

BAB III

METODE PENELITIAN

3.1 Analisis Kebutuhan

Analisis kebutuhan berguna untuk menentukan keperluan dalam pembuatan aplikasi pemberi rekomendasi tempat wisata, yang terdiri dari analisis kebutuhan proses, analisis kebutuhan perangkat keras, dan analisis kebutuhan perangkat lunak.

3.1.1 Analisis Kebutuhan Proses

Analisis kebutuhan proses pada pembuatan aplikasi rekomendasi tempat wisata adalah sebagai berikut:

- a. Proses menampilkan daftar tempat wisata.
- b. Proses perhitungan kesamaan/*similarity* antar tempat wisata.
- c. Proses perhitungan prediksi nilai/*rating* pada tempat wisata.
- d. Proses menampilkan daftar rekomendasi tempat wisata.

3.1.2 Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras yang dibutuhkan dalam pembuatan aplikasi rekomendasi tempat wisata:

- a. Laptop
 1. Processor intel core i3
 2. RAM 8 GB
 3. HDD 500 GB

3.1.3 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak yang dibutuhkan dalam pembuatan aplikasi rekomendasi tempat wisata:

- a. Sublime Text 3
Aplikasi yang digunakan untuk menulis kode.
- b. Firefox
Browser yang digunakan untuk menyajikan tampilan sistem.
- c. Xampp Control Panel v3.2.2

Web server untuk menampilkan halaman web yang dinamis.

d. Justinmind Prototyper 8.5.0

Aplikasi yang digunakan untuk membuat desain rancangan *user interface*.

e. PhpmyAdmin

Manajemen basisdata

f. Bootstrap

Front-end framework untuk mempermudah dan mempercepat penulisan kode html.

g. CodeIgniter

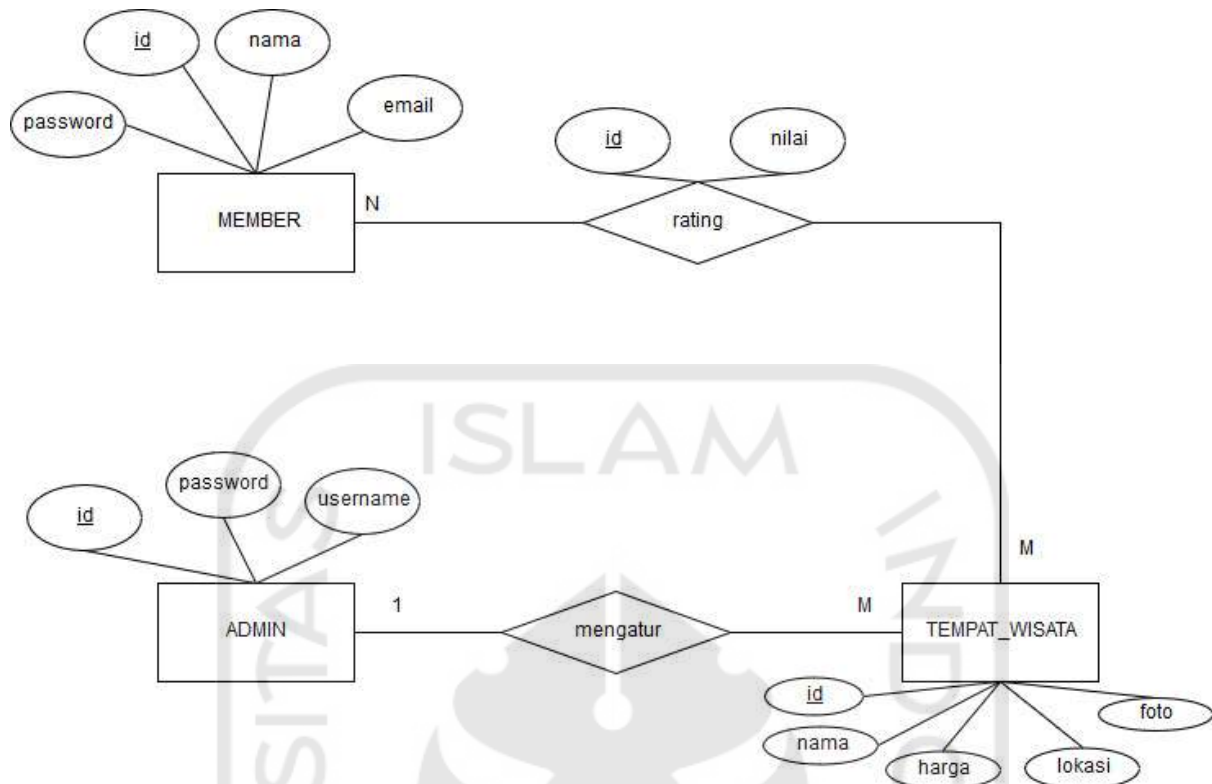
Framework PHP yang digunakan untuk membangun web dinamis.

3.2 Perancangan Sistem

Setelah dilakukan analisis kebutuhan, tahap selanjutnya adalah melakukan perancangan sistem yang akan dibuat. Dalam pembuatan sistem ini rancangan yang dibuat adalah *entity relationship diagram*, *use case diagram*, *activity diagram*, perancangan *interface* dan perancangan pengujian.

3.2.1 Entity Relationship Diagram Sistem Rekomendasi Tempat Wisata

Entity Relationship Diagram adalah analisis pemodelan data yang membantu penyusunan data ke dalam entitas-entitas dan relasi antar entitas. Pada pembuatan *ERD* akan menghasilkan struktur basisdata yang baik sehingga dapat disimpan dan digunakan (Wibowo, 2014). Pada Gambar 3.1 dapat diketahui bahwa sistem rekomendasi tempat wisata memiliki 3 entitas, yaitu *MEMBER*, *ADMIN*, dan *TEMPAT_WISATA*. Pada entitas *MEMBER* memiliki empat atribut yaitu *id* (*primary key*), *password*, nama, dan *email*. Pada entitas *ADMIN* memiliki 3 atribut yaitu *id* (*primary key*), *username*, dan *password*. Sedangkan pada entitas *TEMPAT_WISATA* memiliki 5 atribut yaitu *id* (*primary key*), nama, harga, lokasi, dan foto. Pada relasi *rating* menyimpan 2 atribut yaitu *id* (*primary key*) dan nilai. Kardinalitas untuk entitas *MEMBER* dengan entitas *TEMPAT_WISATA* adalah *many to many* (N:M) yang berarti setiap member dapat memberi *rating* pada banyak tempat wisata dan setiap tempat wisata dapat diberi *rating* oleh banyak member. Sedangkan kardinalitas untuk entitas *ADMIN* dengan *TEMPAT_WISATA* adalah *one to many* (1:M) yang berarti setiap admin dapat mengatur banyak tempat wisata dan setiap tempat wisata hanya bisa diatur oleh satu admin.



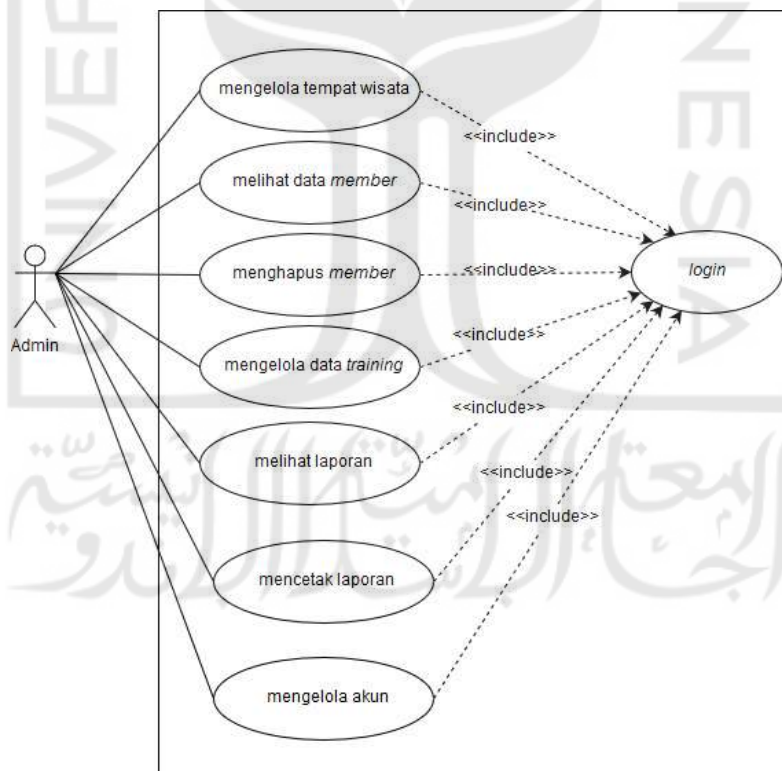
Gambar 3.1 ERD sistem rekomendasi

3.2.2 Use Case Diagram Sistem Rekomendasi Tempat Wisata

Use Case Diagram adalah gambaran *graphical* dari beberapa atau semua *actor*, *use case*, dan interaksi diantaranya yang memperkenalkan suatu sistem. *Use case diagram* tidak menjelaskan secara detail tentang penggunaan *use case*, tetapi hanya memberi gambaran singkat hubungan antara *use case*, aktor, dan sistem (Pratama, 2019). Pada *use case diagram* sistem pemberi rekomendasi tempat wisata memiliki dua aktor yaitu member dan admin. Pada member memiliki tujuh *use case*, yaitu member dapat mencari tempat wisata, melihat tempat wisata, *register* akun, *login*, memberi *rating* tempat wisata, melihat rekomendasi tempat wisata, dan mengelola akun. Sementara itu pada admin memiliki tujuh *use case* yaitu admin dapat *login*, mengelola tempat wisata, melihat data member, menghapus member, melihat laporan, mengelola data *training*, dan mengelola akun. *Use case diagram* untuk sistem pemberi rekomendasi tempat wisata dapat dilihat pada Gambar 3.2 dan Gambar 3.3.



