale. Although the paper prototype cannot represent the whole concept in the same way as the software but still the participants liked the whole idea and showed their interest by their feedback and suggestions. Following are some findings we extracted from this test

- Menu Design : As we implemented a menu screen, which allows students to perform different tasks on the current slide like posting questions or marking slide unclear etc, most of the people didn't like to go to another screen/menu bar to choose the right option. In their point of view, It would be great if they have direct buttons on the tool bar or status bar for such options.
- Menu For Current slide : One of student was not using the application in Synchronized mode, so he was not aware of which slide was current on the view. He just wanted to ask a question on that slide so the only option at that time was to go in synchronized mode and after fetching the current slide post your question on that slide. This might be not a suitable option for those students who don't want to use the synchronized mode all of the time.

5.5 Summary

Overall the paper prototype session proved a good approach to test the application at very early stage. It not only helps us, get the first user impression about the application but also provides positive feedback and suggestions to improve it. We also observed different user behavior such as user interactions, most used sections, ambiguities and confusion they faced during test.

Different new features are planned to implement the ap-

Participant's feedback helped to refine the requirements plication after reviewing the user feedback in this session. Features like Providing a direct link for posting questions and marking slides unclear are implemented in NoteCarrier. In this way the students are not required to use the synchronized mode for posting questions. The new options will be accessible on the page main menu which allows the student to post questions to the current slide and eliminate the extra steps.

Chapter 6

Software Prototype

> *" I think that in the discussion of natural problems we ought to begin not with the Scriptures, but with experiments, and demonstrations."*
>
> — *Galileo Galilei*

To transform our ideas into reality, a small software prototype of NoteCarrier is developed. The resultant prototype will help us to assess the usability of the software and also provide us the opportunity to use it in a classroom environment. In the following chapter, the detail work on the software prototype will be described. The whole chapter will cover different aspects of the software including system requirements, design, features and software prototype.

The first prototype of application is developed for specific hardware and software. The system requirements are mentioned in chapter 3—"Requirements" in the respective sections 3.1—"Hardware Requirements" and 3.2—"Software Requirements" .

This chapter will be explained in three sections. The first section will be about the design requirements of the software. A list of design requirements will be described which should be followed during the implementation of the software.

The second part of the chapter will describe the software and hardware architecture of the application. I will also specify all the features and functionalities offered by the the software.

The last part of this chapter will explain the nomadic framework that is used in application to support all nomadic operations.

6.1 Design Requirements

Main design requirements for the application

With the help of the related research study and requirements gathering phase, a set of design requirements are defined for the software. These requirements are prioritized according to their importance in the software, high demanded requirements are then selected for the first prototype of the application. In the following sections, only those requirements will be discussed which are implemented in the first prototype.

6.1.1 Support for Available Presentation Tool

As related work shows, most of the applications require us to use their own built-in presentation tool for presentation. To work with that software, one has to either create new set of slides using their provided tools or use a specialized plug-ins to import already built presentations. In contrast to this approach, the new design approach should make the software compatible with existing presentation tools i.e. Keynote instead developing them from scratch. The new software should be able to use an existing presentation tool without importing or converting already built presentations and have ability to communicate with that presentation tool. This feature will make the application easier to use for the lecturers and minimize the investigation time that is always required to adopt new solutions. The new design will make the application more adoptive for the presenter since they still have an option to switch back to Keynote and deliver the lecture in their traditional way.

6.1.2 Provide Simple Interaction Techniques

In a related work section, I mentioned the applications that provide functionalities for posting questions and collecting student feedback. To display this information to the lecturer, dedicated applications are being implemented for the lecturer machine. The lecturer has to switch back and forth between the presentation and the application window to look student's posted data. It not only interrupts the whole lecture sequence but also puts the lecturer in an annoying situation by requiring the selection of different windows again and again.

To address this issue, the software should support a simple interaction technique that does not required manual work for checking the question list periodically. Rather the lecturer should only be notified in case someone posts a question. The lecturer should also be able to take a look at these posted questions without interrupting the active presentation.

6.1.3 Implementing of Bidirectional Communication

One of the known limitations in the current classroom environment is the communication gap. Without student participation, the lecture does not make any sense and will affect the overall learning process. To avoid such situations, the software should define a communication mechanism that encourages the students to participate in the lecture. The solution provides an easy to use set of features for active participation in the class, for instance asking questions or reporting unclear contents presented during lecture. The software must have the ability to link the questions with the relevant slide. In this way the lecturer will be able to fully understand the question in its relevant context.

6.1.4 Anonymous Class Participation

Another problem involved in class participation is student hesitance in asking questions.MATT RATTO and GRISWOLD explains this student behavior due to increased class size and the diversity of students. He also mentions that in some cases students also feel that raising too many question might convey a message of impolite behavior. Its also hard for some students to ask questions in front of fellow students due to their shy personality.

To deal with this social problem, the proposed application must provide a convenient way to encourage these students to actively participate. One suitable option will be posting questions anonymously. Students who feel reluctant to ask questions can get a benefit from this feature by posting their questions without raising their voice. Questions are forwarded to the lecturer without their identity which helps these students to take part in the lecture effectively without any hesitance.

6.1.5 Ability To write on slides

Observations were made during the paper prototype sessions that many students preferred to write on printed slides rather on a separate notebook. This will allow students to a keep link between the slide content and the notes they wrote down and also minimize the effort required to copy the slide contents. To keep this habit in practice, a new software design should support writing down notes on a slide's image. Students will be able to store their notes on the slide view in the same manner but in a soft copy format.

6.1.6 Student Control over slides

In the classroom students often ask the lecturer to go back to some specific slide. Either they missed some point on that slide or they want to write down some important notes related to that slide. To help students in this situation, the application must provide the student an ability to select a specific slide on their local view. Such quick control over slides not only saves the students and the lecturer time but also minimizes the overall interruption caused by these requests.

6.2 System Design

By keeping in mind all the non functional requirements and design requirements. The system design for NoteCarrier is proposed. The main goal behind the design was to make the software compatible with current presentation tools. Instead of implementing a specific presentation tool for the application. NoteCarrier was designed in a way so it can be used with existing presentations tool e.g. Keynote. By adopting this approach, the additional step of providing an interface for converting existing presentations to software readable formats will also be eliminated. Figure 6.1 shows a high level design architecture of the system. To keep the legacy system on its place, the NoteCarrier application server is installed as a background process on the lecturer mac book, which also allows the lecturer to use their notebook in the same manner. Additionally an IOS based version of the NoteCarrier application is installed on the lecturer and the students smart devices. To establish a communication channel, a wireless network connection is placed between the application server and IOS based applications.

NoteCarrier consists of three software components, an application server and two IOS based applications

The current high level design architecture consists of two main components

- Cocoa based server application.
- IOS based Iphone/Ipod/Ipad Application

In the following sections I will describe these components in more detail with additional information about the subcomponents involved in the design.

