**SMART GRACE MARK ALLOCATOR**

**PROJECT REPORT**

***Submitted by***

|  |  |
| --- | --- |
| CB.EN.U4CSE17601 | AAKASH RISHISWAR D |
| CB.EN.U4CSE17607 | AJAY RAJ S |
| CB.EN.U4CSE17610 | ARUNA VISWANATHAN |
| CB.EN.U4CSE17617 | SAI TEJA C |
| CB.EN.U4CSE17663 | SRIRAAM A |

***is partial fulfilment of the requirements for the Course – 15CSE313 (Software Engineering)***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**AMRITA SCHOOL OF ENGINEERING**

**AMRITA VISHWA VIDYAPEETHAM**

**COIMBATORE -641112**

**MARCH 2020**

**AMRITA VISHWA VIDYAPEETHAM**

**AMRITA SCHOOL OF ENGINEERING, COIMBATORE**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**BONAFIDE CERTIFICATE**

This is to certify that the report entitled “Smart Grace Mark Allocator” submitted by

|  |  |
| --- | --- |
| CB.EN.U4CSE17601 | AAKASH RISHISWAR D |
| CB.EN.U4CSE17607 | AJAY RAJ S |
| CB.EN.U4CSE17610 | ARUNA VISWANATHAN |
| CB.EN.U4CSE17617 | SAI TEJA C |
| CB.EN.U4CSE17663 | SRIRAAM A |

is partial fulfilment of the requirements for the award of grade in course – **15CSE313 (Software Engineering)** is a bonafide record of the work carried out at Amrita School of Engineering, Coimbatore.

Evaluated on:

Course Faculty Faculty Examiner

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SLNO** | **TOPIC** | **PAGE NO** |
| 1 | **Preview of The Project** | 4 |
|  | Abstract | 4 |
|  | Need | 4 |
|  | Motivation | 4 |
|  | Tools and Technologies used | 4 |
| 2 | **Software Requirements Specifications** | 5 |
|  | Functional and Non-Functional Requirements | 5 |
|  | Weekly Plan for Sprint 1 | 6 |
|  | Weekly Plan for Sprint 2 | 6 |
|  | Boundaries and Constraints | 7 |
| 3 | **Product Backlog** | 8 |
|  | Report of Product Backlog from JIRA | 8 |
|  | EPIC 1: Login Credentials | 8 |
|  | EPIC 2: Grace Mark Allocation | 9 |
|  | EPIC 3: Course Mark Allocation | 9 |
|  | EPIC 4: Allocating and Retrieving Intervals | 10 |
|  | EPIC 5: Adding Grace Marks | 11 |
| 4 | **Diagrams** | 12 |
|  | DFD | 12 |
|  | Use Case Diagram | 14 |
|  | Activity Diagram | 15 |
|  | Sequence Diagram | 16 |
| 5 | **Scrum Activities** | 17 |
|  | Sprint 1 backlog | 17 |
|  | Sprint 1 stand-up notes | 26 |
|  | Sprint 1 retrospective notes | 30 |
|  | Sprint 2 backlog | 31 |
|  | Sprint 2 stand-up notes | 32 |
| 6 | **Implementation** |  |
| 7 | **Sample Code** |  |
| 8 | **Software Engineering Tools Used** |  |
|  | Static Code Testing |  |
|  | Unit Testing |  |
|  | UI Testing |  |
|  | Continuous Integration |  |
|  | Other Testing Practices |  |
| 9 | **Screen Shots of Project** |  |
| 10 | **Conclusion** |  |

**1. PREVIEW OF THE PROJECT**

**ABSTRACT:**

Every year the grace mark allocation becomes a very hectic job for the class advisors after the end of the semester. So, what we have done is an application which allocates grace marks in a very smart way and reduces the work of allocating grace marks by the class advisors.

**1.1 Need:**

The very own purpose of developing this application is to reduce the time taken for allocating grace marks and also send the new updated report after the grace marks are effectively added. This also reduces the discrepancies in adding grace mark manually because manual updation is prone to errors.

**1.2 Motivation:**

To ease the work of faculties who spend a lot of time in allocating grace marks, allocating the examination marks, finding the grades for the intervals decided and also mainly adding grace marks in a very proper way according to the credits of the subjects.

In the world where technology makes life simple we aim to make the grace mark allocation process more simpler by providing the faculties with our application which is very useful and handy in the process of grading and grace mark allocation.

**1.3 Tools and Technologies used:**

* Android Studio
* MySQL DB (php)
* XML : Frontend
* Java : Backend
* JIRA
* SonarQube : Static code testing
* Espresso : UI Testing
* Junit : Unit testing
* Travis : Continuous Integration

**2. Software Requirements Specifications**

**2.1 Functional And Non-Functional Requirements**

|  |  |
| --- | --- |
| **FUNCTIONAL REQUIREMENTS** | **NON-FUNCTIONAL REQUIREMENTS** |
| Grace mark calculation for adding grace marks to respective subjects | Time for the allocation(calculation) of the grace marks |
| Only the Grace Mark Allocator can allocate grace marks to students | Efficiency of the algorithm framed |
| Only the faculty will be able to retrieve the grace marks allocated | Space and time complexities of the algorithm |
| Grace Marks are added to the subjects which improve the grade of the student | Security Issues |

**2.2 Weekly Plan For Sprint 1:**

|  |  |
| --- | --- |
| **WEEK** | **PLAN OF ACTION** |
| 1 | Database Creation |
| 2 | Login screen UI creation and login by establishing connection between app and database |
| 3 | Display all student records as per requirement to specified faculties |
| 4 | Adding grace marks to applicable students and updating it in database |
| 5 | Reflection of the added grace marks in corresponding class advisor’s portal |

**2.3 Weekly Plan For Sprint 2:**

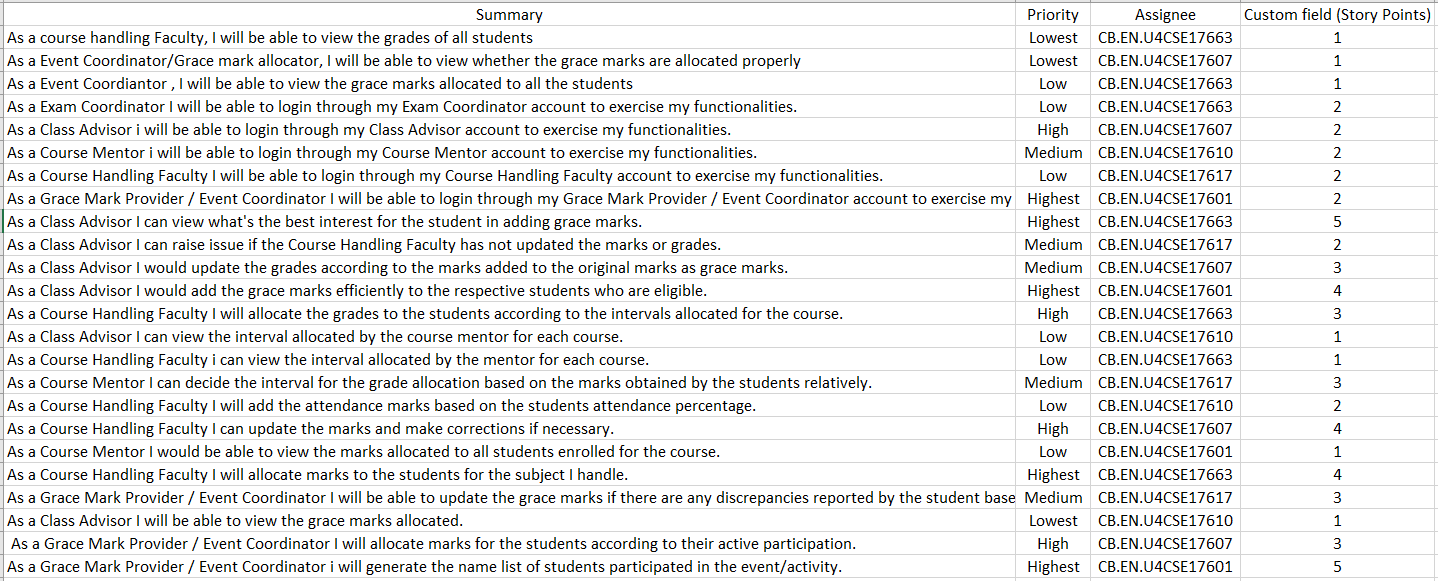
|  |  |
| --- | --- |
| **WEEK** | **PLAN OF ACTION** |
| 1 | Making the CHF be able to update the marks of the students |
| 2 | Making the course mentor able to separate the intervals based on the marks provided |
| 3 | Formulate the algorithm for smart grace mark calculation. |
| 4 | Making the class advisor choose the best mark to be allotted based on the algorithm. |
| 5 | Combining all functionalities together to get the final goal completed. |

**2.4 Boundaries and Constraints:**

* Once the grace mark are allocated, the course mark cannot be changed by the course handling faculty without the knowledge of the class advisor.
* The deadlock arises when the same user logs in through the same account at the same time.
* The redundancy of the database will be cleared once it is normalized.
* There can be 2 people with the same name , so we go wit recognizing people according to their roles allocated
* Also the login credentials for each faculty and other user will be given by us.
* A class advisor can also be a course handling faculty so he will be given two different logins to exercise two different functionalities.

