

SISTEM INVENTARISASI UKM PANAHAH

Mata Kuliah: Pemrograman Web Lanjut



Disusun Oleh:

Ajeng Sakinah Wulandari	F1D022001
Ida Bagus Brahmanta Jayana	F1D022052
Sucitasari Rahmadani	F1D02310138

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MATARAM
TAHUN 2025**

BAB I - DESKRIPSI PROJECT

1. Latar Belakang

Perkembangan teknologi informasi yang semakin pesat telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk dalam pengelolaan organisasi kemahasiswaan. Unit Kegiatan Mahasiswa (UKM) Panahan Universitas Mataram sebagai salah satu wadah pengembangan minat dan bakat mahasiswa memiliki berbagai aset inventaris yang perlu dikelola dengan baik, mulai dari busur (bow), anak panah (arrow), target, hingga perlengkapan keselamatan lainnya..

Pada praktiknya, pengelolaan inventaris UKM masih dilakukan secara manual menggunakan pencatatan di buku atau spreadsheet sederhana. Metode ini menimbulkan berbagai permasalahan seperti kesulitan dalam monitoring stok barang, pencatatan kondisi peralatan yang tidak terstruktur, serta sulitnya melacak riwayat pembelian dan pemeliharaan aset. Kondisi ini dapat berdampak pada hilangnya data inventaris, kesulitan dalam perencanaan pengadaan barang, serta kurang optimalnya pengelolaan aset organisasi.

Berdasarkan permasalahan tersebut, dikembangkan sebuah Sistem Inventarisasi UKM Panahan berbasis web. Sistem ini dirancang untuk membantu pengurus UKM dalam mencatat, mengelola, dan memantau seluruh aset inventaris secara digital. Dengan adanya sistem ini, diharapkan pengelolaan inventaris dapat dilakukan secara lebih efisien, terstruktur, dan transparan.

Pengembangan sistem berbasis web dipilih karena kemudahan aksesnya yang dapat digunakan kapan saja dan di mana saja melalui perangkat yang terhubung dengan internet. Selain itu, sistem ini juga diharapkan dapat menjadi solusi digital yang mendukung tata kelola organisasi yang lebih profesional dan akuntabel di lingkungan perguruan tinggi.

2. Tujuan Project

Tujuan dari pengembangan Sistem Inventarisasi UKM Panahan ini adalah sebagai berikut:

1. Mengembangkan sistem berbasis web yang dapat membantu pengurus UKM dalam mengelola inventaris secara terstruktur dan terintegrasi.
2. Menyediakan platform digital untuk pencatatan data barang, kategori, stok, kondisi, dan harga peralatan.
3. Membantu pengurus memantau kondisi dan ketersediaan barang inventaris secara real-time.
4. Mengurangi risiko kehilangan data dan kesalahan pencatatan akibat pengelolaan manual.

5. Meningkatkan transparansi dan akuntabilitas dalam pengelolaan aset organisasi.
3. Teknologi yang Digunakan (*Backend, Frontend, Database*)
 1. *Backend*

Backend berfungsi sebagai penyedia layanan API dan logika utama aplikasi. Teknologi yang digunakan meliputi:

 - a. Node.js & Express.js

Digunakan sebagai kerangka kerja utama server untuk menangani permintaan HTTP dan *routing*.
 - b. JSON Web Token (JWT)

Digunakan untuk autentikasi pengguna yang aman (sistem *login* dan proteksi *route*).
 - c. Bcryptjs

Digunakan untuk enkripsi (*hashing*) kata sandi pengguna sebelum disimpan ke dalam database demi keamanan.
 - d. MySQL2

Library driver untuk menghubungkan aplikasi Node.js dengan database MySQL.
 - e. CORS

Middleware untuk mengatur Cross-Origin Resource Sharing, memungkinkan frontend dan backend berkomunikasi meskipun berada di domain berbeda.
 - f. Dotenv

Untuk manajemen variabel lingkungan (*environment variables*).
 2. *Frontend*

Frontend bertugas menangani antarmuka pengguna (*User Interface*) dan interaksi pengguna. Teknologi yang digunakan meliputi:

 - a. Vue.js 3

Kerangka kerja JavaScript utama untuk membangun antarmuka yang reaktif dan berbasis komponen.
 - b. Vite

Build tool yang digunakan untuk mempercepat proses pengembangan dan kompilasi aset.
 - c. Vue Router

Mengatur navigasi antar halaman (seperti *Dashboard, Login, Categories, Items*) tanpa memuat ulang halaman (*Single Page Application*).

d. Axios

Klien HTTP untuk melakukan permintaan data (GET, POST, PUT, DELETE) ke server *Backend*.

e. Bootstrap 5

Framework CSS untuk styling dan komponen UI yang responsif dan modern.

f. Pinia

State management library untuk Vue.js, digunakan untuk mengelola state global aplikasi (*authentication state*).

3. *Database*

a. MySQL

Sistem manajemen basis data relasional (RDBMS) yang digunakan untuk menyimpan data pengguna (*users*), kategori (*categories*), dan barang inventaris (*items*). Struktur database mendukung relasi antar tabel (seperti *Foreign Key* *category_id* pada tabel *items*) untuk menjaga integritas data.

BAB II - PERANCANGAN SISTEM

1. Deskripsi Singkat Sistem

Sistem yang dirancang dalam proyek ini adalah sebuah sistem inventarisasi berbasis web yang bertujuan untuk membantu pengurus UKM Panahan dalam mengelola aset organisasi secara digital dan terstruktur. Aplikasi ini dibangun menggunakan arsitektur Client-Server yang memisahkan logika bisnis di sisi backend (menggunakan Node.js dan Express.js) dengan antarmuka pengguna yang responsif di sisi frontend (menggunakan Vue.js), serta memanfaatkan MySQL sebagai basis data relasional untuk penyimpanan informasi.

