IRIS

SWEG Capstone 2016

Project Members

- Scott Arnette
- Joseph LaCava
- Derek Ouzia
- Bryan Smith

Project Overview

- Iris is an application that will provide users the ability to be read emails received to the Google Accounts they choose to use during application use on a device running the Android operating system.
- After being read an email, users will additionally be prompted whether to keep or delete the previously read message.

Project Planning

- Project Development Plan.
- Team Dynamics.
- Project Documentation.
- Project Risks.
- Project Reporting.

Project Development Plan

- Software Development Life Cycle
  - Planning.
  - Analysis.
  - Design.
  - Implementation.
  - Maintenance.

Team Dynamics

- Work Design
  - Autonomy.
  - Skill Variety.
- Composition
  - Ability.
  - Personality.
  - Roles and Diversity.
  - Size.
  - Flexibility.
- Content
  - Adequate Resources.
  - Leadership.
  - Climate of Trust.
- Process
  - Common Purpose and Goals.
  - Confidence.
  - Very low conflict level.

Project Documentation

- Project Vision and Scope.
- Software Requirements Specification Document.
- Software Design Document.
- Source Code.
- Software Maintenance Records.

Project Risks

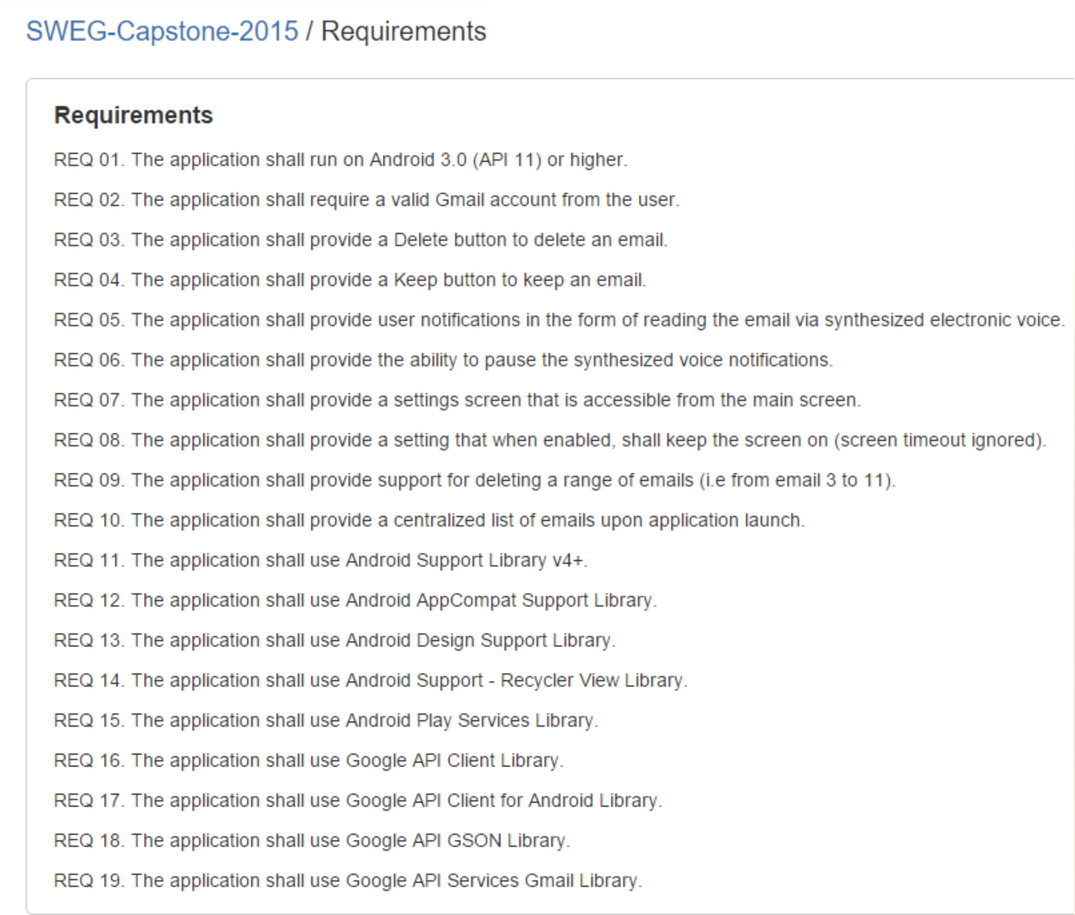
- Identification of Project Risks.
- Qualitative and Quantitative Risk Analysis.
- Risk Response Planning.
- Risk Monitoring and Control.

