IPL Fantasy League

TEAM MEMBERS

Kushaggra Sharma

Onkar Borawake

Nikhil Dharaskar

Sourabh Gupta

Srikanth Gadde

Gaurav Chauhan

Project Statement

- IPL Fantasy League System is a web based application for bidding.
- This application will have admin and bidder.
- Both users have their own set of privileges to use the application.
- This system should allow new users to register themselves and use all the authorized functionalities.

Project Statement (Bidder)

- A bidder can sign in with the help of username and password.
- Bidders can view the match schedules and bid on a team for a match before the start of the match. When the bidder bids on a team he/she would be able to change/cancel his/her team before the match starts.
- The application shows the match details which includes the playing team, the venue of the match and current standing of the teams on the points table.

Project Statement (Admin)

- Admin is responsible for scheduling and rescheduling of matches, updating team statistics.
- Application should also provide facility to admin for managing the tournament.
- When bidders bid on a match, the admin should be able to view all the bidders bidding on a particular team and the percentage of users supporting a team.
- When the match results are announced, the system should update the standings in the team points table and bidder leaderboard.

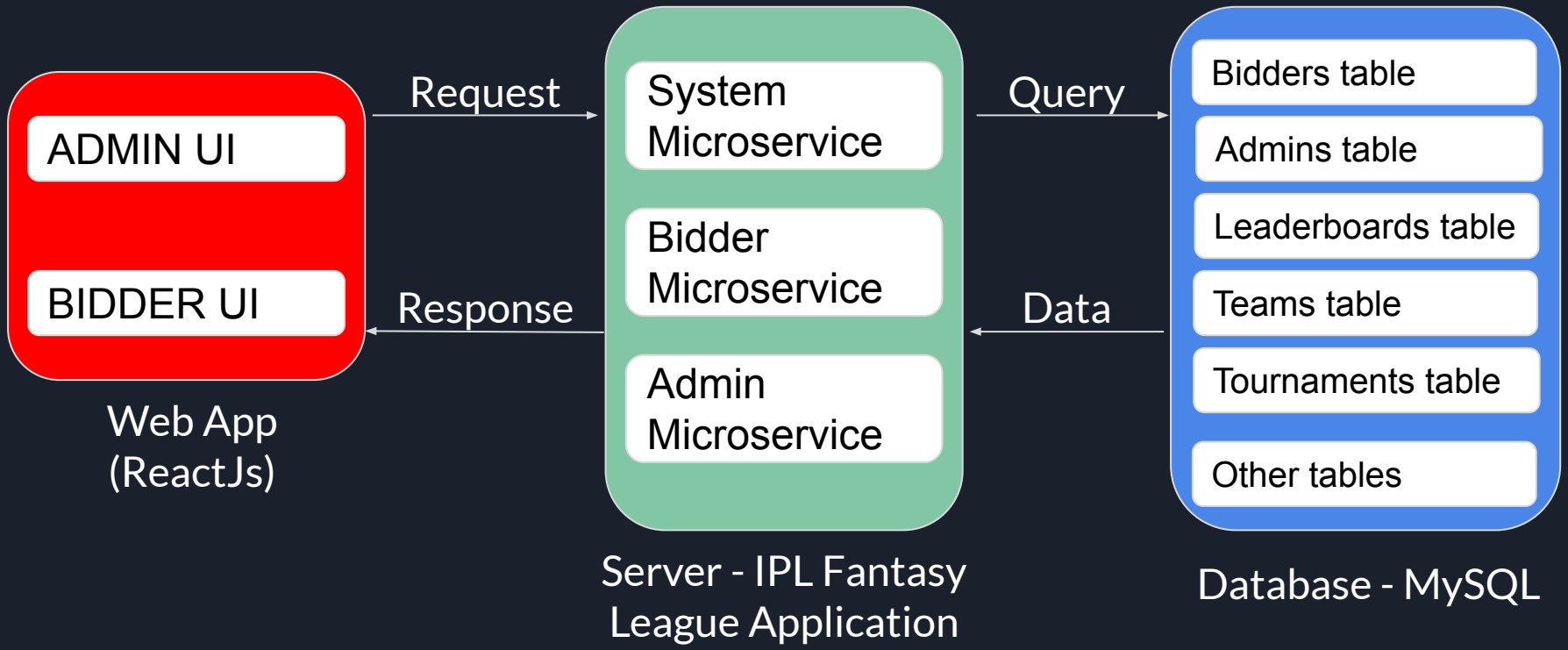
Technology Details

- Build Tool: *Apache Maven*
- JDK: *11*
- Spring Framework – *Spring Boot*
- Routing and discovery: *Feign, Eureka*
- Client-side load balancing: *Ribbon*
- Logging: *Slf4j with Log4j*
- Security: *BCrypt password hashing*
- DB: *MySQL*
- IDE: *Spring Tool Suite*
- Data Layer: *Spring Data JPA*

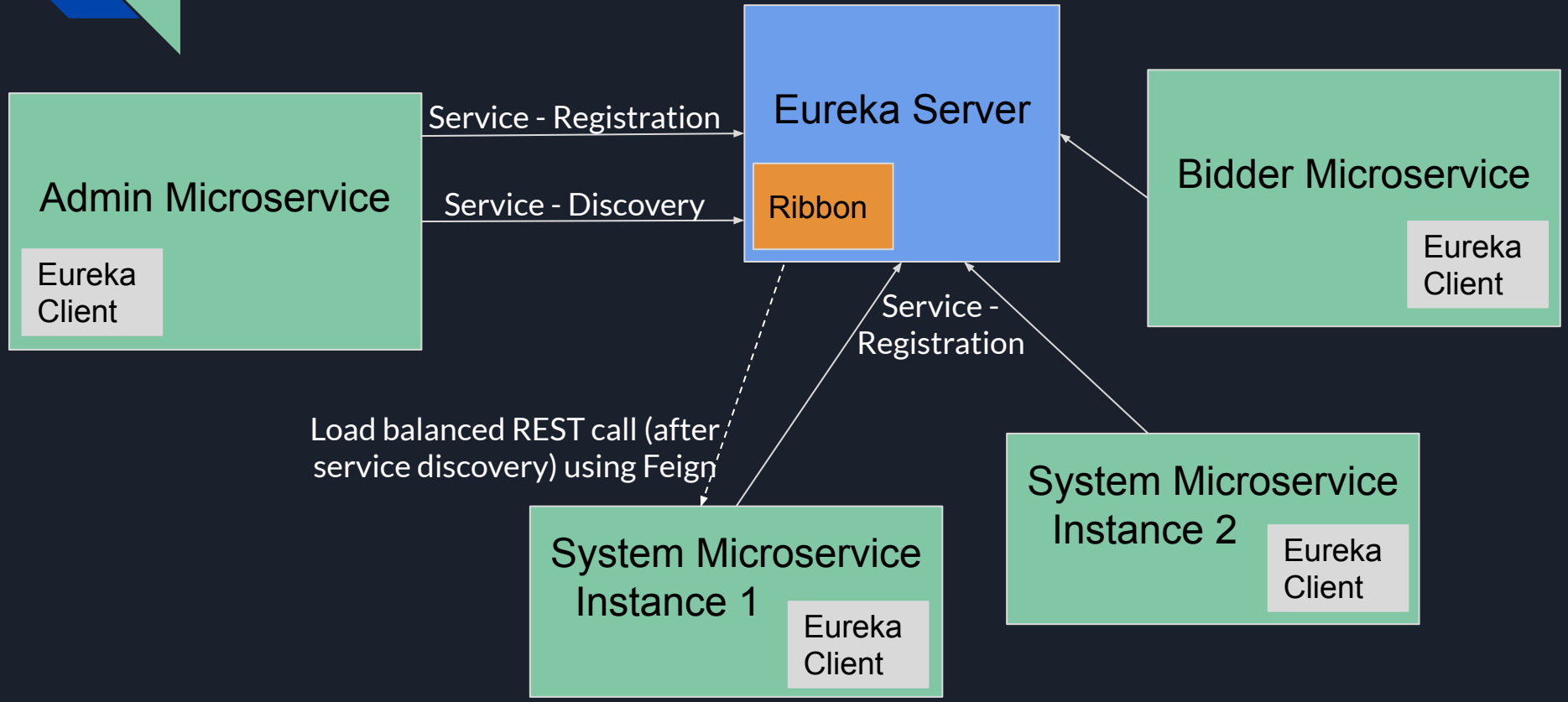
Technology Details - Sprint 2

- Authentication and Authorization: *Spring Starter Security*
- JWT Parser: *JJWT*
- Mail: *Spring Mail*
- Unit Testing: *JUnit 5 & Mockito*
- Static Code Analysis: *SonarQube*
- Automation Testing: *Selenium*
- UI: *ReactJs ,Bootstrap, Material UI*
- Cloud deployment: *AWS EC2 instances, Amazon RDS*