Demi menjaga privasi dan keamanan data, sistem menerapkan mekanisme autentikasi berbasis JSON Web Token (JWT), di mana setiap pengguna diwajibkan melakukan registrasi dan login untuk dapat mengakses sistem. Sistem ini membedakan dua tingkat hak akses: **Pengurus** yang memiliki akses penuh (CRUD - Create, Read, Update, Delete) dan **Anggota** yang hanya dapat melihat data inventaris (Read Only).

Secara fungsional, sistem ini menyediakan fitur pengelolaan kategori barang dan data inventaris yang komprehensif. Pengurus dapat menambahkan kategori baru (seperti Busur, Anak Panah, Target & Bantalan, Perlengkapan Safety), mengelola data barang lengkap dengan atribut seperti nama, stok, harga, tanggal pembelian, kondisi barang, dan deskripsi. Sistem juga dilengkapi dengan dashboard yang menampilkan informasi umum dan navigasi menu yang user-friendly.

Untuk mendukung transparansi dan akuntabilitas, semua anggota dapat melihat data inventaris yang tersedia, sehingga memudahkan koordinasi dalam penggunaan peralatan. Sistem ini dirancang dengan antarmuka yang modern dan responsif menggunakan Bootstrap 5, sehingga dapat diakses dengan nyaman dari berbagai perangkat.

2. Desain Database (ERD dan penjelasan singkat tabel)

Gambar 1. *Entity Relationship Diagram*

Berdasarkan Entity Relationship Diagram (ERD) yang dirancang, sistem ini memiliki tiga tabel utama, yaitu **users**, **categories**, dan **items**. Tabel **users** digunakan untuk

menyimpan data pengguna sistem, yang terdiri dari pengurus dan anggota UKM. Data yang tersimpan meliputi username sebagai identitas login, password yang telah dienkripsi menggunakan bcrypt, nama lengkap (full_name), nomor telepon (opsional), role untuk membedakan hak akses (pengurus/anggota), serta timestamp pembuatan dan pembaruan data. Tabel ini berperan sebagai entitas utama untuk sistem autentikasi dan otorisasi.

Tabel categories digunakan untuk menyimpan jenis-jenis kategori barang inventaris. Setiap kategori memiliki nama (name), deskripsi (description), kode unik (code) untuk identifikasi cepat, status aktif/non-aktif (is_active), serta timestamp. Tabel ini berfungsi sebagai pengklasifikasi barang-barang inventaris untuk memudahkan pengelompokan dan pencarian.

Selanjutnya, tabel items digunakan untuk menyimpan data detail barang inventaris. Setiap item terhubung dengan satu kategori melalui atribut category_id (Foreign Key), sehingga menunjukkan bahwa satu kategori dapat memiliki banyak item (relasi one-to-many). Data yang disimpan meliputi nama barang, jumlah stok, harga satuan, tanggal pembelian, status kondisi barang (Baik/Rusak Ringan/Rusak Berat), deskripsi tambahan, serta timestamp. Tabel ini merupakan inti dari sistem inventarisasi.

Secara keseluruhan, relasi antar tabel dalam ERD ini membentuk hubungan one-to-many antara categories dan items. Struktur ini mendukung pengelolaan inventaris yang terorganisir, memudahkan pencatatan, monitoring, dan pelaporan aset UKM.

BAB III - IMPLEMENTASI

1. Implementasi *Backend* (struktur folder, API, *middleware*)

1. Struktur Folder



Gambar 2. Struktur Folder

Berikut adalah struktur direktori yang diterapkan:

a. config/

Berisi konfigurasi koneksi basis data menggunakan pustaka mysql2 untuk menghubungkan aplikasi dengan server database MySQL.

b. controllers/

Menyimpan logika bisnis utama. Setiap fungsi dalam controller bertanggung jawab menerima permintaan (*request*), memproses data, dan mengirimkan respons. Contohnya adalah authController.js untuk autentikasi, categoryController.js untuk manajemen kategori, dan itemController.js untuk manajemen barang.

c. middleware/

Berisi fungsi perantara yang dieksekusi sebelum mencapai controller utama, seperti verifikasi token JWT (auth.js), validasi input (validate.js), dan penanganan kesalahan global (errorHandler.js).

d. models/

Berisi fungsi-fungsi untuk berinteraksi dengan database, seperti query SELECT, INSERT, UPDATE, dan DELETE untuk setiap entitas (user, category, item).

e. routes/

Mendefinisikan titik akhir (endpoints) API dan memetakan metode HTTP (GET, POST, PUT, DELETE) ke fungsi controller yang sesuai.

f. index.js

Titik masuk utama aplikasi yang menginisialisasi server Express, mengatur CORS, dan memuat variabel lingkungan.

2. Implementasi API

Sistem menyediakan layanan RESTful API yang memungkinkan komunikasi data antara klien dan server. Berikut adalah daftar endpoint utama yang telah diimplementasikan:

1. Autentikasi (/auth)

- a. POST /auth/register - Mendaftarkan pengguna baru dengan enkripsi kata sandi.
- b. POST /auth/login - Memvalidasi kredensial dan menerbitkan JSON Web Token (JWT).

2. Manajemen Kategori (/api/categories)

- a. GET /api/categories - Mengambil semua data kategori (publik).
- b. GET /api/categories/:id - Mengambil detail kategori berdasarkan ID.
- c. POST /api/categories - Membuat kategori baru (protected, role: pengurus).
- d. PUT /api/categories/:id - Mengupdate data kategori (protected, role: pengurus).
- e. DELETE /api/categories/:id - Menghapus kategori (protected, role: pengurus)

3. Manajemen Barang (/api/items)

- a. GET /api/items - Mengambil semua data barang dengan informasi kategori (publik).
- b. GET /api/items/:id - Mengambil detail barang berdasarkan ID.
- c. POST /api/items - Membuat data barang baru (protected, role: pengurus).
- d. PUT /api/items/:id - Mengupdate data barang (protected, role: pengurus).
- e. DELETE /api/items/:id - Menghapus barang (protected, role: pengurus).