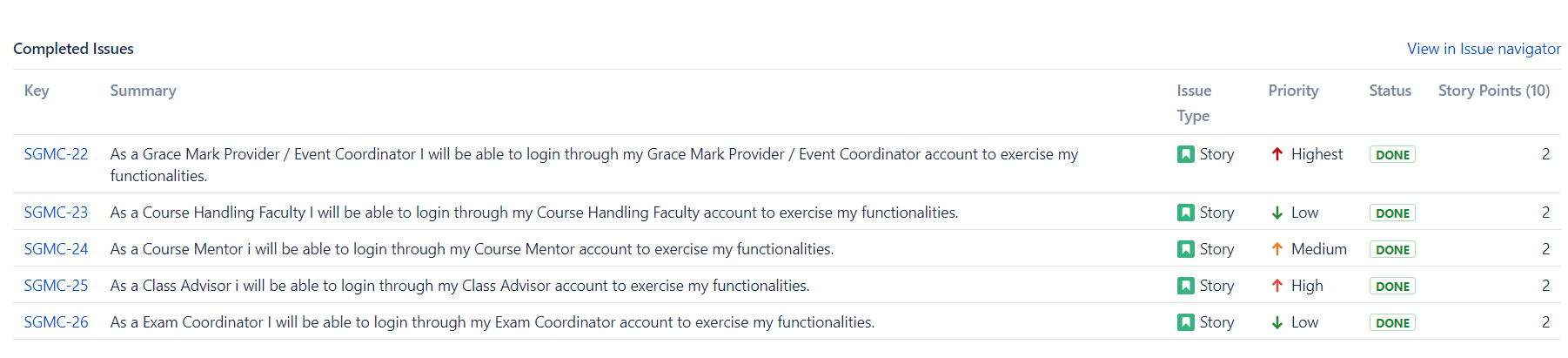
**3. Product Backlog**

**3.1 Report of your product backlog from JIRA**



**3.2 EPIC 1: Login Credentials**

**3.2.1 EPIC 1 Stories:**

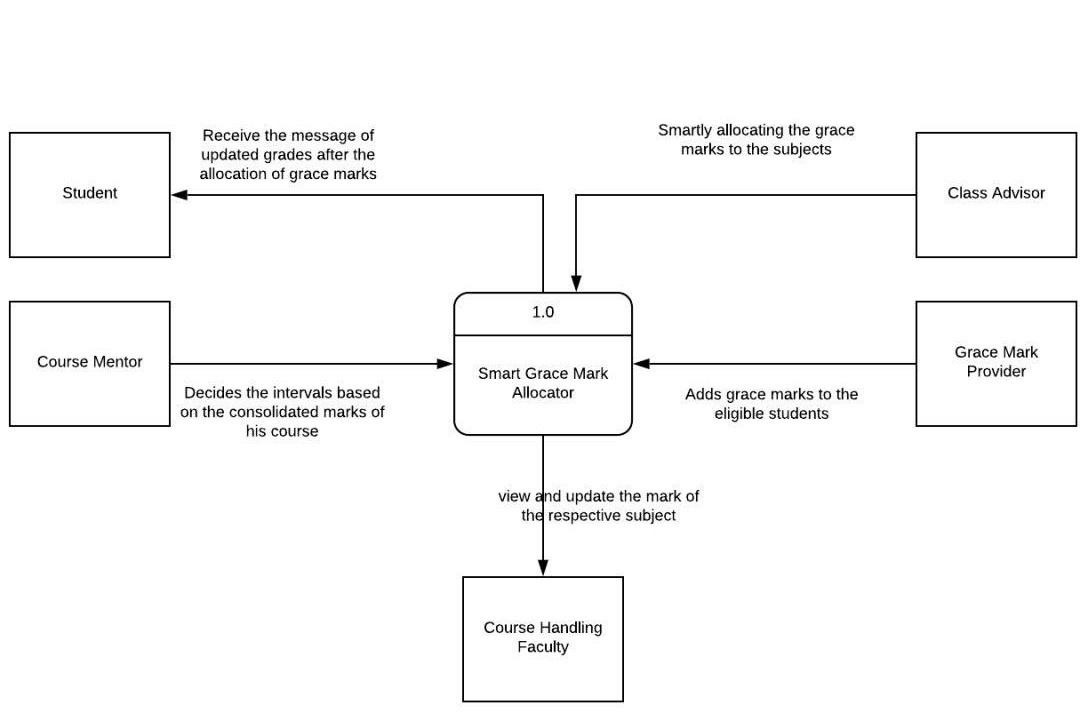


**……**

**4.Diagrams**

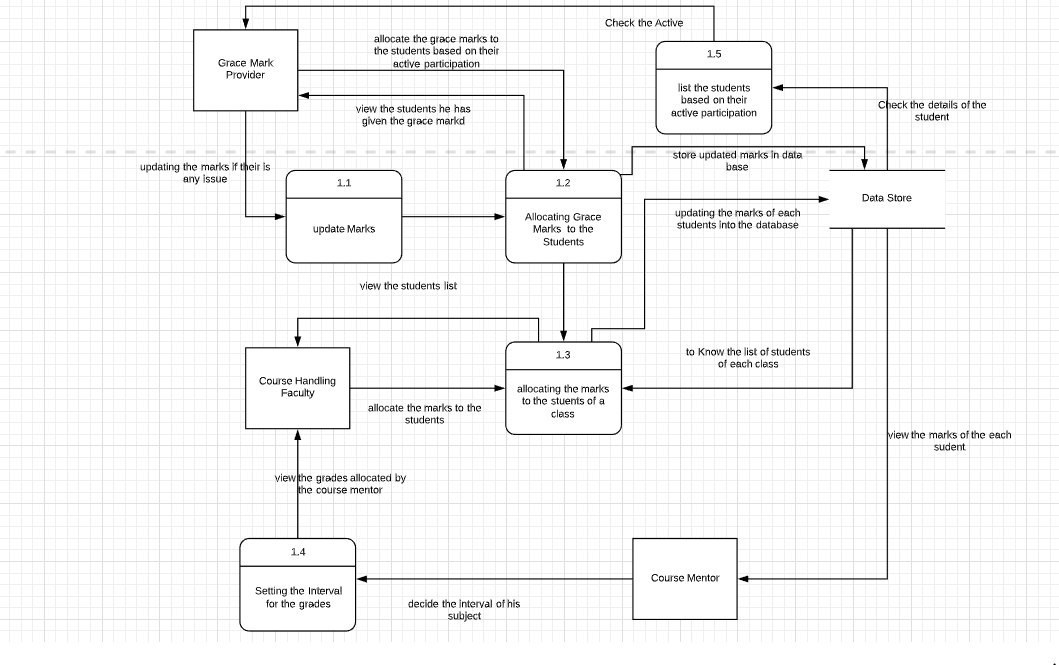
**4.1 Data Flow Diagram**

**Level 0**

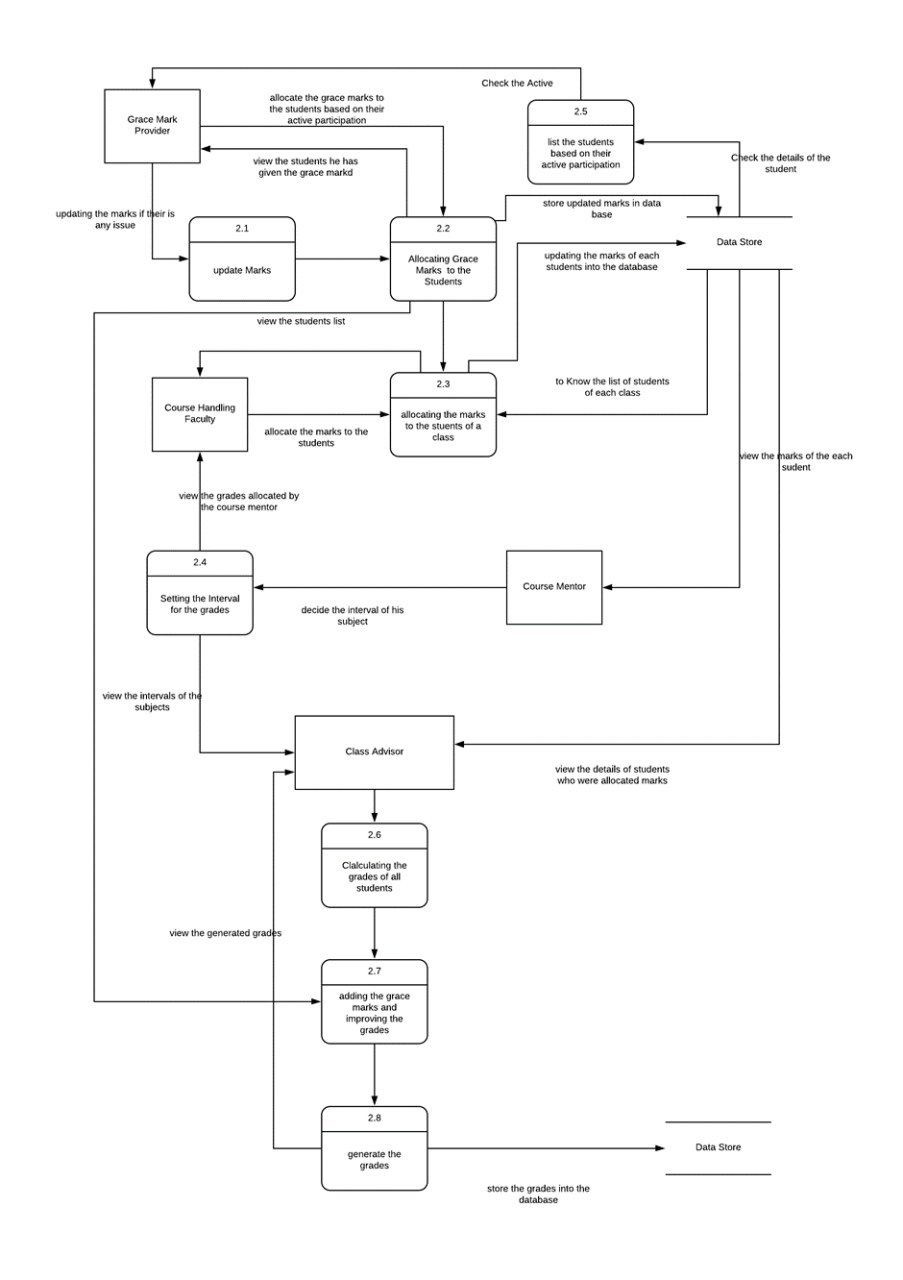


**0.0**

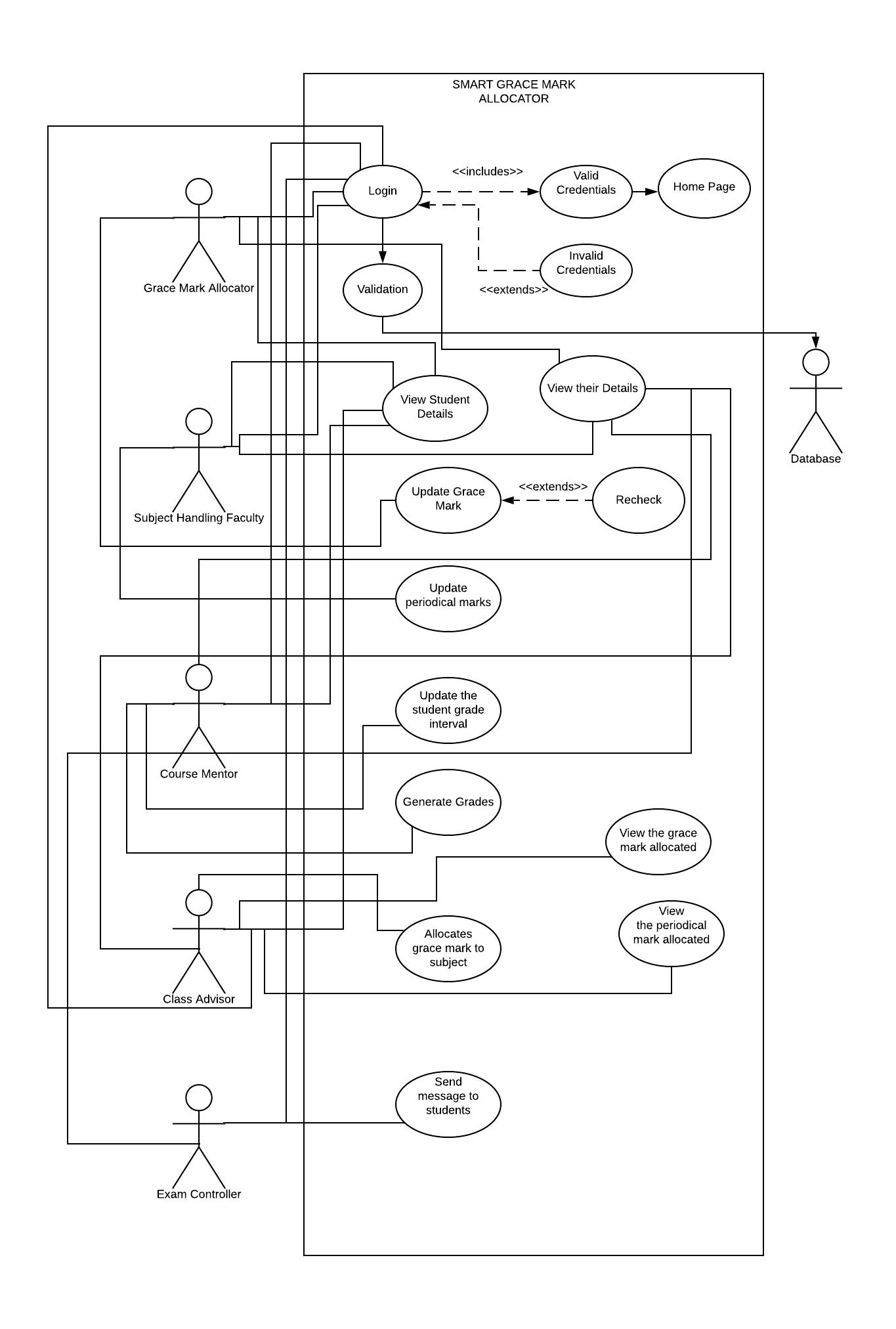
**Level 1**



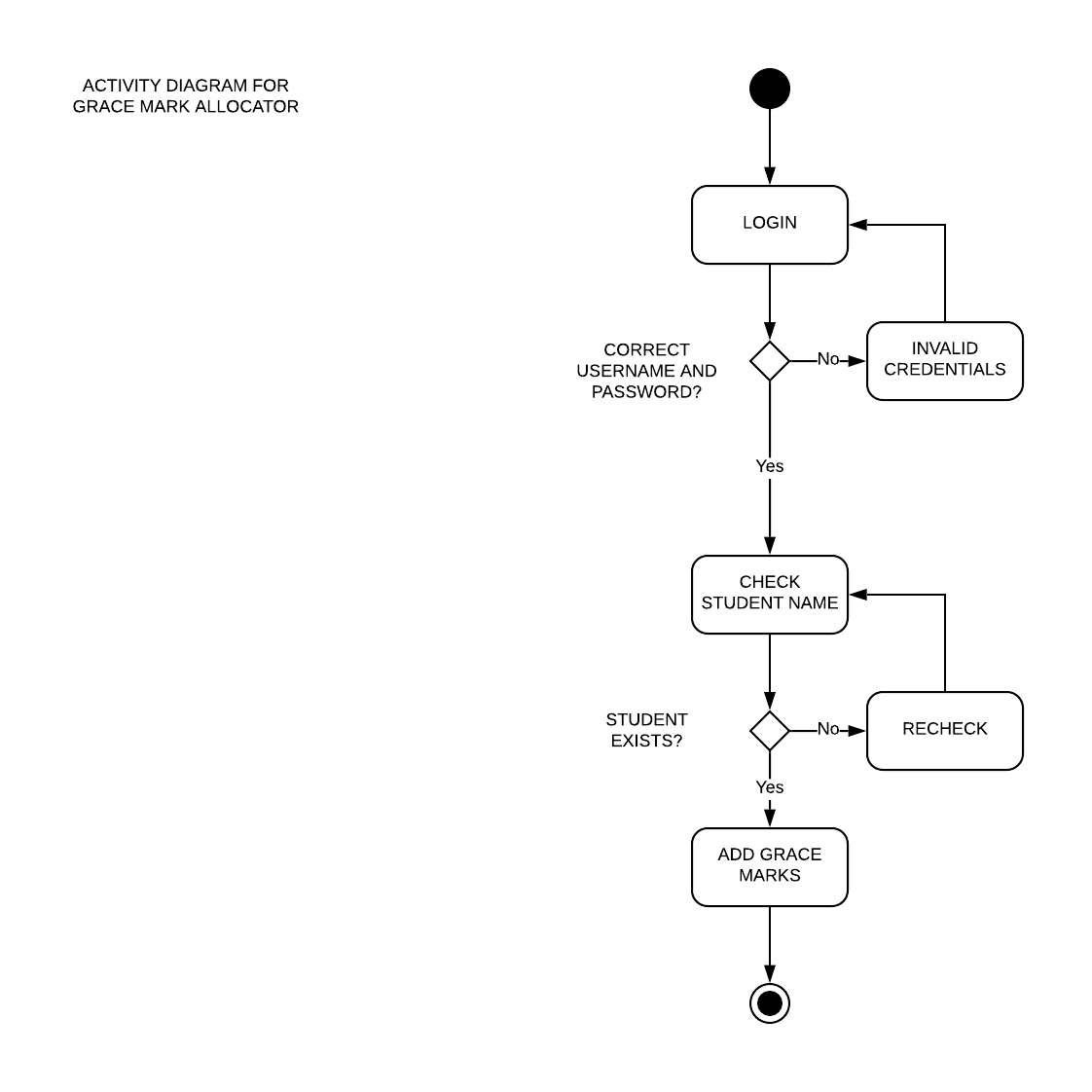
**Level 2**

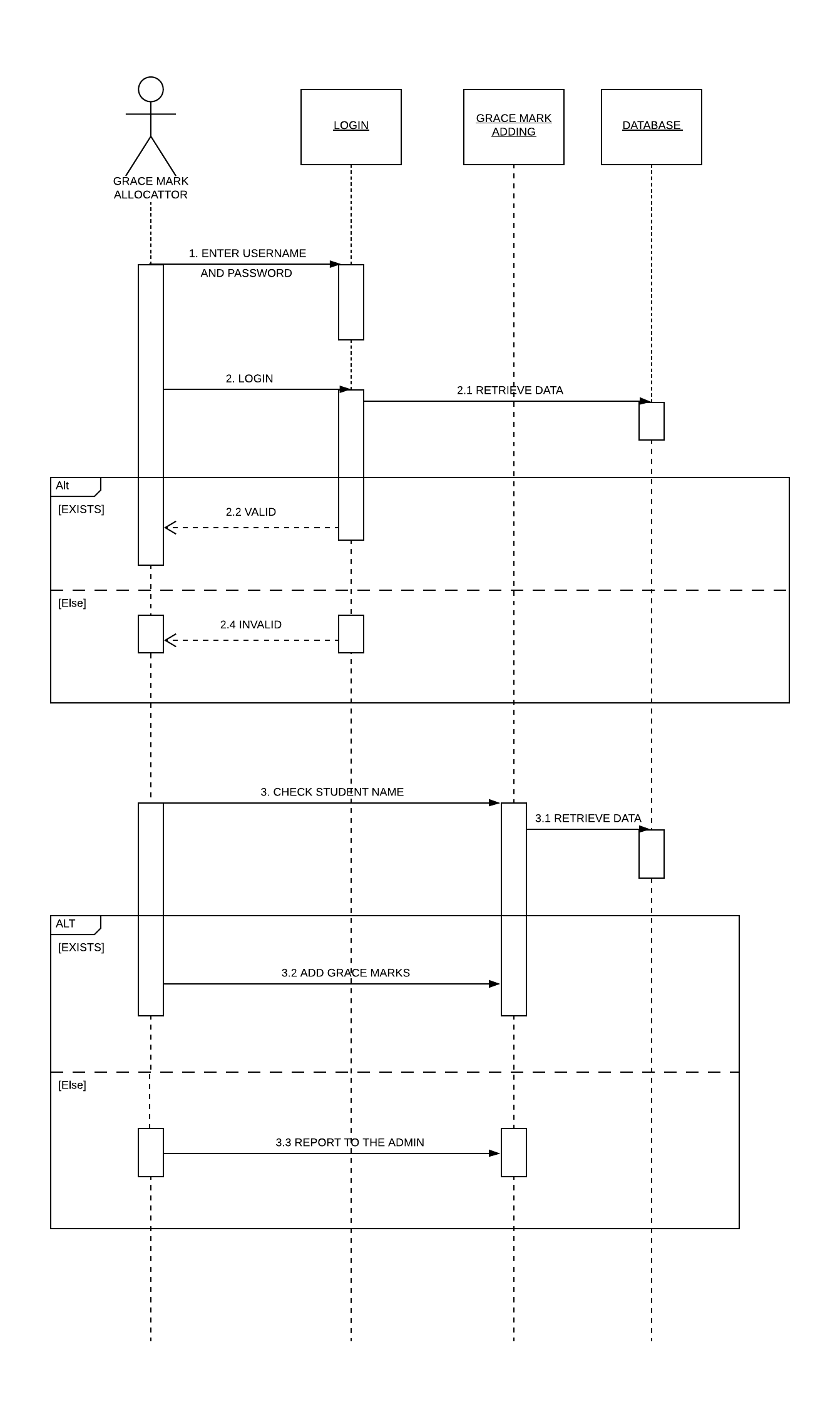


**4.2 Use Case Diagram**



**4.3 Activity Diagram**



**4.4 Sequence Diagram**

**5. Scrum activities(To paste Jira Statistics)**

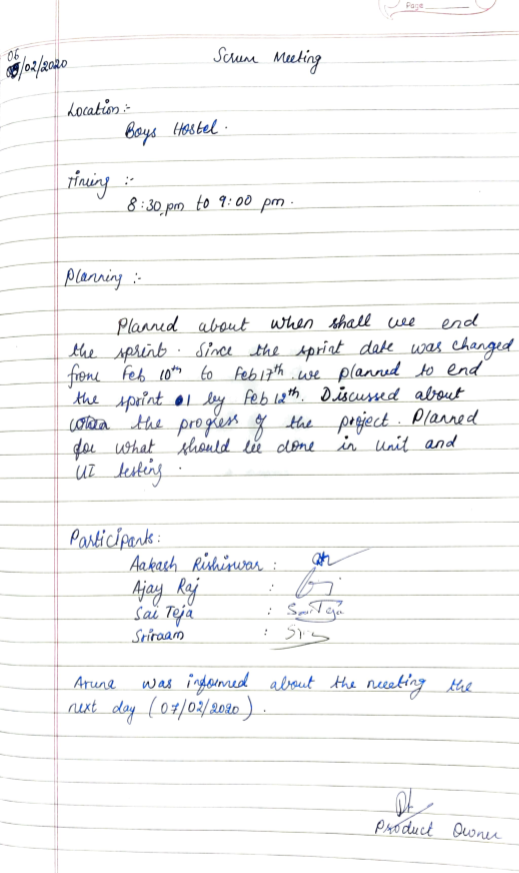
**5.1 Sprint 1 backlog sample(to post all details similarly in a format of your choice)**

|  |  |
| --- | --- |
| [SGMC-6] [As a Grace Mark Provider / Event Coordinator i will generate the name list of students participated in the event/activity.](http://jira.amritanet.edu:8080/browse/SGMC-6) Created: 12/Dec/19  Updated: 11/Mar/20  Resolved: 20/Jan/20 | |
| **Status:** | Done |