Data flow


ADMIN UI
BIDDER UI
Web App
(ReactJs)
Request
Response
System
Microservice
Bidder
Microservice
Admin
Microservice
Server - IPL Fantasy
League Application
Query
Data
Bidders table
Admins table
Leaderboards table
Teams table
Tournaments table
Other tables
Database - MySQL

Microservices pattern



Special Highlights and Suggestions

- Code Review
- Collaborative work on Github
- Deletion of a Record

Code Testing

- Selenium Testing using TestNG
- Static Code Analysis using SonarQube

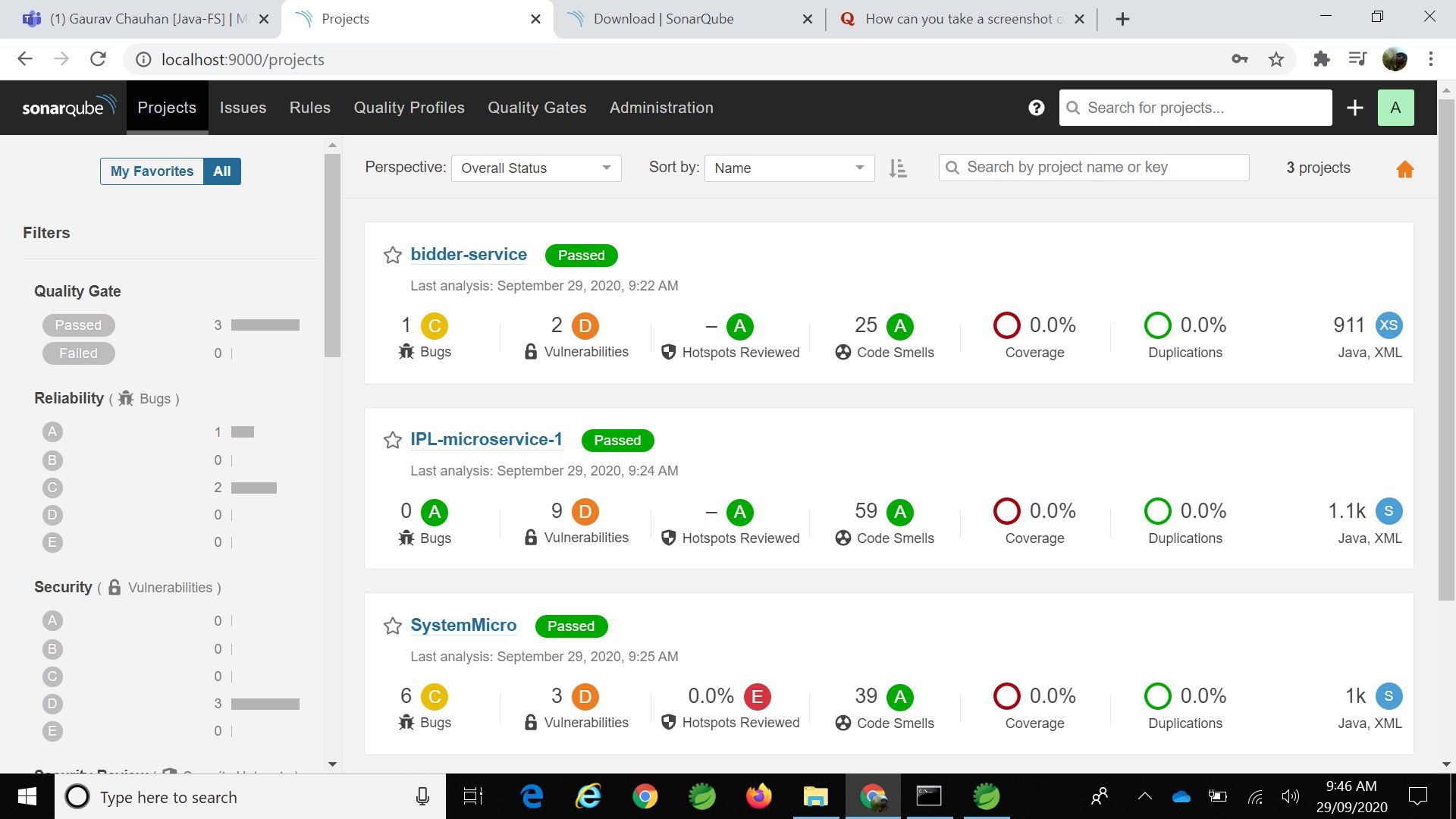
Previous Progress

API documentation - bit.ly/ipl-fl

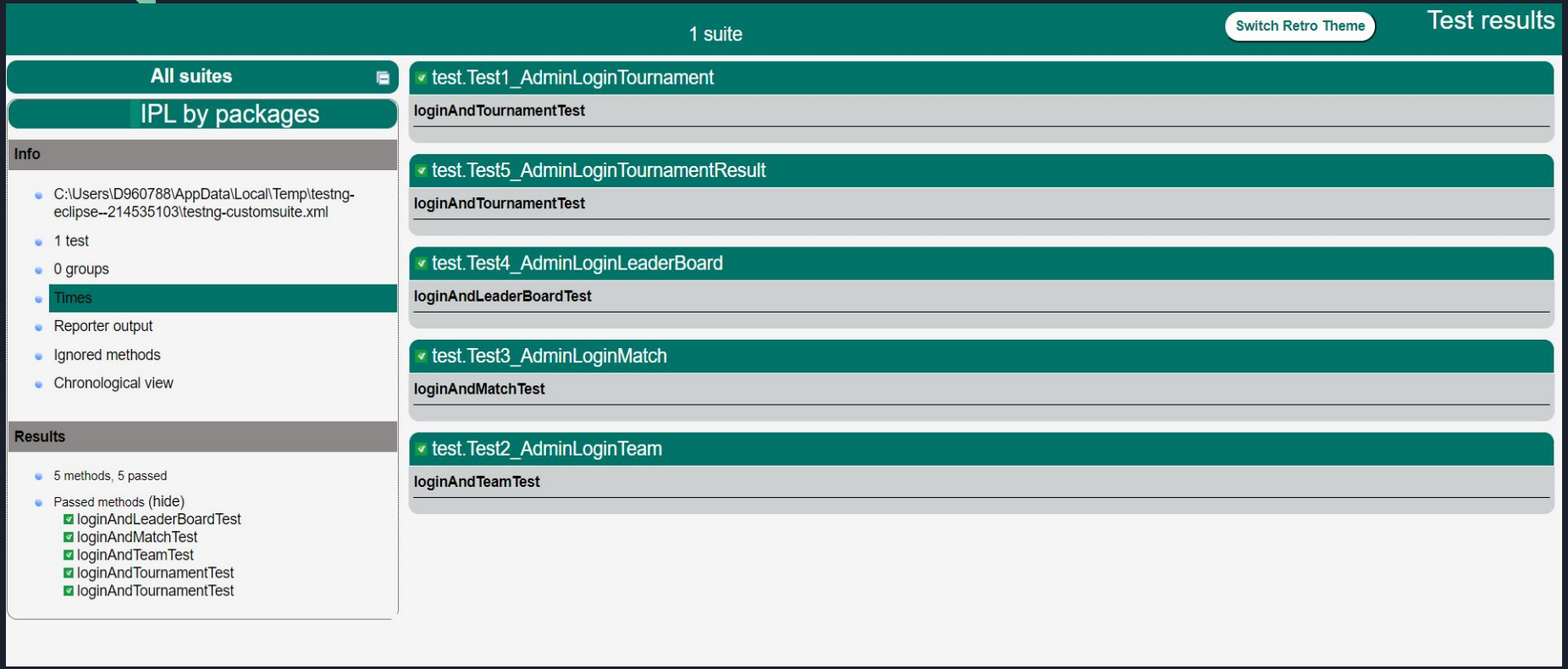
Features implemented during this sprint

- Authentication using Spring Security and saving state (in JWT) using cookies on the client-side
- Adding UI using ReactJS
- Front-end validations for the data
- Testing and analysis
- Cloud deployment