Project Reporting

- Weekly Status Report.
- Communication of project status to the project manager.
- Identification of issues with the project.
- Prompt planning and action.

Requirements Elicitation

- Client wanted Apple App
- Wanted buttons to keep or delete a message as it was read.
- Ability to stop or start the service.
- Determine APIs and libraries needed.
- Determine the minimum version of Android that supported APIs and libraries needed for development.



SWEG-Capstone-2015 / Requirements

Requirements

REQ 01. The application shall run on Android 3.0 (API 11) or higher.

REQ 02. The application shall require a valid Gmail account from the user.

REQ 03. The application shall provide a Delete button to delete an email.

REQ 04. The application shall provide a Keep button to keep an email.

REQ 05. The application shall provide user notifications in the form of reading the email via synthesized electronic voice.

REQ 06. The application shall provide the ability to pause the synthesized voice notifications.

REQ 07. The application shall provide a settings screen that is accessible from the main screen.

REQ 08. The application shall provide a setting that when enabled, shall keep the screen on (screen timeout ignored).

REQ 09. The application shall provide support for deleting a range of emails (i.e from email 3 to 11).

REQ 10. The application shall provide a centralized list of emails upon application launch.

REQ 11. The application shall use Android Support Library v4+.

REQ 12. The application shall use Android AppCompat Support Library.

REQ 13. The application shall use Android Design Support Library.

REQ 14. The application shall use Android Support - Recycler View Library.

REQ 15. The application shall use Android Play Services Library.

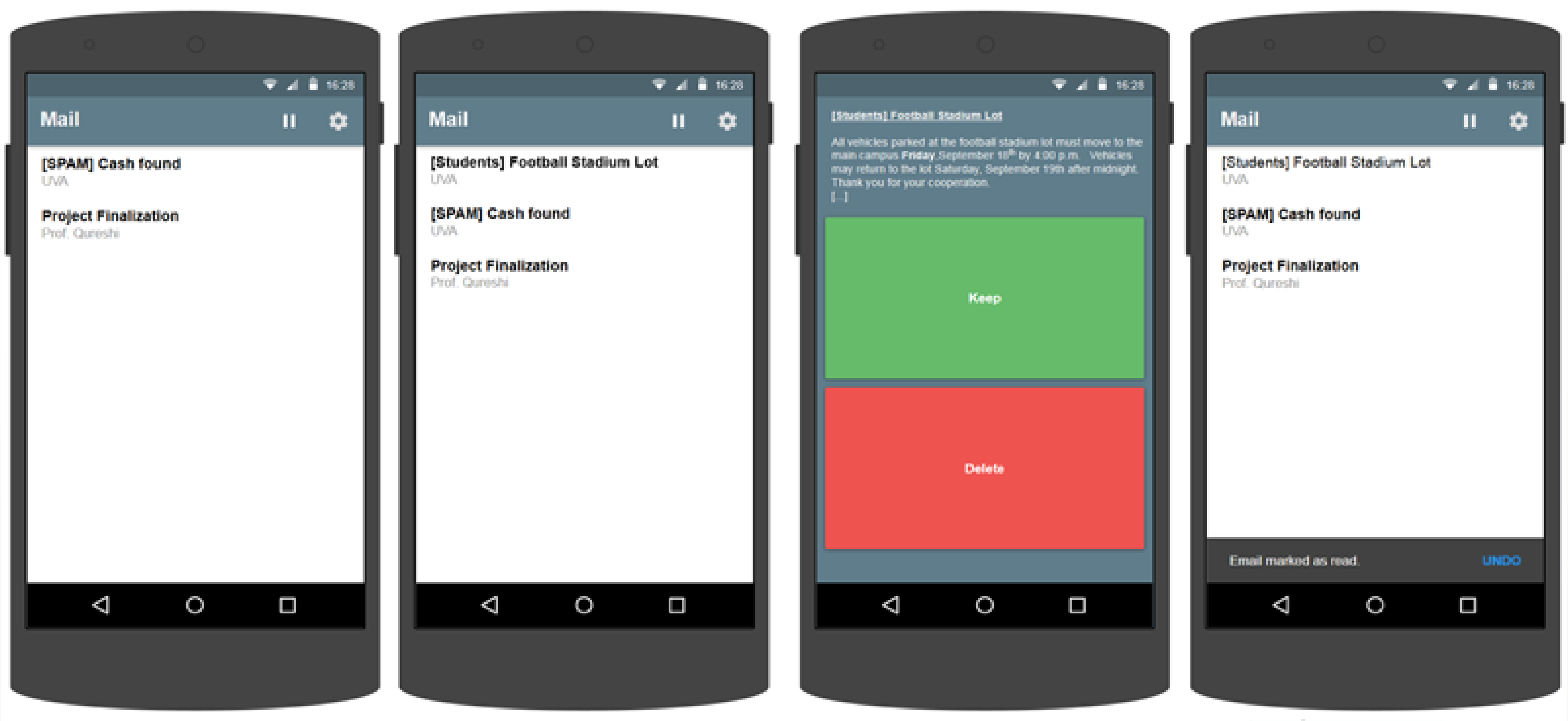
REQ 16. The application shall use Google API Client Library.

