CSE 304 Practice of Programming
Course Project Evaluation Report

(Chaitanya(2012031), Danish(2012032), Manan(2012054), Sarthak(2012088))

Title of the project: coDrive (Code here)

Project Description:
Often offline storage runs out, and the user then needs to store data either on cloud, or on external media. We propose a solution, wherein the user needn't be concerned about the security of their data in the cloud, by simply storing it onto a dedicated space on one of his trusted peers, at the luxury of a network. Another use case could be sharing of media with another peer, via that shared space.
We have implemented an application that allows multiple users to share some space on each of their drives (offline storages). This is achieved by initially setting up a fixed amount of space on each user's local drive, required for the purpose of the application. Once a user issues a request to his list of "friends", it can be accepted for storing the user's data.

Team members:
- Chaitanya (2012031)
- Danish (2012032)
- Manan (2012054)
- Sarthak (2012088)

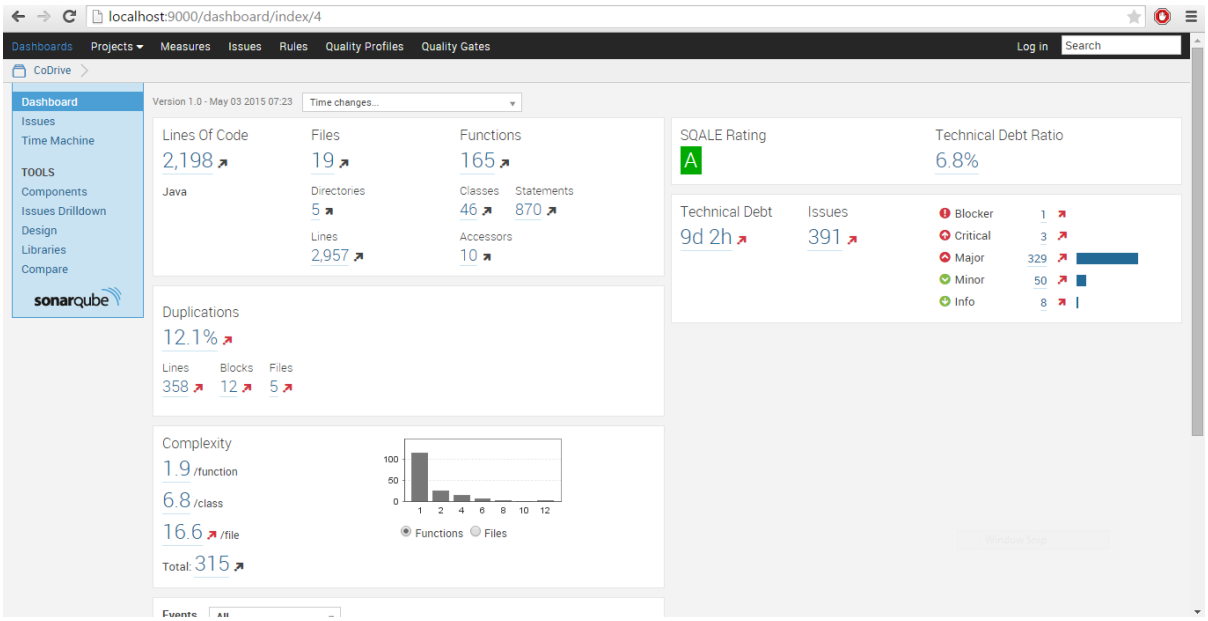
Languages/Tools/Platform (hardware/software)
- Hardware requirements: None
- Language: Java
- Platform: Android OS
- IDE: Android Studio
- Server: Parse database

Technical Challenges Faced:
The project relies on Parse, a NoSQL Server and was developed using Android Studio. However, Unit testing could not be done as the Parse API gave some technical glitches, for the same. Despite looking up for the same, no fixes could be found. The testing was therefore done by actual execution of the application, exhaustively.

Lines of Code: About 2,200 (on Android Studio); About 700 cloud code (on Parse)

Design Patterns:
Adapter: viewPager class, CustomList adapter
Observer: ActionListeners
Factory: Inflater.inflate() from xml to View class object;
Builder: AlertDialog creation
Note: A lot of android code has these adapters in-house, and during development extensive use of these happens naturally.

