**Spartan Drive - Project 3 (Canvas)**

Ankit Devani, Aparna Gupta, Fareen Magdum, Mario Madan Vinay Prakash, Rashmi Hadli, Roshan Srinivas

**Abstract**

With the rapid adoption of cloud computing, it has quickly become the standard for hosting applications, services, data, environments and more. Mobile is another area that has seen tremendous growth and innovation in the last decade. These two technologies along with improvements in communication technologies have enabled newer markets and business opportunities.

Cloud storage is one such area that has seen tremendous growth and innovation. On the enterprise side, cloud has become the de-facto standard. On the consumer side, adoption of cloud for storage is increasing. Mobile applications such as Google Drive, DropBox, Box, OneDrive, and iCloud Drive have enabled consumers to store and share large amounts of data. These applications are available on mobile platforms such as Android, iOS, Windows Phone and Blackberry.

Spartan Drive is an android application that aims to simplify managing and sharing files among users. It provides a user friendly interface and functionality that is not available or easily accessible in existing cloud storage mobile applications. Spartan Drive uses Google Drive as the backend storage and enables a user to easily upload, view information and share files with other users. Some features provided uniquely by Spartan Drive include usage reports, push notifications to alert users when files are shared with them and simplified folder management.

**Features**

1. Login with social platforms - Google
2. File upload. A user can upload files into any folder
3. Folder management
   1. A folder will have a title
   2. A user can create, update, and delete folders
4. File information
   1. File name
   2. File description
   3. Supported file types, use different icon for each type

doc, xls, pdf, ppt

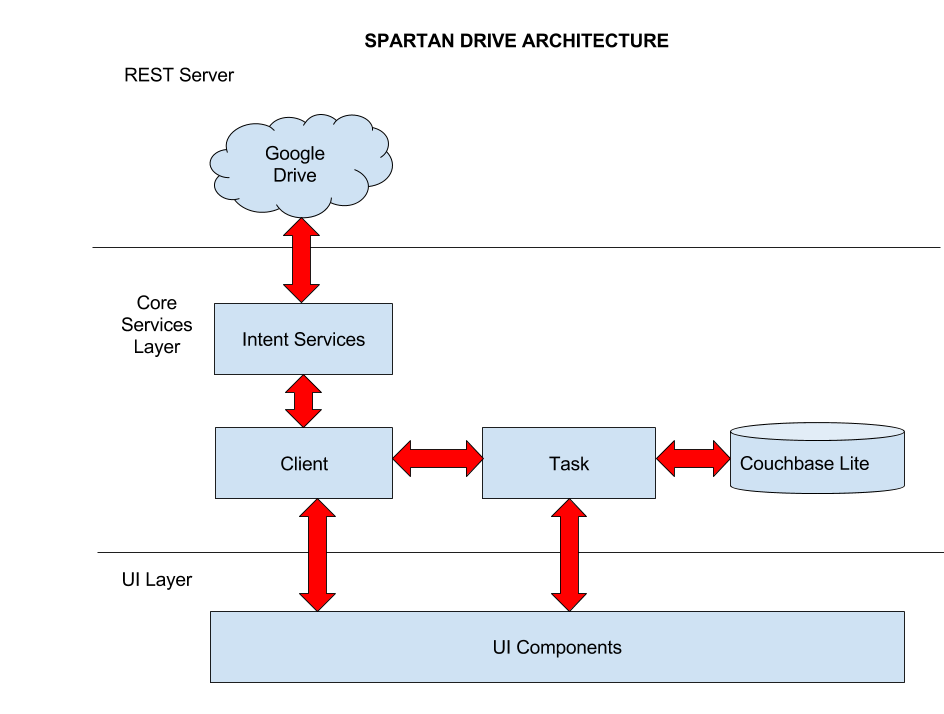
1. File sharing
   1. Share and unshare a single file (read-only)
   2. Share the whole folder
2. A user can search files (owned or shared) by name and/or text in description
3. A user can browse files(owned or shared)
   1. Browse list of folders and list of files within a folder
   2. Supports icons for folder and different file types
   3. View each file(only the supported types)
   4. Email the current file
   5. Navigate into any level of directories one by one
4. Usage report: the user will be able to see how many files and folders he owns, and their total size.
5. Push notification: when a file is shared with you or unshared with user, user will get a push notification.

**Technology Stack**

The application is divided into two layers Frontend and Backend or as we call them -

* **UI Layer** - Android
* **Core Service Layer** - Java
* **Google API​**: For User Login
* **Couchbase Lite** - Persisting Folder and File Structure
* **Google Drive** - For storing files

**Architecture Diagram**



We will create the core services layer as a library project and the UI as a main project. This helps a lot to separate these two layers and to distribute the work.

On the UI layer we get everything in charge of managing the interface: activities, fragments, views, adapters, etc. All of them are data consumer objects.

On the other side, on the core services layer (the one with the main part) we get the parts managing the data: beans, clients, Couchbase lite and tasks (and some other things such as helpers)

**Milestones**

|  |  |
| --- | --- |
| **Feature** | **Completion Date** |
| Login Module | 11/18/2015 |
| Folder & File Management | 11/25/2015 |
| File Sharing | 11/27/2015 |
| Search and Browse Files/Folders | 11/30/2015 |
| Usage Reports and Push Notification | 12/2/2015 |
| Integration and Testing | 12/3/2015 |
| Project Report and Bug Fixing | 12/13/2015 |

ParsePush.subscribeInBackground("fareen.magdum@gmail.com", new SaveCallback() { @Override public void done(ParseException e) { if (e == null) { Log.d("com.parse.push", "successfully subscribed to the broadcast channel."); } else { Log.e("com.parse.push", "failed to subscribe for push", e); } } });