REQ 17. The application shall use Google API Client for Android Library.

REQ 18. The application shall use Google API GSON Library.

REQ 19. The application shall use Google API Services Gmail Library.

Design - 1st Semester



[SPAM] Cash found
UVA

Project Finalization
Prof. Qureshi

[Students] Football Stadium Lot
UVA

[SPAM] Cash found
UVA

Project Finalization
Prof. Qureshi

[Students] Football Stadium Lot

All vehicles parked at the football stadium lot must move to the main campus **Friday**,September 18th by 4:00 p.m. Vehicles may return to the lot Saturday, September 19th after midnight. Thank you for your cooperation.
[...]

[Students] Football Stadium Lot
UVA

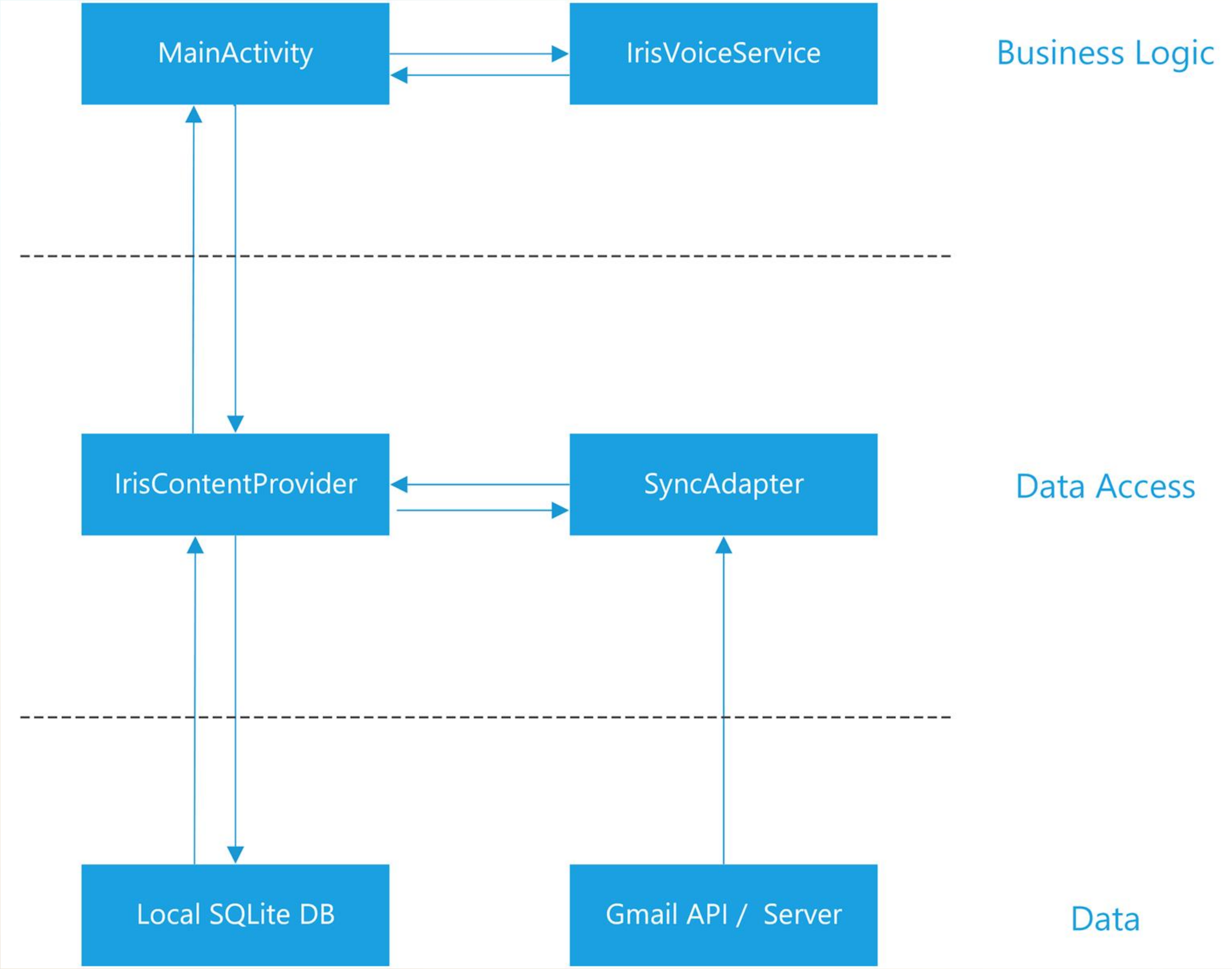
[SPAM] Cash found
UVA

Project Finalization
Prof. Qureshi

Email marked as read.

System Architecture

- Divided into four main parts: business logic, data access, data, and views. Three of those parts are visible in the diagram here.
- **Business logic:** What will the app do with data or input it receives?
- **Data Access:** How will the application receive and manage data and input?