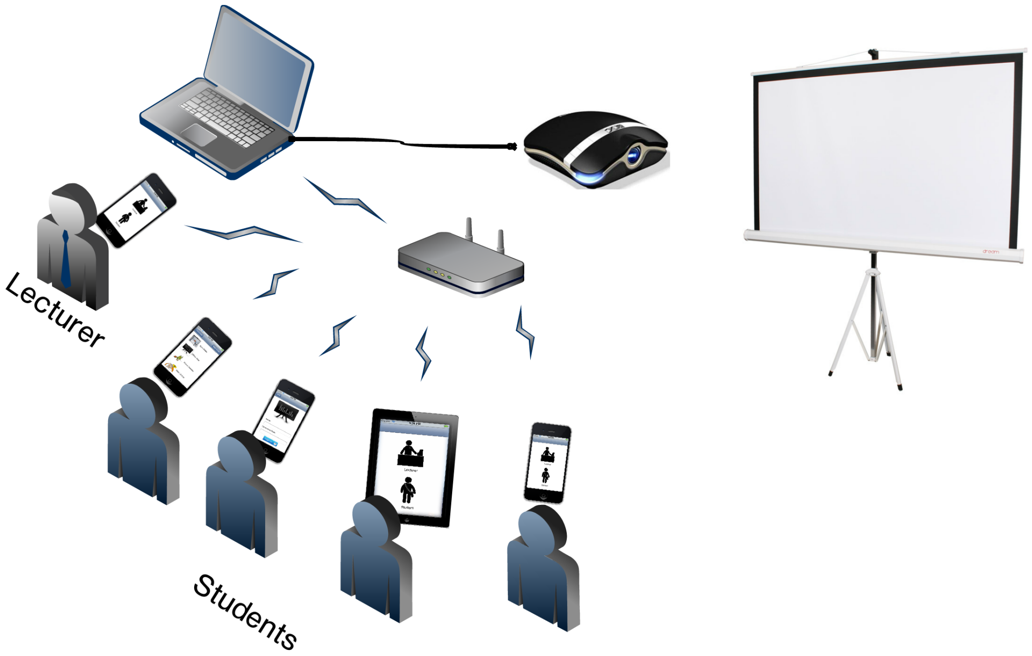


Figure 6.1: NoteCarrier System Design : Figure shows a high level design for the NoteCarrier application, In which the presenter system is attached to a projector. All IOS devices and the presenter notebook is connected over a wireless network connection for intercommunication

6.2.1 Cocoa Based Server Application

The application server acts as a communicate bridge between Keynote and client applications

The server application of NoteCarrier performs two main activities, controlling network traffic and communication with the Keynote application. The server application is responsible handling all the queries sent by the client devices. There are a number of queries a client can post to the server, these queries might be a request for data or to get some information related to the presentation tool. The following are supported queries a client device can sent to the server.

- Login request as student
- Login request as lecturer

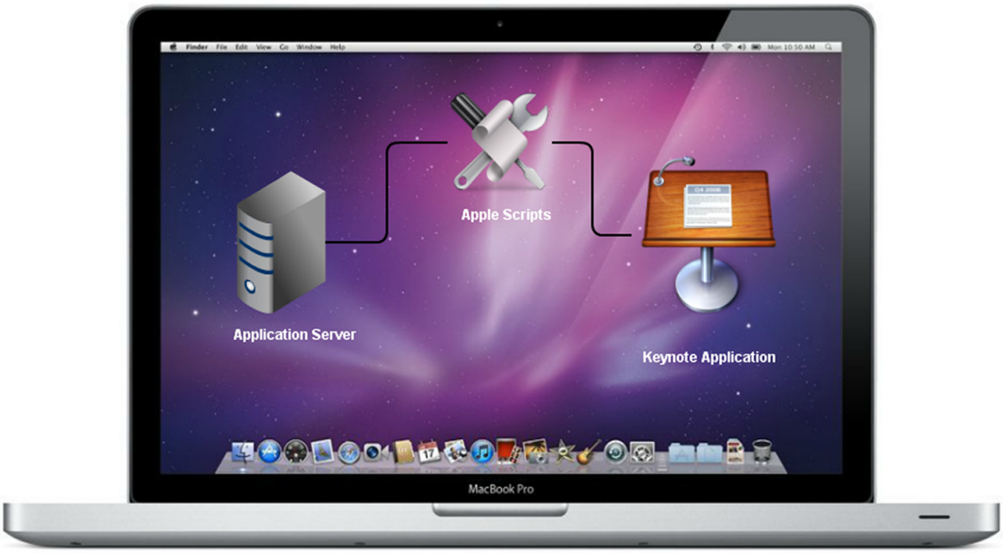


Figure 6.2: High level architecture on the presenter machine

- Get total slides count
- Get current slide number on the view
- Get a slide as an image with slide number
- Get current slide as image
- Post a question related to current slide
- Post a question with slide number
- Marking current slide unclear
- Marking a specific slide unclear with slide number

There are two major categories of these queries, information posting queries and information retrieval queries. In information posting queries, students send information to server, this information might be in the form of questions or a flag to indicate that slide contents are not clear to them. Students can post these queries either from the off-line slide view section or the main menu in which all operations are attached to the current slide view.

The other type of queries are information retrieval queries. The purpose of these queries are to fetch information or data from the server application. Loading the current slide or getting the total count for slides are some example of these categories.

AppleScript is used to control Keynote application

Except in off-line mode posting queries, the application server has to communicate with the Keynote application to fetch information about the current presentation. This fetched information is needed to prepare the response for the information retrieval queries. AppleScripts are implemented to build a communication channel between the NoteCarrier application server and the Keynote application. Figure 6.2 shows a high level architecture of the NoteCarrier application on the presenter machine. According to this architecture the NoteCarrier server application consists of three main components

1. Keynote
2. AppleScripts
3. NoteCarrier Application server

KeyNote

Keynote [1] is a standard tool for presentation for OS X operating systems. The Keynote is available in iWork [2] suit, an office suite which provides tools for documentations, presentation and spreadsheets etc. Keynote provides an ability to present or create presentations. The big advantage of using the Keynote application for the NoteCarrier is that it provides an interface for communication via AppleScripts. This interface provides a mechanism to communicate with running Keynote application using these scripts.

Another plus point for using Keynote is its import functionality, which allows the presenters to import presentations made in other tools e.g. Microsoft Power Point. In

[1] http://www.apple.com/iwork/keynote/

[2] http://www.apple.com/iwork/

this way the software can also be used with those presentations made in tools other than Keynote.

AppleScript

AppleScript is a scripting language provided by Apple that is used to automate tasks in OS X operating system.

AppleScript is a scripting language that makes possible direct control of scriptable applications and many parts of the Mac OS. A scriptable application is one that makes its operations and data available in response to AppleScript messages, called Apple events. Source [3]

NoteCarrier application server provides an interface between client devices and AppleScripts

In NoteCarrier, AppleScripts are primarily used to communicate with the Keynote application. Since the Keynote application already provides an interface for communication via AppleScript, no additional development was required to manage external commands in the Keynote. Since Keynote only possesses limited functionalities via its scripting interface, I developed an application server as a wrapper over these scripts to provide additional functionalities. For instance, keynote does not provide an export functionality in its presenter mode, to deal with this problem the application server exports all the slides at the start of the lecture and maintains this information on its own. Any request regarding slide fetching is directly handled by the application server without any interaction with AppleScript and Keynotes.

Following are some of the Keynote functionalities utilized via AppleScript in the application.

- **Total Slide Numbers :** Keynote can return the total number of slides from the current presentation file. This count is fetched by the AppleScript wrapper application at the start of the session and stored locally. The application uses this count for many decision based operations in both the server and the client

[3] http://tinyurl.com/cr5k8yg

applications. For instance, switching the slide in lecturer mode sends next slide call to the server on the basis of the current slide and the total slide count.

- **Exporting Slides :** Another important feature of Keynote is an *Export* function. In the export function, Keynote will export all or a specific number of slides in a specified format on a provided path. Currently Keynote supports different file formats for its export functionality. In current implementation of the NoteCarrier application, all slides are exported as image files to a specific folder stored on the file system.

- **Current Slide Number :** AppleScripts are also used to fetch the current slide number that is on the presenter view. This is one of the most frequently used requests that application server posts to Keynote via a script. The fetched value is used in many functionalities such as posting questions, marking the slide unclear and the synchronization view.

- **Adding Presenter Notes :** In NoteCarrier, when a student posts a question related to any slide, the lecturer immediately gets a notification and question on his IOS device. Besides this mechanism, the lecturer can also see the same question in the footnotes section in the Keynote presenter view. This is an optional feature with some limitations since it might confuse those lecturers who use the notes section for their personal points. But in case the lecturer doesn't want to use a separate IOS device and prefers to deliver a lecture in the traditional way, NoteCarrier will post a question directly to the notes section. Adding functionality for injecting questions as slides notes is also achieved using AppleScript.