Gambar 3.2 Use case diagram member



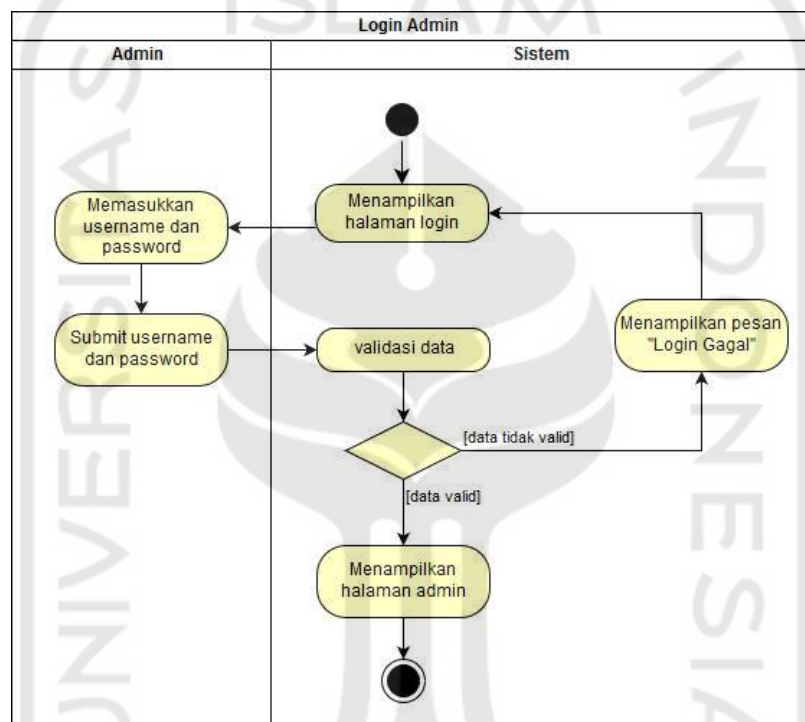
Gambar 3.3 Use case diagram admin

3.2.3 Activity Diagram Sistem Rekomendasi Tempat Wisata

a. Activity diagram admin

1. Activity diagram login admin

Pada Gambar 3.4 merupakan *activity diagram* untuk *login* admin. Admin diminta untuk memasukkan *username* dan *password* kemudian sistem akan melakukan validasi data, jika sesuai maka sistem akan menampilkan halaman utama admin. Jika tidak sesuai maka sistem akan menampilkan pesan “Login Gagal” dan menampilkan halaman login kembali.



Gambar 3.4 Activity diagram login admin

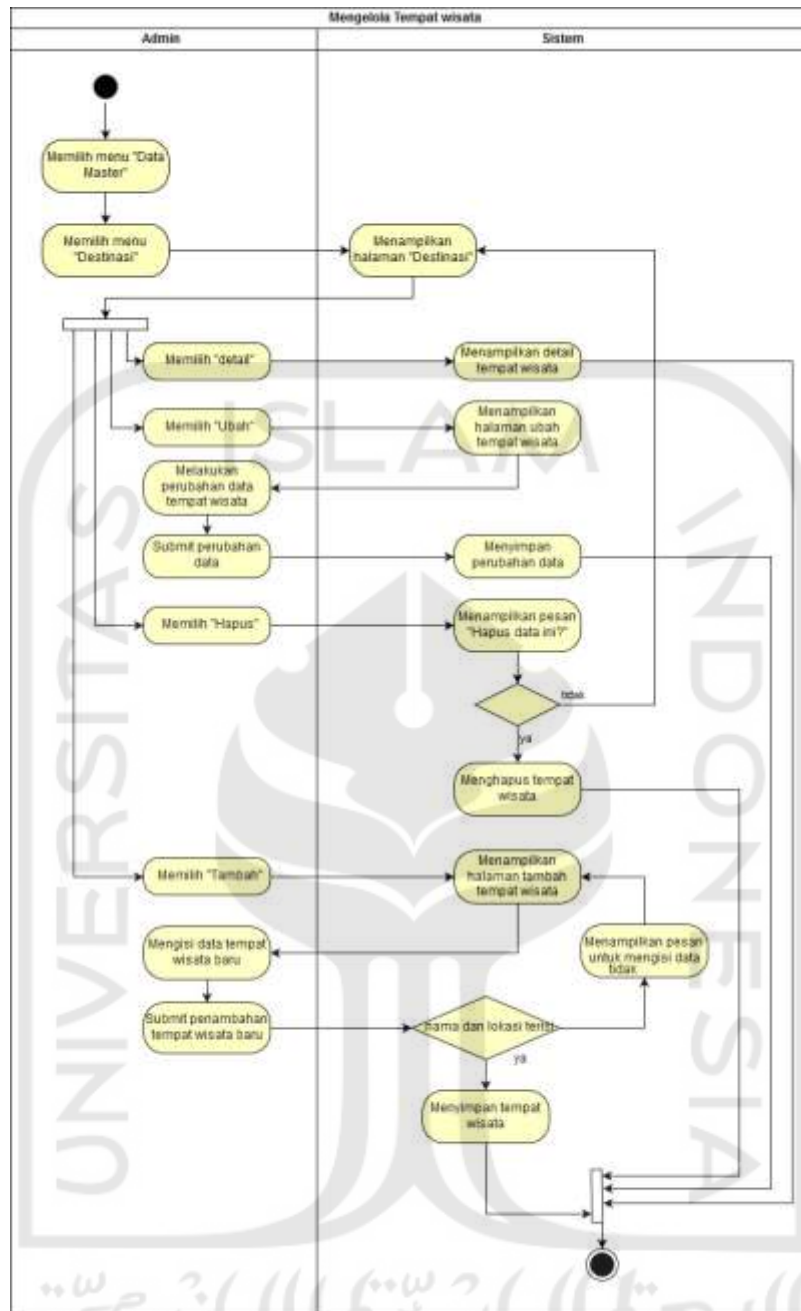
2. Activity diagram mengelola destinasi

Pada Gambar 3.5 merupakan *activity diagram* admin dalam melakukan pengelolaan data tempat wisata. Admin dapat melihat detail tiap tempat wisata, dapat mengubah data tempat wisata, menghapus tempat wisata, dan menambah tempat wisata.

Saat ingin melihat detail tempat wisata, admin harus menekan tombol detail kemudian sistem akan menampilkan detail tempat wisata yang dipilih. Untuk melakukan perubahan data tempat wisata admin harus menekan tombol ubah, kemudian sistem akan menampilkan halaman ubah tempat wisata yang berisi *form* dengan isi data sebelumnya, setelah itu admin dapat mengubah data tempat wisata dan sistem akan menyimpannya.

Apabila admin ingin menghapus tempat wisata, maka admin harus menekan tombol hapus, kemudian sistem akan menampilkan pesan “hapus data ini?”, jika ya maka sistem akan menghapusnya dan sebaliknya jika tidak maka sistem tetap menampilkan halaman destinasi. Untuk menambah tempat wisata baru, admin harus memilih tombol tambah dan kemudian sistem akan menampilkan halaman yang berisi form untuk menambahkan data tempat wisata baru. Setelah admin mengisi *form* dan melakukan *submit* data pada sistem, sistem akan melakukan pengecekan apakah *form* nama dan lokasi dari tempat wisata yang ditambahkan sudah terisi. Jika sudah maka sistem akan menyimpannya, jika belum maka sistem akan memberi pesan untuk mengisi *form* nama dan lokasi yang kosong.

