Ubiquitous and Mobile Computing
CS 528: LockNote

Boya Zhou, Chu Wang, Jin Huang, Roger Wirkala, Yuhan Liu

Computer Science Dept.
Worcester Polytechnic Institute (WPI)
The Existing Inconvenience

• Hard to write down the notes on the board quickly
• Hard to record the key points according to the speaker
• Hide some password or other tips in mobile phone during a class while listening the speech for own use
Motivation

• Improve the efficiency of making plan and taking notes
• Note taking is necessary, make note-taking easier
• Helps forgetful people or those with poor memory
• Helps people who is used to misplacing the stuff (Hand written notes are easy to lose)
Why solve it through mobile app

- Students do not always take PC everywhere
- The weight of mobile phone can be ignored
- Non-mobile devices cannot record media as easily
- Much more convenient to take photos on mobile phone
- Mobile devices are more inconspicuous
Methodology

• Securely store passwords and record notes in various formats (e.g. text, picture, audio)

• Require user to sign in with password/social media accounts (facebook, google, twitter, etc). Extra layer of encryption for password and confidential notes.

• Take audio notes or pictures and transcribe them to text files in app (media recognition)

• Search through notes by keywords, e.g. time, location (#wpi#)

• Organize notes by date, location

• Event reminder (exam dates, interviews, birthdays)
Difficulty Points

• **Level 1:**
  - Every 5 android screens, (4)
  - Audio playback, (4)
  - Maps, (4)
  - Location sensing, (4)
  - Camera (4)

• **Level 2:**
  - Speech recognition, (6)
  - Media recorder (6)

Total difficulty points: 32
FirebaseUI is a library built on top of the Firebase Authentication SDK that provides drop-in UI flows for use in app directly.
Local Database for note security
Database—Firebase Database

Store and sync data with NoSQL cloud database. Data is synced across all clients in realtime, and remains available when the app goes offline.
ML Kit's text Recognition

Cloud Speech-to-Text

Quiz!!!
Quiz!!!
OMG!!!
QUIZ AGAIN!!!

Quiz!!!
Quiz!!!
OMG!!!
QUIZ AGAIN!!!
Timeline:

**Week 1**: Complete the design of the UI interface and understand the functions and implementation methods of each API.

**Week 2**: Complete the login function using firebase authentication and design local database to implement note locking function.

**Week 3**: Complete the design of the database using firebase real time database and implement the functions of the corresponding page.

**Week 4**: Using ML kit’s text recognition and cloud speech-to-text API to complete text conversion of voice and pictures.

**Week 5**: Complete the assembly of the various parts of our application.

**Week 6**: Test the whole app and finish the final report and preparation for the final presentation.
Related Works

- Aim to combine the best features from various note taking apps and relevant APIs
- Evernote
  - Capture and convert verbal notes
  - Uses SSL and AES-256 encryption
- Google API used in Word Lens to detect text in images
Related Works

- Take aspects of smart albums from Google Photos to organize notes
Evaluation Plan

● User testing will be valuable to gauge this application
● Start with peers, transition to real world use in classroom setting (3-5 students per test)
● Focus on task completion within the app
● For helpful results and feedback you need to test the system, not the users
● Carefully observe reactions to “smart” features of the app that are executed without much user control