- **Slide Switching :** The lecturer mode of the application provides an opportunity to use a smart device as a presentation remote control for slide switching. Using a swipe gesture the lecturer jumps to the next or previous slide of the presentation. In implementation of this feature AppleScript is responsible for communicating with Keynote for slide switching.

6.2.2 IOS Based Application

Another completely separate and major part of the development is an IOS based application for students and lecturers. Since the application server and AppleScript work as background processes, this is the only part that is visible to students and lecturers. All interaction with the system is only possible through the interface of the IOS application. This application has two different operating modes, a student mode and a lecturer mode (Shown in Figure 6.3). Each mode is secure with a password and can only be accessed via valid credentials. The password policy is imposed to secure the lecturer contents and to avoid unwanted access on the lecturer view.

IOS based application can be used either in Lecturer mode or Student mode

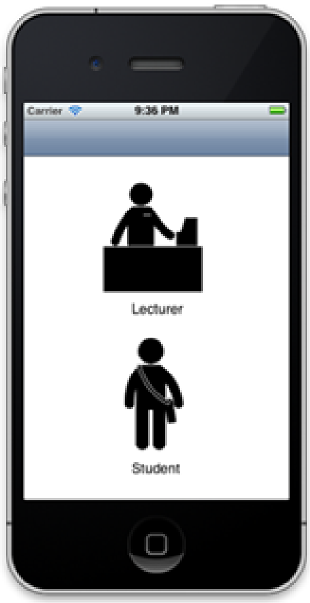


Figure 6.3: NoteCarrier: Figure shows a startup screen of the application, where the user has to select an appropriate mode to use the application

The current prototype of the application can be used in to two different operating modes. *Lecturer Mode* and *Student Mode* are two modes developed to assist the lecturer and students respectively. Each mode provides a completely different set of features according to its needs.

We will split the the following section according to these two modes. In each section, all those features and functionalities will be described which are offered by the respective operating mode.

6.2.3 Lecturer Mode

The lecturer mode is designed to serve the lecturer in the classroom. The application provides very limited but attractive features in this mode which not only make the presentation process easy for the lecturer but also improve the overall learning process. The features offered by the application specifically for lecturers are listed below.

Slide View on Device

NoteCarrier provides the same slide view on the smart device as it is on the projector screen. It gives more room to the lecturer to move around the class without tying himself to the presenter desk to look at slide content. Although the projector screen can be used as an option for viewing slides, it put an additional burden on the lecturer to look back again and again at the projector screen while presenting. This NoteCarrier feature allows the lecturer to have a glimpse of the slide content while facing the students.

Using Device as Presentation Remote

This software prototype can also be used by the lecturer as a traditional presentation remote. By swiping a finger right or left on the screen, the lecturer can switch between slides on the projector view. In this way the lecturer doesn't have to use the additional remote for slide switching. Using swipes also make the slide switching process easier as compared to push buttons on conventional remote controls. Swiping is a more natural way for humans to change pages while reading books or papers. It will be more convenient for the lecturer to use a swipe gesture rather than pushing buttons on the remote. It also remove the hassle of remembering the right button for the appropriate direction.

Notification for Question

The lecturer mode for NoteCarrier is not just a remote for the presentation rather it offers extra features which make it different and more functional. Getting notifications about the questions raised by the students, is another exciting feature of this application. The lecturer has an on screen notification on his device, if students post a question related to current slide. The application stores all these questions with their relevant slide numbers. In this way the lecturer can view the slide specific questions in one place. Keeping a reference between the question and the slide also makes it possible for the lecturer to respond to the question as soon as the student posts it or entertain the questions at a later stage.

Notification for Unclear

Another feature that creates a distinction between NoteCarrier and other applications is the ability to visually notify the lecturer if the content of the slides are not clear to students. This unique feature allows the lecturer to assess the complexity of the slide material by knowing the number of students who are unable to understand it. If the view of the current slide which is being delivered by the lecturer starts getting red. It means the lecture is not clear. The more red it gets, the more unclear it is. The lecturer′s view also shows the count for unclear marked i.e. the number of times students mark the slide unclear. The software keep track of these counts with each slide, which helps the lecturer analyze his lecture contents. Using this analysis data, lecturers can improve their slide contents or can allot more time to these contents during a lecture.

Lecture Summary

Questions are one of the most fundamental parts of lectures. They not only show the interest of the students towards the lecture but also clarify the ambiguous aspect of the lecture and improve the learning process. Questions might be raised at any time during a lecture or in some cases the lecturer may prefer to have a Q&A *"Question and Answer"* session at the end of the lecture. NoteCarrier saves all questions posted by the students with the relevant slide numbers. This allows the lecturer to look at all the questions at the end of the lecture in the lecture summary section. The lecture summary section improves the readability of these questions by sorting them according to relevant slide numbers.

6.2.4 Student Mode

One of the main goals for designing NoteCarrier was to assist the students in their activities during a lecture. To provide a complete set of functionalities, the application not only offers attractive features in the *Lecturer Mode* but also proposes a set of functions to assist students in the *Student Mode*. Figure 6.4 shows a main menu of the IOS application in *Student Mode*. The following section will explain the *Student Mode* of the application.

Synchronized View Mode

The application allows students to have slide views on their own devices. In synchronized view mode the application will show the same slide on the device as is it on the projector screen. Slides will be synchronized with the projector view. As soon as the lecturer switches the slide on his machine all of the client devices will update their views to that current slide. All the slides are downloaded on the client device at the start of the client session, this will improve the application's response time since it will not be required to download each slide separately during the lecture.

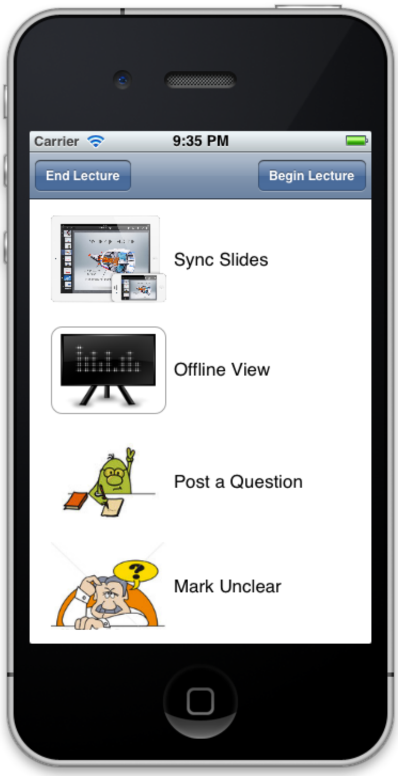


Figure 6.4: Application provides two different slide view modes, Off-line and Synchronized slide view. Application also provides quick options to post questions related to current slide or mark it unclear.

Off-line View Mode

In contrast with the synchronized view, the off-line view mode gives more freedom to student to switch between slides. Students have a full set of slides on their own device which they can view, swipe left or swipe right gestures can be used to select next or previous slides respectively. The off-line view provides an opportunity to consult slides other than on the projector view, in this way students don't have to ask the lecturer to go back to a previous slide to view the content once again.

Posting Question

Students can also post questions using the NoteCarrier. The application provides two possible ways to post question during a lecture. Students can post the questions related to a slide currently on the presenter view by choosing "Post a Question" option from the main menu. To ask questions about slides other than the current slide, a student can choose a slide in Off-line mode and post a question against that slide. All posted questions are immediately sent to the lecturer's device, if the question is related to current slide, the lecturer will get an on screen notification for that question. In other cases the lecturer will be notified with the posted question once that specific slide is selected for the presenter view. The lecturer can also view the list of all posted questions in the "Lecturer Summary Section" available in the lecturer mode.