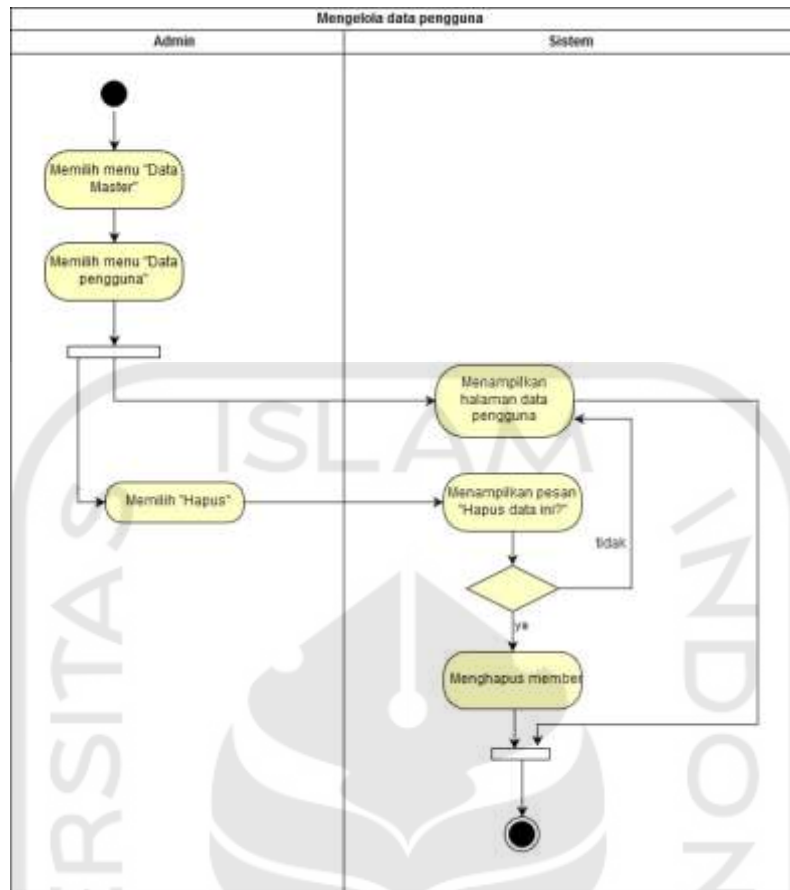




Gambar 3.5 Activity diagram mengelola tempat wisata

3. Activity diagram data pengguna

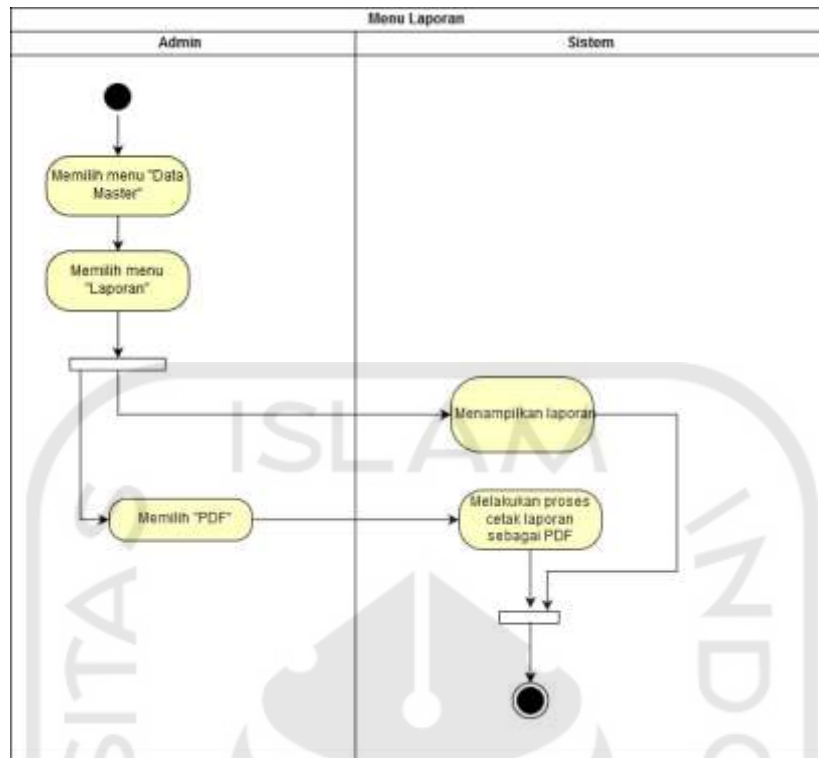
Pada Gambar 3.6 menggambarkan saat admin memilih menu "Data Pengguna" maka sistem akan menampilkan data nama pengguna sistem dalam bentuk tabel. Kemudian saat admin menekan tombol hapus, maka sistem akan menampilkan pesan "Hapus data ini?", jika ya maka sistem akan melakukan penghapusan data, jika tidak sistem tetap menampilkan halaman data pengguna.



Gambar 3.6 Activity diagram untuk menu data pengguna

4. Activity diagram menu laporan

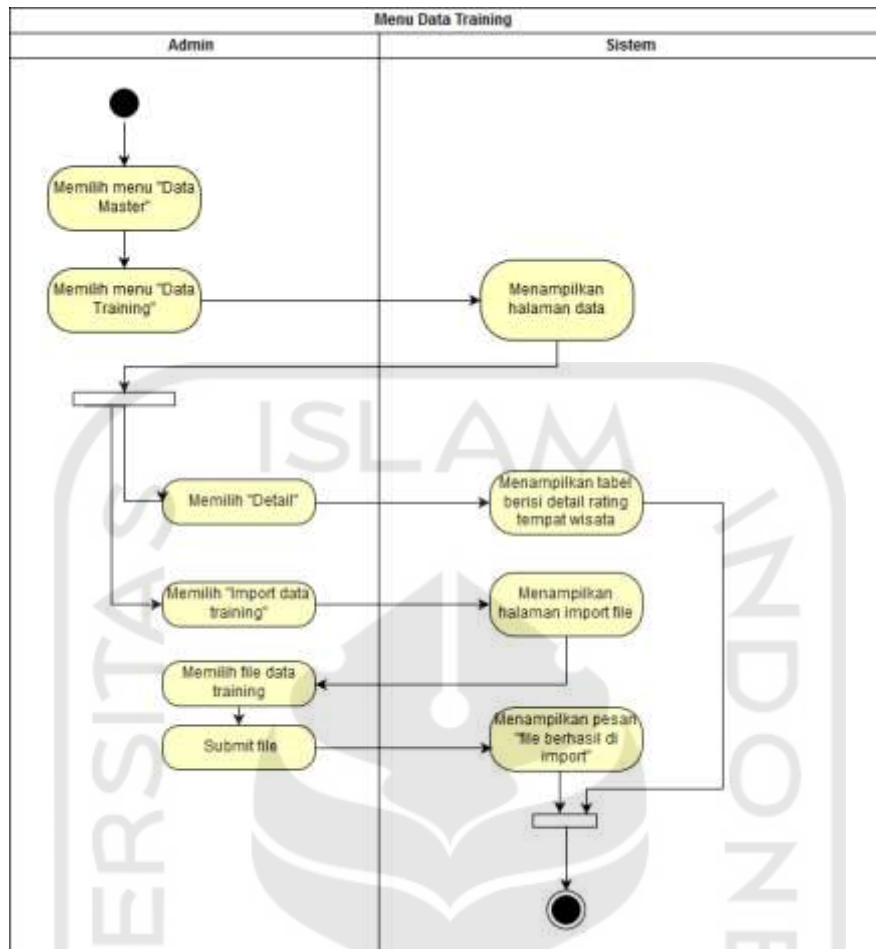
Gambar 3.7 menggambarkan activity diagram dari menu laporan. Saat admin memilih menu "Laporan" sistem akan menampilkan halaman laporan. Pada menu ini sistem juga dapat melakukan proses cetak laporan ke dalam PDF saat admin menekan tombol "PDF".



Gambar 3.7 Activity diagram menu laporan

5. Activity diagram menu data training

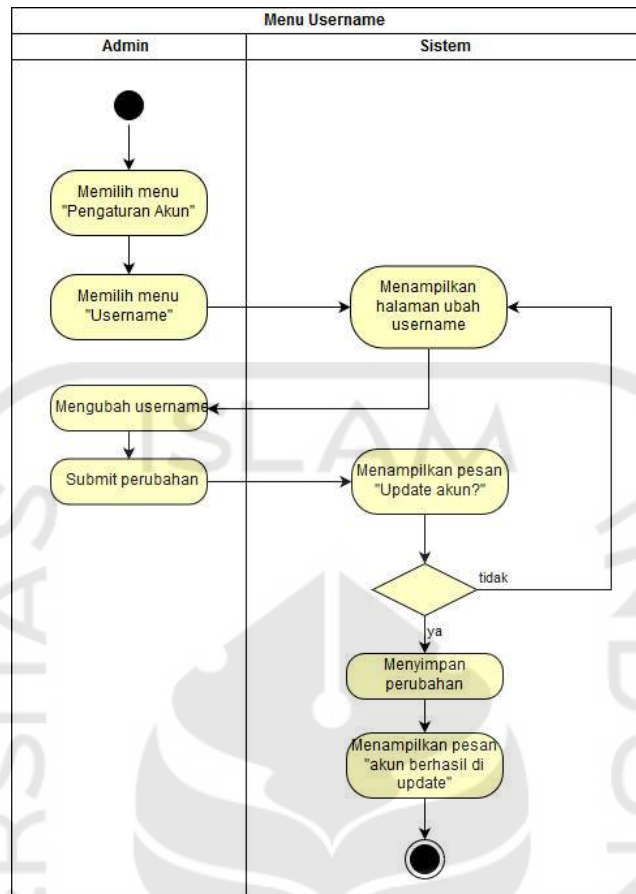
Gambar 3.8 menunjukkan *activity diagram* admin pada menu data training. Saat admin memilih menu data *training* maka sistem akan menampilkan halaman data training. Pada halaman ini admin dapat menekan tombol detail dan kemudian sistem akan menampilkan tabel yang berisi detail rating tempat wisata yang dipilih. Pada menu data *training* admin juga dapat menekan tombol “*import data training*” dan selanjutnya sistem akan menampilkan halaman untuk impor *file*. Untuk melakukan impor admin harus memilih file dan kemudian melakukan *submit file* tersebut ke sistem, selanjutnya sistem akan menampilkan pesan “file berhasil di import”.



