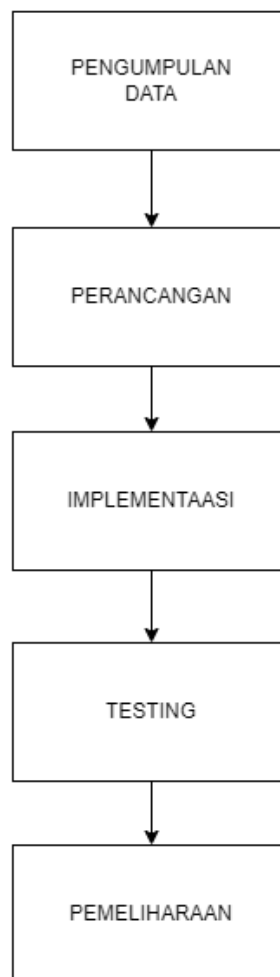


## BAB III

### METODE PENELITIAN

#### 3.1 Alur Penelitian

Alur penelitian merupakan konsep atau gambaran penelitian yang akan dilakukan. Penjelasan alur penelitian dapat dilihat pada gambar diagram 3.1



**Gambar 3.1** Alur Penelitian

## **3.2 Pengumpulan data**

Pengumpulan data dilakukan untuk mengumpulkan data dan berbagai informasi yang berkaitan dengan penelitian. metode pengumpulan data dilakukan melalui beberapa tahap yang terinci sebagai berikut:

### **3.2.1 Pengamatan (*Observasi*)**

Pengumpulan data diawali dengan observasi langsung terhadap Direktorat Kemahasiswaan IBI Darmajaya tentang bagaimana cara pengaduan yang dilakukan oleh mahasiswa. Melalui pengamatan ini, peneliti dapat memperoleh pemahaman yang lebih mendalam tentang cara mahasiswa melakukan pengaduan, termasuk pengalaman pengguna dan skenario penggunaan aplikasi secara real-time.

### **3.2.2 Wawancara (*Interview*)**

Dilakukan wawancara terhadap mahasiswa, dan pengelola Direktorat Kemahasiswaan IBI Darmajaya tentang pengaduan mahasiswa. Wawancara ini bertujuan untuk mengumpulkan pandangan dan umpan balik langsung dari pengguna aplikasi dan pemangku kepentingan mengenai proses pengaduan siswa. Pertanyaan terfokus diajukan untuk mendapatkan pemahaman lebih dalam tentang kebutuhan pengguna, tantangan yang dihadapi, dan harapan terkait penerapan teknologi Web Socket..

## **3.3 Alat dan Bahan Penelitian**

Agar suatu sistem dapat berfungsi dengan baik, pembuatannya memerlukan alat pendukung hardware dan software tertentu, adalah sebagai berikut:

1. Perangkat Keras
  - a. *AMD Ryzen 3 3250U (2C / 4T, 2.6 / 3.5GHz, 1MB L2 / 4MB L3)*
  - b. *Graphics Integrated AMD Radeon Graphics*
  - c. *Memmmory 20 gb*
  - d. *Ssd 1 tb*

## 2. Perangkat Lunak

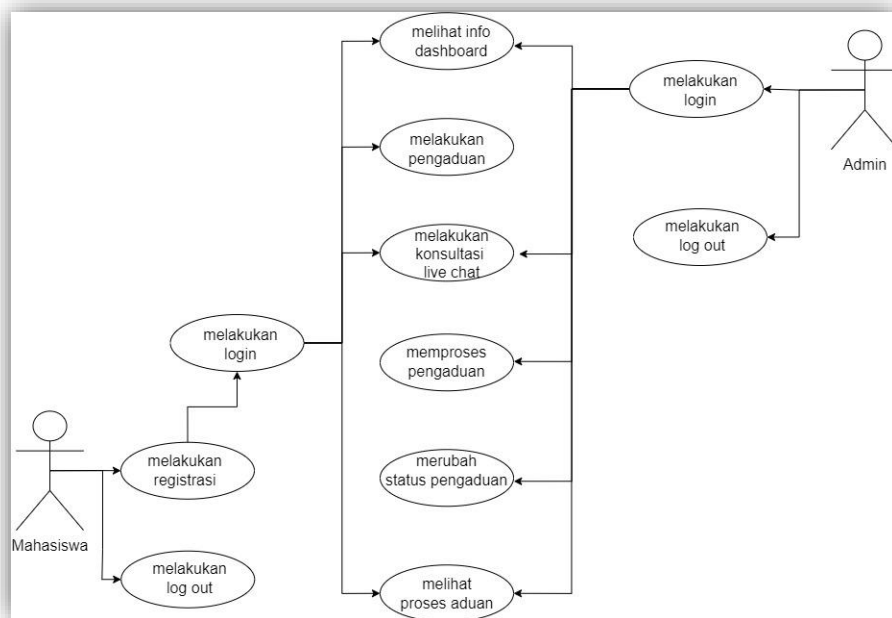
- a. *Visual Studio Code*
- b. *PostgreSQL*
- c. *NodeJS*

### 3.4 Perancangan (*Design*)

Pada tahap ini dilakukan perancangan yang dimulai dari desain perancangan yaitu menentukan desain use case diagram, Tujuannya untuk mengecek apakah prototype yang dibuat apakah sudah sesuai dengan kebutuhan klien

#### 3.4.1 Use Case Diagram

Dalam perancangan site pengaduan mahasiswa, telah dirancang usecase chart yang menggambarkan fungsi-fungsi yang dimiliki oleh masing-masing role yaitu admin dan mahasiswa. Rancangan use case diagram dapat dilihat pada Gambar 3.2 sebagai berikut:



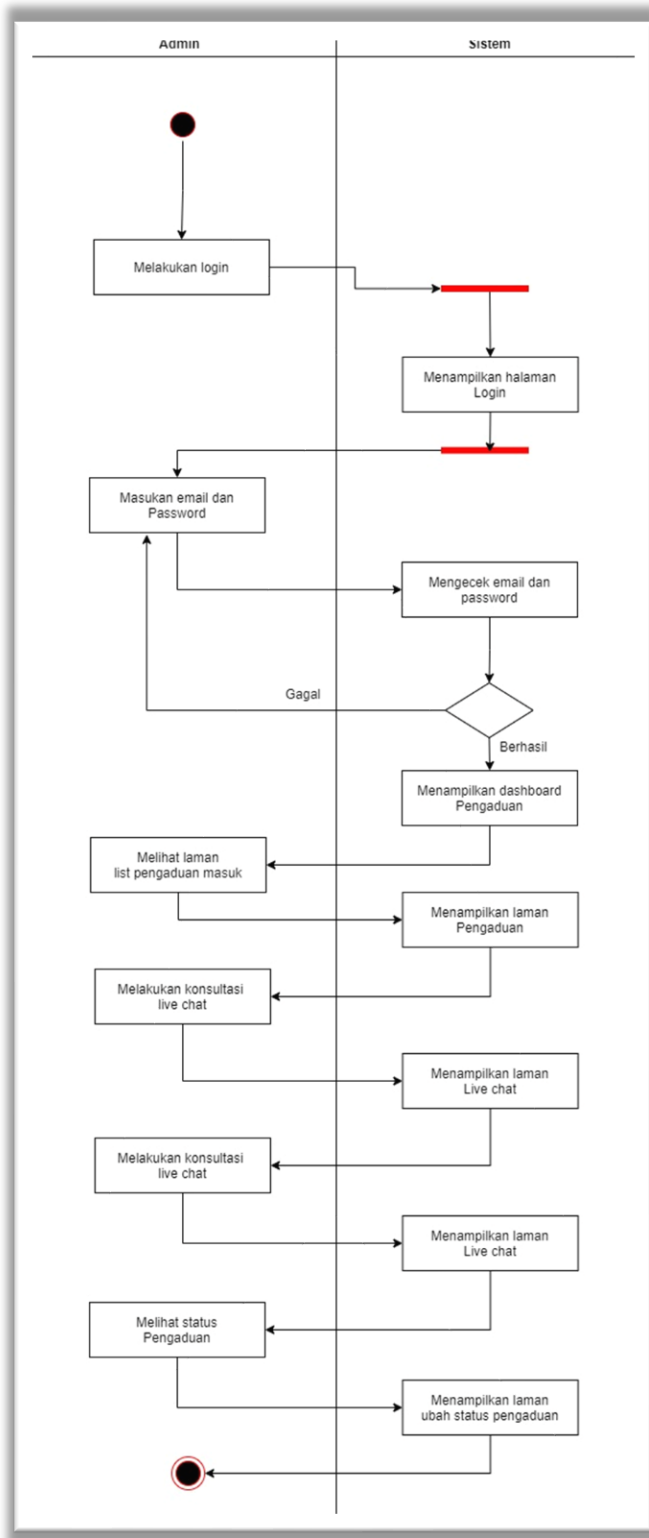
**Gambar 3.2 Use Case Diagram**

### **3.4.2 Activity Diagram**

Activity diagram menggambarkan alur proses dan urutan aktivitas dalam sebuah proses. Activity diagram digunakan untuk memodelkan sebuah proses dalam membantu memahami proses keseluruhan. Activity Diagram dibuat berdasarkan usecase diagram yang sesuai dengan perangkat lunak. Berikut activity diagram dari web site Pengaduan Mahasiswa:

#### **a. Activity Diagram Admin**

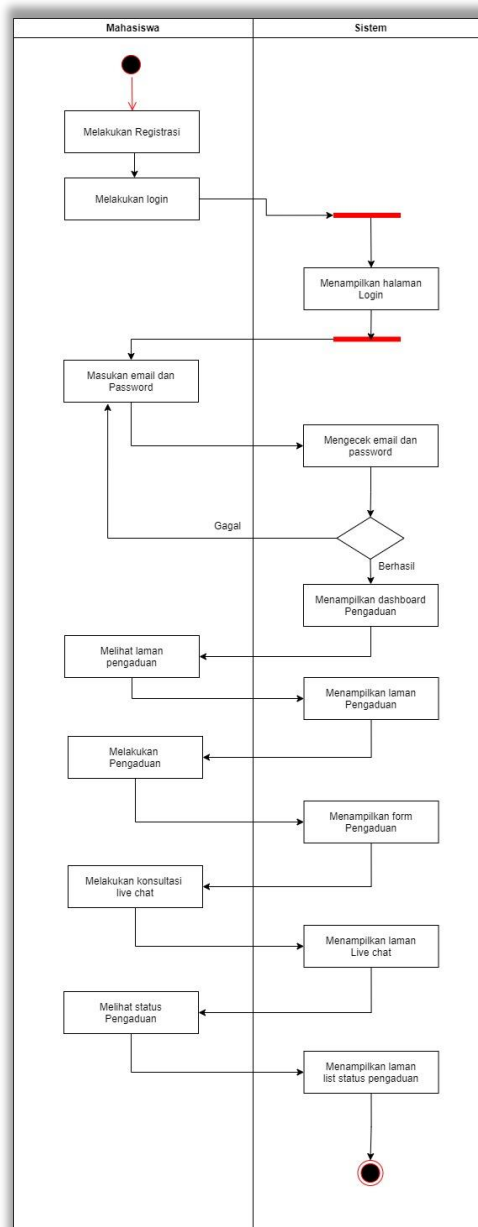
*Activity diagram Admin* menjelaskan bagaimana proses dan aliran kerja *Admin* dimulai dari proses login dan verifikasi login sampai dalam memvalidasi aduan yang sudah dibuat. Proses dari activity diagram *Admin* dapat dilihat pada gambar 3.3 dibawah ini.