3. Penerapan Middleware

1. Autentikasi (Auth Middleware)

Middleware ini memverifikasi keberadaan dan kevalidan token JWT pada header permintaan. Jika token valid, informasi pengguna akan disisipkan ke dalam objek req.user, memastikan bahwa pengguna hanya dapat mengakses resource yang dilindungi jika sudah login.

2. Validasi Input (Validate Middleware)

Middleware ini memeriksa kelengkapan field yang wajib diisi sebelum data diproses oleh controller, mencegah error akibat data yang tidak lengkap.

3. Penanganan Kesalahan (Error Handler)

Middleware ini ditempatkan di akhir rantai eksekusi aplikasi untuk menangkap kesalahan yang tidak terduga dan mengembalikan respons format JSON yang standar kepada klien.

2. Implementasi *Frontend* (halaman, *routing*, konsumsi API)

1. Halaman (Views)

Antarmuka aplikasi dibagi menjadi beberapa komponen halaman utama (*Views*) yang berada di direktori src/views:

a. LoginView.vue

Halaman formulir login dengan validasi username dan password. Menampilkan pesan error jika kredensial salah dan link ke halaman registrasi.

b. RegisterView.vue

Halaman pendaftaran anggota baru dengan opsi mendaftar sebagai pengurus (memerlukan kode rahasia). Sistem role-based registration yang membedakan hak akses sejak pendaftaran.

c. DashboardView.vue

Halaman untuk mengelola kategori barang inventaris. Menampilkan tabel kategori dengan informasi nama, deskripsi, kode, dan status. Pengurus dapat menambah, edit, dan hapus kategori melalui modal form.

d. CategoriesView.vue

Halaman untuk mengelola kategori barang inventaris. Menampilkan tabel kategori dengan informasi nama, deskripsi, kode, dan status. Pengurus dapat menambah, edit, dan hapus kategori melalui modal form.

e. ItemsView.vue

Halaman untuk mengelola data barang inventaris. Menampilkan tabel barang lengkap dengan kategori, stok, kondisi, dan harga. Pengurus dapat melakukan CRUD operations, sedangkan anggota hanya dapat melihat data.

2. Manajemen Routing

Navigasi antar halaman dikelola oleh Vue Router yang memungkinkan perpindahan tampilan tanpa memuat ulang halaman browser (SPA). Konfigurasi rute didefinisikan dalam berkas router/index.js. Sistem routing dilengkapi dengan

Navigation Guard yang memeriksa autentikasi pengguna sebelum mengakses halaman yang dilindungi.

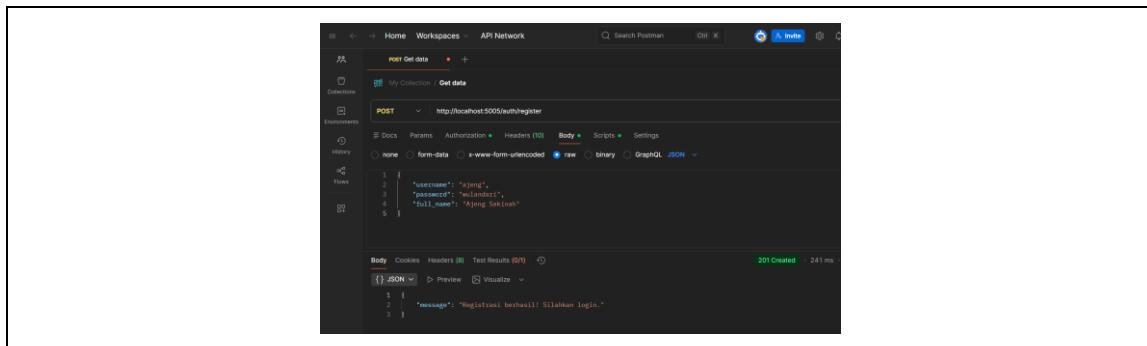
3. Konsumsi API dan Integrasi Data

Komunikasi data dengan backend dilakukan menggunakan pustaka HTTP Client **Axios**. Pola integrasi yang diterapkan meliputi:

- a. Penyisipan Token Otomatis: Pada halaman yang membutuhkan otorisasi (seperti create, update, delete), token JWT yang tersimpan di localStorage diambil dan disisipkan ke dalam header Authorization pada setiap permintaan HTTP.
- b. State Management dengan Pinia: State global seperti token dan data user dikelola menggunakan Pinia store (authStore.js), sehingga dapat diakses dari berbagai komponen.
- c. Reaktivitas Data: Data yang diterima dari API disimpan dalam variabel reaktif Vue (ref()). Perubahan pada data ini secara otomatis memicu pembaruan tampilan (DOM), seperti saat data kategori atau barang dimuat atau diperbarui.
- d. Role-Based UI Rendering: Tampilan tombol aksi (tambah, edit, hapus) ditampilkan secara kondisional berdasarkan role user:

