

BACKEND TESTER DOCUMENTATION

1. Test Strategy (Backend)

The purpose of backend testing is to verify that all API endpoints function correctly according to role-based authorization rules, data integrity constraints, and business logic defined in the system.

Tested Roles

- another
- student
- advisor
- admin

Tested Features

- Authentication & Authorization
- Thesis management
- Role-based access control
- Comments & approval logic
- Statistics & analytics
- Favorites, downloads, citations

2. Test Environment (Backend)

Component	Description
Runtime	Node.js
Framework	Express.js
Database	PostgreSQL
Testing Tool	Postman
OS	Windows
Server Port	5000

3. Backend Test Cases

ID	Endpoint	Method	Role	Test Scenario	Expected Result	Status
BE-01	/api/auth/login	POST	All	Valid login	JWT token returned	Pass
BE-02	/api/auth/login	POST	All	Invalid password	401 Unauthorized	Pass
BE-03	/api/theses	POST	Admin	Upload thesis	Thesis created	Pass
BE-04	/api/theses	POST	Student	Upload thesis	403 Forbidden	Pass
BE-05	/api/theses/{id}	DELETE	Admin	Delete thesis	Thesis deleted	Pass
BE-06	/api/theses/{id}	DELETE	Advisor	Delete thesis	Access denied	Pass
BE-07	/api/theses	GET	All	List theses	Thesis list returned	Pass
BE-08	/api/theses/{id}/assign	POST	Admin	Assign student & advisor	Assignment successful	Pass
BE-09	/api/comments	POST	Advisor	Add comment	Comment saved	Pass
BE-10	/api/comments	POST	Student	Add comment	Pending approval	Pass
BE-11	/api/comments/{id}/approve	PUT	Admin	Approve comment	Comment visible	Pass

BE - 12	/api/comments/{id}	DELETE	Admin	Delete comment	Comment removed	Pass
BE - 13	/api/analytics	GET	Admin	View site stats	Statistics returned	Pass
BE - 14	/api/analytics	GET	Advisor	View thesis stats	Limited stats returned	Pass
BE - 15	/api/favorites	POST	All	Add thesis to favorites	Favorite saved	Pass

4. Backend Test Results

- Role-based access control works correctly.
- Unauthorized actions are blocked successfully.
- Data integrity is preserved in all operations.
- Business rules (comment approval, thesis upload restrictions) are enforced