**Gambar 3.3** Activity Diagram Login

**b. Activity Diagram Mahasiswa**

Pada *Activity diagram* Mahasiswa input merupakan proses dan alirankerja dimulai dengan menampilkan system utama sekaligus melakukan pembuatan akun untuk Mahasiswa melakukan login. Proses dari activity diagram Mahasiswa dapat dilihat pada gambar 3.4 berikut ini:



**Gambar 3.4** Activity Diagram Mahasiswa

### 3.5 Struktur Database

Pada pembuatan sistem pengaduan mahasiswa memerlukan sebuah database, berikut adalah rancangan database yang akan digunakan dapat dilihat pada table 3.1 & 3.2 sebagai berikut:

**Tabel 3. 1 Database Pengguna**

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Id pengguna	Uuid	32	Id
Nama	Varchar	255	Nama
Email	Varchar	255	Email
NPM	Int	255	NPM
Password	Varchar	255	Password
Role	Varchar	255	Level
Id report	Uuid	32	Id
Title	Varchar	255	Judul aduan
Subject	Varchar	255	Subjek aduan
Image	Varchar		Image aduan
File	Varchar		File aduan
Description	Varchar	255	Deskripsi aduan
Created_at	Timestamp	-	Membuat
Update_at	Timestamp	-	Memperbarui

**Tabel 3. 2 Database Admin**

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Id admin	Uuid	32	Id
Email	Varchar	255	Email
Password	Varchar	255	Password
Role	Varchar	255	Level
Id report	Uuid	32	Id
Image	Varchar		Image aduan

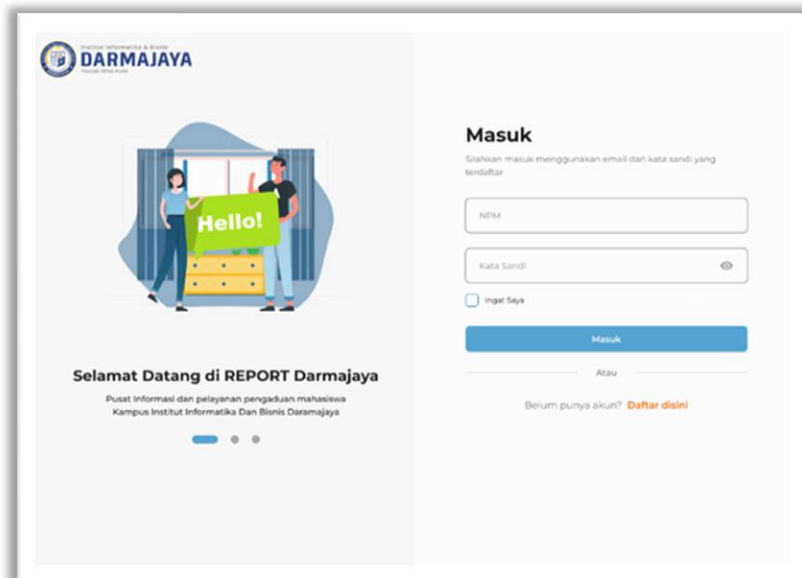
File	Varchar		File aduan
Created_at	Timestamp	-	Membuat
Update_at	Timestamp	-	Memperbarui

### 3.6 Tampilan Antarmuka

Pada tampilan anatar muka (Interface) web site Pengaduan Mahasiswa adalah sebagai berikut:

- **Halaman Login**

Halaman login merupakan halaman awal Ketika user membuka website, fungsi dari halaman login ini adalah agar user dapat login terlebih dahulu sebelum dapat melakukan pengaduan. Tampilan halaman login dapat dilihat pada gambar 3.5 berikut:



**Gambar 3.5** Halaman *Login*



- **Halaman Register**

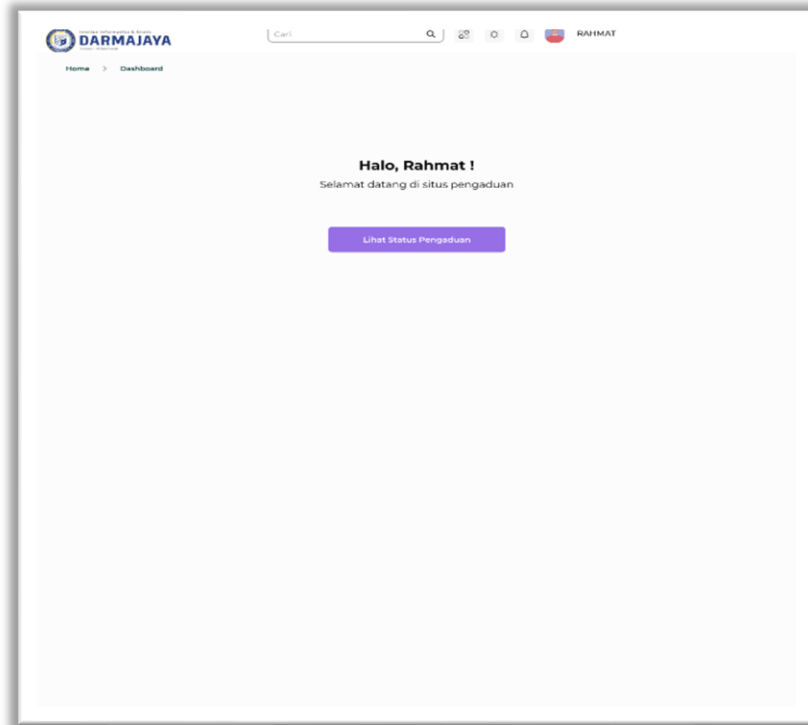
Halaman register merupakan halaman apabila user belum memiliki akun, halaman ini adalah berfungsi sebagai awal bagi user yang belum melakukan pendaftaran akun, disini juga user hanya bisa daftar menggunakan email domain kampus IIB Darmajaya. Tampilan halaman register dapat dilihat pada gambar 3.6 sebagai berikut:

The image shows a web registration page for IIB Darmajaya. On the left side, there is a graphic with the text "Selamat Datang di REPORT Darmajaya" and "Pusat Informasi dan pelayanan pengaduan mahasiswa Kampus Institut Informatika Dan Bisnis Darmajaya". The right side contains a "Daftar Akun" form with the following fields: "Nama Lengkap", "NPM", "Email" (with a note "Wajib menggunakan Email kampus"), "Kata Sandi" (with a note "Minimal 8 karakter terdiri huruf besar, huruf kecil dan angka"), and "Konfirmasi Kata Sandi" (with a note "Minimal 8 karakter terdiri huruf besar, huruf kecil dan angka"). Below the form is a blue "Daftar" button, followed by "Atau" and a link "Sudah punya akun? Masuk disini".