Gambar 3.8 Activity diagram menu data training

6. Activity diagram menu ubah username

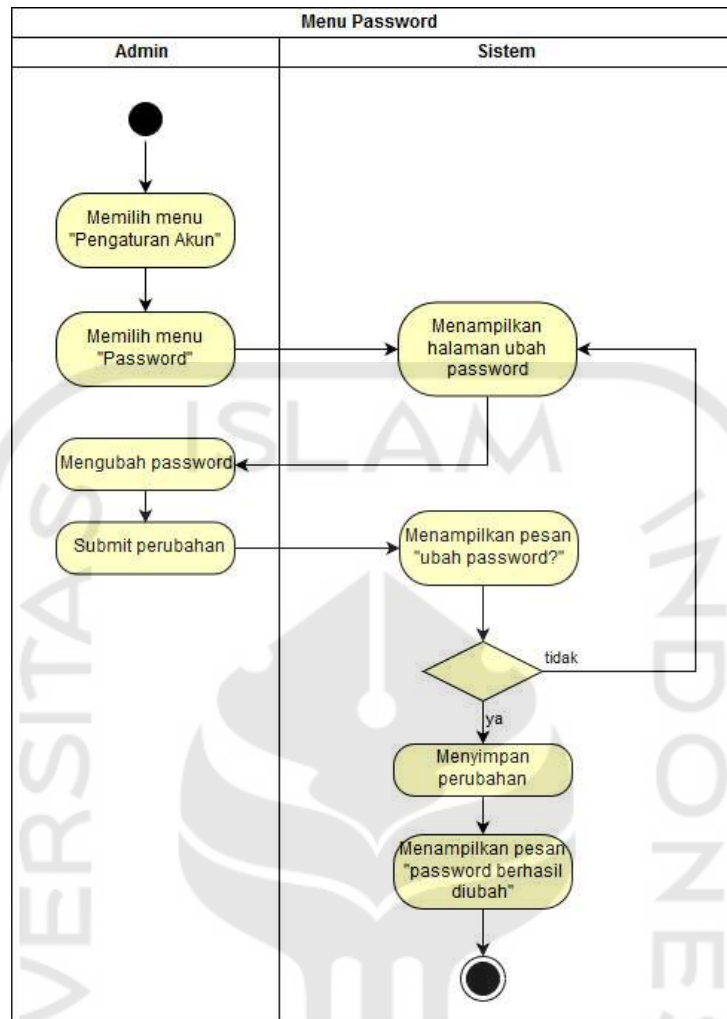
Gambar 3.9 menggambarkan saat admin memilih menu *username* maka sistem akan menampilkan halaman ubah *username* berupa *form* yang sudah terisi dengan nama *username* admin yang sedang *login*. Admin dapat melakukan perubahan *username* dan kemudian menyimpannya. Saat admin menekan tombol untuk menyimpan perubahan, selanjutnya sistem akan menampilkan pesan "*update akun?*" jika ya, maka sistem akan menyimpan perubahan tersebut. Jika tidak maka sistem akan tetap menampilkan halaman ubah *username*.



Gambar 3.9 Activity diagram pada menu username

7. Activity diagram menu ubah password

Gambar 3.10 menggambarkan saat admin memilih menu *password* maka sistem akan menampilkan halaman ubah *password*. Admin dapat melakukan perubahan *password* dan kemudian menyimpannya. Saat admin menekan tombol untuk menyimpan perubahan, selanjutnya sistem akan menampilkan pesan "ubah *password*?" jika ya, maka sistem akan menyimpan perubahan tersebut. Jika tidak maka sistem akan tetap menampilkan halaman ubah *password*..

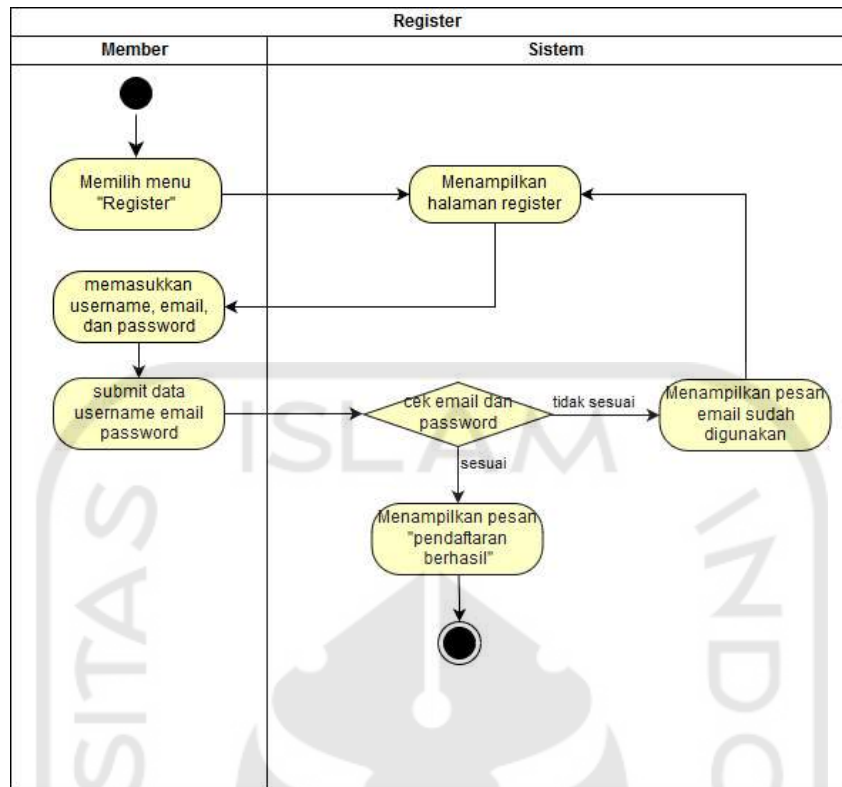


Gambar 3.10 Activity diagram pada menu password

b. Activity diagram member

1. Activity diagram register

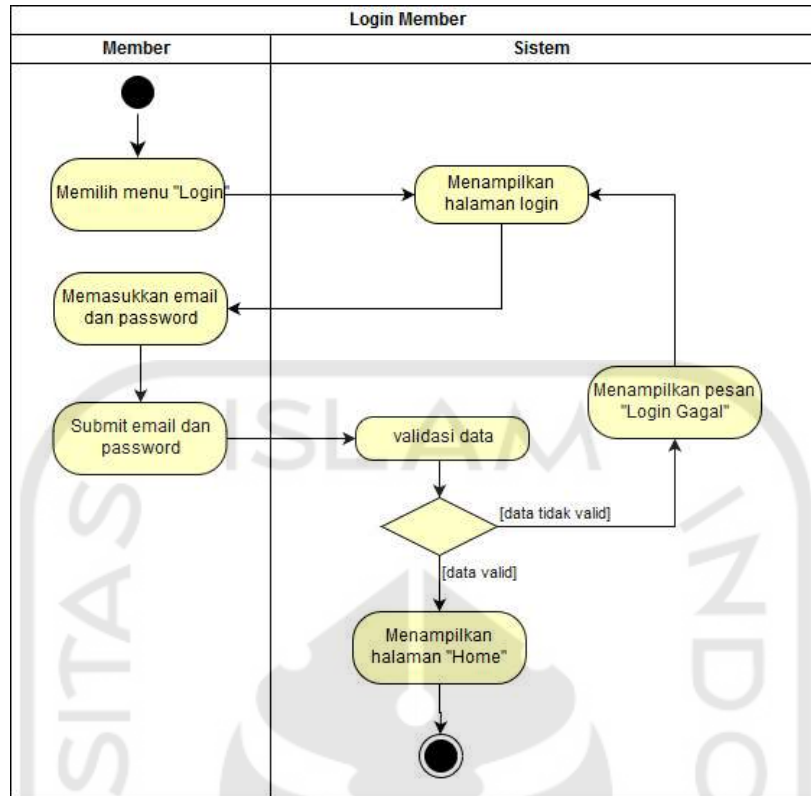
Pada Gambar 3.11 dapat dilihat proses pengguna saat melakukan *register* pada sistem. Saat pengguna memilih menu *register* sistem akan menampilkan halaman untuk melakukan pendaftaran. Pengguna harus mengisi *username*, *email*, dan *password* untuk melakukan pendaftaran. Setelah itu sistem akan mengecek apakah email sudah pernah digunakan dan password terdiri dari minimal 6 karakter. Apabila sudah sesuai maka sistem akan menampilkan pesan bahwa pendaftaran berhasil, jika tidak sesuai maka sistem akan menampilkan halaman register kembali.