| **Project:** | [Smart grace mark calculator](http://jira.amritanet.edu:8080/secure/BrowseProject.jspa?id=10506) |
| **Component/s:** | None |
| **Affects Version/s:** | None |
| **Fix Version/s:** | None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Type:** | Story | **Priority:** | Highest |
| **Reporter:** | [CB.EN.U4CSE17601](http://jira.amritanet.edu:8080/secure/ViewProfile.jspa?name=CB.EN.U4CSE17601) | **Assignee:** | [CB.EN.U4CSE17601](http://jira.amritanet.edu:8080/secure/ViewProfile.jspa?name=CB.EN.U4CSE17601) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original Estimate:** | Not Specified | | |

|  |  |
| --- | --- |
| **Epic Link:** | [Grace Mark Allocation](http://jira.amritanet.edu:8080/browse/SGMC-1) |
| **Sprint:** | SGMC Sprint 1 |

**5.2 Sprint 1 stand-up notes**



**5.3 Sprint 1 retrospective notes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.no** | **START** | **CONTINUE** | **STOP** |
| 1 | UPDATION OF JIRA ACCOUNT REGULARLY | HAVING SCRUM MEETINGS TWICE A WEEK | HAVING SCRUM MEETING IN THE HOSTEL SINCE ONE PERSON IS NOT INVOLVED PROPERLY |
| 2 | UPDATING CHART REGULARLY | NOTING THE NUMBER OF STORY POINTS COMPLETED | DOING TESTING AFTER ALL COMPONENTS ARE INTEGRATED |
