

# Documentation du Processus de Test Backend

## 1. Implémentez au moins un test unitaire dans le backend avec Mocha/Chai

Installation des dépendances de test

```
npm install --save-dev mocha chai supertest nyc
```

```
PS C:\Users\eiman\user-management-app-main> npm  
install --save-dev supertest  
>>
```

## 2. Fichiers de test

Testez une route ou une logique simple

test unitaire

```
JS user.test.js X
```

```
test > JS user.test.js > ...
```

```
1 // test/user.test.js
2 const chai = require('chai');
3 const expect = chai.expect;
4
5 describe('Exemple de test unitaire', () => {
6   it('devrait retourner vrai si 2 + 2 = 4', () => {
7     const result = 2 + 2;
8     expect(result).to.equal(4);
9   });
10 });
11
```

```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + v [ ] [ ] ...
PS C:\Users\eiman\user-management-app-main> npm
test
>>

> user-management-app@1.0.0 test
> mocha ./test/**/*.test.js

Connecté à la base de données SQLite.

✓ devrait retourner vrai si 2 + 2 = 4
```

## test/userRoutes.test.js (test d'intégration)

```
test > JS userRoutes.test.js > describe('Tests d'intégration des routes utilisateurs') cal
1 const chai = require('chai');
2 const chaiHttp = require('chai-http');
3 const app = require('../server'); // Assurez-vous que
4 const expect = chai.expect;
5
6 chai.use(chaiHttp);
7
8 describe('Tests d'intégration des routes utilisateurs') {
9   it('GET /users doit retourner un tableau', (done) => {
10     chai.request(app)
11       .get('/users')
12       .end((err, res) => {
13         expect(res).to.have.status(200);
14         expect(res.body).to.be.an('array');
15         done();
16       });
17   });
18 };
```

```
PS C:\Users\eiman\user-management-app-main> npm
test
>>

> user-management-app@1.0.0 test
> mocha ./test/**/*.test.js

Connecté à la base de données SQLite.

✓ devrait retourner vrai si 2 + 2 = 4

Tests d'intégration des routes utilisateurs
✓ GET /users doit retourner un tableau (41ms)
)

2 passing (53ms)
```

## Configurez la couverture de code

```
Run npm audit for details.
PS C:\Users\eiman\user-management-app-main> npm
run coverage
>>

> user-management-app@1.0.0 coverage
> nyc npm test

> user-management-app@1.0.0 test
> mocha ./test/**/*.test.js

Connecté à la base de données SQLite.

Exemple de test unitaire
✓ devrait retourner vrai si 2 + 2 = 4

Tests d'intégration des routes utilisateurs
✓ GET /users doit retourner un tableau (55ms)
)

2 passing (74ms)

-|-----|-----|-----|-----|-----
-|-----|-----|-----|-----|-----
vered Line #s
-|-----|-----|-----|-----|-----
-|-----|-----|-----|-----|-----
| 36.36 | 15.62 | 23.07 | 36.36 |
| 36.36 | 15.62 | 23.07 | 36.36 | ...
07-116,125-126
-|-----|-----|-----|-----|-----
-|-----|-----|-----|-----|-----
```

```
er.test.js  {} package.json X
package.json > ...
{
  "name": "user-management-app",
  "version": "1.0.0",
  "main": "server.js",
  Debug
  "scripts": {
    "start": "node server.js",
    "test": "mocha ./test/**/*.test.js",
    "coverage": "nyc npm test"
  },
}
```