Posting questions by device encourages those students who are hesitant to ask questions during a lecture. This can also be used as an alternative way for those students who don't want to interrupt during a lecture and choose to note their questions for the Q&A session.

Marking Slide Unclear

"Is it clear to you?", is one of the most common questions lecturer used to ask at the end of each slide. Unfortunately most of time they don't get a true answer from students. Some students are reluctant to raise theirs voice if they are unable to understand or in other cases students think that it might create a bad impression with the lecturer.

The application provides very easy way to handle this situation. On each slide the student has an option to mark it unclear without raising their voice. Each time a student marks the slide unclear, the lecturer will get a visual notification on his screen. This will encourage students to notify the lecturer about the unclear content without having to point it out.

Taking Notes

NoteCarrier, as the name suggests is an application that provides students the ability to take lecture notes. In the application students can take their handwritten notes on the images of slides, this method provides the same mechanism to students who used to take notes on printed slides. Students can take notes using a stylus pen or finger on the client devices. The application also provides an option to save these notes into device memory so they can use them at later stage.

Taking notes on a slide, also keeps a track of your notes and slide contents and saves time by eliminating the time needed to copy the contents from the slide to your notebook.

Saving Slides

Although the application downloads the latest slides from the server at the start of each lecture session, these slides are not permanently saved on your device. Rather the slides are only available until the end of that specific session. The application will remove all the slides from device once you create a new session for a lecture. The student can also remove the data from device by using the ″End Lecture″ option available in both the lecturer and the student modes.

To explicitly save the content of a lecture, the application provides a save functionality in the off-line view mode. Students can save any particular slide while taking notes on the slide, all saved slides are stored in the form of pictures under the ″Photos″ Section.

6.3 Nomadic Operations

To establish a communication channel between the IOS based application and the application server, all devices are connected to the same wireless network. In the application, an open source framework async-network [4] is used to implement networking related functionalities to support communication between the application server and the IOS applications. The framework is based on Apple's open source technology Bonjour[5] . This network technology provide a zero-configuration networking support which automatically discovers devices and services on the same local network. With the help of this technology, in NoteCarrier when a new IOS based application connects to the wireless network, it will automatically search and connect with the application server service hosted on the same network. Each server will be implemented with a unique service name, this service name acts as a key for all client devices. To implement multiple servers on the same network, each server will have a unique service name and clients can connect to a specific server by using its unique key.

Asynchronous network connections is used between IOS devices and application server

To separate the handling of network requests posted by students and lecturers the application server uses two different server keys for students and lecturers. This mechanism also ensures that only an authorized device can post a certain request. An IOS device registered in student mode can not access or post those requests that are reserved for a lecturer device only. As soon as a new IOS device connects with a wireless network, the application server automatically detects its operating mode (i.e. student device or lecturer device) with the help of a unique service key. A new network connection is established between the application server and a newly connected IOS device according to its operating mode.

Lecturer mode and student mode requests are handled by two separate network servers

The async-network framework is based on Client Server architecture using TCP/IP socket programming. This client server infrastructure uses an asynchronous approach for handling each request. Client machines send the request asynchronously to the server where as the server handles each request in a new thread without blocking itself for new incoming requests.

In a traditional classroom environment, one lecturer delivers a lecturer to many students. According to this scenario

[4] https://github.com/jdiehl/async-network

[5] https://developer.apple.com/bonjour/

one dedicated request handler is sufficient for the lecturer. But handling all student network traffic with a single server may affect its performance and students may face a slight delay in getting responses back from the server. This delayed response will be more visually noticeable in a synchronized slide view section where the students might experience a short delay in updating the slide view. To overcome this issue, the current client server architecture needs to be upgraded. Introducing more active servers or replacing the client server architecture with a peer to peer methodology can improve the network performance in this scenario.

Chapter 7

Evaluation

> *" Measurement is the first step that leads to control and eventually to improvement. If you can't measure something, you can't understand it. If you can't understand it, you can't control it. If you can't control it, you can't improve it."*
>
> *— H. James Harrington*

NoteCarrier is evaluated during real classroom session

To judge the usability of NoteCarrier, a small user study was conducted in the real classroom environment. The main purpose of this case study was to investigate the feasibility of the project in the real environment. Gathering people specifically for evaluation may influence the results of the user study due to a preset mindset about project. To avoid this situation, NoteCarrier was presented to students at the start of a lecture without any prior knowledge about the project. Only a small introduction was given to the students about the use of the application. The stage was set to get a very first impression from the students about NoteCarrier and to assess its usability.

7.1 Study Plan

User study was conducted at University of Bonn

The user study was conducted in "Object Oriented Software Construction" [1] class on Wednesday 11th July 2012 from 11:15 to 13:00 on the premises of "B-IT building Bonn"[2] . It was the last lecture of the semester and planned as a summary lecture for the course taught during the semester. The lecture was fifteen minutes longer than a routine lecture since lecturer had to cover the whole course summary. The lecture was conducted by Daniel Speicher [3] , who covered approximately 147 slides in that lecture. The lectures slides were made in Microsoft Power Point but since the current implementation of NoteCarrier only supports Keynotes as a presentation tool, those slides were imported into the Keynote application. Macbook pro was provided to the lecturer to use for the presentation. The macbook pro was also connected to a projector, so students could have a larger view of the slides on the wall.

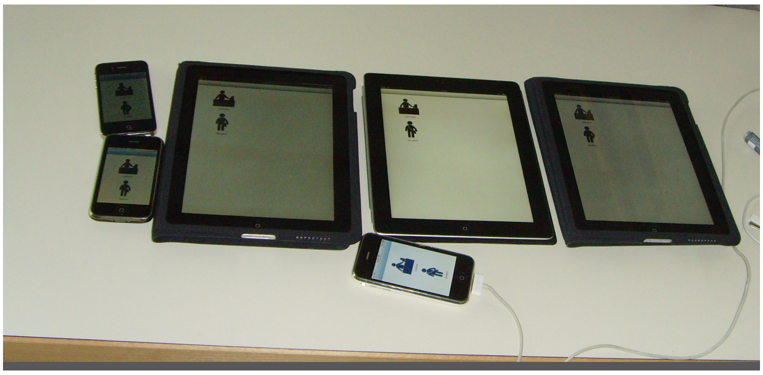


Figure 7.1: User Study : Configuring NoteCarrier on the client devices

6 IOS devices are distributed among students

To assist the audience of the lectures three Ipads and three Iphones were configured as client devices (See Figure 7.1). All devices were updated to IOS 5.0, to support the client application of NoteCarrier. A dedicated wireless network was also configured as a communication channel between the client and the server application.

[1] https://sewiki.iai.uni-bonn.de/teaching/lectures/oosc/2012/start/

[2] http://www.b-it-center.de/

[3] https://sewiki.iai.uni-bonn.de/private/daniel/public/start?s[]=daniel

A pre-demo of NoteCarrier was presented to the lecturer before the lecture. To save the lecture time all server and network related settings were configured before the students arrived. To ease the students, network connections were also configured on all client device before handing them to the students. A small introduction was given to all students about the working of NoteCarrier at the start of the lecture.

A brief demonstration of application presented before start

7.2 Application Setup

One of the main design features for the NoteCarrier is to propose a solution which is easy to integrate with minimum human interaction. To accomplish this, the application is designed in a way that it doesn't require extra configuration time before lecture start. Once the NoteCarrier is installed on each client device the whole setup took maximum one to two minutes to setup. During user study, most of the time was used to setup the dedicated wireless network. Since currently most of the classrooms are already equipped with such facilities so the setup time can be reduced by using already installed wireless setup.