**Gambar 3.6** Halaman *Register*

- **Halaman Dashboard**

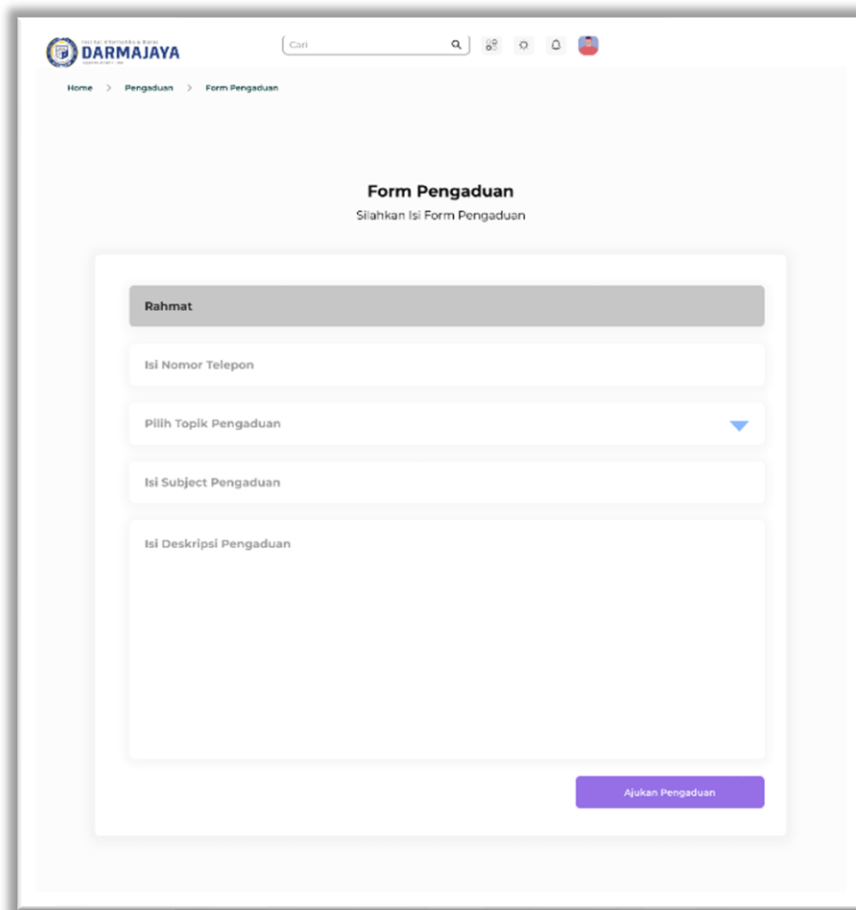
Halaman dashboard merupakan halaman setelah mahasiswa memasukan NPM dan password pada halaman login sebelumnya. Tampilan halaman dashboard dapat dilihat pada gambar 3.7 sebagai berikut:



**Gambar 3.7** Halaman *Dashboard*

- **Halaman Form Pengaduan Mahasiswa**

Halaman form pengaduan mahasiswa adalah halaman untuk mahasiswa menginput data pengaduan seperti topik dan deskripsi pengaduan. Tampilan halaman form pengaduan mahasiswa dapat dilihat pada gambar 3.8 sebagai berikut:

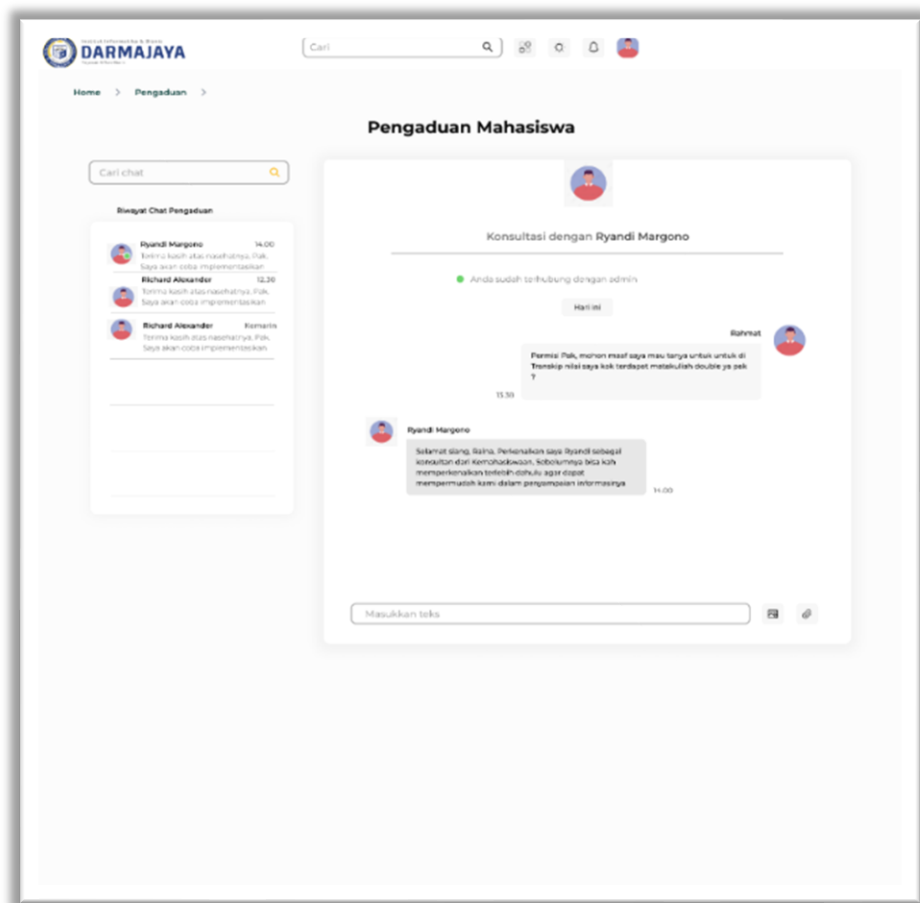


The screenshot shows a web browser window displaying the 'Form Pengaduan' page. At the top left is the DARMAJAYA logo. A search bar is located at the top right. Below the header, there is a breadcrumb trail: Home > Pengaduan > Form Pengaduan. The main heading is 'Form Pengaduan' with the subtext 'Silahkan Isi Form Pengaduan'. The form itself is a white box with a grey header 'Rahmat'. It contains five input fields: 'Isi Nomor Telepon', 'Pilih Topik Pengaduan' (a dropdown menu), 'Isi Subject Pengaduan', and 'Isi Deskripsi Pengaduan' (a larger text area). A purple button labeled 'Ajukan Pengaduan' is positioned at the bottom right of the form.

**Gambar 3.8** Halaman Form Pengaduan Mahasiswa

- **Halaman Live chat**

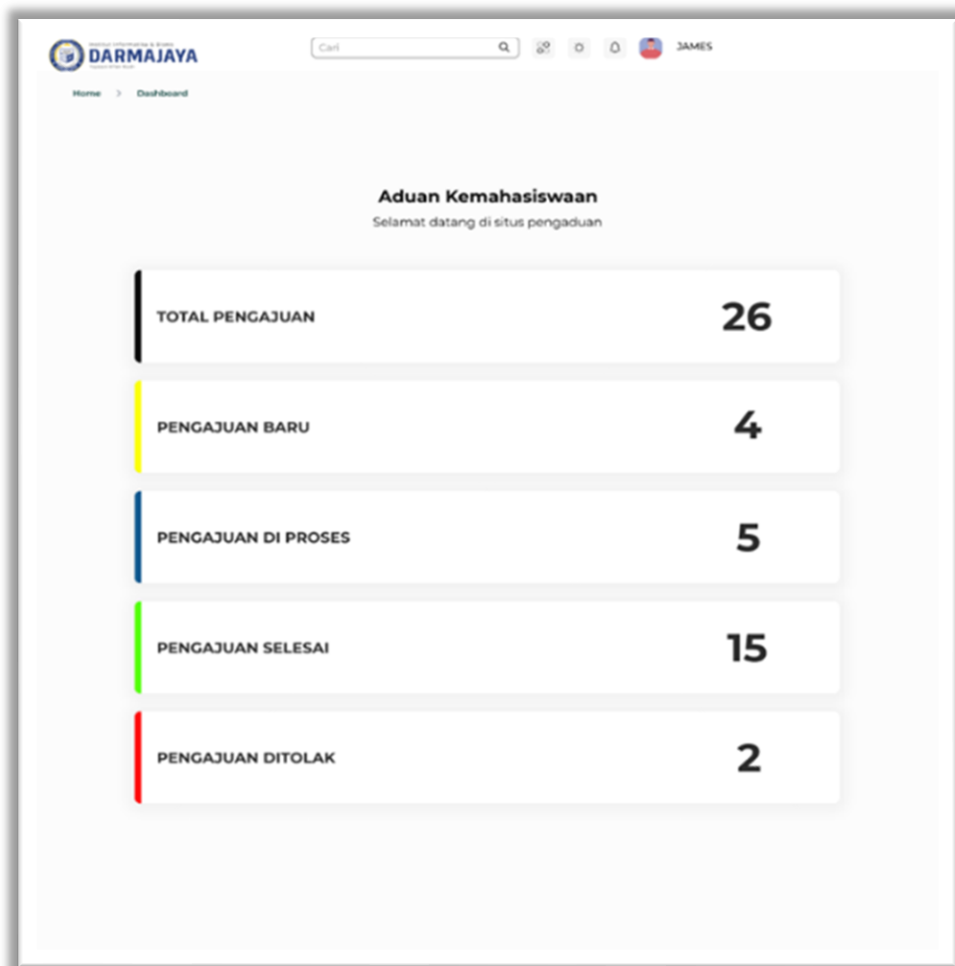
Pada halaman live chat merupakan halaman untuk mahasiswa atau admin. Setelah mahasiswa mengisi form pengaduan di halaman ini dapat melakukan konsultasi pengaduan dengan cara live chat. Tampilan halaman live chat dapat dilihat pada gambar 3.9 sebagai berikut:



**Gambar 3. 9** Halaman *Live chat*

- **Halaman Dashboard Admin**

Pada halaman dashboard admin merupakan halaman awal admin Setelah melakukan login, di sini admin dapat melihat jumlah pengaduan yang masuk dan pengaduan yang selesai. Tampilan halaman dashboard admin dapat dilihat pada gambar 3.10 sebagai berikut:



**Gambar 3.10** Halaman *Dashboard Admin*

### 3.7 Implementasi (*Implementation*)

Tahapan ini mencakup penulisan kode program dan implementasi setiap modul atau komponen. Setelah itu, modul-modul tersebut akan diintegrasikan ke dalam satu sistem. Implementasi program dapat dilihat pada gambar 3.11 & 3.12

```
10 module.exports = {
11   login(req, res) {
12     res.render("login", {
13       url: 'http://localhost:5050/',
14       colorFlash: req.flash('color'),
15       statusFlash: req.flash('status'),
16       pesanFlash: req.flash('message'),
17     });
18   },
19   loginAuth(req, res) {
20     let email = req.body.email;
21     let password = req.body.pass;
22     if (email && password) {
23       pool.getConnection(function(err, connection) {
24         if (err) throw err;
25         connection.query(
26           'SELECT * FROM table_user WHERE user_email = ? AND user_password = SHA2(?,512)',
27           [email, password], function (error, results) {
28             if (error) throw error;
29             if (results.length > 0) {
30               req.session.loggedin = true;
31               req.session.userid = results[0].user_id;
32               req.session.username = results[0].user_name;
33               res.redirect('/');
34             } else {
35               req.flash('color', 'danger');
36               req.flash('status', 'Oops..');
37               req.flash('message', 'Akun tidak ditemukan');
38               res.redirect('/login');
```

