



# ShareUp - Project Modules & Team Assignment



## Final Module List (What to Build)

You will build **6 main backend modules** and **1 frontend module**:

1. Authentication Module
2. User / Profile Module
3. Item Management Module
4. Rental (Borrow & Return) Module
5. Rating & Penalty Module
6. Notification (Email) Module
7. Frontend Module

## Team Assignment

| Member   | Assigned Modules                       |
|----------|--|
| Member 1 | Authentication + User/Profile          |
| Member 2 | Item Management + Frontend             |
| Member 3 | Rental + Rating/Penalty + Notification |



## MODULE 1: Authentication Module

**Owner:** Member 1

**Service:** auth-service

### What to Build

- User Registration (Owner / Borrower)
- Login
- JWT Token Generation
- Role Handling (OWNER / BORROWER)

### Database Table – User

| Field |
|-------|
| id    |

| Field    |
|----------|
| name     |
| email    |
| password |
| role     |

## APIs

- POST /register
- POST /login
- GET /me

## Tech Stack

- Spring Boot
- Spring Security
- JWT
- BCrypt Password Encoder

JWT token must be returned on successful login.

---

## MODULE 2: User / Profile Module

**Owner:** Member 1

**Service:** auth-service (same service as authentication)

## What to Build

- Store pickup address (for item owners)
- View profile
- Update profile

## APIs

- GET /profile
  - PUT /profile
- 



## MODULE 3: Item Management Module

**Owner:** Member 2

**Service:** item-service

## What to Build

- Owner adds items
- Browse all items
- Filter items by category
- Update item status
- Upload item image

## Item Fields

- Name
- Category
- Price
- Image
- Pickup Address
- Status

## Database Table – Item

| Field         |
|---------------|
| id            |
| ownerId       |
| name          |
| category      |
| price         |
| imageUrl      |
| status        |
| pickupAddress |

## APIs

- POST /items
- GET /items
- GET /items/category/{category}
- PUT /items/{id}/status

## Image Upload Options

- Cloudinary
- OR
- Multipart file upload



# MODULE 4: Rental (Borrow & Return) Module

**Owner:** Member 3

**Service:** rental-service

## Borrow Flow

1. Borrower sends request
2. Owner approves or rejects
3. Item status updates

## Return Flow

1. Borrower uploads return image
2. Owner approves return

## Database Table – RentalRequest

| Field          |
|----------------|
| id             |
| itemId         |
| borrowerId     |
| ownerId        |
| status         |
| returnImageUrl |

## APIs

- POST /rentals/request
- PUT /rentals/{id}/approve
- PUT /rentals/{id}/reject
- POST /rentals/{id}/return
- PUT /rentals/{id}/approve-return

## Service Communication

Rental Service communicates with Item Service:

- GET /items/{id}
- PUT /items/{id}/status



# MODULE 5: Rating & Penalty Module

**Owner:** Member 3

**Service:** rental-service or separate service

## What to Build

- Borrower rates owner
- Owner rates borrower
- Owner applies penalty

## Database Tables

### Rating

| Field      |
|------------|
| id         |
| fromUserId |
| toUserId   |
| score      |
| review     |

### Penalty

| Field    |
|----------|
| id       |
| rentalId |
| amount   |
| reason   |

## APIs

- POST /ratings
- POST /penalties
- GET /ratings/{userId}



## MODULE 6: Notification Module

**Owner:** Member 3

**Service:** notification-service

### What to Build

Send email notifications for:

- New rental request (to owner)
- Approval (to borrower)
- Return request
- Return approved

### Tech Stack

- Spring Boot
- JavaMailSender
- Gmail SMTP

### API

- POST /notify/email

Rental Service will call this API.

---



## FRONTEND MODULE

**Owner:** Member 2

**Service:** React Application

### Owner Pages

- Login
- Dashboard
- Add Item
- Requests
- My Items

### Borrower Pages

- Login
- Home
- Category Filter
- My Rentals

- Return Request
- Rating

## Tech Stack

- React
  - Axios
  - React Router
  - JWT Storage (LocalStorage / SessionStorage)
-