Easy to integrate NoteCarrier without the need of specialized hardware and software

To begin the lecture, the lecturer only have to run NoteCarrier application server on his mac machine, once the server is started all client devices are ready to use. In contrast to other applications in which lecturer has to broadcast the presentation material at the start of the lecture, NoteCarrier client can join and fetch the slides at any stage during the lecture. Student who joined the lecture lately can also connect their client device with server and use the application in same manner.

In our specific user study at one stage unfortunately the wireless router get disconnected which interrupts the whole communication channel between clients devices and application server. Without the network connection, client devices were unable to communicate with server and perform their functionality. Since it took little time for me to investigate the root cause for this failure but to keep the lecture flow continue, Lecturer decided not to use NoteCarrier

No additional time is required to unplug NoteCarrier in case of failure until I figure out the issue. As it was a very minor issue and fixed very quickly but the whole situation helped me to observe application plug and play support during failures. As in case of failure, lecturer did not have to perform any extra setup to switch back to his old presentation setup. The lecturer just had to put away the NoteCarrier client device, and used his presented machine to control the flow of the application in his same routine way.

7.3 Participants

Lack of devices influenced the overall evaluation process

Since NoteCarrier was presented in a routine lecture (See Figure 7.2 and 7.3), there were twenty students present in the class during the user study. Fourteen of the twenty students participating in this study were males and six were females. Due to a shortage of client devices we could not offer a dedicated client device to each participant. Five out of six devices were distributed to students to use NoteCarrier in the student mode and one was given to the lecturer to use NoteCarrier in the lecturer mode. To manage a large

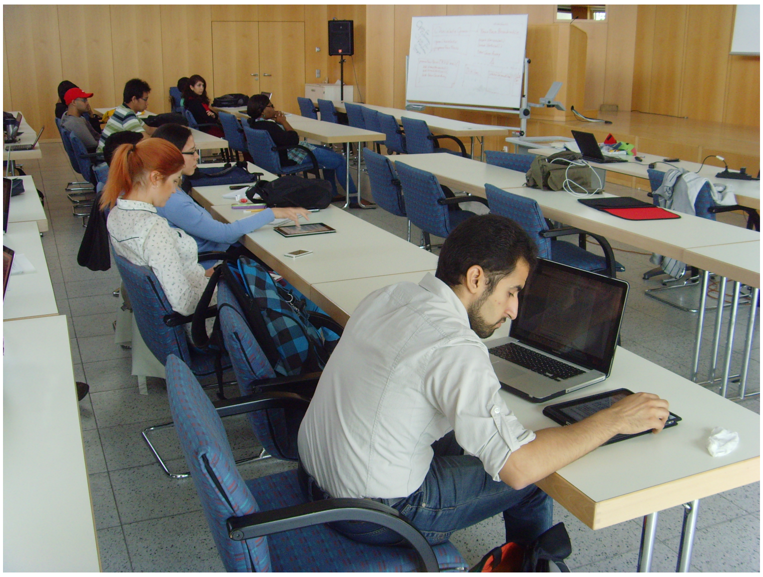


Figure 7.2: User Study : Students are using NoteCarrier in Student Mode

number of students with fewer devices, these devices were passed to those people who were sitting closer and formed a group so they could share a device. After one hour these devices were redistributed to cover those people who did not get an opportunity in first hour to experience NoteCarrier.

One device was assigned to the lecturer to use NoteCarrier in lecturer mode. To facilitate the lecture an Iphone device was used instead of an Ipad, which allowed the lecturer to easily move around during lecture. The lecturer also had an option to use a macbook as the presenter pc to deliver a lecture in the traditional way without using NoteCarrier.

The whole evaluation session spanned over two hours

The students and the lecturer used the NoteCarrier application over the whole lecture time span. Although the client devices were short in number we somehow managed with group sharing. A questionnaire was distributed to the students at the end of the lecture to evaluate the performances of NoteCarrier. Personal interviews were also conducted with the participants at the end of the lecture.

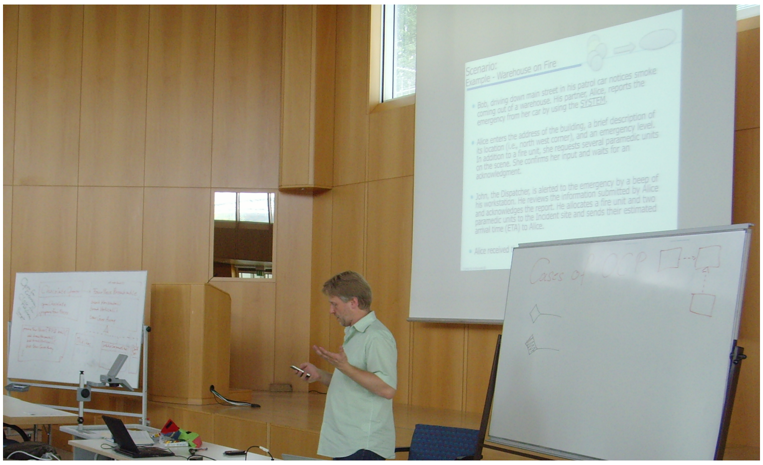

Scenario:
Example - Warehouse on Fire
Bob, driving down main street in his patrol car notices smoke coming out of a warehouse. His partner, Alice, reports the emergency from her car by using the SYSTEM.
Alice enters the address of the building, a brief description of its location (i.e., north west corner), and an emergency level. In addition to a fire unit, she requests several paramedic units on the scene. She confirms her input and waits for an acknowledgment.
John, the Dispatcher, is alerted to the emergency by a beep of his workstation. He reviews the information submitted by Alice and acknowledges the report. He allocates a fire unit and two paramedic units to the Incident site and sends their estimated arrival time (ETA) to Alice.
Alice received
Cases of OCP

Figure 7.3: User Study : Lecturer is using NoteCarrier in Lecturer Mode

7.4 Observations

The whole lecture was observed during this user study to determine the user's point of view of the project. At the end of the lecture, small discussions were held with individual students about their observations. In the following section we will discuss these observations in more detail.

7.4.1 NoteCarrier In Lecturer Mode

Lecturer showed a great interest in application features

In this user study, the lecturer really appreciate the efforts and the features provided in form of NoteCarrier. The lecturer utilized almost all of the functionalities offered by application. Especially marking slide unclear was a different experience for the lecturer when on one slide the view of the slide turns to red. The lecturer also switched the slides using his Iphone as presentation remote and mentioned that for him it was an easier approach to use swipe gesture as compared to the traditional remote buttons. During the lecture, the lecturer also used the application to view all the slides on his smart device rather using the projector view. In the interview session, the lecturer supported the application features with his positive feedback.

7.4.2 NoteCarrier In Student Mode

Difficult to fully evaluate with limited devices

The most exciting feature of the NoteCarrier application is the *Student Mode*. In this user study we could not investigate this mode in every detail due to a shortage of devices. But we were able to judge the first impressions of students about such an application. In the following section, I will discuss those aspects of the application that I observed during the user study.

Notes Writing

Students were not interested in taking notes on test devices

NoteCarrier provides a functionality to take down the notes on the slides using the finger or a stylus. In this specific user study, observations show that this feature was not used by the students as expected. Very few students used this feature, or rather used it just for test purpose.

There were two main reasons students told at the end in interview session. One student explained this to me as

"I don't own this device, if I take notes on this device, then all those notes will not be with me after the lecture. Since this is an important lecture for me I don't want to lose these notes"

The other major reason I observed was due to shortage of devices, since the devices were used in groups students didn't get an opportunity to use it to its full capacity. But students showed their interest in this feature and said if such features are available on their own devices, it might be a different scenario for them as well.

On Device Slide View

Slide view on device proves a very interesting feature

On device view was one of major features offered by NoteCarrer , and students used it very extensively during the whole exercise. I observed that students lose eye contact with the lecturer while using their own device to view slides. Which is not a good approach during a lecture. I think it would be the same practice for student to look at projector screen or a device. This activity might be influenced by the common human nature, in which a person gets more attracted by new things at the start. Further case studies may helps us to investigate this behavior in more detail.

