Rookie-plays  
moview

May 21, 2018

# Overview

## Project Background and Description

So we are going to create a project to test how we will work as a unit

In this app we are going to work on an app that works as streaming apps.

The app should allow the user to add a video clip watch a movie clip, like the clip and follow the creator

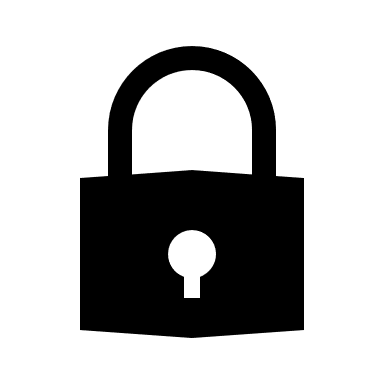
Each movie is stored in a database in the format of Title, Description, Release Date, Rating, Genre, uploader, Runtime, Movie Link

## JOBS

* (ollie and someone else) DATABASE Handling
* (ollie and someone else) Login and sign up
* (everyone) UI Design
* (all programmers) Logic and events aka coding
* (2 programmers) Debugging
* (3 members) Testing and bug reporting..

## DATABASE HANDLINGDatabase

Use either Firebase or SQLite

Authentication

Email

Password

UID

Movie

🡪MID(String)

* Title(String)
* Description(String)
* Release Date(Long)
* Rating(int)
* Genre(String)
* Uploader(UID)
* Likes(int)
* Link(String)

User Information

🡪UID(String)

* Email(String)
* Username(String)
* Followers (UIDs)
* Following(UIDs)
* Liked videos(MIDs)
* Favourite Videos(MIDs)

## Login and sign-up

Create an authentication for user login sign in and login send registration info to the database

Retrieve login details from data base…

authentication

Sign up!

Email..

Password..

Username…

User information

DONE

Username / email

password

LOGIN!

## UI DESIGN

Aims

* Fast - Flow
* Dynamic
* Themed
* Attractive

## Core Programming

Language: Java

## Debugging

Done by programmers

Use these test cases to debug the application

Login Test Classes

|  |  |
| --- | --- |
| Test class no. | 1 |
| Title | Login |
| Description | All required Fields entered |
| Validity | Valid |
| Example | Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

|  |  |
| --- | --- |
| Test class no. | 2 |
| Title | Login |
| Description | User found |
| Validity | Valid |
| Example | Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

|  |  |
| --- | --- |
| Test class no. | 3 |
| Title | Login |
| Description | User found and correct password |
| Validity | Valid |
| Example | Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

Registration Test classes

|  |  |
| --- | --- |
| Test class no. | 1 |
| Title | Registration |
| Description | All required Fields entered |
| Validity | Valid |
| Example | Email: olamidepeters@gmail.com  Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

|  |  |
| --- | --- |
| Test class no. | 2 |
| Title | Registration |
| Description | Correct Email Format |
| Validity | Valid |
| Example | Email: olamidepeters@gmail.com  Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

John544

Deba322

Ollie244

|  |  |
| --- | --- |
| Test class no. | 3 |
| Title | Registration |
| Description | Correct Username Format |
| Validity | Valid |
| Example | Email: olamidepeters@gmail.com  Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

|  |  |
| --- | --- |
| Test class no. | 4 |
| Title | Registration |
| Description | Correct password Format |
| Validity | Valid |
| Example | Email: olamidepeters@gmail.com  Username: ollie2224  Password: Abcdef123 |

|  |  |
| --- | --- |
| Test class no. | 5 |
| Title | Registration |
| Description | Email confirmed |
| Validity | Valid |
| Example | Email: olamidepeters@gmail.com  Username: ollie2224  Password: \*\*\*\*\*\*\*\*\*\* |

|  |  |
| --- | --- |
| Test class no. | # |
| Title |  |
| Description |  |
| Validity |  |
| Example |  |

## Testing

* Test for bugs
* Create bug reports
* When creating bug reports follow this template

|  |  |
| --- | --- |
| Name of Reporter: |  |
| Date: | 20052018(ddmmyyyy) |
| Title: | Ui crashing |
| Importance of bug : red=critical, amber=minor |  |
| (Short) summary of bug: | Ui crashing when user enters app |
| Steps to reproduce: | * Open app * Click the screen * Observe crash |
| BugID: Date\_Title | 20052018\_Ui Crashing |

|  |  |
| --- | --- |
| Bug ID | 20052018\_ui |
| Status | solved |
| Assignee | ollie |

# Approval and Authority to Proceed

We approve the project as described above, and authorize the team to proceed.

|  |  |  |
| --- | --- | --- |
| Name | Title | Date |
| Ollie | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Gabe | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Josh | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Vanco | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Deba | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Mko | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| Dami | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
| GB | Database Handling  Login and sign up  UI design  Core programming  Debugging  Testing | 20/05/2018 |
|  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ollie |  |  | 20/05/18 |  |  |  |  |  |
| Approved By |  |  | Date |  | Approved By |  |  | Date |
|  |  |  |  |  |  |  |  |  |
| Approved By |  |  | Date |  | Approved By |  |  | Date |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| Approved By |  |  | Date |  | Approved By |  |  | Date |