Gambar 3.11 Activity diagram register member

2. Activity diagram login member

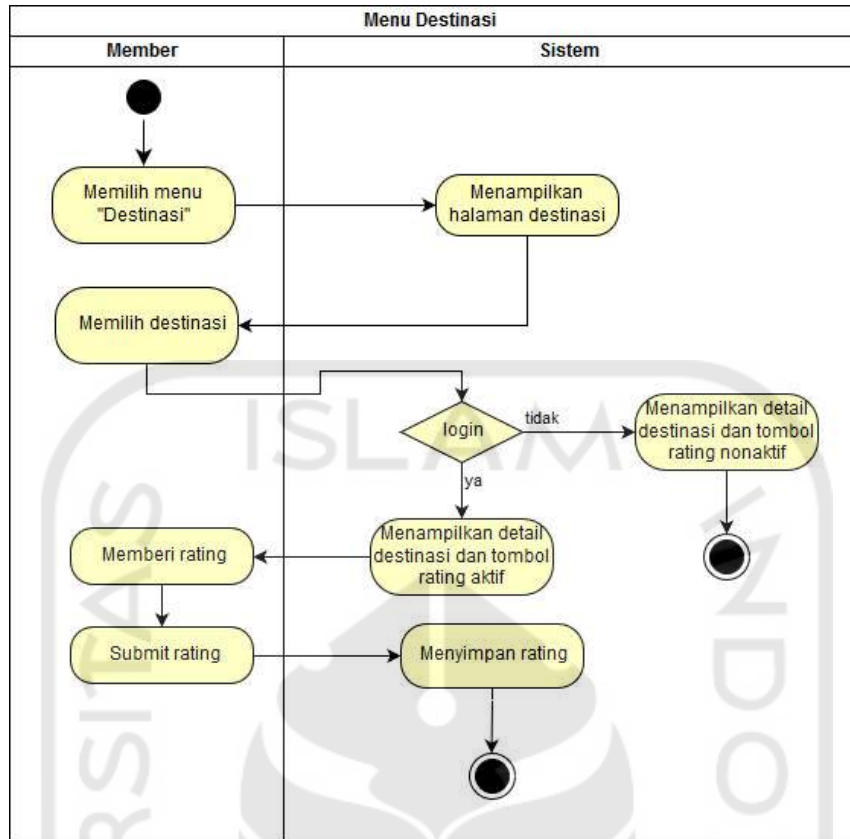
Gambar 3.12 merupakan activity diagram untuk login member. Member diminta untuk memasukkan email dan password kemudian sistem akan melakukan validasi data, jika sesuai maka sistem akan menampilkan halaman utama home. Jika tidak sesuai maka sistem akan menampilkan pesan “Login Gagal” dan menampilkan halaman login kembali.



Gambar 3.12 Activity diagram login member

3. Activity diagram menu destinasi

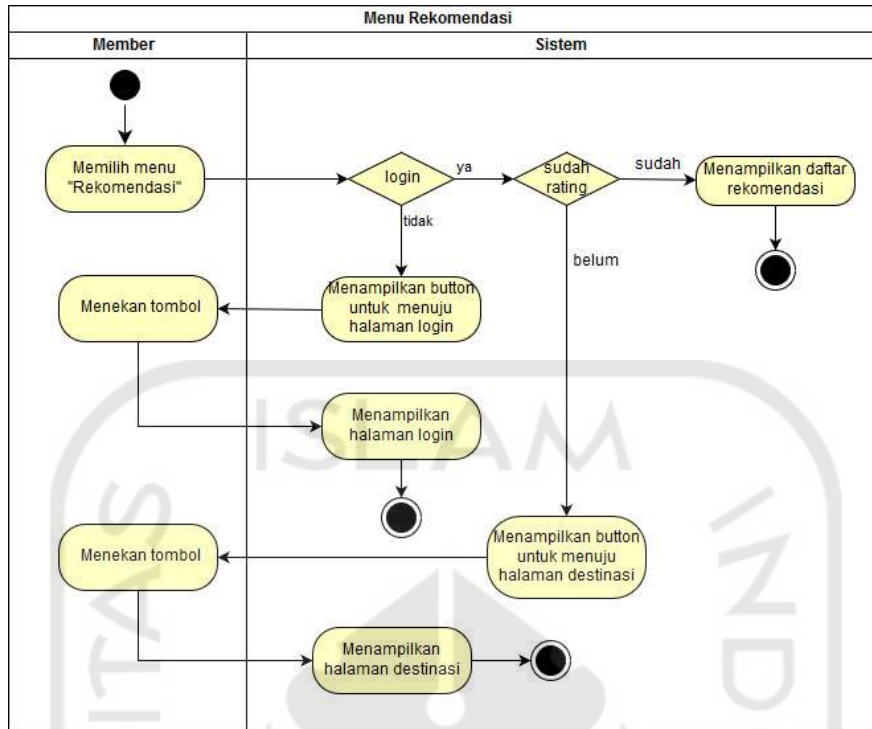
Saat pengguna sistem memilih menu “Destinasi”, sistem akan menampilkan halaman destinasi. Kemudian pengguna dapat memilih salah satu tempat wisata dan sistem akan melakukan pengecekan apakah pengguna sudah melakukan *login* atau belum. Jika sudah makan sistem akan menampilkan detail tempat wisata yang dipilih dengan tombol pemberian rating aktif. Sedangkan bagi pengguna yang tidak *login*, sistem menampilkan halaman detail tempat wisata dengan tombol pemberian rating tidak aktif. Penggambaran *activity diagram* dapat dilihat pada Gambar 3.13.



Gambar 3.13 Activity diagram menu destinasi

4. Activity diagram menu rekomendasi

Pada Gambar 3.14 menunjukkan alur *activity diagram* saat pengguna memilih menu rekomendasi. Ketika pengguna ingin melihat hasil rekomendasi dengan cara memilih menu rekomendasi, sistem akan melakukan pengecekan yang pertama yaitu apakah pengguna sudah *login* atau belum. Jika belum maka sistem akan menampilkan halaman yang berisi tombol untuk mengarahkan pengguna melakukan *login*. Jika sudah melakukan *login*, sistem akan melakukan pengecekan yang kedua yaitu apakah *member* yang sudah *login* tersebut sudah pernah memberikan *rating*. Jika belum maka sistem akan menampilkan halaman yang berisi tombol untuk mengarahkan pengguna melakukan *rating*. Jika sudah maka sistem akan menampilkan hasil rekomendasi untuk *member* tersebut.