| 3 | DOING TESTING REGULARLY AFTER EVERY COMPONENT IS IMPLEMENTED | WORKING WITH THE SAME TEAM SPIRIT | STOP IRREGULAR STANDUPS |
| 4 | DRAWING DIAGRAMS FOR THE PROJECT | DOING INTEGRATION AS AND WHEN A COMPONENT IS IMPLEMENTED | COMMUNICATION GAP BETWEEN TEAMMATES |
| 5 | STICKING TO DEADLINES | EXECUTING THE PROJECT AS PER WEEKLY PLAN | MAKING FREQUENT CHANGES IN PLAN |

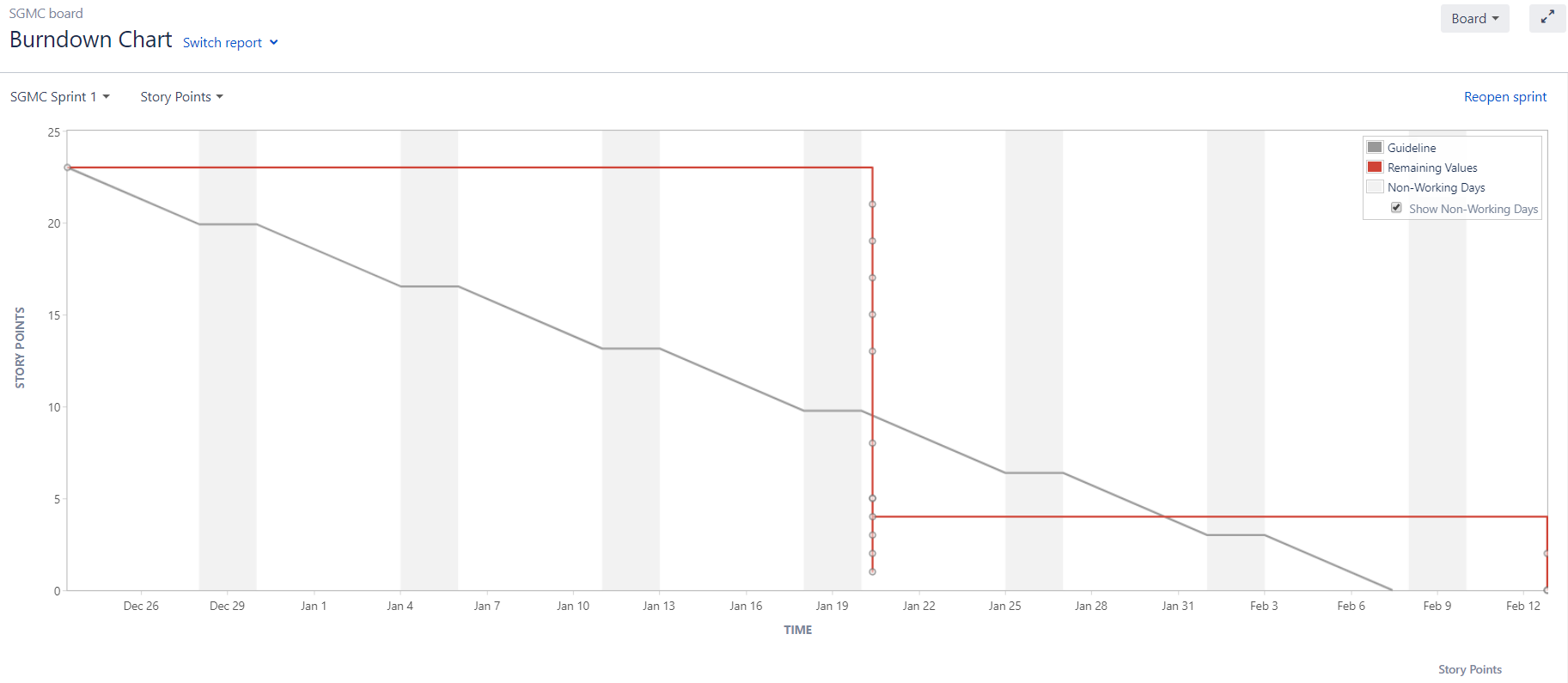
**5.4 Sprint 2 backlog**

Before the start of sprint 2:

**5.5 Sprint 2 stand-up notes**

**5.6 Burndown Charts based on Story Points:**

**Sprint 1:**



**Sprint 2:**

**6. Implementation**

Grace mark allocator is a simple yet efficient app which assists the teachers in allocating periodical marks to students and also grades them based on their performance. ……

(example)

The algorithm which we have implemented works by allocating grace marks based on the credit of the subject. It gives more preference to higher credit subjects and ………

**Architecture Diagram**

**7. Sample Code**

**#1(Display all details)**

<?php

require "gmaconn.php";