Posting Questions and Marking slides Unclear

Is discussion is better than posting questions ?

In NoteCarrier, students can post questions using smart phones, In this user study many students utilized this option during this session. It was a nice surprise for students when the lecturer start addressing their questions during the lecture. Many students also used the option "Mark

Unclear" during the lecture to inform the lecturer that the current content was not clear to them. The lecturer also responded to these requests by explaining the contents of slide.

Post interviews also exhibits students high interest in these two feature as compared to other features offered by the application. Many students admitted that they feel hesitant to ask questions during the lecture and the application could be a suitable solution for them to post their questions.

In contrast to these student, some students totally denied the need of such system. They explained, its much easier for them to ask questions directly to the lecturer rather then writing it down. Also asking questions directly often leads toward a proper discussion which is more beneficial for them in solving their confusions.

7.5 Results

Questionnaires and post interviews are conducted at the end of session

To evaluate the application in a more detail way, questionnaires were distributed to the students and lecturer. Students were asked to explain their experiences with the application and provide suggestions and improvements for the NoteCarrier application. The result of this activity not only helped us assess the current features of the application but also made it easy for us to judge the productivity and usability of the application. All of the participants also suggested some improvements which might help them in their learning process.

In the questionnaire, some questions were added to gain some knowledge about student activities during the lecture. The purpose of these questions was to collect some data about those activities in which NoteCarrier can provide some assistance. The gathered information will be useful to determine the target group of our user study.

For a better understanding, summarized information is shown in the form of graphs. This would help us to focus on those features which are most commonly used by stu-

Questions	Available Options			
How often do you take notes during lecture ?	Every Time	Some Times	Rarely	Never
Which medium do you use for writing down notes	Printed Slides	NoteBook	Plain Paper	Not specific.
Is it easy for you to ask questions during lecture	Yes	Sometime	I feel shy to ask	Never.

Table 7.1: Questionnaire : Some selected question related to Student Activities

dents during a lecture. The information can also be used to determine the overall usability of the NoteCarrier application.

Graph 7.4 shows, student interest in taking notes during a lecture. As the graph shows more than 75% students take notes during a lecture with according to their needs. Since NoteCarrier provides an option to the student to take important notes on the slide, this feature could can be utilized by more than 75% students in a class.

Since majority of students selected *Rarely* as an option for taking notes, NoteCarrier might be a good option to note down very few notes over the slide rather writing on paper.

A high percentage of students are hesitant to ask questions during a lecture

Graph 7.5 depicts another interesting fact about asking questions during a lecture. Approximately 50% of the students feel hesitant to ask questions during a lecture. NoteCarrier might be a very suitable solution for them to handle this problem. The application provides different ways to notify the lecturer for questions and unclear content of the lecture. The post question feature encourages students to ask questions during the lecture. Very productive results can be achieved with NoteCarrier by raising this percentage

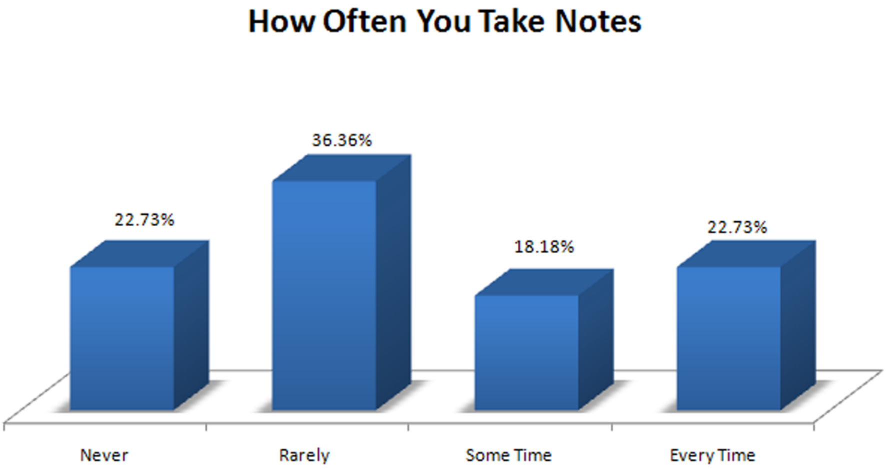


Figure 7.4: User Study : Shows the percentage students take notes during lecture

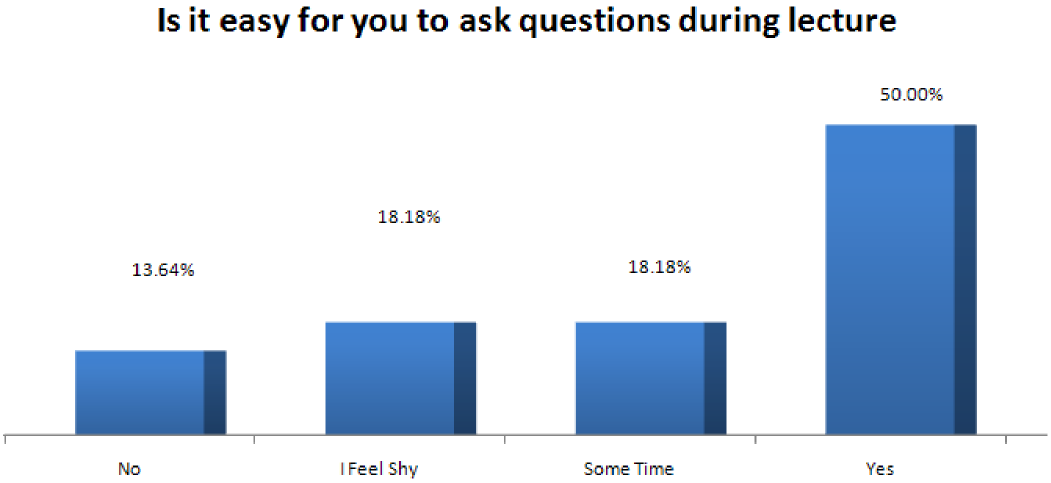


Figure 7.5: User Study : Shows the percentage students comfortable in asking questions during lecture

In addition to get student activities information, the questionnaire also includes some post study questions about the application. Most of the questions are scaled base questions to find out the student evaluation of specific feature of the application. These results show the students positive feedback towards the application. The suggestion and comments sections also depict student's high interest in the application which indicates the need for such an application specifically for the audience of the lecture.

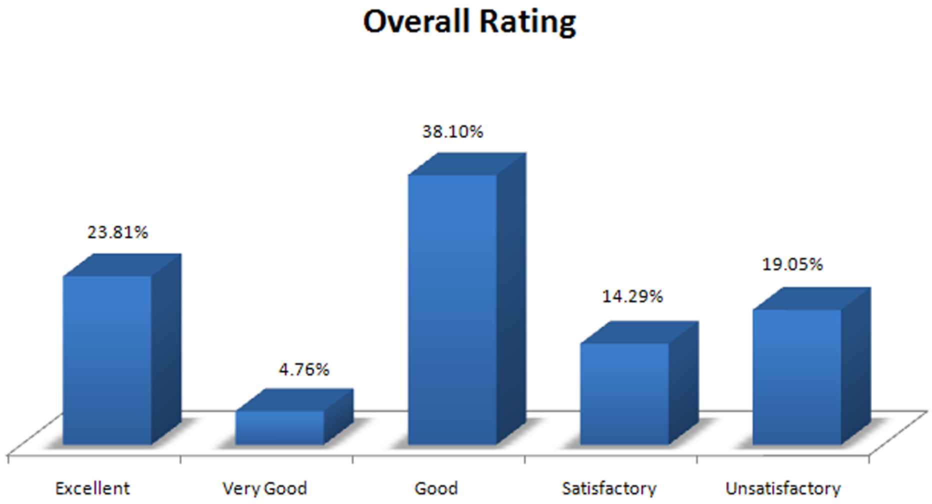


Figure 7.6: User Study : Overall rating for NoteCarrier application

Graph 7.6 shows an overall ranking for the NoteCarrier application by students. The result depicts that more than 66% students liked the overall idea behind the application. Many students agreed that it would be suitable solutions for better class participation and encouraged them to keep their interest in the lecture. As result shows in 7.5 that many students still feel hesitant in asking question during lecture, so application can be used as an alternative source for communication. The post interviews also helped me to closely observed those students who feel uncomfortable during class participation and considered the NoteCarrier a convenient solution for their problem.