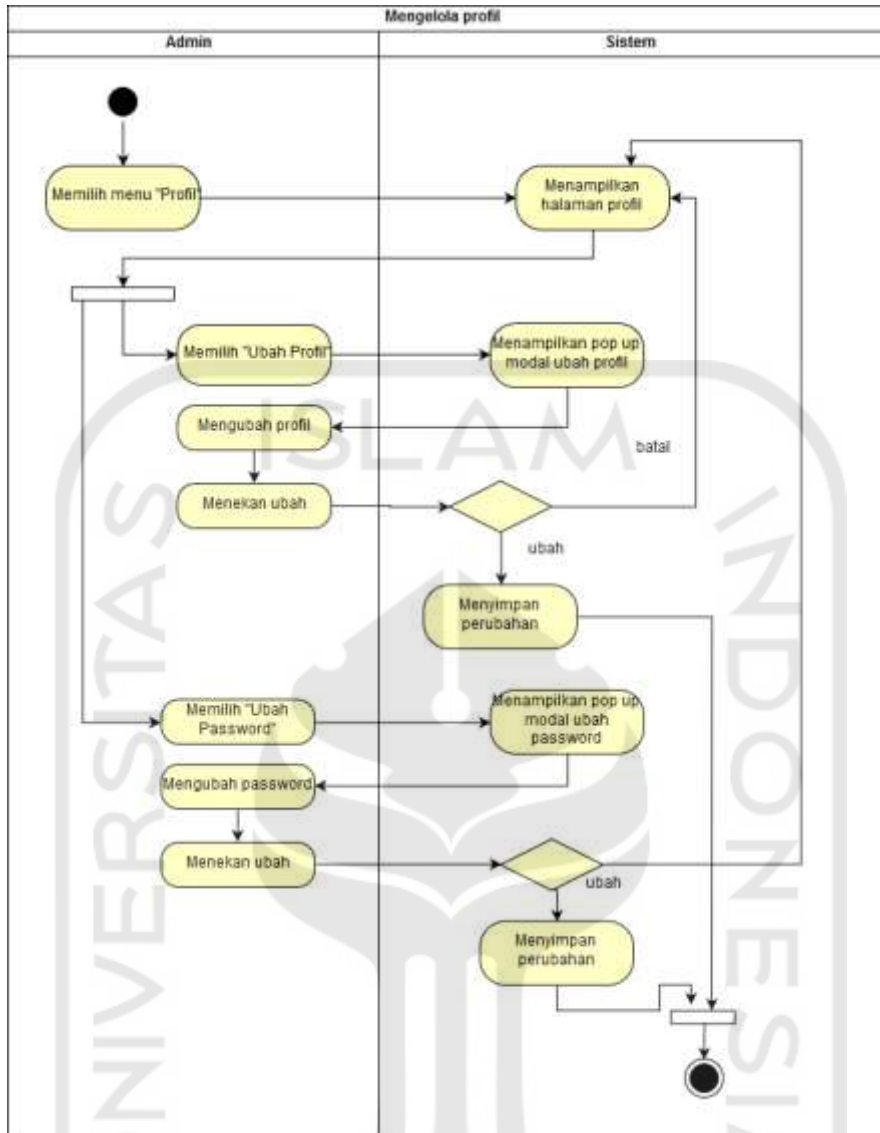


Gambar 3.14 Activity diagram menu rekomendasi

5. Activity diagram mengelola akun

Saat *member* memilih menu profil maka sistem akan menampilkan halaman profil. Pada halaman profil ini *member* dapat melakukan perubahan pada *username*, *email*, dan *password*. Ketika *member* memilih tombol “ubah profil” maka sistem akan menampilkan *pop up model* berisi *form* yang telah terisi dengan data *email* dan *username member*. Pada *pop up model* ini *member* dapat mengubah *email* dan *username* nya. Saat *member* melakukan *submit* perubahan, maka sistem akan menyimpan perubahan. Sedangkan jika *member* memilih membatalkan perubahan, maka sistem akan tetap menampilkan halaman profil.

Ketika *member* memilih tombol “ubah password” maka sistem akan menampilkan *pop up model* berisi *form* untuk melakukan perubahan *password*. Saat *member* melakukan *submit* perubahan, maka sistem akan menyimpan perubahan. Sedangkan jika *member* memilih membatalkan perubahan, maka sistem akan tetap menampilkan halaman profil.



Gambar 3.15 Activity diagram menu profil

3.3 Contoh Kasus Item Based Collaborative Filtering

Untuk memberikan gambaran mengenai cara kerja sistem rekomendasi wisata dengan *item based collaborative filtering* berikut adalah contoh kasusnya :

Tabel 3.1 Rate tempat wisata dari pengguna

Pengguna	I1	I2	I3	I4	I5	\bar{U}
U1	4	-	4	4	4	4
U2	3	-	-	3	3	3
U3	3	-	-	-	-	3
U4	4	-	-	-	-	4
U5	5	4	4	4	4	4,2
U6	5	4	4	-	5	4,5
U7	4	-	-	-	4	4

Langkah-langkah dalam menghasilkan rekomendasi adalah:

- Mengumpulkan data pengguna yang telah memberi *rate* dan menghitung rata-rata *rate*. Pemberian *rate* dilakukan oleh pengguna yang sudah terdaftar dan *login* sebagai member. Simulasi data pengguna yang telah memberi *rate* dapat dilihat pada Tabel 3.1 Rate tempat wisata dari pengguna. Pada tabel tersebut U melambangkan pengguna, I melambangkan tempat wisata, dan \bar{U} melambangkan nilai rata-rata *rate* dari tiap pengguna.
- Menghitung nilai kemiripan. Untuk menghitung nilai kemiripan (*similarity*) antar tempat wisata digunakan rumus *adjusted cosine similarity* sebagai berikut:

$$sim(i, j) = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_u)(r_{u,j} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_u)^2}} \quad (3.1)$$

Dengan :

$sim(i, j)$: nilai kemiripan antara produk i dan produk j

$u \in U$: himpunan pengguna u yang memberikan *rate* pada produk i dan produk j

$r_{u,i}$: *rating* pengguna u pada produk i

$r_{u,j}$: *rating* pengguna u pada produk j

\bar{r}_u : rata-rata *rating* pengguna u

Dalam menghitung nilai kemiripan, nilai yang dihasilkan akan berkisar antara +1,0 dan -1,0. Nilai kemiripan mendekati +1,0 maka produk memiliki kemiripan yang tinggi dan nilai kemiripan mendekati -1,0 maka produk saling bertolak belakang

Contoh perhitungan nilai kemiripan (*similarity*) untuk $I1$ dan $I2$ adalah

$$\begin{aligned} sim(I1, I2) &= \frac{\sum_{u \in U} (r_{u,I1} - \bar{r}_u)(r_{u,I2} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,I1} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,I2} - \bar{r}_u)^2}} \\ &= \frac{(r_{u5,I1} - \bar{r}_{u5})(r_{u5,I2} - \bar{r}_{u5}) + (r_{u6,I1} - \bar{r}_{u6})(r_{u6,I2} - \bar{r}_{u6})}{\sqrt{(r_{u5,I1} - \bar{r}_{u5})^2 + (r_{u6,I1} - \bar{r}_{u6})^2} \sqrt{(r_{u5,I2} - \bar{r}_{u5})^2 + (r_{u6,I2} - \bar{r}_{u6})^2}} \\ &= \frac{(5 - 4,2)(4 - 4,2) + (5 - 4,5)(4 - 4,5)}{\sqrt{(5 - 4,2)^2 + (5 - 4,5)^2} \sqrt{(4 - 4,2)^2 + (4 - 4,5)^2}} \\ &= \frac{-0,41}{\sqrt{0,89} \sqrt{0,29}} \end{aligned}$$

$$\begin{aligned}
 &= \frac{-0,41}{0,508} \\
 &= -0,81
 \end{aligned}$$

Simulasi nilai kemiripan antar tempat wisata dapat dilihat pada Tabel 3.2 dengan I melambangkan tempat wisata. Berdasarkan Tabel 3.2 terdapat empat pasang tempat wisata yang memiliki nilai kemiripan lebih dari 0,5 yaitu $I2$ dengan $I3$, $I2$ dengan $I4$, $I3$ dengan $I4$, dan $I4$ dengan $I5$.

Tabel 3.2 Nilai kemiripan antartempat wisata

Tempat wisata 1	Tempat wisata 2	Nilai Kemiripan
$I1$	$I2$	-0,81
$I1$	$I3$	-0,807
$I1$	$I4$	-1
$I1$	$I5$	0,177
$I2$	$I3$	1*
$I2$	$I4$	1*
$I2$	$I5$	-0,72
$I3$	$I4$	1*
$I3$	$I5$	-0,72
$I4$	$I5$	1*

*Pasangan tempat wisata yang memiliki *similarity* lebih besar dari 0,5

- c. Menghitung nilai prediksi *rate* tempat wisata menggunakan *weighted sum*

Daftar tempat wisata yang dijadikan rekomendasi adalah pasangan tempat wisata yang nilai kemiripannya lebih besar dari 0,5 ($I2$, $I3$, $I4$, dan $I5$). Untuk menentukan rekomendasi bagi tiap pengguna, maka perlu dilakukan perhitungan nilai prediksi *rate* menggunakan rumus *weighted sum* sebagai berikut:

$$P(u, j) = \frac{\sum_{i \in I} (R_{u,i} * S_{i,j})}{\sum_{i \in I} |S_{i,j}|} \quad (3.2)$$

Dengan:

$P(u, j)$: prediksi untuk pengguna u pada produk j .

$i \in I$: himpunan produk yang mirip dengan produk j .

$R_{u,i}$: *rate* pengguna u pada produk i .

$S_{i,j}$: nilai kemiripan antara produk i dan produk j .