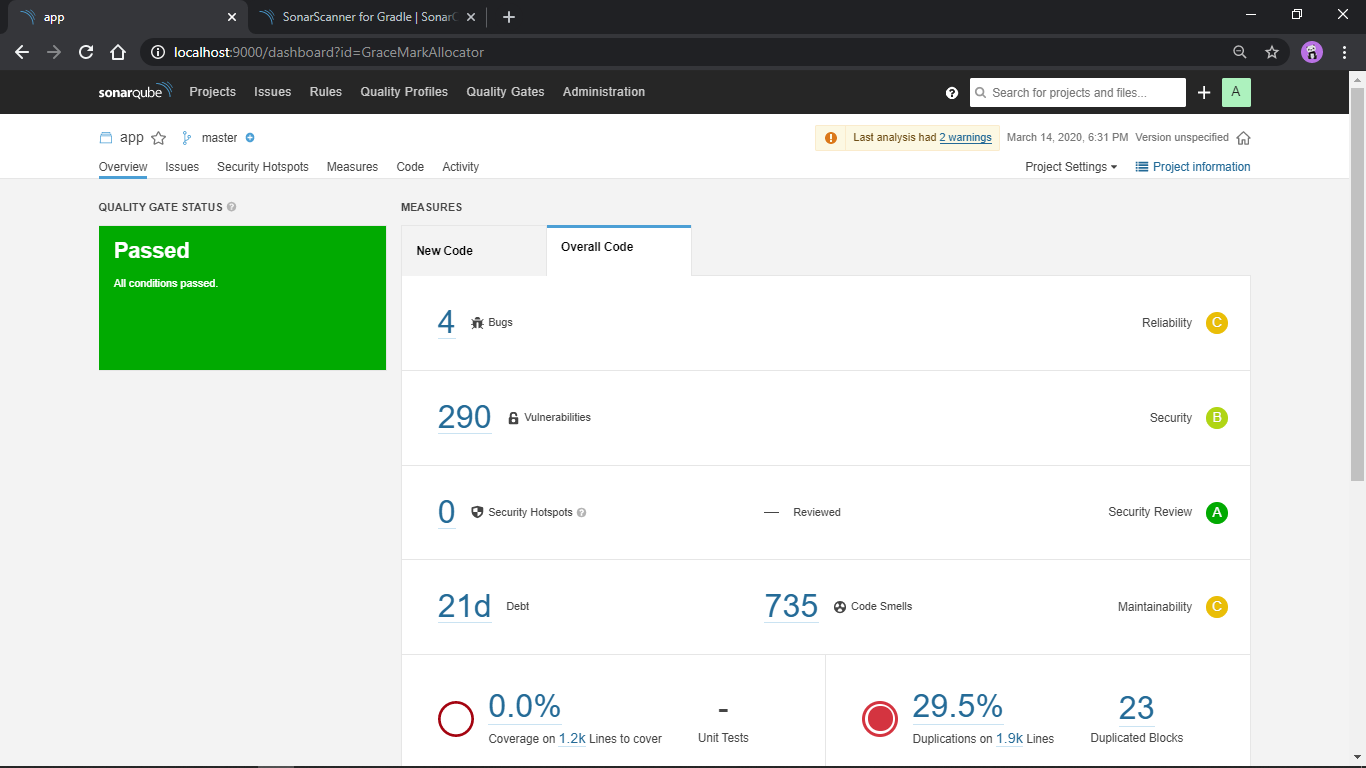
$user\_name = $\_POST["username"];

/……

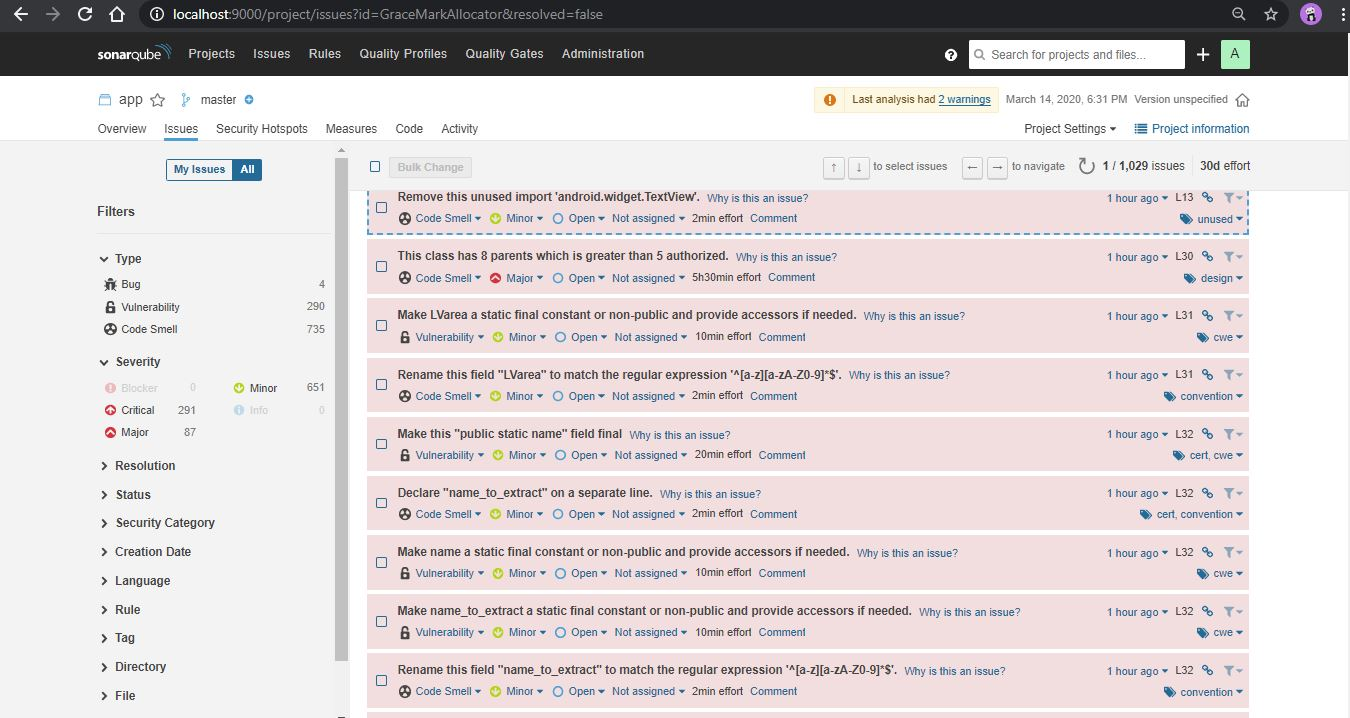
**8. Software Engineering tools used**

**8.1 Static Code analysis**

**8.1.1 Screenshots/report of the scanning results**



**8.1.2 Actions Taken**



**8.1.3 Additional Information**

**8.2 Unit Testing**

**8.2.1 Tool used and its setup details**

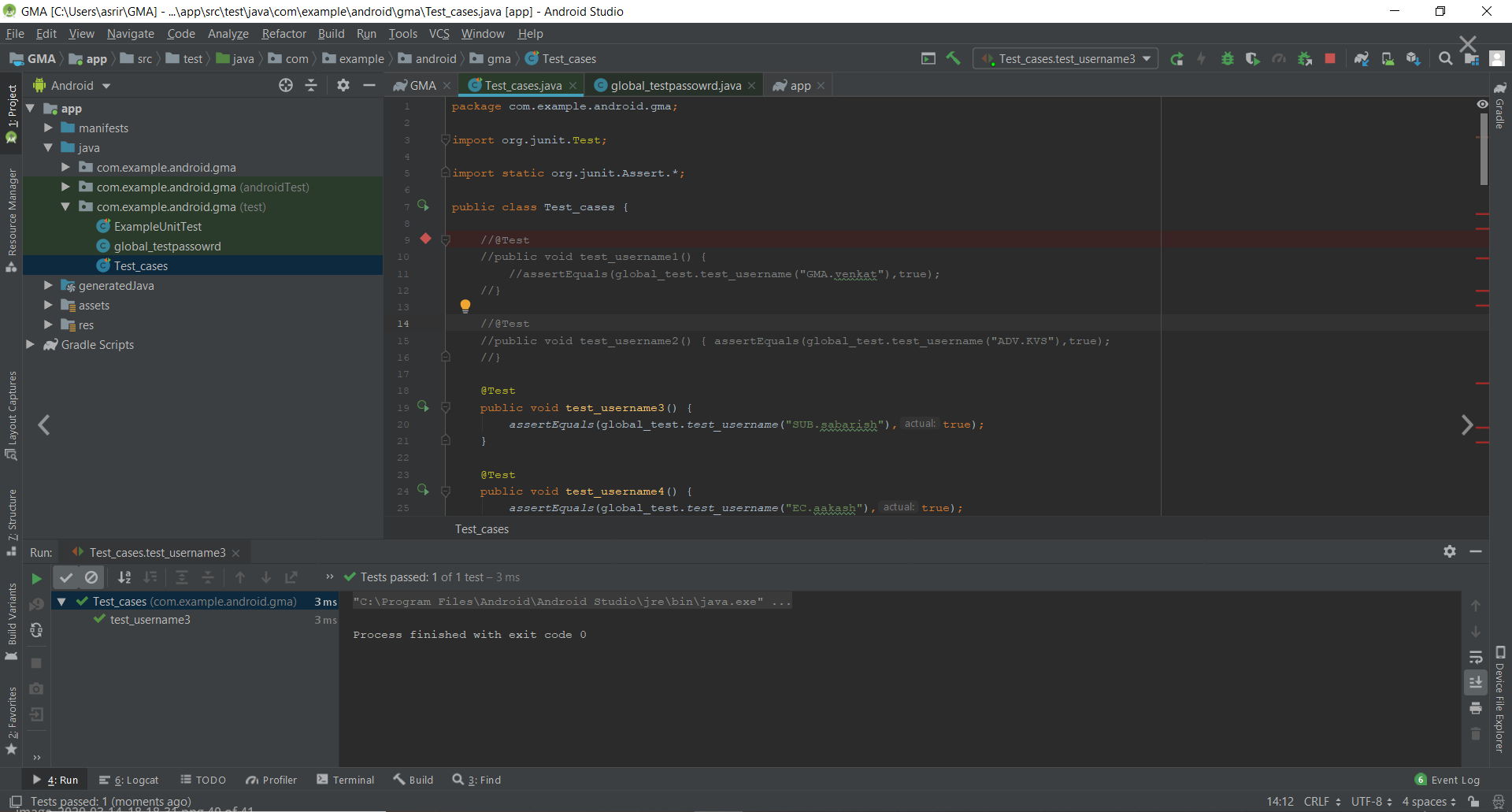
We have used Junit for unit testing.