Static Analysis Tool used: SonarQube
Error Analysis:
Lint was used for the error analysis of the code, within Android Studio. Besides, the app was given for Alpha testing among the 4 developers.
The code on the Parse database went through 23 revision, spanning across about 700 lines of code. Parse handles all exceptions, and we made use of its Log Console to keep track of all such situations.
Report:



CoDrive

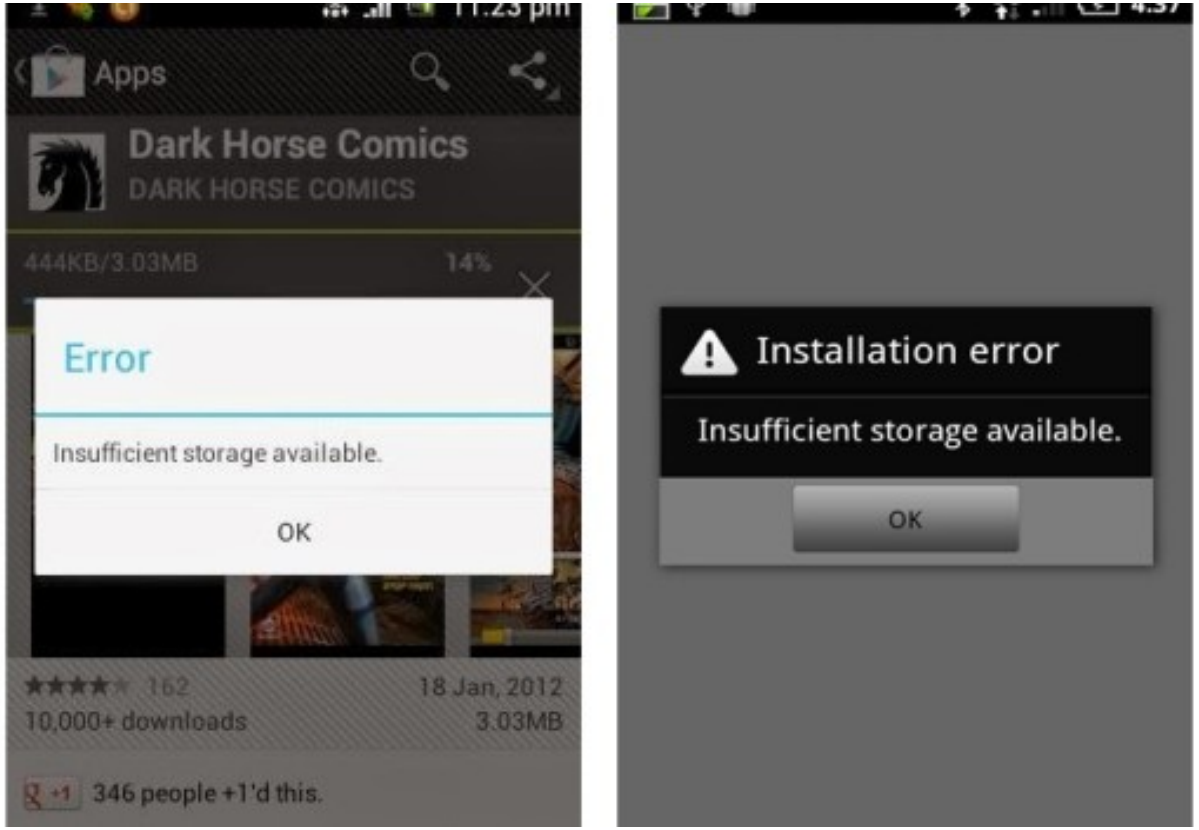
CSE 304 Project Presentation

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

-

Chaitanya	2012031
Danish	2012032
Manan	2012054
Sarthak	2012088

Motivation


Apps
Dark Horse Comics
DARK HORSE COMICS
444KB/3.03MB
14%
Error
Insufficient storage available.
OK
162
18 Jan, 2012
10,000+ downloads
3.03MB
346 people +1'd this.
Installation error
Insufficient storage available.
OK