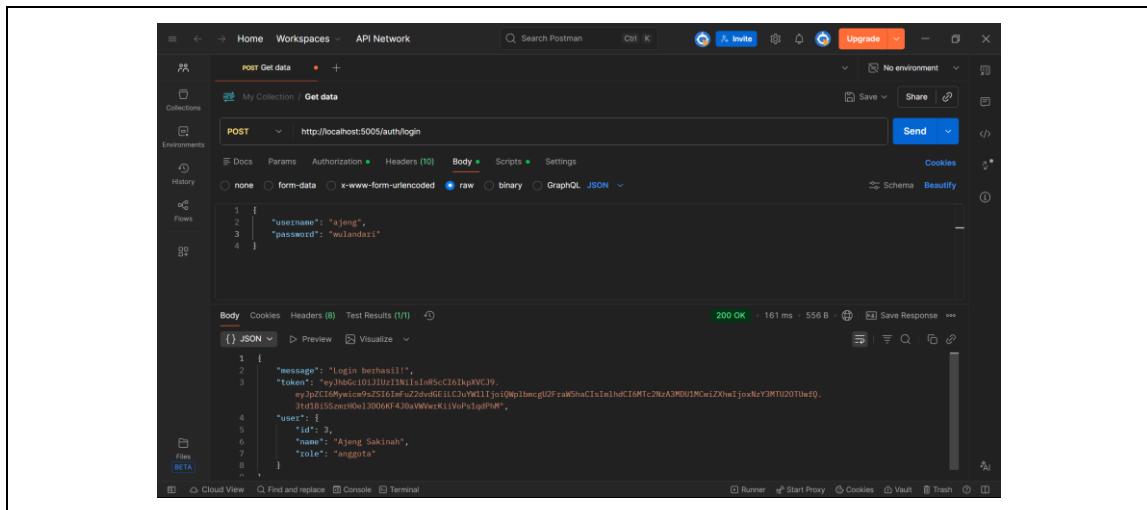
BAB IV - PENGUJIAN

1. Pengujian API Menggunakan Postman



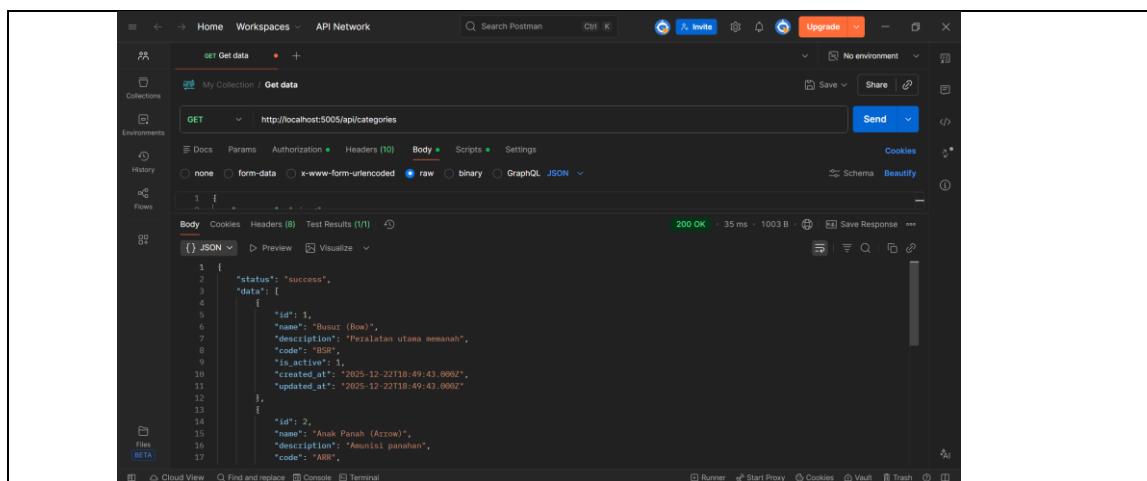
Gambar 3. Tampilan *Register User* pada Postman

Pengujian endpoint **POST /auth/register**.



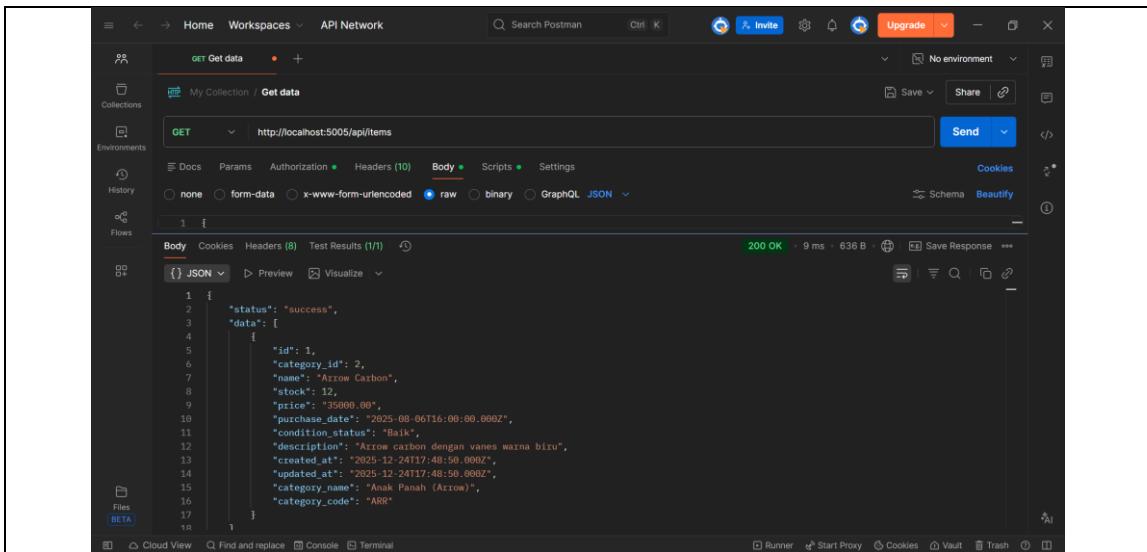
Gambar 4. Tampilan *Login User* pada Postman

Gambar di atas menunjukkan hasil pengujian endpoint **POST /auth/login**.



Gambar 5. Tampilan *Get All Categories* pada Postman

Gambar di atas menampilkan hasil pengujian endpoint **GET /api/items**.