We created a new java class and defined functions

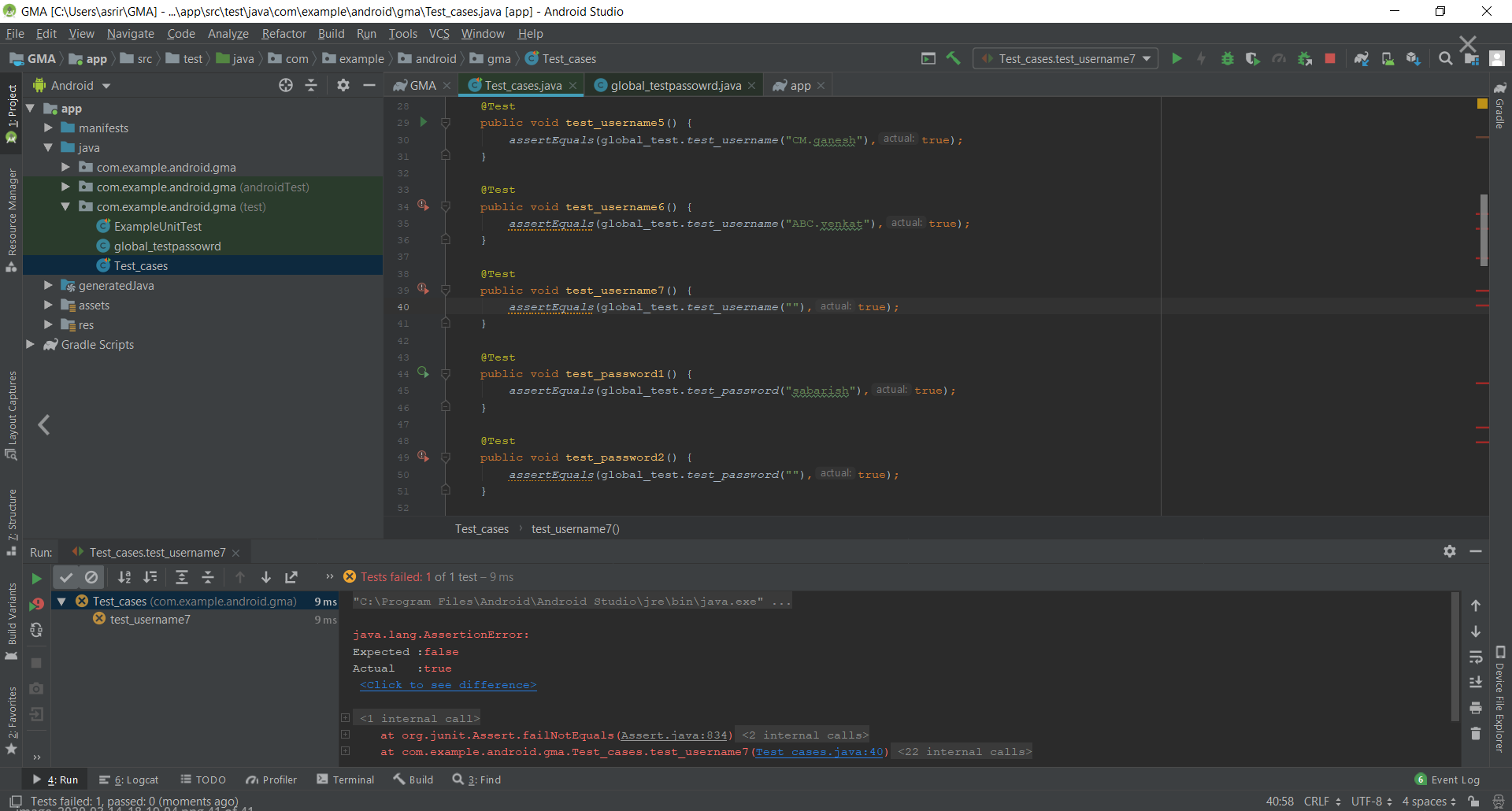
to test the validity of username and password.

**8.2.2 Test cases statistics**

**Pass:**



**Fail:**



**8.2.3 Test cases code**

**Pass:**

(fill your code here)

**Fail:**

(fill your code here)

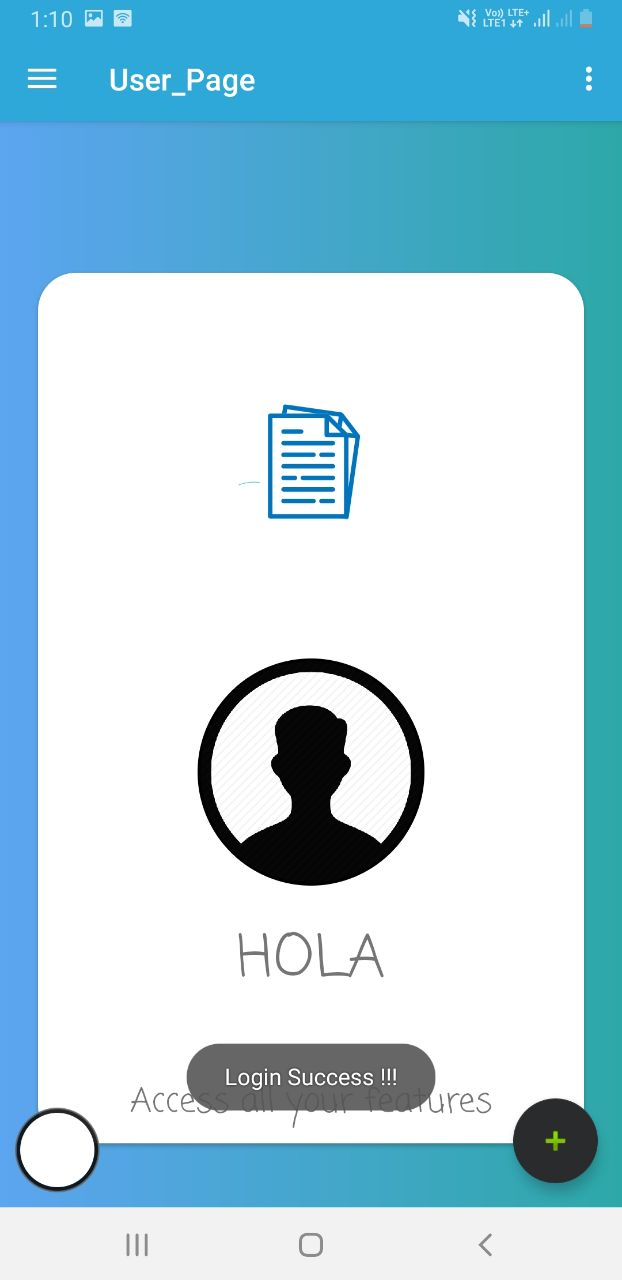
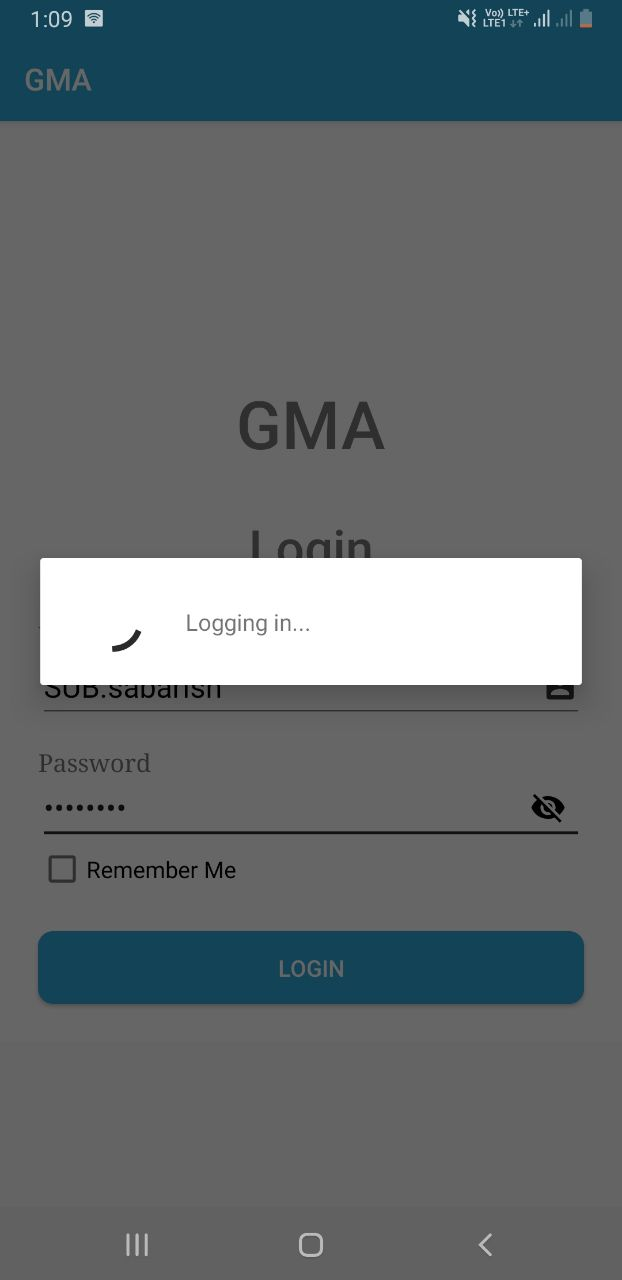
**8.3 UI Testing**

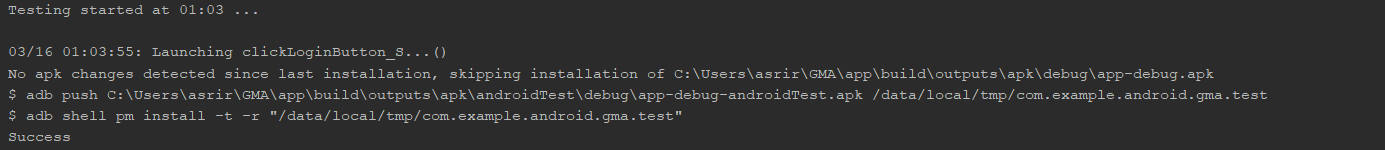
**8.3.1 Tool used**

We have used Espresso for UI testing.

