



KASUS 8: LOGIN DI IONIC

**Catatan:**

Pastikan kalian telah mengerjakan Latihan pada Kasus 7 sebelumnya. Pada Latihan 8 ini akan dibahas mengenai proses login dengan membaca data yang tersimpan di dalam database MySql melalui backend Laravel. File ini akan digunakan untuk melakukan komunikasi dengan backend yang sudah dibuat sebelumnya.

**SOAL:**

1. Pertama lakukan instalasi axios, vuex:

```
npm install axios
```

Catatan:

**axios** – akan kita gunakan utk melakukan request ke server.

**Pastikan proses instalasi AXIOS sukses.**

2. Buka halaman **Login.vue** kemudian tambahkan script untuk menangani proses login seperti di bawah ini:

```
<script lang="ts" setup>
  import AuthLayout from "@/components/AuthLayout.vue";
  import { IonCard, IonCardContent, IonList, IonItem, IonInput, IonIcon, IonButton,
    useIonRouter, alertController, loadingController } from "@ionic/vue";
  import { lockClosed, mail } from 'ionicons/icons';
  import { Form, Field, ErrorMessage, defineRule } from 'vee-validate';
  import { required, email, min } from '@vee-validate/rules';
  import axios from 'axios';

  defineRule('required', required);
  defineRule('email', email);
  defineRule('min', min);

  const judulpage = "USER AUTH";
  const router = useIonRouter();
  const handleSubmit = async (data: any) => {
    //loading
    const loading = await loadingController.create({
      message: 'Please wait ...',
    });

    loading.present();

    axios.get('http://localhost:8000/sanctum/csrf-cookie').then(()=>{
      axios.post('http://localhost:8000/api/login', data).then(async (res)=>{
        if(res.data.status == 'Success'){
          localStorage.setItem('loggedIn', 'true');
          localStorage.setItem('token', res.data.access_token);
          localStorage.setItem('user', res.data.user);
          router.replace('/tabs');
        } else {
          localStorage.removeItem('loggedIn');
          localStorage.removeItem('token');
          localStorage.removeItem('user');
          // alert
          const alert = await alertController.create({
            header: 'Login Gagal',
            message: 'User Account tidak ditemukan!',
            buttons: ['OK'],
          });
          await alert.present();
        }
      }).finally(() => {
        loading.dismiss();
      });
    });
  }
</script>
```

3. Silakan dijalankan proses login ketikan dulu email dan password yang salah kemudian klik tombol Login lihat apakah muncul loading dan alert di layar. Kemudian ketikan email dan password yang benar klik tombol Login lihat apakah muncul loading dan jendela baru terbuka. Awasi juga apakah localStorage terisi dengan variable.
4. Silakan coba untuk halaman Register !

----- SELAMAT MENCOBA -----