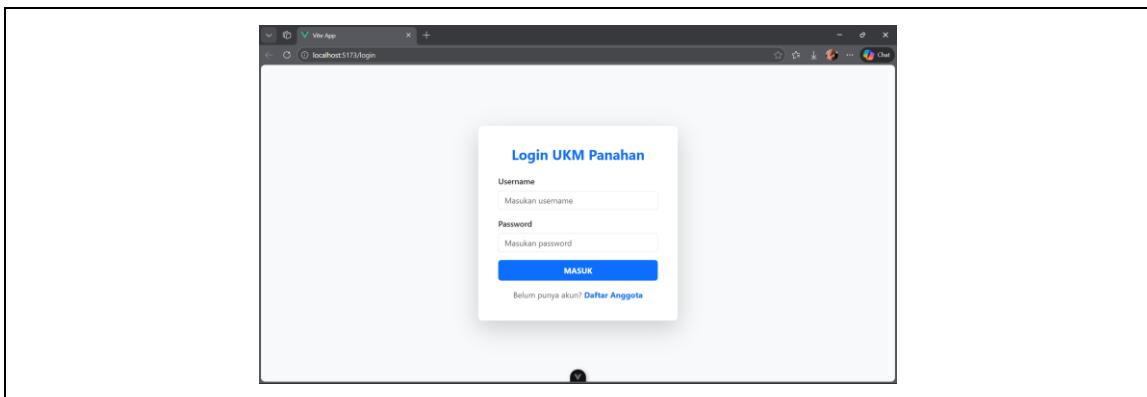


```
1 {
  "status": "success",
  "data": [
    {
      "id": 1,
      "category_id": 2,
      "name": "Arrow Carbon",
      "stock": 12,
      "price": "156000.00",
      "purchase_date": "2025-08-06T16:00:00.000Z",
      "condition_status": "Baru",
      "description": "Arrow carbon dengan vanes warna biru",
      "created_at": "2025-12-24T17:48:58.000Z",
      "updated_at": "2025-12-24T17:48:58.000Z",
      "category_name": "Awan Panah (Arrow)",
      "category_code": "ARR"
    }
  ]
}
```

Gambar 6. Tampilan *Get All Items* pada Postman

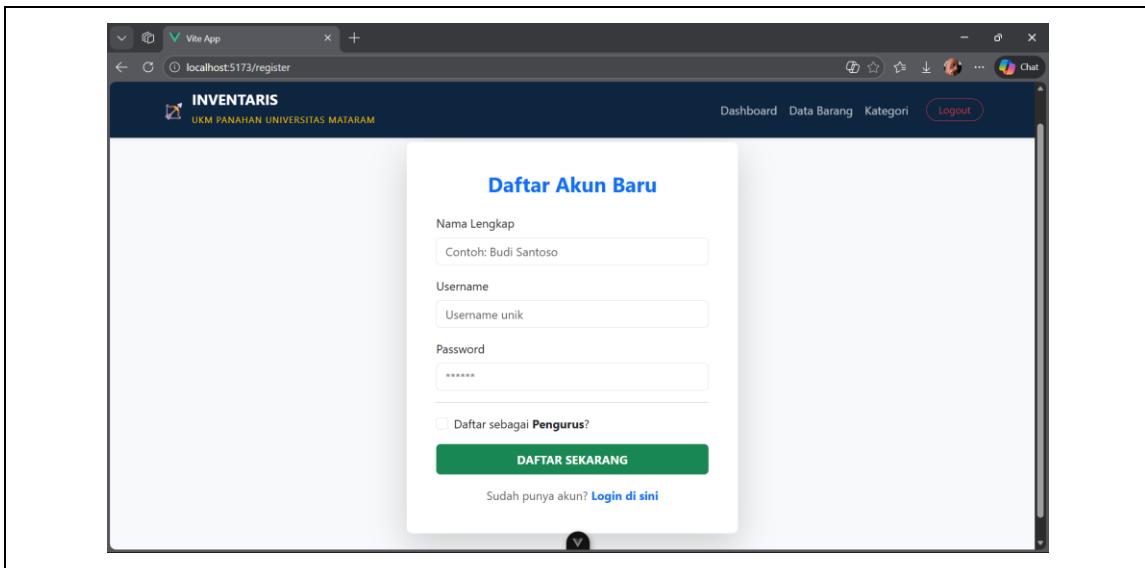
Gambar di atas menampilkan hasil pengujian endpoint **GET/api/items**.

2. Pengujian Aplikasi (*screenshoot* fitur utama)



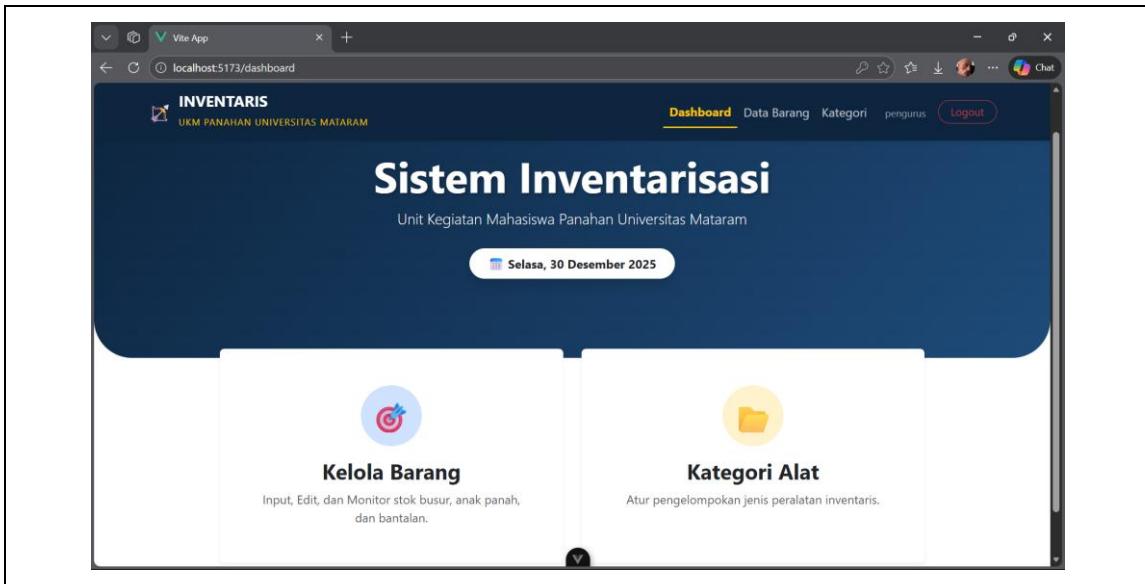
Gambar 7. Tampilan Website Halaman *Login*.