Classes

Interaction

private name:ParseObject
private fromUser:User
private toUser:User
private Status:Status
private Type:Type

public getType() : Type
public setType(Type)
public getStatus() : Status
public setStaus(Status)
public getFromUser() : User
public setFromUser(User)
public getToUser() : User
public setToUser(User)
public setFile(String)
public getFileLink() : String
public removeFile():boolean
public getInteractions() : List<Interaction>
public getAllInteractions() : List<Interaction>
public getLog() : List<InteractionLog>
public getAvailableSpace(User):Long
public storeFile(File)

MyData

private Name:String
private User:User
private Size:long

public getName() : String
public setName(String)
public getUser() : User
public setUser(User)
public getSize() : long
public setSize(long)

InteractionLog

private Status:status
private Date:date

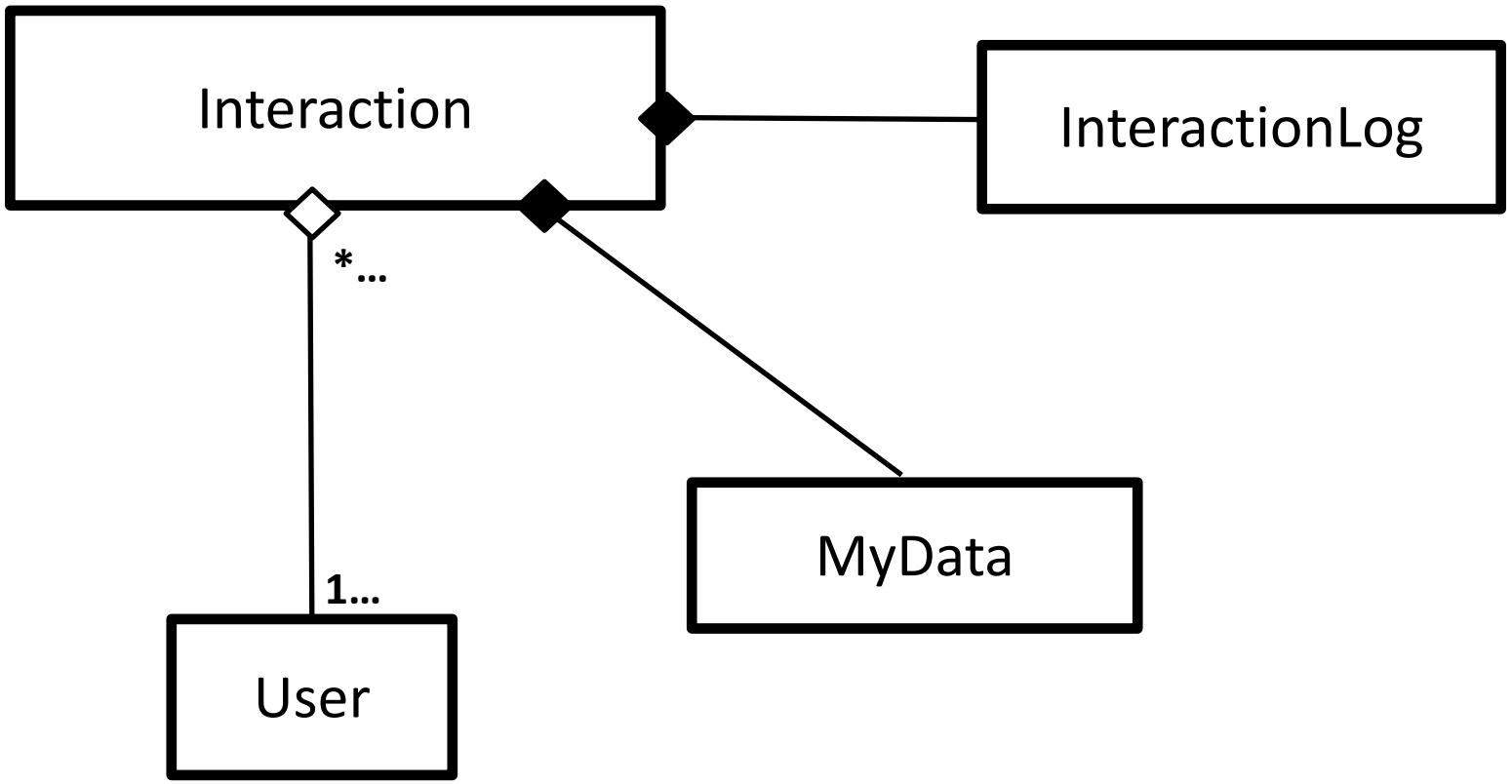
public getStatus() : Status
public setStatus(String)
public getDate() : Date
public setDate(Date)