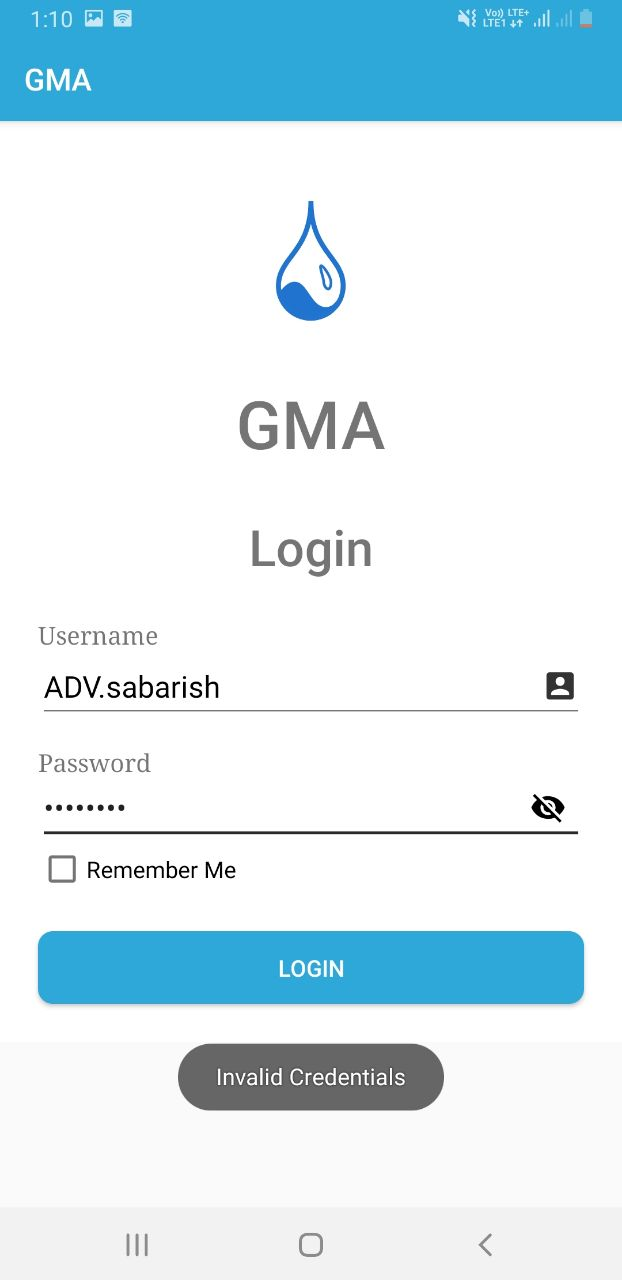
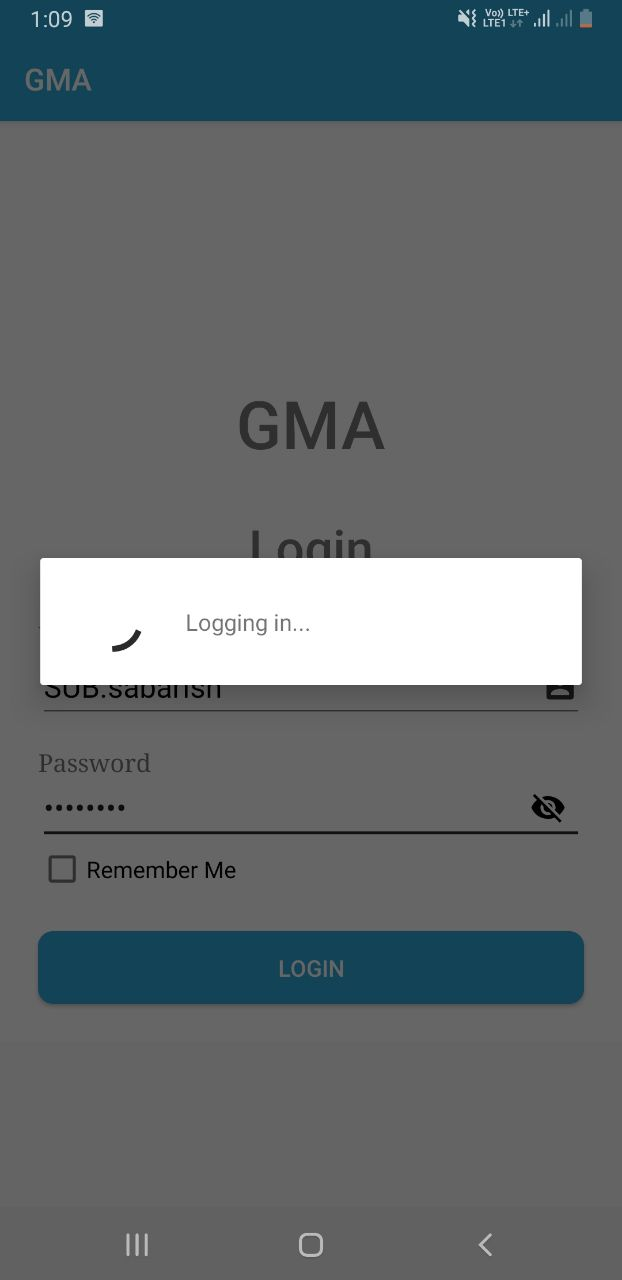
**8.3.2 Test cases statistics**