Halaman login menampilkan form *input username* dan *password* dengan validasi. Terdapat link ke halaman registrasi untuk pengguna baru. Pesan error ditampilkan jika kredensial salah. Dengan berbagai fitur yaitu Input username dan password, Validasi form, Error handling dengan pesan yang jelas, Link ke halaman register, Design modern dengan card shadow.



Gambar 8. Tampilan Website Halaman *Register*

Halaman registrasi untuk mendaftarkan anggota baru dengan opsi mendaftar sebagai pengurus. Pengurus memerlukan kode rahasia (UKM2025) untuk validasi role. Dengan berbagai fitur, seperti, Input nama lengkap, username, dan password, Checkbox untuk mendaftar sebagai pengurus, Input kode rahasia yang hanya muncul jika memilih role pengurus, Validasi kode rahasia, Feedback success/error message, Link ke halaman login.



Gambar 9. Tampilan Website Halaman *Dashboard*

Halaman utama setelah login yang menampilkan hero section dengan gradient background dan card navigasi ke menu utama. Terdapat fitur-fitur yaitu, Hero section dengan informasi tanggal real-time, Card navigasi “Kelola Barang” dengan icon , Card navigasi “Kategori Alat” dengan icon , Hover effect pada card, Gradient background yang menarik, Responsif untuk berbagai ukuran layer.

No	Nama	Deskripsi	Kode	Status	Aksi
1	Busur (Bow)	Peralatan utama memanah	BSR	Aktif	<button>Edit</button> <button>Hapus</button>
2	Anak Panah (Arrow)	Amunisi panahan	ARR	Aktif	<button>Edit</button> <button>Hapus</button>
3	Target & Bantalan	Sasaran tembak	TRG	Aktif	<button>Edit</button> <button>Hapus</button>
4	Perlengkapan & Safety	Pelindung diri	SFT	Aktif	<button>Edit</button> <button>Hapus</button>

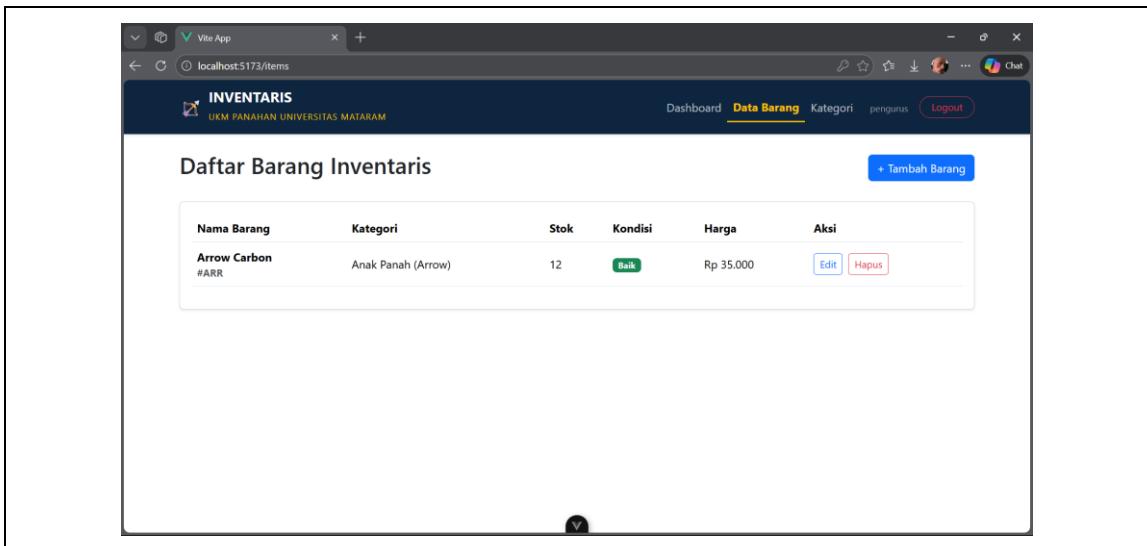
Gambar 10. Tampilan Website Halaman *Categories* (Pengurus View)

Halaman untuk mengelola kategori barang inventaris. Pengurus memiliki akses penuh untuk CRUD operations. Dengan fitur-fitur, seperti, Tabel kategori dengan kolom: No, Nama, Deskripsi, Kode, Status, Aksi, Tombol "+ Tambah Kategori" (hanya untuk pengurus), Badge untuk status aktif/non-aktif, Badge untuk kode kategori, Tombol Edit dan Hapus pada setiap baris (hanya untuk pengurus), Modal form untuk create dan edit kategori, Konfirmasi delete dengan dialog