User

private User:ParseUser
private AvailableSpace:Long

public getEmail() : String
public login(String,String)
public signUp(String,String)

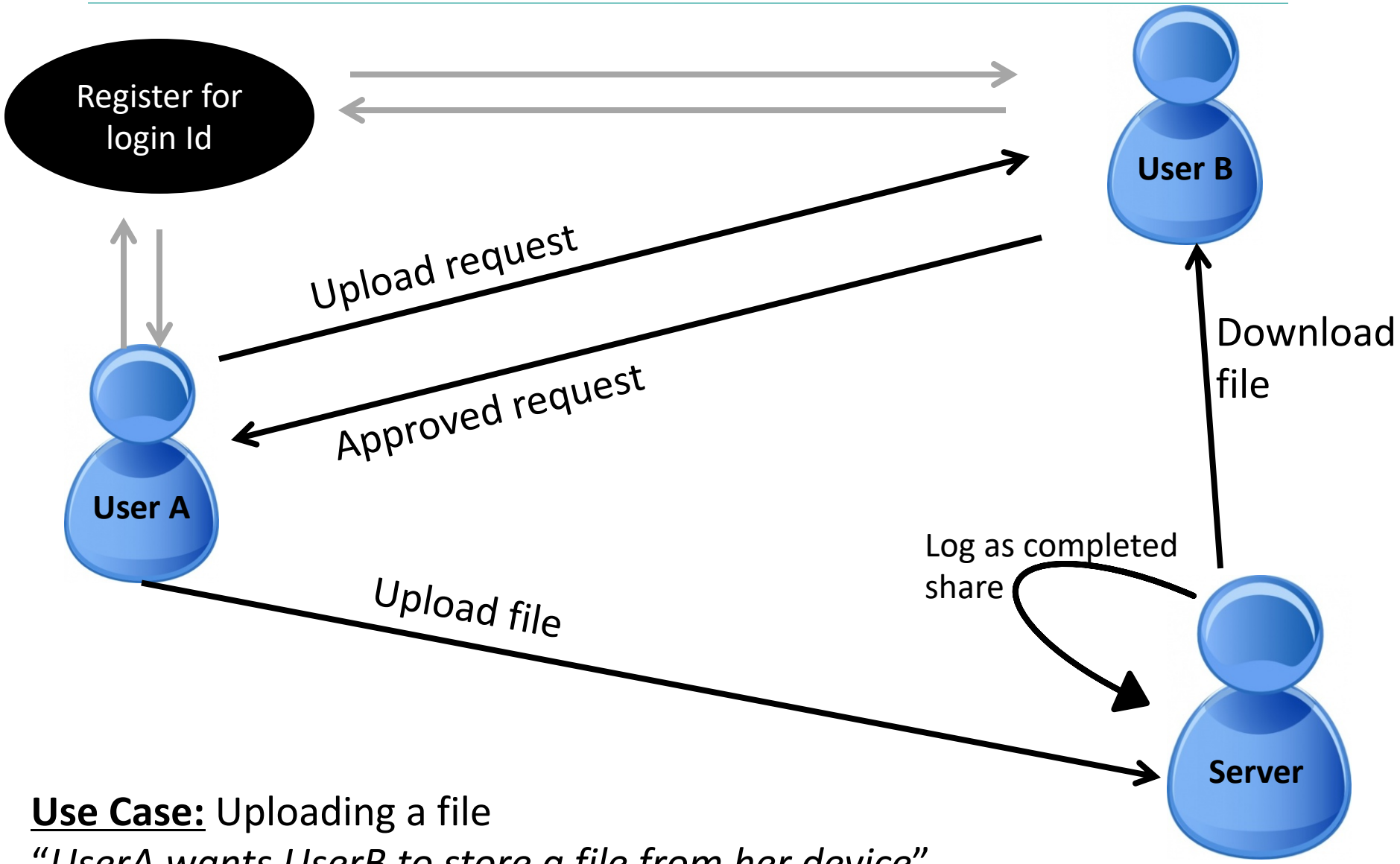
Classes



Design Patterns implemented:

- Adapter: CustomList adapter
- Observer: ActionListeners
- Factory: Inflater.inflate() from xml to View class object;
- Builder: AlertDialog creation

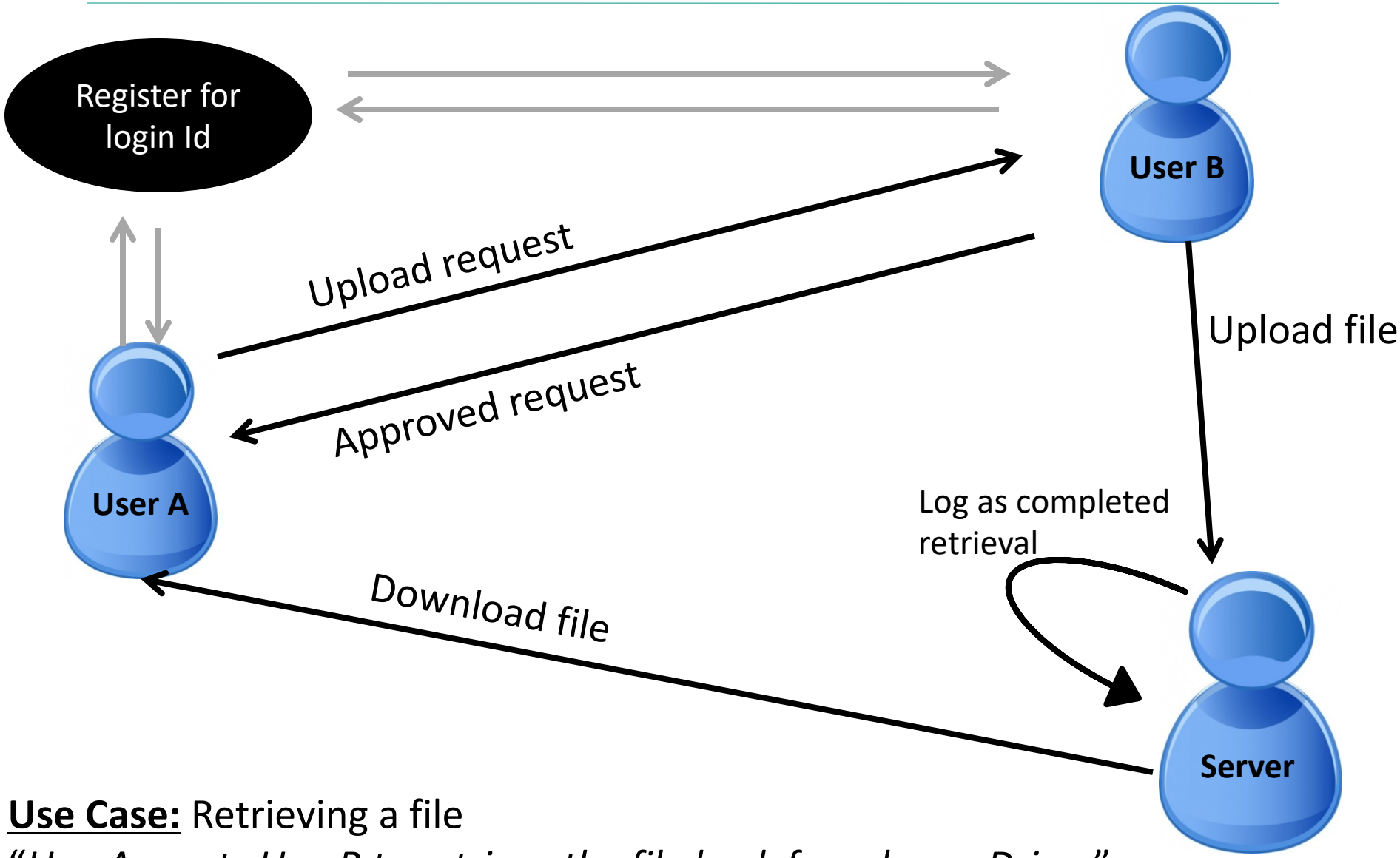
Use Case1



Use Case: Uploading a file

"UserA wants UserB to store a file from her device"