Multiple user studies are needed to evaluate the application

Since it is not an easy job to evaluate the overall application progress with one user study, I strongly believe that more user studies are needed to truly assess the application strength. But still I tried to investigate the issues and problems users faced during this session. Many students who marked the application unsatisfactory thinks that its has very limited audience and only addresses those student who owns expensive Iphone mobiles. They suggested that system should have capability to integrate laptops as optional client device. Some other students also states that its easier for them to start discussion directly with lecturer rather using third medium for communication as there is chance that they might not be able to explain the question properly.

Due to lack of user studies, I am also unable to completely figure out the actual impact of the application on overall student's class participation. To support my hypothesis, I need to do multiple user studies to capture the difference made by the NoteCarrier. Since I was unable to collect multiple user studies due to semester break, in this case it is very difficult for me to exactly measure the difference made by the application in class participation activity. In current situation I can only mention those questions which are reported by the NoteCarrier but there is possibility that the questions count will be the same as compared to those lectures when students don't have the application as an option. I will recommend to have more case studies in future to clearly identify the influence of the application on student participation.

Despite of evaluation of the application I was also able to collect the student's general idea about application and suggestions for new features and enhancements. The following suggestions are mentioned by students and the lecturer with respect to their operating mode.

7.5.1 Suggestions and Improvements

The following section includes two sub sections to cover all the suggestions related to the *Student Mode* and the *Lecturer Mode* of the application. A detail overview of these features are discussed in 8.2.2—"Feature Enhancement"

Lecturer Mode

- An off-line view mode for the lecturer, so the lecturer can go back and forth in a slide without interrupting the projector view
- Provide a mechanism to use the device as pointer
- Allow the lecturer to write on the slide view which is available to students in real time

Student Mode Mode

- Landscape view of slide should be available in both the off-line and the synchronized view
- An option to jump to the current slide in Off-line Mode
- Passing Information back to the Lecturer
- Discussion Board for students

Chapter 8

Summary and future work

"So many of our dreams at first seem impossible, then they seem improbable, and then, when we summon the will, they soon become inevitable."

— Christopher Reeve

The communication gap and minimal students participation are major problems of today's classroom infrastructure. New technologies come with new potentials to face such challenges in our routine life. NoteCarrier provides a new concept to approach these traditional problems involved in the classroom environment. The thought behind this project is to enhance the communication channel between the the students and the lecturers and encourage students for active participation in the class with the support of those tools and technologies which are already tightly integrated in current environment.

In following chapter, I will summarize my whole work on the NoteCarrier application and the evaluation results we achieved by this project. I will also provide some base lines for the future work and motivation for those who want to enhance this project for further development.

8.1 Summary and contributions

NoteCarrier promotes a two way communicate channel between student and lecturer

The main purpose behind this thesis is to research and present a solution for the classroom environment which will truly focuses and provides a two way communication channel between the students and the lecturers. With the growing size of classes, the teachers notices a negative trend in student's class participation. This minimal participation not only influences the overall learning process but also makes the whole lecture senseless.

NoteCarrier is presented as a solution to avoid such a situation and motivate the students to actively participate in lectures. The first step towards this goal was to investigate and analyze those systems which are already developed to overcome the same underlying issue.*Ubiquitous Presenter* by Michelle Wilkerson and Simon and *Active Campus* by William G. Griswold and Boyer are a few examples of such projects covering the same domain. More such projects and related work is presented in the section 2—"Related work" .

The architecture design of the application, mainly took it one step ahead of other available solutions. In contrast to other available solutions, NoteCarrier does not require any own created presentation tool or software rather it provides support for those tools currently used for presentations.

After investing a reasonable amount of time in the research phase, efforts were made to set achievable standards and goals for the application. To keep the distinction between NoteCarrier and other available solutions, the application design and goals were finalized with the idea that NoteCarrier should provide a solution which is more practical and usable in to real environment. I wanted to overcome those development issues which still make these applications hard to use in a real environment.

Storyboards are used to demonstrate the need of application in real scenarios

Later on, to illustrate the proposed solution within its specified context, storyboards were created on the basis of those scenarios that I wanted to address in this project. These storyboards acted as a first prototype of the project and were presented to the target audience of the project to determine the students' first impressions about the idea behind this application. To show the distinct impact of NoteCarrier, two sets of storyboards were formulated. One set of storyboards depicts the current situation in the classroom when the students and the lecturer are not equipped with NoteCarrier. The second part of the storyboards introduces NoteCarrier classroom environment. With this approach, I was able to portray the drawbacks attached to the current process and what difference NoteCarrier can make in such situations. A noticeable interest was exhibited by the students who evaluated these storyboards.

Paper prototype is used to illustrates initial design of the application

In the next step, I created a paper prototype for the NoteCarrier application to judge its usability at an early design stage. This paper prototypes consisted of hand drawn prototype for the IOS application. Because the server application, AppleScript are used as background processes, these components were not included in this prototype. In this prototype, students were presented with a hand drawn user interface of the IOS application. Small menus and icons were also added in this prototype to analyze the user interaction techniques with the system. To handle Nomadic operations, I used sticky notes to handle the information sharing operations of the application. Paper prototypes were also evaluated with students in a controlled environment. The students played a vital role in this session and exhibited their interest in the application by their positive feedback and suggestions.

Software prototype is implemented for Mac machines and IOS devices

With the confidence I gained from the positive feedback, I took the application into its next phase in which I developed the first software prototype of the NoteCarrier application. NoteCarrier consists of three development components: two IOS based applications for students and lecturers, a cocoa based application server for the lecturer mac machine and a set of AppleScripts deployed on the application server which forms a communication bridge between the NoteCarrier application and the Keynote presentation tool.

Asynchronous network connection was used as communication channel between the IOS based applications and the NoteCarrier application sever deployed on the mac machine. The software prototypes were implemented into different small development cycles with a continuous feedback approach. These small development life cycles not only improved the overall quality of the application but also helped me to integrate the user suggested features during development.

Evaluation results confirms the need of NoteCarrier in real environment

To evaluate the NoteCarrier in a real classroom environment, I conducted a user study with the help of the students and the lecturer at University Of Bonn. The students were provided with needed devices during the class so they could use them freely during the lecture. Questionnaires and interviews were conducted with the students and lecturer at the end of the evaluation session. Evaluation results confirms that NoteCarrier can be a good option for the students as well for the lecturer to enhance the communication during the lecture. It will definitely be a good choice for those students who are hesitant to participate in a lecture and the application will encourage them toward active participation.

My research contributions is a successful demonstration of a system for classroom environment

The contributions I made via this thesis verify that no doubt NoteCarrier is a suitable application for the current classroom environment. Its capacious architecture not only covers those activities which are related to students but also facilitates lecturers delivery. Its design allows NoteCarrier to support available presentation tool, which makes this system more adoptive for the environment. The user's interest and satisfaction with application's performance also show a positive impact of NoteCarrier on the overall learning process. Specifically the evaluation session proves that its integration into the current classroom environment will encourage students to actively participate and decrease the communication gap between lecturer and students. With the overall NoteCarrier performance and evaluation results, I can conclude that in this thesis I have accomplished a major part of the goals mentioned in the section 1.1—"Research Goals" But still due to lack of user studies, I am unable to investigate all research goals. I believe, there is more room available for improvements and to make application more productive. As software prototype of application is developed to demonstrate the idea so it is not in very refined state as it should be. Further development is needed to improve its current user interaction techniques and to make application more easy to use. User evaluations also shows student lack of interest in writing down notes, as mentioned in previous chapter one possible reason might be the ownership of the device. As students did not want to lose their notes, so they used their writing pads to note down their notes. Some students also said that for them its more easy to keep notes on one place rather writing notes on the different slides. In presence of projector screen, synchronized slide view did not gain that much attention as expected. In my of view more user studies might be very useful to extract more about students behavior and to judge application creditability.