No	Nama	Deskripsi	Kode	Status
1	Busur (Bow)	Peralatan utama memanah	BSR	Aktif
2	Anak Panah (Arrow)	Amunisi panahan	ARR	Aktif
3	Target & Bantalan	Sasaran tembak	TRG	Aktif
4	Perlengkapan & Safety	Pelindung diri	SFT	Aktif

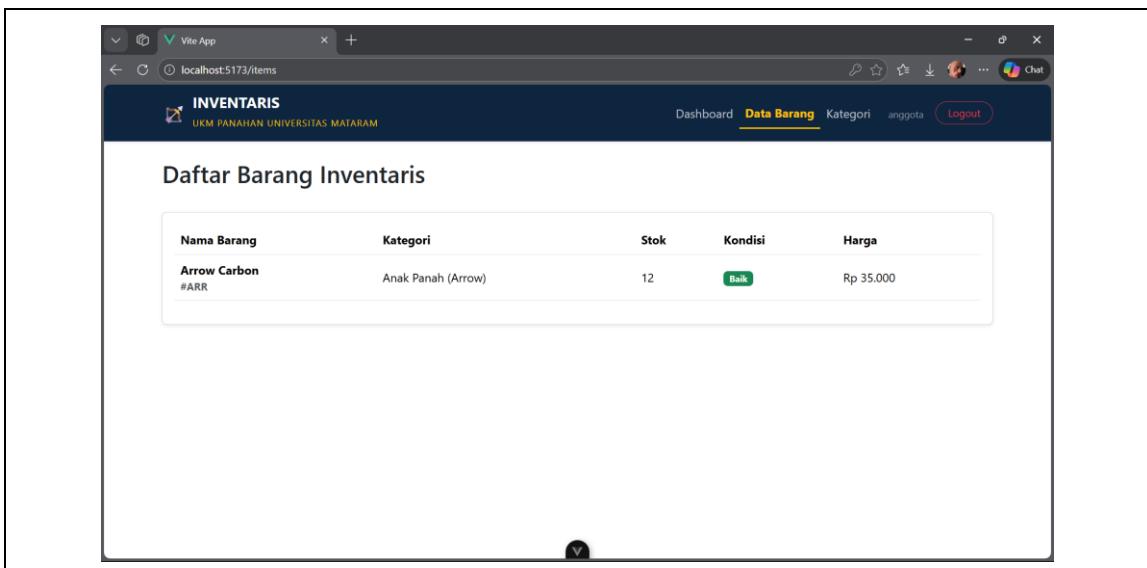
Gambar 11. Tampilan Website Halaman *Categories* (Anggota View)

Tampilan yang sama seperti pengurus tetapi tanpa tombol aksi (Tambah, Edit, Hapus). Anggota hanya dapat melihat data kategori. Dengan fitur-fitur, seperti, Tabel kategori (read-only), Tidak ada tombol aksi CRUD, Badge status dan kode tetap ditampilkan



Gambar 12. Tampilan Website Halaman *Items*/Barang (Pengurus *View*)

Halaman untuk mengelola data barang inventaris dengan fitur lengkap. Yaitu, Tabel barang dengan kolom: Nama Barang, Kategori, Stok, Kondisi, Harga, Aksi, Tombol "+ Tambah Barang" (hanya untuk pengurus), Badge untuk status kondisi (Baik/Rusak Ringan/Rusak Berat), Format harga dengan pemisah ribuan (Rupiah), Kode kategori ditampilkan di bawah nama barang, Modal form untuk create dan edit barang, Form lengkap: kategori (dropdown), nama, stok, harga, tanggal beli, kondisi, deskripsi, Tombol Edit dan Hapus (hanya untuk pengurus), Konfirmasi delete.



Gambar 13. Tampilan Website Halaman *Items*/Barang (Anggota *View*)

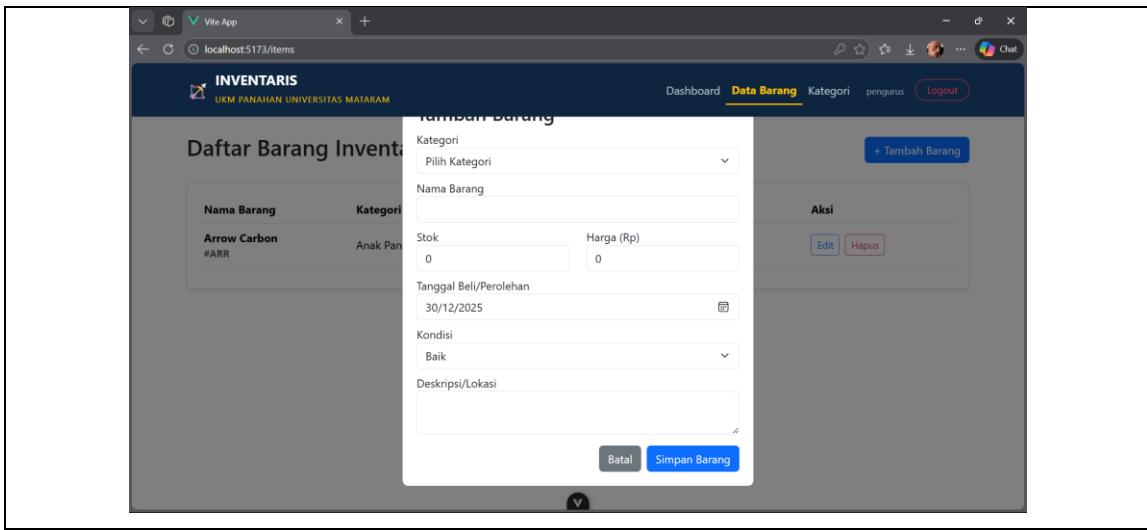
Tampilan yang sama tetapi tanpa tombol aksi CRUD. Yaitu, Tabel barang (read-only) Dapat melihat semua informasi barang, Tidak ada tombol aksi.