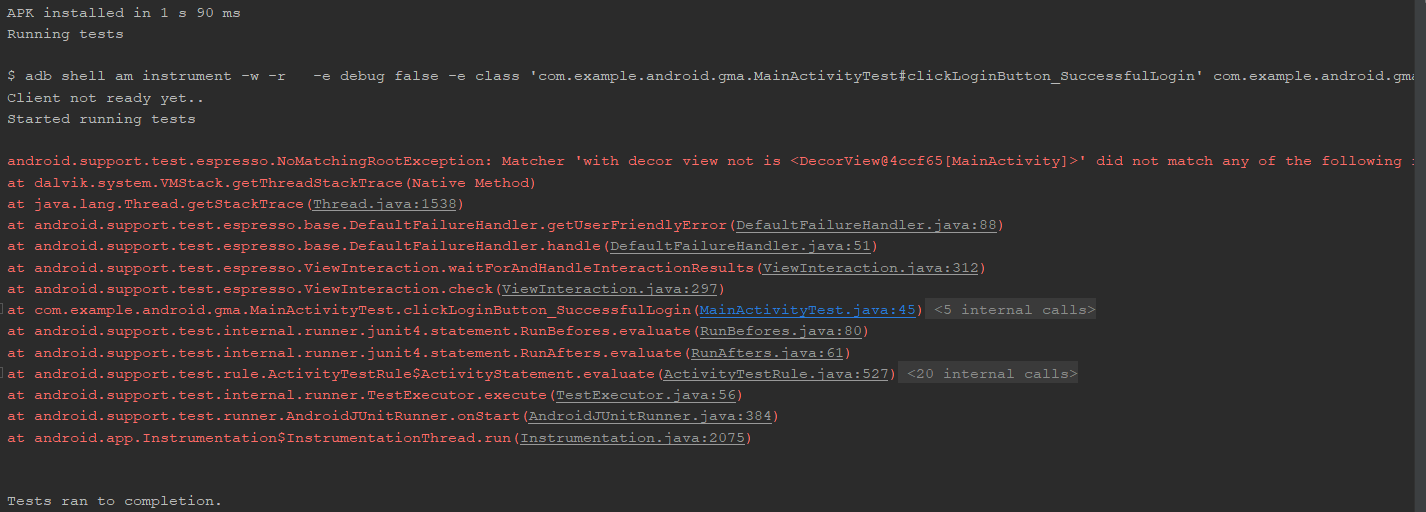
**Pass:**





**Fail:**





**8.3.3 Test case code**

**Pass:**

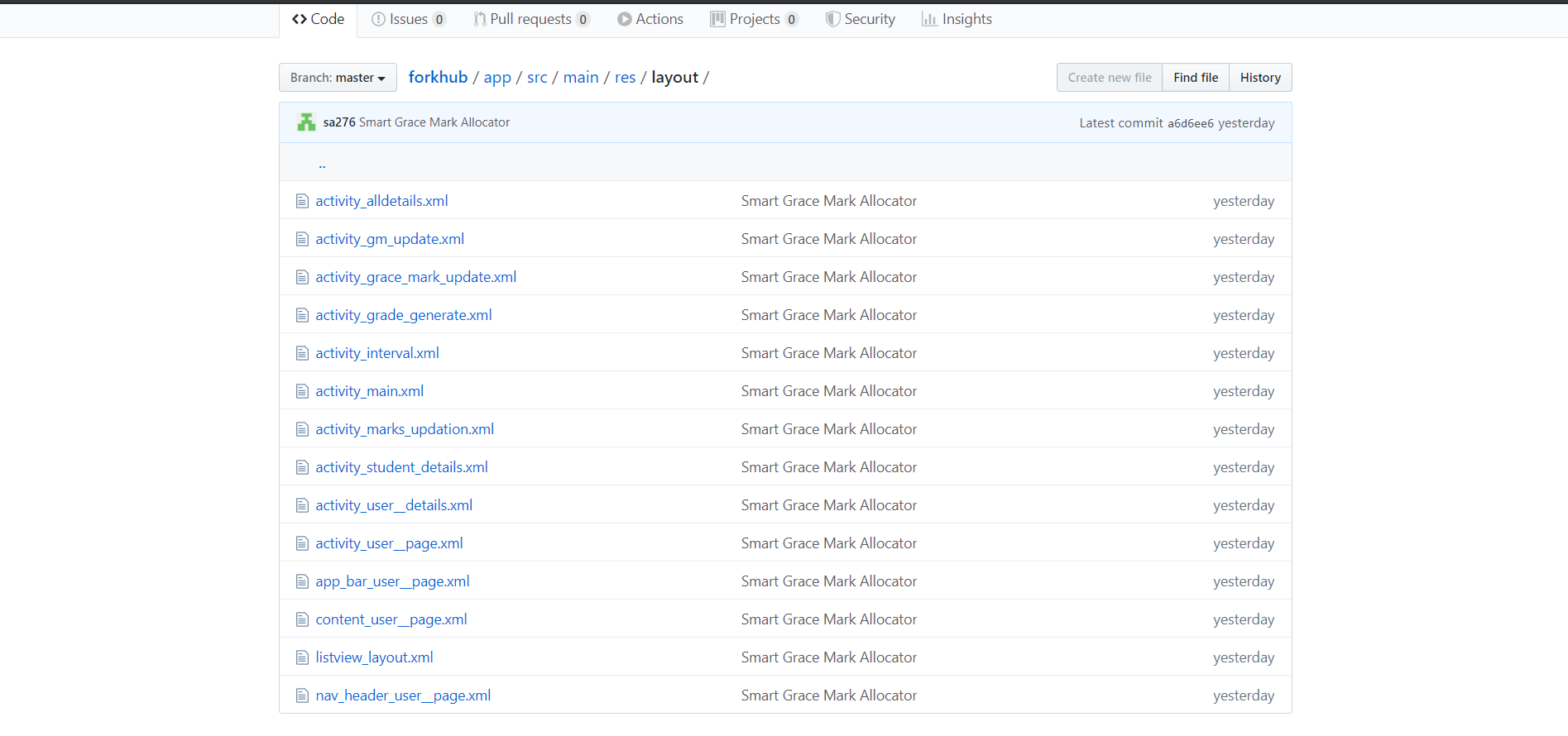
**(fill your code here)**

**Fail:**

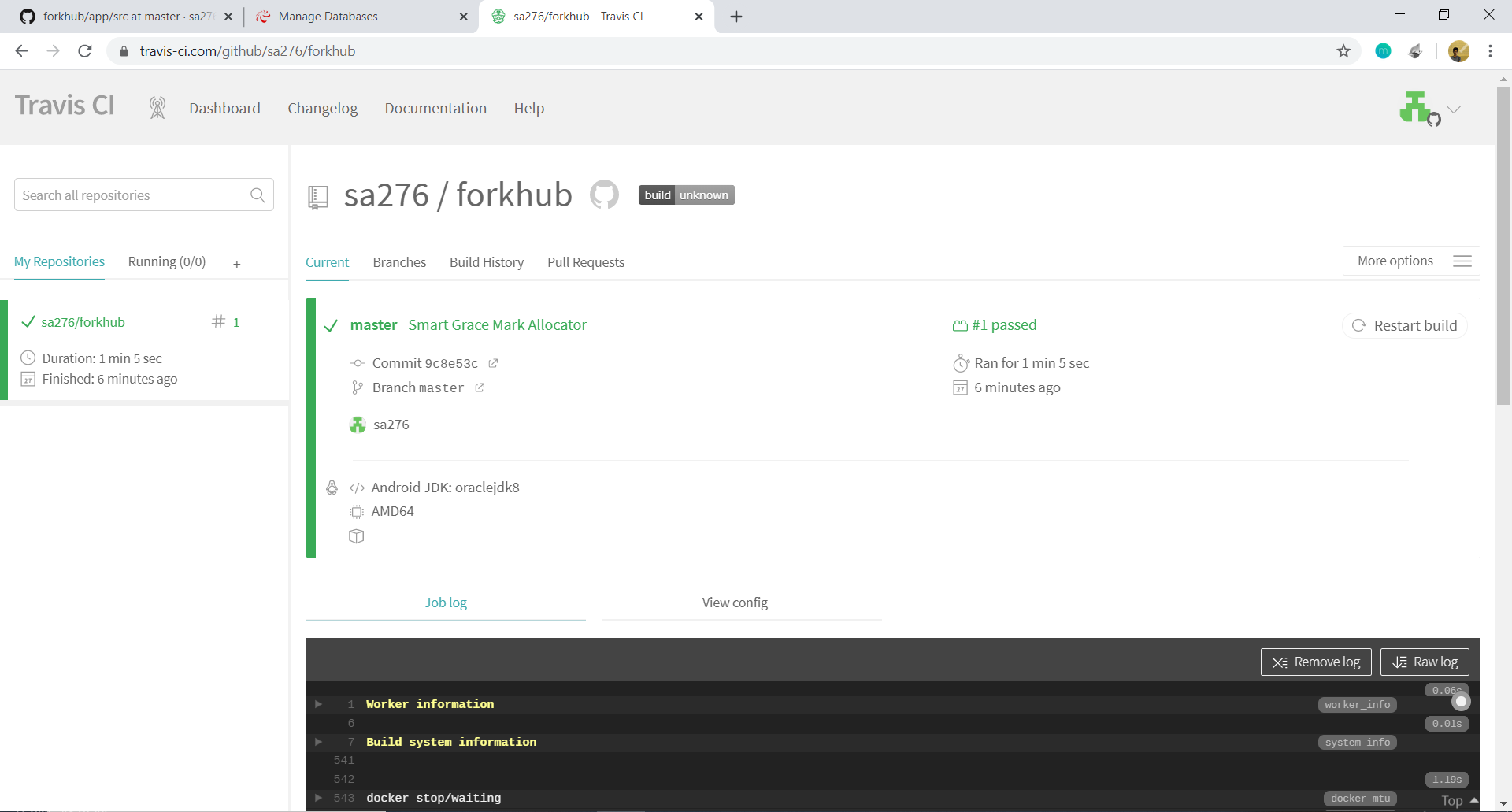
**(fill your code here)**

**8.4 Continuous Integration**

**8.4.1 Tool used**

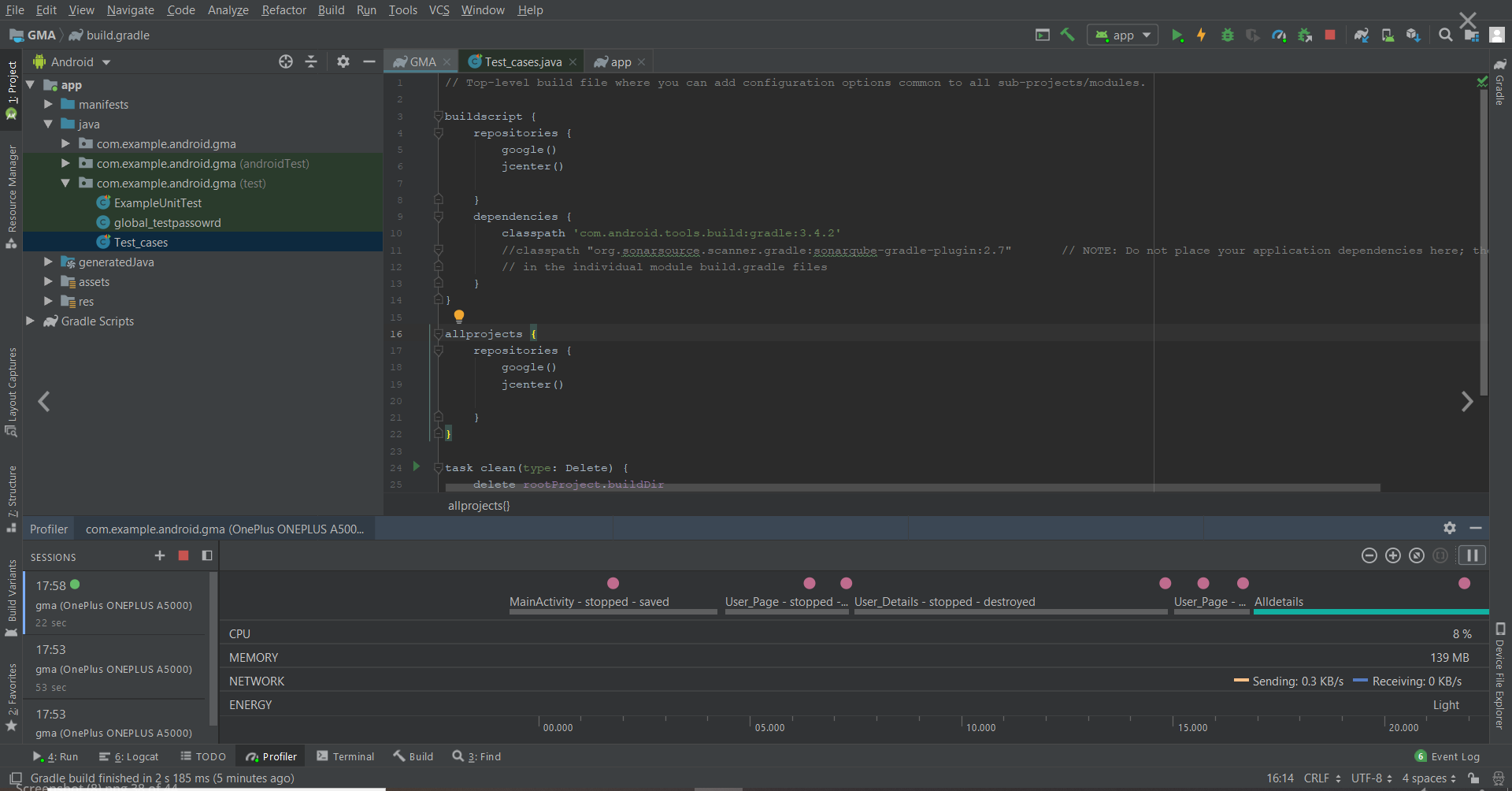


**8.4.2 Brief description of how CI has been done**



**8.5 Other Testing Practices**

We have done Performance Testing as an additional testing practice.



**9. Screen shots of Project**

**10. Conclusion**

…..