In this thesis I tried to introduce a new approach to integrate current presentation tools into the classroom system. The idea is supported by its limited implementation in the form of NoteCarrier application. I am sure that further development using this approach will improve its credibility and enhance the learning process for the target audience in a real environment.

8.2 Future work

NoteCarrier was developed by following DIA (design, Implement and Analysis) iterative approach, each cycles ended with an analysis phase in which I evaluated the whole application with a group of people. During the evaluation process, users presented different ideas and suggestions for the improvement of the system. In the following section, I will present some of the enhancements that we can make in the future

8.2.1 Enhancement Support

In the current software prototype of NoteCarrier, the support for the software and hardware is very limited. To maintain the adaptability of the application, further development is required to support additional hardware and software systems.

A number of enhancements can be made in an iterative way to broaden the scope of the application in terms of software as well as hardware. The goal can be achieved in successive iterative steps to add support for further devices or tools one by one. Following are some examples:

Support For PDF Presentations

The current version of NoteCarrier, can only communicate with a Keynote presentation tool via AppleScript. Since PDF presentations are also extensively used in many lectures, writing AppleScripts for communicating with PDF tools might be a major step towards the goal. With this additional support, NoteCarrier will be able to cover a large range of presentations already prepared in PDF formats.

Web Based Clients

Another interesting feature might be adding support for web based clients. This additional feature will allow students to interact and communicate with the NoteCarrier application server via a web browser. This will allow those students to interact and participate in the class who don't own an IOS device. Since most of students already carry their notebooks to the classroom, it will be an good option to enable such student to communicate via their notebooks.

Microsoft Technology Support

Another milestone would be providing support for Microsoft Technologies. Currently NoteCarrier is only compatible with Mac OS X, since a large percentage of users use Windows based machines, providing support for Windows based systems would be a big plus for the application.

8.2.2 Feature Enhancement

The evaluation phase at the end of each prototype not only improved the overall application but also provided an opportunity to collect valuable suggestions from the user's perspective. At the end of the software prototype evaluation, questionnaires and interviews were conducted to gather more ideas for the NoteCarrier application. In section 7.5.1—"Suggestions and Improvements" a list was prepared with a high priority feature list recommended by the students and lecturer.

For instance, adding an of off-line view beside the synchronized view in the lecturer mode will be a handy feature for the lecturer. Although NoteCarrier provides an off-line slide view in student mode but this feature is currently not available under the lecturer mode. With this additional feature, the lecturer can jump between slides without disturbing the projector view. This new feature will be very beneficial in situations when the lecturer want to look at upcoming slides without altering the projectors view.

Off-line slide view will help lecturer to look next coming slide without interrupting the projector view

Another useful feature in the lecturer mode might be to add a laser pointer function. In user study I observed that the lecturer used external laser pointers to point to specific areas on the slide view. It might be hard for the lecturer to mange multiple devices during a lecture, As a solution, development can be made in a way that lecturer can point out specific areas by using a finger on the smart device. That would eliminate the need for using an external laser pointer during the lecture.

Adding ability to use lecturer IOS device as a laser pointer

The students and lecturer also suggested implementing a mechanism in which they can share the notes with each other. Since NoteCarrier already provides a feature for taking personal notes on the slide view, it would be more effective for the students and lecturer if application was equipped with a sharing notes feature.

Sharing personal notes with other client devices

8.2.3 Additional User Studies

No doubt user feedback and suggestions play a vital role in application development. Although NoteCarrier was developed with an iterative approach with the involvement of continuous user feedback and proper evaluation sessions, Improvements are needed to make the application more stable. One option for making an application more stable and reliable would be to invest more time in the user studies in an actual environment. Conducting multiple user studies with heterogeneous groups of students would help to improve the overall application performance.

Examining the application for a longer period will also help to verify its usability. The extracted results of these case studies will help us determine its usability in actual classroom environment and to assess its overall impact on the learning process. The invested time in these studies will provide a sufficient base to determine its influence over student progress and especially for those who are hesitant to participate.

The current prototype of the application is developed with attention to classroom scenarios, but the domain of this application can be extended by involving presentations other than the lectures. Presentations are widely used in many other scenarios such as offices, group discussions and marketing etc. Efforts can be made to conduct user studies for these different scenarios and to extend NoteCarrier according to new domains.

8.2.4 Nomadic Operations

At this stage the nomadic operations in the application are not mature enough. There is plenty of room available to improve the current Nomadic architecture of the application. In order to make the application more generic and to make it compatible with web based clients, the network architecture of the NoteCarrier needs to be revised accordingly. The request and response formats also need to be more standardized into proper formats e.g. XML or JSON.

In addition to the current nomadic operation, the new nomadic feature can be introduced into the current application. Sharing notes and student discussion panels were already suggested by the students and lecturer in the evaluation session. Investigating more user studies can help us to figure out valuable possible features in this domain.

Implementing a load balance to improve application server's performance

Suggesting an effective solution for network load sharing will be another interesting research point for this application. In the current implementation all networks calls are handled by single server instance which might cause a bottle neck in case of active participation from larger groups of students. Investigating a solution for handling large network traffic with multiple servers or with help of peer devices will improve the application response time in a dramatic way.

8.2.5 Form Factors

The NoteCarrier software prototype was developed for a very limited set of hardware. Ipads and Iphones are the only potential client devices at this stage. It would be an interesting research field to add additional support for other devices in NoteCarrier.

Laptops and notebooks are the most commonly used devices in the current era. Extending NoteCarrier for laptop client support would be a major achievement towards application adaptability in terms of available hardware support. This exciting feature would definitely increase the user participation from those students who own a laptop instead of an IOS device.

I hope with new coming technologies, we can extend the idea of the NoteCarrier application in a more systematized way. I am looking forward towards enriching NoteCarrier into the current classroom environment so it will find itself an obligatory tool for all lecture rooms.

Bibliography

J Brooke. Sus-a quick and dirty usability scale.

Dr. Chris Buckalew and Alan Porter. The lecturers assistant.

TAYSHENG JENG. Toward a ubiquitous smart space design framework. *JOURNAL OF INFORMATION SCIENCE AND ENGINEERING*.

TAN MINH TRUONG MATT RATTO, R. BENJAMIN SHAPIRO and WILLIAM G. GRISWOLD. The activeclass project: Experiments in encouraging classroom participation.

William G. Griswold Michelle Wilkerson and Beth Simon. Ubiquitous presenter: Increasing student access and control in a digital lecturing environment.

Buntarou Shizuki Motoki Miura, Susumu Kunifuji and Jiro Tanaka. Augmented classroom: A paper-centric approach for collaborative learning system.

Jakob Nielsen. Iterative user interface design. *Originally published in IEEE Computer*, 26(11):32–41, 11 1993.

Mark Weiser. The computer for the 21st century.

Steven W. Brown William G. Griswold, Patricia Shanahan and Robert T. Boyer. Activecampus - experiments in community-oriented ubiquitous computing.

Steven W. Brown Robert T. Boyer Matt Ratto R. Benjamin Shapiro William G. Griswold, Patricia Shanahan and Tan Minh Truong. Activecampus - experiments in community-oriented ubiquitous computing.

Index

Typeset September 3, 2012