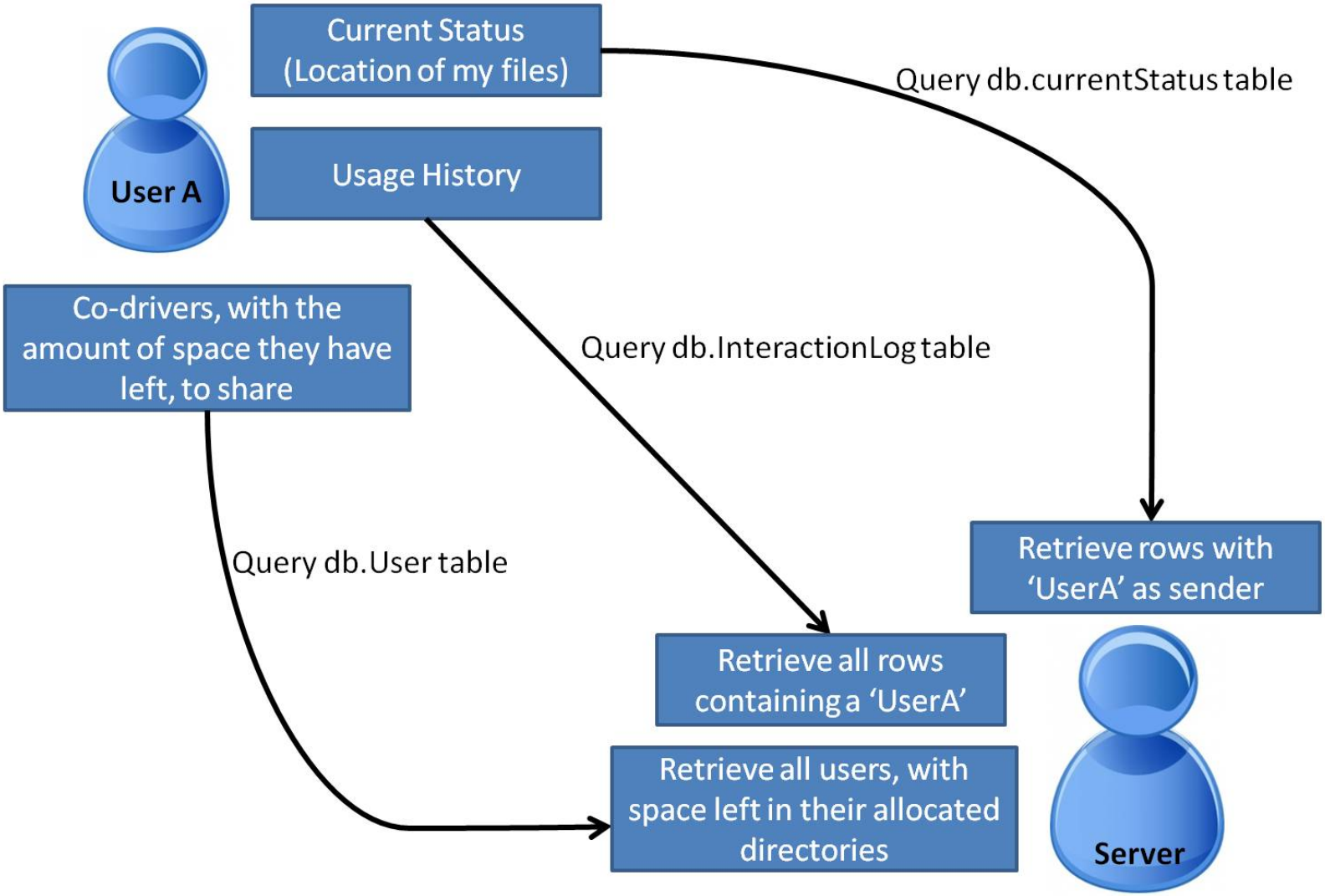
Use Case2



Use Case: Retrieving a file

"UserA wants UserB to retrieve the file back from her coDriver"