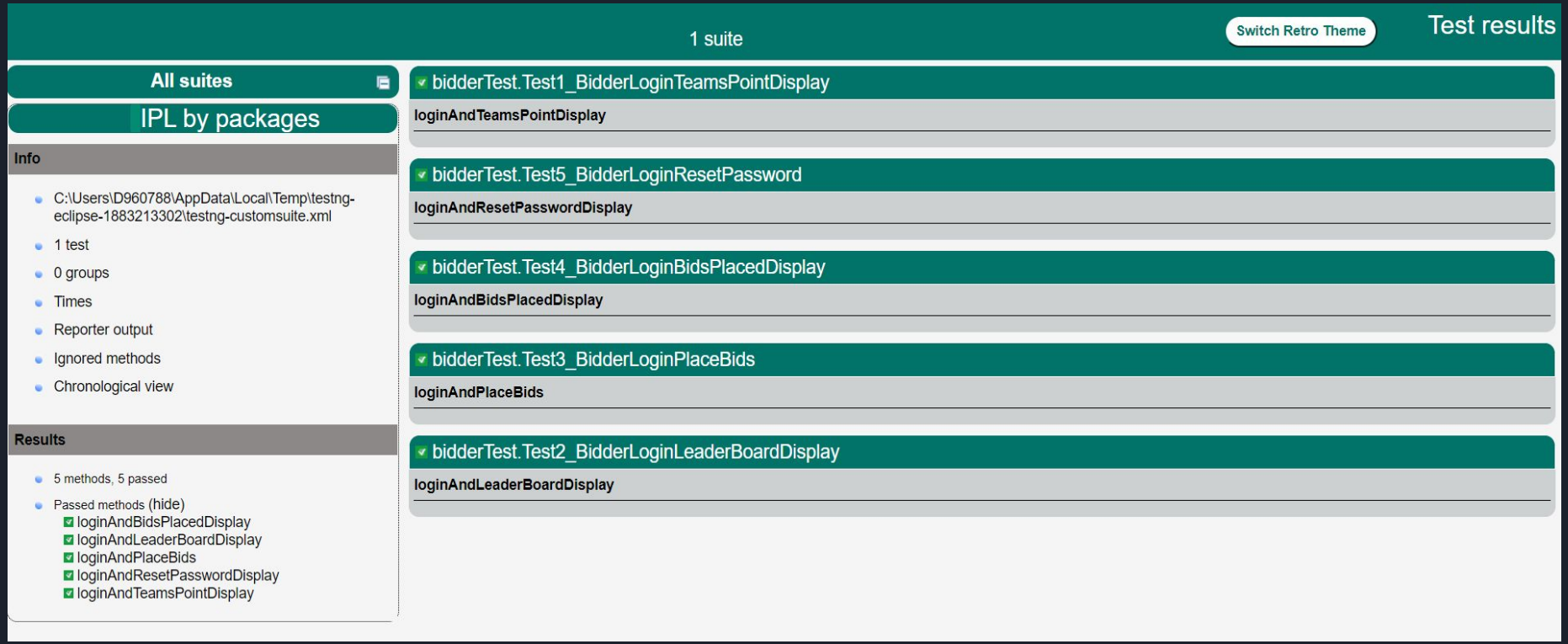
DEMONSTRATION


(1) Gaurav Chauhan [Java-FS] | M
Projects
Download | SonarQube
How can you take a screenshot o
localhost:9000/projects
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
A
My Favorites
All
Perspective:
Overall Status
Sort by:
Name
Search by project name or key
3 projects
Filters
Quality Gate
Passed 3
Failed 0
Reliability (Bugs)
A 1
B 0
C 2
D 0
E 0
Security (Vulnerabilities)
A 0
B 0
C 0
D 3
E 0
bidder-service
Passed
Last analysis: September 29, 2020, 9:22 AM
1 C Bugs
2 D Vulnerabilities
– A Hotspots Reviewed
25 A Code Smells
0.0% Coverage
0.0% Duplications
911 XS Java, XML
IPL-microservice-1
Passed
Last analysis: September 29, 2020, 9:24 AM
0 A Bugs
9 D Vulnerabilities
– A Hotspots Reviewed
59 A Code Smells
0.0% Coverage
0.0% Duplications
1.1k S Java, XML
SystemMicro
Passed
Last analysis: September 29, 2020, 9:25 AM
6 C Bugs
3 D Vulnerabilities
0.0% E Hotspots Reviewed
39 A Code Smells
0.0% Coverage
0.0% Duplications
1k S Java, XML
Type here to search
9:46 AM
29/09/2020