- **Data:** Local and remote data sources or stores.
- **Views:** How is the data displayed to the user?



Development - Technologies

- **Java** - Main Programming Language
- **XML** - Extensible Markup Language defines the UI views.
- **SVG** - Scalable Vector Graphics for our images
- **Android SDK** - Software Development Kit that integrates with Java to allow execution on Android Devices

Development - Libraries

- **Gmail API for Android**
- **Google Play Services v8.3**
- **Google Identity Library** (Dependence of Gmail API)
- **Javax.Mail** - Allowed for easily parsed email headers (find the sender etc)
- **Android Support Design Library v7** - Allowed backwards compatible design on older versions of Android
- **Android App Compat v23** - Allowed backwards compatible features on older versions of Android

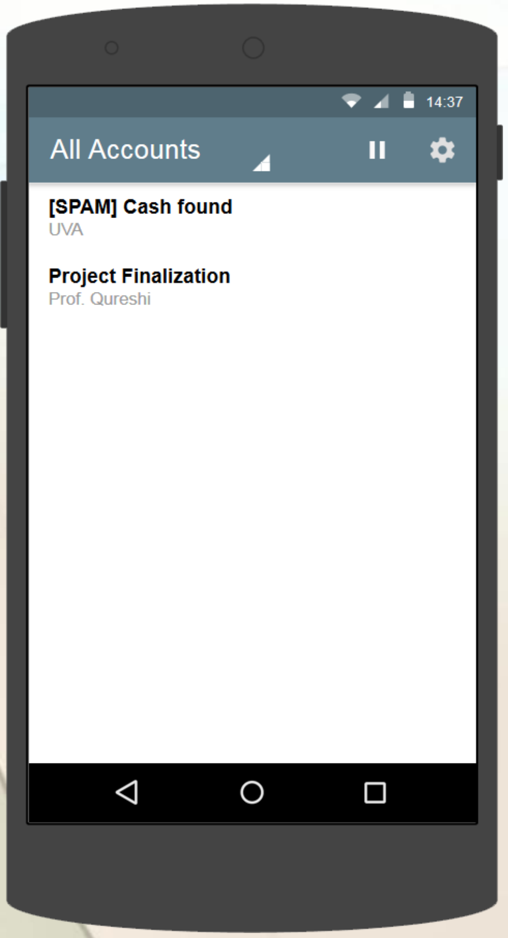
New This Semester

- Reviewed Project Schedule
- Reviewed Project Requirements
- Reviewed Project Risks
- Add new feature requirements
- Update design to accommodate new features