Gambar 3. 11 Implementasi kode program *websocket*

```
15 CREATE DATABASE /*!32312 IF NOT EXISTS*/ 'pengaduan' /*!40100 DEFAULT CHARACTER SET utf8mb4 */;
16
17 USE 'pengaduan';
18
19 /*Table structure for table `table_user` */
20
21 DROP TABLE IF EXISTS `table_user`;
22
23 CREATE TABLE `table_user` (
24   `user_id` int(11) NOT NULL AUTO_INCREMENT,
25   `user_name` varchar(50) DEFAULT NULL,
26   `user_email` varchar(50) DEFAULT NULL,
27   `user_password` text DEFAULT NULL,
28   PRIMARY KEY (`user_id`)
29 ) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4;
30
31 /*Data for the table `table_user` */
32
33 insert into `table_user` (`user_id`,`user_name`,`user_email`,`user_password`) values
34 (1,'test','ronzuma@gmail.com','fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cd
35 (2,'zamroni','user@email.com','fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cd
36 (3,'zamroni','ronzuma@gmail.com','fa585d89c851dd338a70dcf535aa2a92fee7836dd6aff1226583e88e0996293f16bc009c652826e0fc5c706695a03cd
37
38 /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
39 /*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
40 /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
41 /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
42
```

Gambar 3. 12 Implementasi kode program Database

### **3.8 Uji (Testing)**

Berdasarkan analisis yang telah dilakukan, maka dapat diketahui apa saja yang akan menjadi masukan sistem, keluaran sistem, fungsi atau metode yang digunakan oleh sistem, kebutuhan perangkat keras, kebutuhan perangkat lunak serta antar muka sistem yang akan dibuat, sehingga sistem yang dibangun sesuai dengan apa yang diharapkan.

1. Analisis Kebutuhan Masukan (Input) Input atau masukan dari system website pengaduan mahasiswa ini berupa :
  - a. Masukkan Username dan Password
  - b. Masukkan Form Pengaduan Mahasiswa
  - c. Masukkan file pengaduan Mahasiswa
  - d. Masukkan konsultasi chat Mahasiswa
2. Analisa Kebutuhan Keluaran (Outout)

Data keluaran yang dihasilkan oleh system website pengaduan mahasiswa ini adalah setelah melakukan penginputan data pengaduan dan dilakukan pemrosesan pengaduan oleh admin yang terkait sesuai pengaduan yang diadakan oleh mahasiswa.

### **3.9 Pemeliharaan (*Maintenance*)**

Pada tahap ini akan dilakukan pemeliharaan terhadap sistem, termasuk perbaikan bug dan peningkatan fitur jika diperlukan, setelah penerapan selesai.

## BAB IV

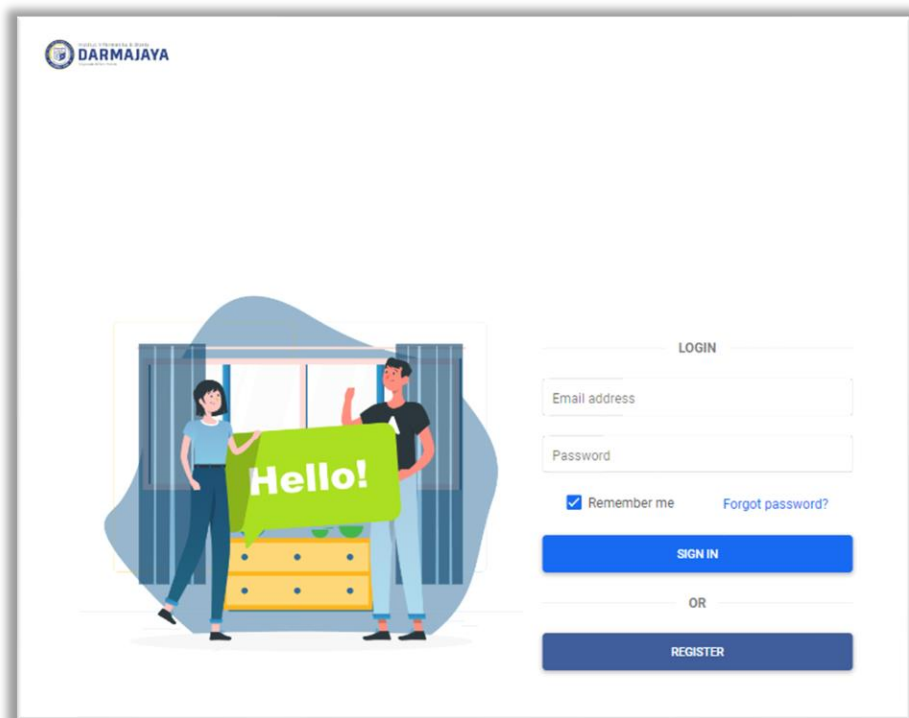
### HASIL PENELITIAN DAN PEMBAHASAN

#### 4.1 Hasil Penelitian

Hasil penelitian ini akan menguraikan dampak dan efektivitas penerapan program yang telah dirancang sebelumnya, dengan fokus penelitian ini yaitu dimana mahasiswa dapat melakukan pengaduan dengan secara langsung. Berikut terkait hasil penelitian yang telah dilakukan:

##### 4.1.1 Halaman *Login*

Halaman login merupakan halaman awal Ketika user membuka website, fungsi dari halaman login ini adalah agar user dapat login terlebih dahulu sebelum dapat melakukan pengaduan. Tampilan pada halaman login dapat dilihat pada gambar 4.1 sebagai berikut:

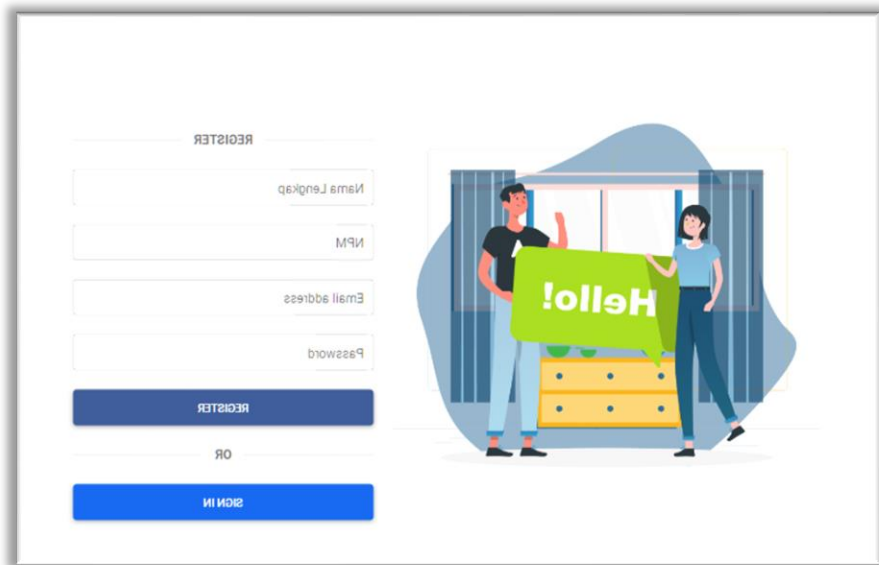


Gambar 4.1 Halaman *Login*



#### 4.1.2 Halaman Register

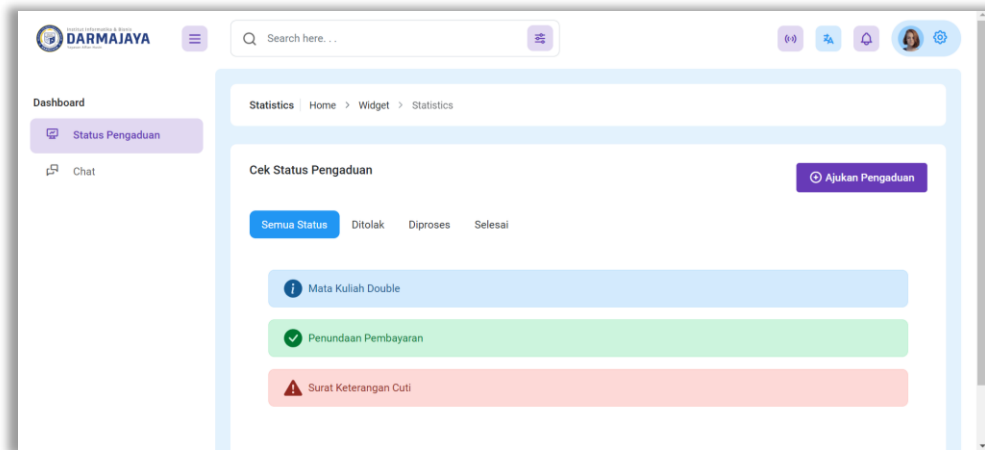
Halaman register merupakan halaman apabila user belum memiliki akun, halaman ini adalah berfungsi sebagai awal bagi user yang belum melakukan pendaftaran akun, disini juga user hanya bisa daftar menggunakan email domain kampus IIB Darmajaya. Tampilan pada halaman register dapat dilihat pada gambar 4.2 sebagai berikut:



**Gambar 4.2** Halaman *Register*

#### 4.1.3 Halaman Dashboard

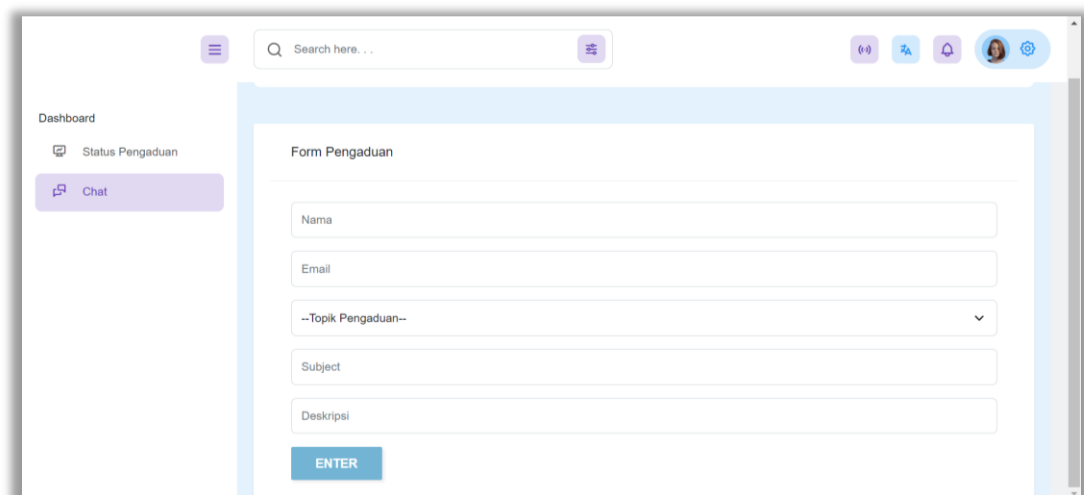
Halaman dashboard merupakan halaman setelah mahasiswa memasukan NPM dan password pada halaman login sebelumnya, untuk melakukan pengaduan mahasiswa dapat menekan tombol ajukan pengaduan. Tampilan pada halaman dashboard dapat dilihat pada gambar 4.3 sebagai berikut:



**Gambar 4.3** Halaman *Dashboard* Mahasiswa

#### 4.1.4 Halaman Form Pengaduan

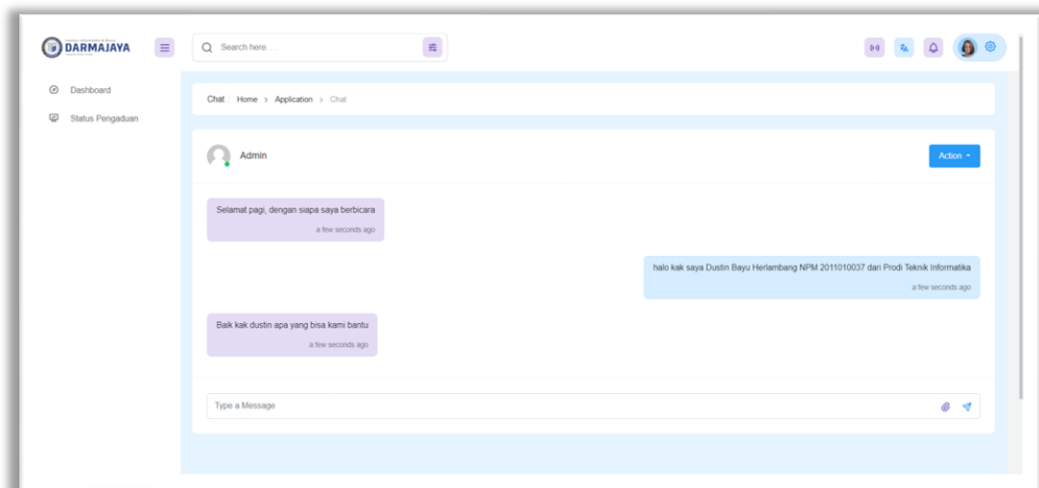
Halaman form pengaduan mahasiswa adalah halaman untuk mahasiswa menginput data pengaduan seperti topik dan deskripsi pengaduan dengan cara menekan bagian topik pengaduan. Tampilan halaman form pengaduan dapat dilihat pada gambar 4.4 sebagai berikut:



**Gambar 4.4** Halaman *Form* Pengaduan

#### 4.1.5 Halaman live chat

Pada halaman live chat merupakan halaman untuk mahasiswa atau admin Setelah mahasiswa mengisi form pengaduan di halaman ini dapat melakukan konsultasi pengaduan dengan cara live chat. Tampilan pada halaman live chat dapat dilihat pada gambar 4.5 sebagai berikut:



Gambar 4.5 Halaman *Live chat*

#### 4.1.6 Hasil Pengujian

Penelitian ini menghasilkan berupa website sistem pengaduan mahasiswa yang memanfaatkan teknologi WebSocket untuk menghubungkan mahasiswa dengan pihak kampus dalam menyampaikan keluhan. Pengujian dilakukan menggunakan black box adalah untuk memastikan perangkat lunak berfungsi sesuai spesifikasi, mendeteksi bug, dan memverifikasi keandalan tanpa melihat struktur internalnya.

**Tabel 4.1** Pengujian Black Box

<b>NO</b>	<b>SKENARIO PENGUJIAN</b>	<b>HASIL YANG DIHARAPKAN</b>	<b>HASIL</b>	<b>KESIMPULAN</b>
1	Mengosongkan Username dan password, lalu klik tombol login	Sistem akan menampilkan peringatan bahwa kolom username dan password harus diisi	Sistem memberikan peringatan bahwa field username dan password harus diisi	Valid
2	Hanya mengisi username dan mengosongkan password, lalu klik tombol Login	Sistem memberikan pemberitahuan dan meminta pengguna melengkapi data.	Sistem memberikan peringatan bahwa field username dan password harus diisi	Valid
3	Mengisi email dan password	Sistem akan mengarahkan ke halaman utama web (admin)	Sistem mengarahkan ke halaman utama web (admin)	Valid
4	Mendaftarkan Akun Mahasiswa	Sistem akan menampilkan halaman register yang terdapat form Nama, NPM, Email dan Password	Sistem memberikan pop up berhasil	Valid
5	Mahasiswa mengklik tombol 'Ajukan Pengaduan' untuk menambahkan pengaduan.	Sistem akan menampilkan halaman pengajuan pengaduan yang terdapat form Nama, Email, Topik pengaduan, Subject dan Deskripsi	Sistem otomatis mengarahkan kehalaman Live chat	Valid
6	Mahasiswa melakukan konsultasi di live chat	Sistem akan menampilkan halaman Live Chat dengan admin pengaduan	Sistem akan menampilkan halaman Live Chat dengan admin pengaduan	Valid

## 4.2 Pembahasan

Pembahasan ini mengulas hasil implementasi teknologi WebSocket pada layanan pengaduan di Unit Direktorat Kemahasiswaan Institut Informatika dan Bisnis Darmajaya. Dengan penerapan teknologi ini, diharapkan para mahasiswa dapat lebih mudah mengakses dan mengajukan pengaduan dalam lingkungan kampus. Teknologi WebSocket memungkinkan komunikasi dua arah yang real-time antara pengguna dan server, sehingga laporan pengaduan dapat diproses dan ditanggapi lebih cepat. Hasil penerapan ini menunjukkan peningkatan efisiensi dalam penanganan pengaduan, dengan waktu respons yang lebih singkat dan transparansi yang lebih baik dalam proses penyelesaiannya. Hal ini diharapkan dapat meningkatkan kepuasan mahasiswa terhadap layanan yang disediakan oleh kampus.

Program ini memiliki Kelebihan sebagai berikut :

- a) Mahasiswa dapat mengetahui notifikasi mengenai status pengaduan mereka, sehingga mereka selalu terinformasi tentang perkembangan penanganan kasus mereka.
- b) Sistem ini dapat memungkinkan interaksi antara mahasiswa dan admin pengaduan, meningkatkan kualitas komunikasi dan pemahaman.
- c) Integrasi dan manajemen data pengaduan menjadi lebih mudah dan terstruktur, mempermudah analisis dan pelaporan.

Program ini memiliki kekurangan yaitu.

- a) Sistem tidak memiliki sistem bot untuk dapat membalas pengaduan ketika admin sedang tidak online.
- b) Sistem ini hanya memiliki topik keuangan dan topik Kemahasiswaan.