Miscellaneous Use Cases



Use Case: Miscellaneous use cases, requiring only User and Server interactions

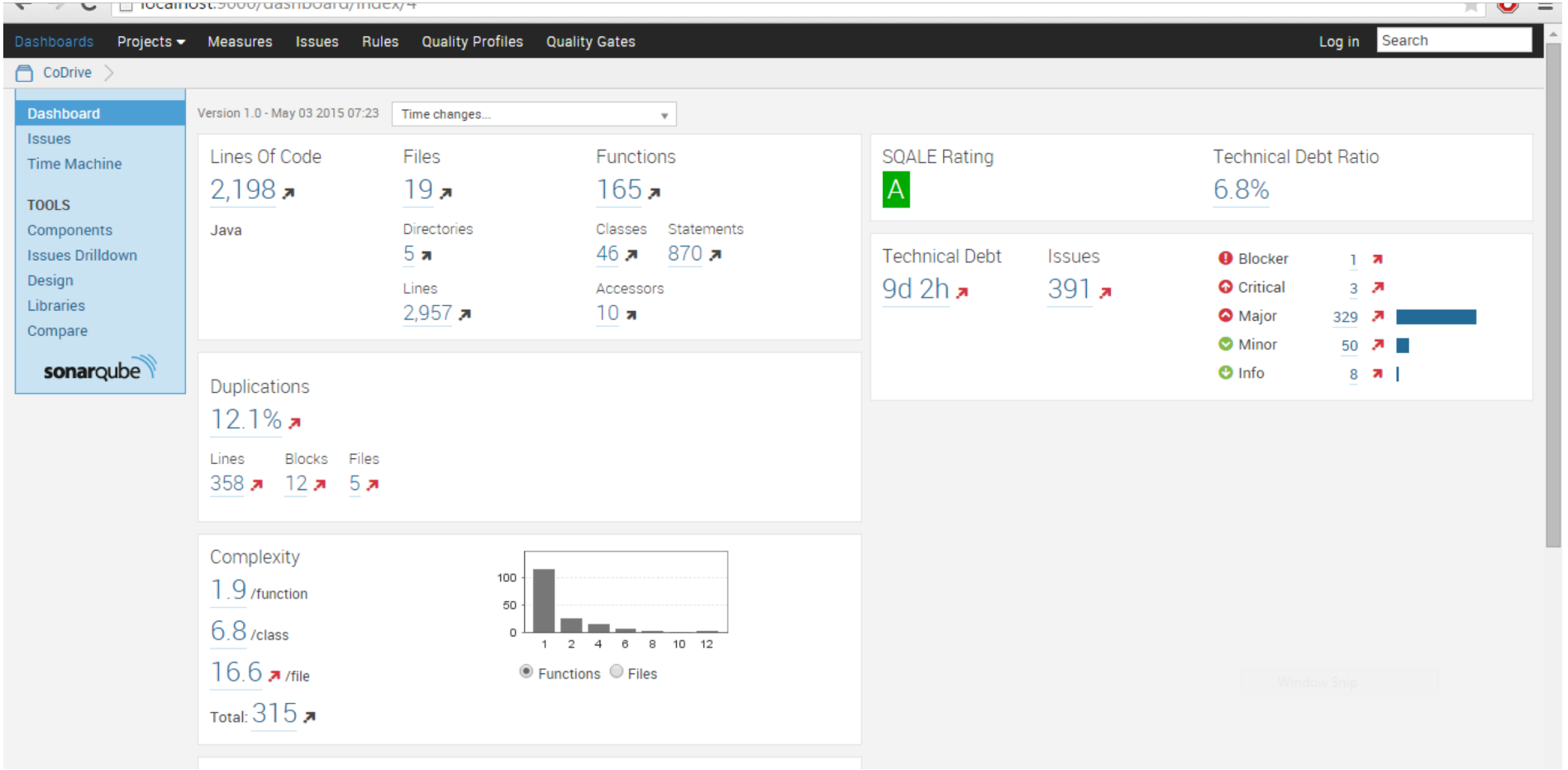
"UserA wants to view her usage history (or location of her files)"

Project Details

- Lines of Code:
  - About 2,200 (Java code)
  - About 700 (cloud code)

- Tools/Technologies Used:
  - Hardware requirements: None
  - Language: Java (offline); Javascript (cloud code)
  - Platform: Android OS
  - IDE: Android Studio
  - Server: Parse database

SonarQube stats



Error Analysis

- Lint was used for the error analysis of the code, within Android Studio. Besides, the app was given for Alpha testing among the 4 developers.
- The code on the Parse database went through 23 revision, spanning across about 700 lines of code. Parse handles all exceptions, and we made use of its Log Console to keep track of all such situations.

Thank you