New Features

- Support multiple Gmail Accounts
- Second battery-saving sync frequency
- Improve syncing logic
- Bug Fixes

Design - 2nd Semester

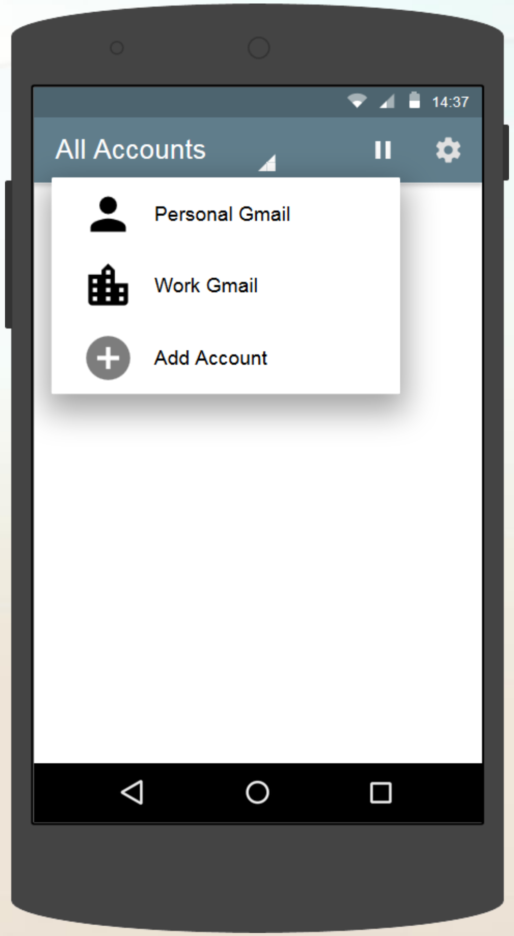


14:37

All Accounts

[SPAM] Cash found
UVA

Project Finalization
Prof. Qureshi



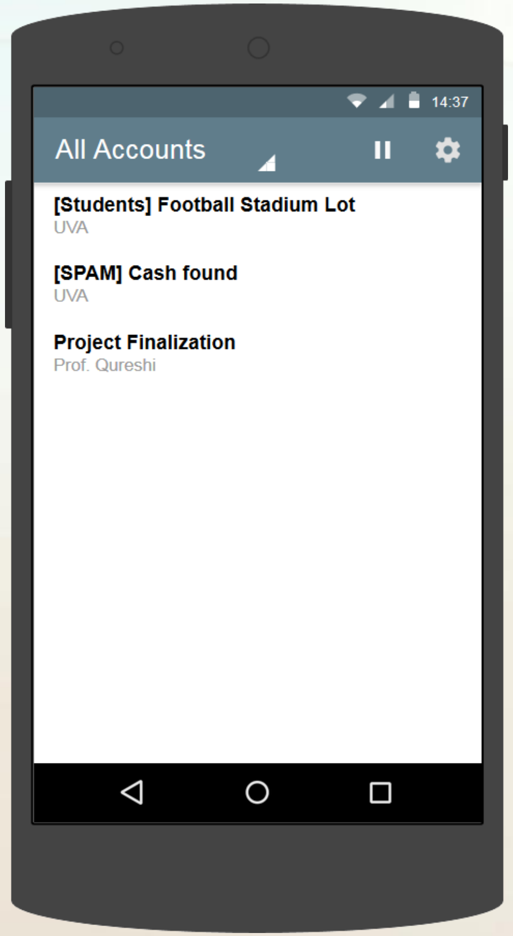
14:37

All Accounts

Personal Gmail

Work Gmail

Add Account



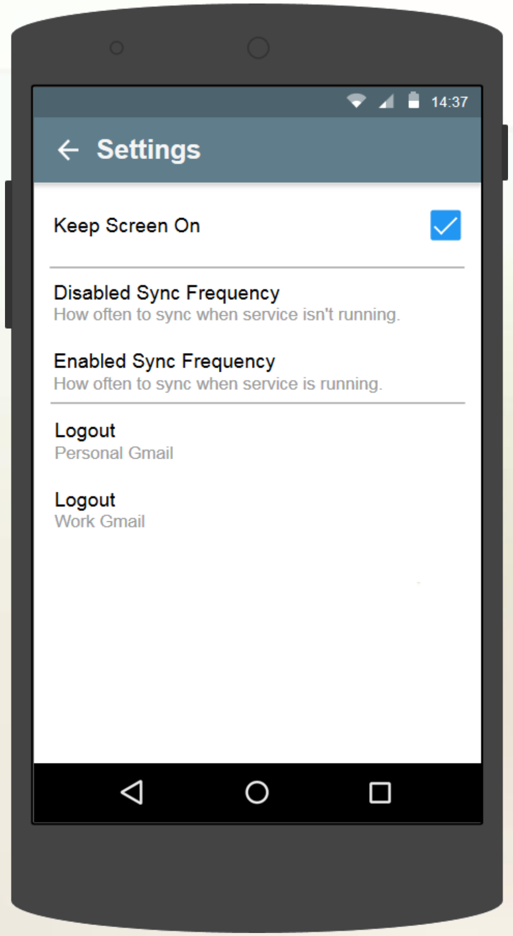
14:37

All Accounts

[Students] Football Stadium Lot
UVA

[SPAM] Cash found
UVA

Project Finalization
Prof. Qureshi



14:37

← **Settings**

Keep Screen On ☑

Disabled Sync Frequency
How often to sync when service isn't running.

Enabled Sync Frequency
How often to sync when service is running.

Logout
Personal Gmail

Logout
Work Gmail

Development - New Semester

- Determine what would need to be changed, added, or removed to accommodate our new requirements.
- We assumed most of the email sync logic would need to be re-designed to work with multiple accounts.
- As we got into implementing multiple accounts, we found that we would be able to leave most of the sync code as-is. Android handled it itself smoothly.
- Bulk of the changes were to the UI to accommodate logging in, out of, and selecting the multiple accounts.

Testing

- Bulk of testing was from an independent testing team who were taking the Software Engineering course, Software Testing and Verification.
- They started testing as we were finishing up development.
- We followed up with our testing after resolving their findings, as to avoid duplicates.
- After the independent testing team had executed their testing and those issues were resolved, the rest of the issues we found were mostly cosmetic UI issues.

Project Progression

Task	Duration	Date to be Started	Date to be Completed
Review and Requirements	15 days	1/12/2016	2/1/2016
Design/Risks	10 days	2/2/2016	2/15/2016
Development	25 days	2/16/2016	3/28/2016
Testing	15 days	3/29/2016	4/18/2016
Demo/Papers/ Etc.	6 days	4/19/2016	4/26/2016

- Application enhancement and testing is completed.
- Project as initially defined by requirements is complete, with some features dropped due to time constraints.

Configuration Management

- IBM ClearQuest Bug Tracking (Shared with independent testing team)
- BitBucket as the code repository
  - Contains branches, wiki, change management, and issue tracking all in one location

Atlassian

Bitbucket

Deployment & Maintenance

- Published to the Google Play Store pending UVA Wise Approval
- Can also be installed via APK file if the Google Play Store is not available.
- No current plan to have continue support unless expanded on or maintained by a class.
- Documentation has been developed giving detailed notes on requirements for distribution and updating.

Social Impacts

- Can help users maintain important focus while simultaneously alerting them to new emails.
- However, Iris still requires some touch interaction and should be used with caution.
- Users should of course abide by all applicable laws (I.E Driving)
- Email messages are never permanently deleted by Iris. (Only ‘trashed’)
- Iris never has access to the user's password for their Gmail accounts. (Android OS Handles the account credentials)

Questions?