SISTEM INFORMASI PERPUSTAKAAN DENGAN ARSITEKTUR MICROSERVICES MENGGUNAKAN STACK TEKNOLOGI MERN (Mongodb, Express, React Js, Node JS)

Penjelasan

Pada project website yang kami buat ini dengan arsitektur microservices. Memisahkan antara backend dan frontend dengan menyediakan REST Server pada sisi backend dan REST Client yang meminta akses atau request kepada REST Server. Pada sisi backend kami menggunakan expressjs dan database mongodb untuk membuat fungsionalitas REST Server / Service yang akan di akses di sisi client / frontend, dan untuk frontend kami menggunakan library javascript React Js untuk membangun user interface juga react router dom untuk berpindah antar halaman dan menggunakan bootstrap versi 5 untuk framework html dan css. Pada authentikasi login kami menggunakan jsonwebtoken (JWT) untuk mengautentikasi user / pengguna.

**User / pengguna**

Pada aplikasi website ini kami membuat dengan tiga role user:

1. Role user 1 === SUPER ADMIN

User role 1 memiliki semua akses aplikasi seperti: Create, Read, Update, Delete data buku dan Read User, Update User Role, Delete User

1. Role user 2 === ADMIN

User role 2 memiliki akses terbatas aplikasi seperti: Create, Read, Update, Delete data buku

1. Role user 3 === User / Pengguna

User role 3 memiliki kases terbatas seperti Read data buku

Fungsionalitas dan services yang terdapat di dalam aplikasi perpus berbasis website :

1. Register / Daftar akun

Pada halaman ini berguna untuk daftar user / pengguna baru ke dalam website perpus. Karena jika belum melakukan registrasi / daftar akun maka tidak akan bisa masuk kedalam aplikasi. Pengguna baru harus memasukan data nama, email, password dan role user. Role user otomatis adalah 3 di input langsung dari backend. Setelah tombol register di klik akan ada pengecekan apakah alamat email sudah ada ? jika email sudah digunakan akan tampil pesan error, jika belum maka user baru berhasil mendaftar. Pada bagian input password user akan di hash menggunakan modul bcrypt lalu masuk ke database

1. Login user / pengguna

Pada halaman login, user yang sudah melakukan registrasi langsung diarahkan ke halaman ini untuk masuk ke aplikasi. Pada login user harus memasukkan email dan password, pada saat tombol login diklik akan mengecek email pada database jika ada dan password benar maka login berhasil, jika tidak maka akan tampil pesan error. Jika berhasil login token jsonwebtoken (JWT) akan dibuat lalu disimpan di local storage browser (chrome, mozila, dll) juga info user login juga akan disimpan di local storage. Jsonwebtoken (JWT) digunakan untuk mengatur waktu akses / expired user di aplikasi. Kemudian info user login digunakan untuk mengetahui siapa yang login juga untuk validasi role user untuk akses tertentu

1. User Lupa password

Pada halaman lupa password user harus memasukkan email aktif yang sudah terdaftar jika tidak maka akan tampil pesan kesalahan. Jika email ada maka akan dikirim link beserta jsonwebtoken dan id user untuk validasi jika token expired atau id user ada atau tidak, minimal password harus 5 character. Jika validasi sukses maka password berhasil diubah . Kemudian user akan diarahkan ke menu login dan bisa lgoin dengan password baru

1. Menu Beranda

Pada menu beranda berisi informasi perpustakaan

1. Menu Daftar Buku

Berisi informasi buku yang ada di perpustakaan seperti : kode barcode, judul buku, penerbit, pengarang, tahun terbit, tempat terbit

1. Menu Tentang perpus

Berisi informasi tentang perpustakaan

1. Menu logout / keluar

Pada menu logout / keluar, token (JWT) dan informasi user login akan di hapus dari local storage lalu diarahkan ke halaman login.