Gambar 24. Tampilan Website Halaman Navbar Navigation

Navbar yang ditampilkan di semua halaman (kecuali login/register). Dengan Fitur, Logo dan nama organisasi “INVENTARIS UKM PANAHAN UNIVERSITAS MATARAM”, Menu navigasi: Dashboard, Data Barang, Kategori, Active link indicator dengan border bawah kuning, Display nama dan role user di pojok kanan, Tombol Logout, Responsive dengan hamburger menu untuk mobile.

The image contains two screenshots of a "Data Kategori" page from a web application. Both screenshots show a table of categories on the left and a modal dialog on the right. In the top screenshot, the modal is titled "Tambah Kategori" and contains fields for "Nama Kategori" (Anak Panah (Arrow)), "Deskripsi" (Amunisi panahan), and "Status" (Aktif). It has "Batal" and "Simpan" buttons. In the bottom screenshot, the modal is titled "Edit Kategori" and shows the same information for the category "Anak Panah (Arrow)". Both screenshots also show a sidebar with a table of categories and edit/handle buttons.



Gambar 14. Tampilan Website Halaman Modal *Form Create/Edit*

Modal form yang muncul untuk operasi create dan edit data. Dengan Fitur untuk modal kategori, Input nama kategori, Textarea deskripsi, Dropdown status (Aktif/Non-Aktif), Tombol Batal dan Simpan, Title dinamis (Tambah/Edit Kategori). Dan fitur untuk modal barang, Dropdown kategori (populate dari API), Input nama barang, Input stok (number), Input harga (number), Date picker tanggal pembelian, Dropdown kondisi (Baik/Rusak Ringan/Rusak Berat), Textarea deskripsi/Lokasi, Tombol Batal dan Simpan Barang, Scrollable untuk form yang Panjang.

3. Hasil Pengujian dan Validasi

a. Pengujian Fungsional

Autentikasi dan Otorisasi:

- Register user baru berhasil dengan password terenkripsi
- Login dengan kredensial valid berhasil mendapat token JWT
- Login dengan kredensial invalid menampilkan error yang sesuai
- Protected route tidak dapat diakses tanpa token
- Token invalid/expired mengembalikan error 403
- Role-based access control berjalan dengan baik (pengurus vs anggota)

Manajemen Kategori:

- Pengurus dapat menambah kategori baru
- Kode kategori auto-generate dari 3 huruf pertama nama
- Pengurus dapat mengupdate data kategori
- Pengurus dapat menghapus kategori (dengan konfirmasi)
- Semua user dapat melihat daftar kategori
- Anggota tidak melihat tombol aksi CRUD

Manajemen Barang:

- Pengurus dapat menambah barang baru dengan semua atribut
- Dropdown kategori populate dari database
- Pengurus dapat mengupdate data barang
- Pengurus dapat menghapus barang (dengan konfirmasi)
- Semua user dapat melihat daftar barang dengan info kategori
- Badge kondisi barang tampil dengan warna sesuai status
- Format harga dengan pemisah ribuan

UI/UX:

- Navbar muncul di semua halaman kecuali login/register
- Active link indicator berfungsi
- Modal form muncul dengan overlay
- Responsive design untuk mobile dan desktop
- Hover effect pada card dan tombol
- Loading state saat fetch data
- Error handling dengan alert/message yang jelas

b. Pengujian Non-Fungsional

Keamanan:

- Password dienkripsi dengan bcrypt sebelum disimpan
- JWT token memiliki expiration time (24 jam)
- Sensitive data tidak terekspos di client-side
- Protected route hanya accessible dengan token valid
- CORS dikonfigurasi dengan benar

Performa:

- Load time halaman cepat (< 2 detik)
- API response time rata-rata < 200ms
- Database query teroptimasi dengan JOIN untuk relasi
- Frontend menggunakan lazy loading untuk komponen besar

Usability:

- Interface intuitif dan mudah dipahami
- Navigasi jelas dengan active indicator
- Error message deskriptif dan membantu

- Konfirmasi untuk aksi destruktif (delete)
- Form validation mencegah input invalid

BAB V – KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem, dapat disimpulkan bahwa:

1. Sistem Inventarisasi UKM Panahan berbasis web telah berhasil dikembangkan dengan menggunakan stack teknologi modern (Node.js, Express.js, Vue.js 3, MySQL) yang memungkinkan pengelolaan data inventaris secara digital, terstruktur, dan efisien.
2. Fitur autentikasi dan otorisasi berbasis JWT berhasil diimplementasikan dengan baik, memungkinkan sistem membedakan hak akses antara pengurus (full CRUD) dan anggota (read-only), sehingga meningkatkan keamanan dan privasi data.
3. Sistem CRUD (Create, Read, Update, Delete) untuk manajemen kategori dan barang inventaris berjalan dengan baik, dilengkapi dengan validasi input dan error handling yang memadai.
4. Antarmuka pengguna (UI) dirancang responsif dan user-friendly menggunakan Bootstrap 5, dengan navigasi yang jelas, feedback visual yang baik, dan design modern yang meningkatkan user experience.
5. Relasi database antara tabel categories dan items dengan foreign key constraint berhasil menjaga integritas data dan memudahkan query data barang beserta informasi kategorinya.
6. Pengujian API menggunakan Postman menunjukkan semua endpoint berfungsi sesuai spesifikasi, dengan response time yang baik dan error handling yang tepat.
7. Sistem ini dapat menjadi solusi digital yang mendukung tata kelola organisasi UKM yang lebih profesional, transparan, dan akuntabel dalam pengelolaan aset inventaris

LAMPIRAN

Link GitHub Repository: https://github.com/LilianaW17/ProjectPWL_InventarisUKM.git

Link Deployment (jika ada):

Screenshoot Tambahan: