

# QUALITY ASSURANCE (QA) PLAN

Nisa Naz Keleşoğlu: 231101040

Zeynep Saçaklı: 221101051

Beril Aydın:231101002

## Task Matrix

Nisa Naz Keleşoğlu	Zeynep Saçaklı	Beril Aydın
Quality Assurance Strategy	Quality Factors & Metrics	Test Plan

## Table Of Contents

- Quality Assurance Strategy
  - Testing Methodologies
  - Automated vs. Manual Testing
- Quality Factors & Metrics
- Test Plan
  - Test Cases
  - Bug Tracking

# 1.Quality Assurance Strategy

## Overwiev

The testing and quality assurance process follows a structured methodology to ensure reliability, security, and usability. A combination of automated and manual testing techniques is used to validate different components of the system. The testing process begins with unit testing to verify individual modules, followed by integration testing to ensure seamless communication between different system components. Further, usability and performance testing are conducted to enhance user experience and optimize system efficiency. A continuous feedback loop, incorporating bug tracking and resolution, ensures that the application meets quality standards before deployment.

## Testing Methodologies

- **Unit Testing:** Jest for frontend and backend testing. Ensures individual components function as expected in isolation.
- **Integration Testing:** API testing using Postman. Verifies that different modules interact correctly.
- **System Testing:** Evaluates the entire application to ensure all components function together correctly.
- **Regression Testing:** Ensures that new code changes do not introduce bugs in previously working functionality.
- **Security Testing:** Identifies vulnerabilities and ensures data protection.
- **Acceptance Testing:** Ensures that the final product meets business requirements and user expectations.

## Automated vs. Manual Testing

Test Type	Automated	Manual
Unit Tests	Jest for frontend and backend testing	Not applicable
API Testing	Automated scripts using Postman	Not applicable
UI Testing	Selenium for automated UI validation	Manual user interface validation
Security Testing	Automated vulnerability scanning	Manual penetration testing
Regression Testing	Automated test suites to validate previous functionality	Manual verification for complex scenarios
Acceptance Testing	Not applicable	Final validation based on business requirements

## 2. Quality Factors and Metrics

Quality Factor	Description	Measurement Metric
Performance	Response time	< 5 seconds
Security	Secure authentication	No security issues in audit
Usability	User-friendliness	80%+ satisfaction
Reliability	System uptime	99.9% uptime

## 3. Test Plan

### Test Cases

Test Name	Description	Expected Outcome
Search Recipe by Ingredients	User enters ingredients and searches for recipes	System returns recipes matching the given ingredients
Search Recipe by Name	User enters a recipe name and searches	System returns recipes matching the entered name
Save to Favorites	User saves a recipe to favorites	Recipe appears in the favorites section
Dietary Filter	User applies dietary filters (e.g., vegan, gluten-free)	System returns only recipes matching selected dietary preferences
User Authentication	User logs in or signs up with e-mail and password	System securely logs in or registers the user

### Bug Tracking

- Bugs will be tracked using GitHub Issues.
- Each bug report must include:
  - Steps to reproduce
  - Expected vs. actual outcome
  - Severity level