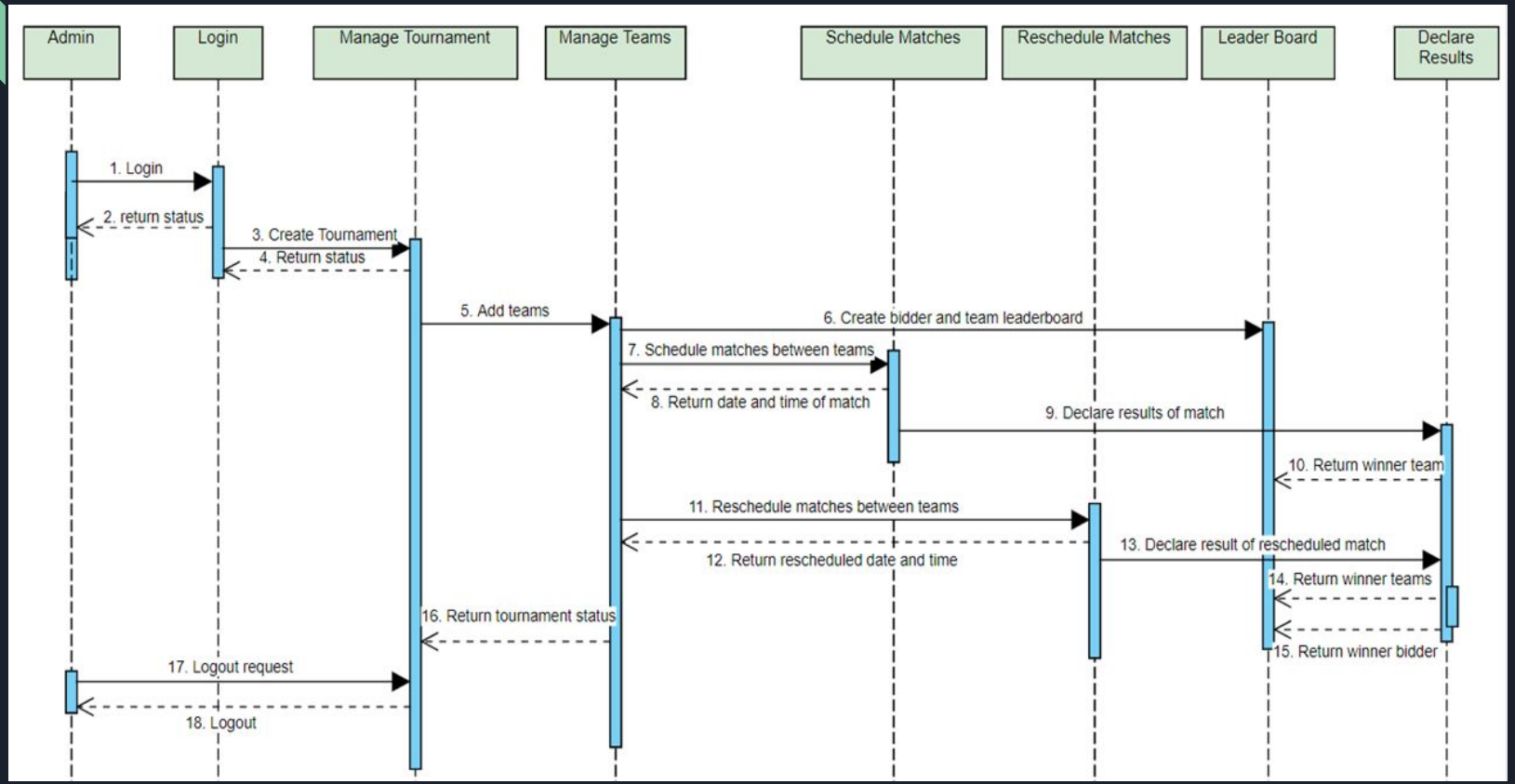
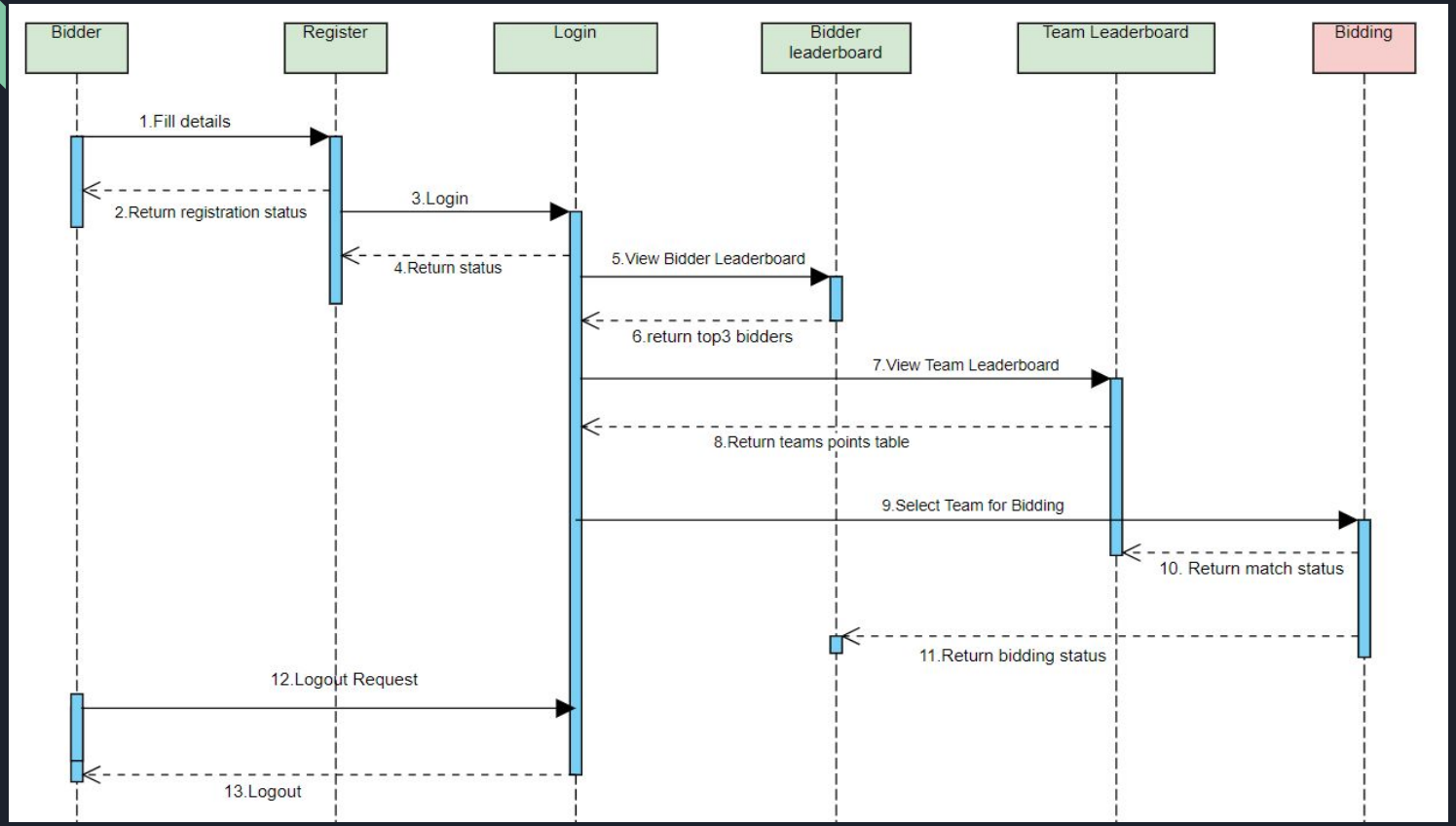
Admin Functionalities Tests Result



Bidder Functionalities Tests Result



Admin Microservice Sequence diagram



Bidder Microservice Sequence diagram