Contoh perhitungan nilai prediksi *rate* (*weighted sum*) untuk U1 adalah:

$$P(u,j) = \frac{\sum_{i \in I} (R_{u,i} * S_{i,j})}{\sum_{i \in I} |S_{i,j}|}$$

$$\begin{aligned} P(U1, I2) &= \frac{(R_{u1,I3} * S_{I2,I3}) + (R_{u1,I4} * S_{I2,I4})}{|S_{I2,I3}| + |S_{I2,I4}|} \\ &= \frac{(4 * 1) + (4 * 1)}{|1| + |1|} \\ &= \frac{8}{1 + 1} \\ &= 4 \end{aligned}$$

Tabel 3.3 Nilai prediksi tempat wisata

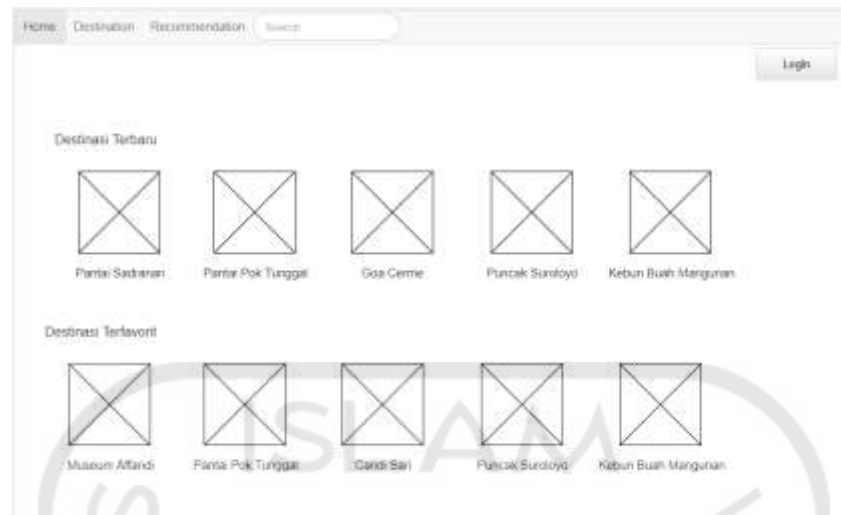
Pengguna	Prediksi rate tempat wisata
<i>U1</i>	(<i>I2</i> , 4)
<i>U2</i>	(<i>I2</i> , 3)(<i>I3</i> ,3)
<i>U6</i>	(<i>I4</i> , 4)
<i>U7</i>	(<i>I4</i> , 4)

Prediksi yang diberikan kepada pengguna adalah prediksi tempat wisata yang belum pernah diberi *rate* oleh pengguna. Pengguna yang memiliki rekomendasi hanya 4 pengguna, yaitu *U1*, *U2*, *U6*, dan *U7*. Beberapa pengguna tidak mendapat rekomendasi karena *rate* yang diberikan bertolak belakang dengan *rate* yang diberikan oleh pengguna lain.

3.3.1 Perancangan Antarmuka

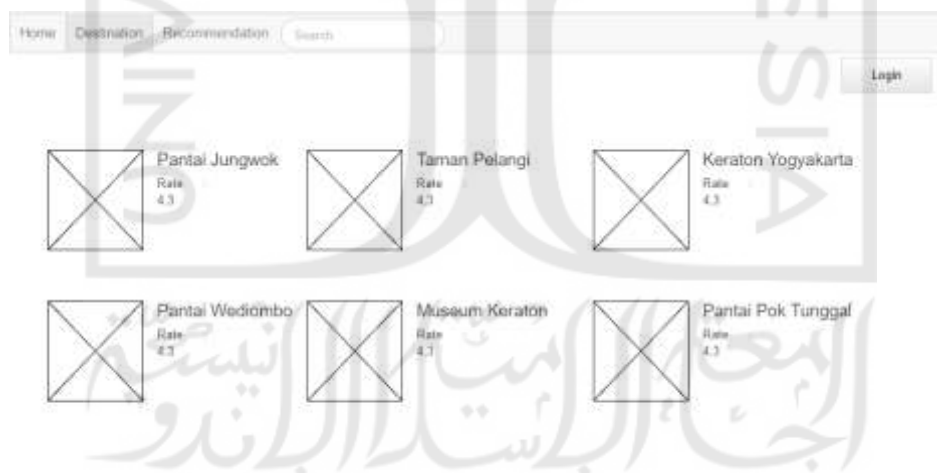
Berikut adalah perancangan antarmuka untuk aplikasi pemberi rekomendasi tempat wisata:

Pada Gambar 3.16 merupakan halaman utama dari sistem rekomendasi tempat wisata. Pada halaman ini menampilkan 5 destinasi yang baru ditambahkan oleh *admin* dan 5 destinasi terfavorit. Destinasi terfavorit menampilkan destinasi yang memiliki rata-rata nilai *rating* tertinggi.



Gambar 3.16 Rancangan antarmuka halaman *home*

Pada Gambar 3.17 merupakan halaman menu Destinasi yang menampilkan seluruh foto dan nama tempat wisata yang tersimpan di dalam *database*. Apabila pengguna memilih salah satu tempat wisata yang ada di menu Destinasi, maka akan tampil detail dari tempat wisata yang dipilih dan *radio button* untuk memberi *rating* tempat wisata yang dipilih. Rancangan antarmuka dari detail tempat wisata tersebut dapat dilihat pada Gambar 3.18.

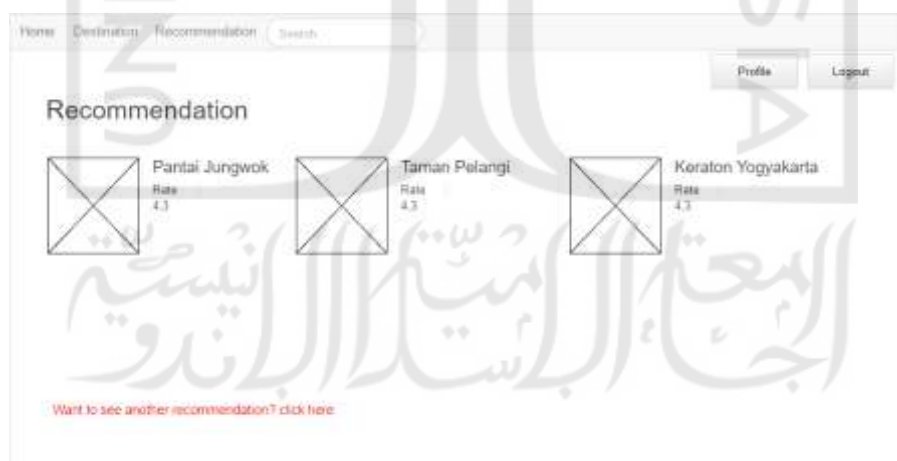


Gambar 3.17 Rancangan antarmuka halaman destinasi

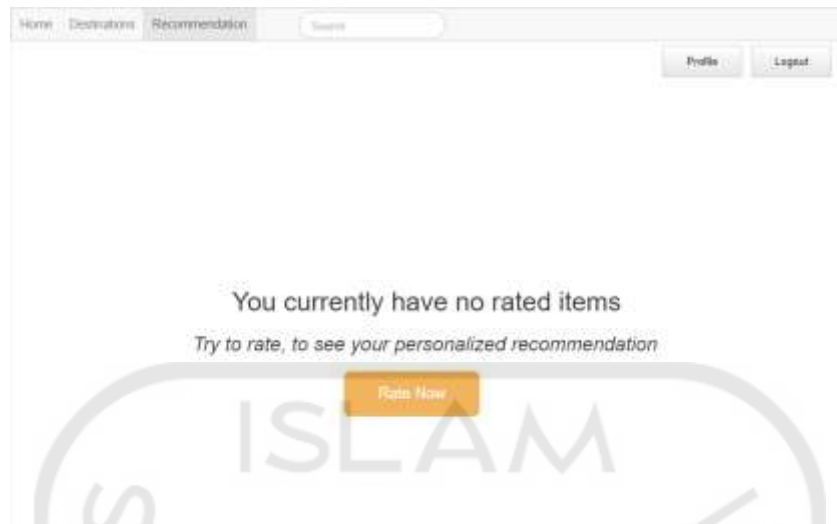


Gambar 3.18 Rancangan antarmuka detail tempat wisata

Pada Gambar 3.19 adalah tampilan daftar rekomendasi tempat wisata untuk pengguna yang sudah *login* dan pernah memberikan *rating* sebelumnya. Daftar rekomendasi tempat wisata yang ditampilkan untuk setiap pengguna akan berbeda-beda sesuai dengan histori *rating* yang sudah pernah dilakukan sebelumnya. Apabila pengguna belum pernah memberi *rating* sebelumnya, maka pada halaman menu *Recommendation* terdapat tombol yang mengarahkan pengguna ke menu *Destination* seperti yang terlihat pada Gambar 3.20. Apabila pengguna belum melakukan *login*, maka pada halaman menu *Recommendation* terdapat tombol yang akan mengarahkan pengguna untuk *login* terlebih dahulu, seperti yang terlihat pada Gambar 3.21.



Gambar 3.19 Rancangan antarmuka halaman *recommendation* untuk *user* yang sudah pernah memberikan *rating*

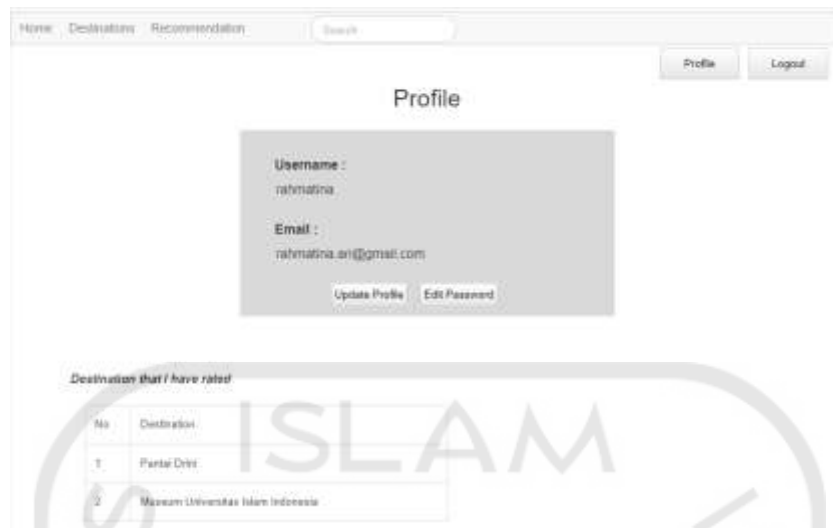


Gambar 3.20 Rancangan antarmuka halaman *recommendation* untuk *user login* yang belum pernah memberikan *rating*



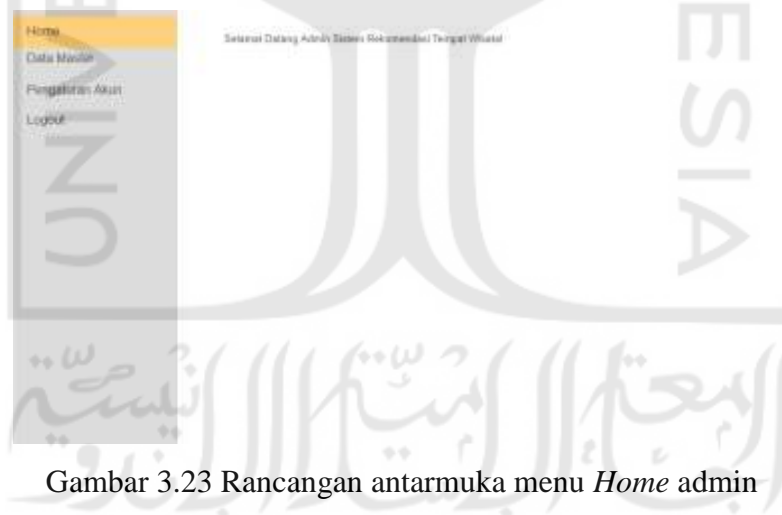
Gambar 3.21 Rancangan antarmuka halaman *Recommendation* untuk *user* tidak *login*

Pada Gambar 3.22 merupakan rancangan antarmuka untuk menampilkan profil pengguna yang meliputi nama, email, dan daftar tempat wisata yang sudah pernah diberi *rate* oleh pengguna.



Gambar 3.22 Rancangan antarmuka halaman *Profile*

Pada Gambar 3.23 merupakan halaman utama pada admin sebagai pengelola sistem. Pada *sidebar* terdapat empat menu yaitu *Home*, *Data Master* yang berisi seluruh data pengguna sistem dan tempat wisata, *Pengaturan Akun* untuk mengatur akun admin, dan *Logout* untuk keluar dari sistem.



Gambar 3.23 Rancangan antarmuka menu *Home* admin

Gambar 3.24 adalah rancangan untuk antarmuka sub-menu dari *Data Master* yaitu *Destinasi*. Pada menu *Destinasi* akan menampilkan seluruh data tempat wisata yang tersimpan dalam *database*. Di bawah tabel terdapat tombol untuk menambahkan tempat wisata baru. Rancangan antarmuka untuk penambahan tempat wisata oleh admin dapat dilihat pada Gambar 3.25.

No	Nama Tempat Wisata	Lokasi	Opsi
1	Pantai West Ombak	Jaya, Ombak, Perancis, Jafa, Ombak, Kabupaten Gunung Kidul, Daerah Istimewa Yogyakarta	Tambah Ubah Hapus
2	Desa Wisata Gunung Kai Ngampelre	Kelurahan Kuri, Ngampelre, Ponor, Kabupaten Gunung Kidul, Daerah Istimewa Yogyakarta 55882	Tambah Ubah Hapus
3	Watu Sema	Bromo Lor, Hargobilo, Klatap, Kabupaten Klatap, Daerah Istimewa Yogyakarta 55503	Tambah Ubah Hapus

Gambar 3.24 Rancangan antarmuka sub-menu Destinasi

Nama Tempat Wisata

Lokasi

Telepon

Harga

Faksi

Gambar 3.25 Rancangan antarmuka tambah tempat wisata oleh admin

Pada sub-menu Data Pengguna menampilkan seluruh nama pengguna yang tersimpan di *database*. Admin hanya bisa menghapus data pengguna. Terlihat pada Gambar 3.26

No	Nama Member	Opsi
1	Guest	Hapus
2	Admin	Hapus
3	Admin	Hapus

Gambar 3.26 Rancangan antarmuka sub-menu Data Pengguna

Pada sub-menu Laporan akan menampilkan laporan terbaru mengenai tempat wisata terfavoriturut berdasarkan nilai *rating* tertinggi. Dapat dilihat pada Gambar 3.27.

No	Nama Tempat Wisata	Rating
1	Pantai Wisata Cotto	4,6
2	Data Wisata Sunang Api Nglanggeran	3,8
3	Waduk Sampe	3,7

Gambar 3.27 Rancangan antarmuka sub-menu Laporan

Sub-menu Data Training menampilkan seluruh data pengguna yang sudah memberikan *rating* pada tempat wisata. Data training merupakan data yang diolah untuk menghasilkan rekomendasi tempat wisata bagi masing-masing pengguna. Terdapat tombol *import* yang berfungsi untuk *import* data training berupa file .xls atau .xlsx.

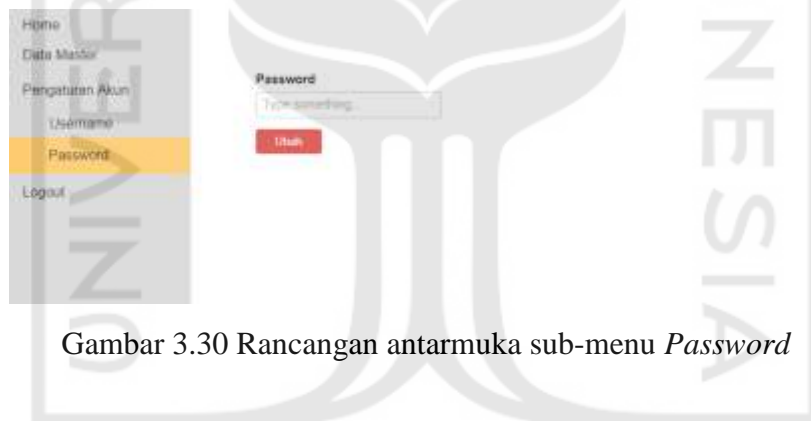
No	Nama Member	Nama Tempat Wisata	Rating
1	Bayu	Pantai Wisata Cotto	4
2	Rahel	Data Wisata Sunang Api Nglanggeran	4
3	Rahel	Waduk Sampe	3

Gambar 3.28 Rancangan antarmuka sub-menu Data Training

Pada menu Pengaturan Akun memiliki dua sub-menu yaitu *Username* dan *Password* yang berfungsi untuk mengubah *username* dan *password* admin. Pada *form input* sub-menu *Username* akan menampilkan *username* yang sedang digunakan oleh admin sebelum melakukan perubahan. Rancangan antarmuka untuk menu tersebut dapat dilihat pada Gambar 3.29 dan Gambar 3.30.



Gambar 3.29 Rancangan antarmuka sub-menu *Username*



Gambar 3.30 Rancangan antarmuka sub-menu *Password*

3.4 Skenario Pengujian

Skenario pengujian yang akan diterapkan pada aplikasi rekomendasi tempat wisata ini adalah *black box testing*, *user acceptance test* (UAT), dan *mean absolute error* (MAE).

3.4.1 *Black Box Testing*

Black box testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak (Irwan, 2019). Pengujian aplikasi dilakukan dengan menggunakan tabel yang berisi deskripsi pengujian, *test case* (sebagai kasus uji *input*), *output* yang diharapkan, hasil pengujian (*output* yang dihasilkan),

dan kesimpulan dari sama tidaknya hasil pengujian dengan *output* yang diharapkan.

3.4.2 Perancangan *User Acceptance Test* (UAT)

User acceptance test atau uji penerimaan pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa *software* yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian (*testing*) sudah bisa dianggap memenuhi kebutuhan dari pengguna (Sidik, 2019).

Pengujian UAT dilakukan dengan cara meminta responden untuk menggunakan aplikasi sebagaimana mestinya, dan selanjutnya responden diminta untuk mengisi beberapa pertanyaan tentang aplikasi yang mereka gunakan dalam bentuk kuesioner. Kuisisioner diisi dengan lima macam jawaban yaitu Sangat Tidak Setuju (STS), Tidak Setuju (TS), Netral (N), Setuju (S), dan Sangat Setuju (SS).

Tabel 3.4 Rancangan kuisisioner aplikasi rekomendasi tempat wisata

No	Pertanyaan	Skala Penilaian				
		STS	TS	N	S	SS
1.	Sistem memberikan kemudahan kepada anda untuk melakukan <i>register</i> .					
2.	Sistem memberi kemudahan dalam mencari referensi tempat wisata di DIY.					
3.	Sistem memberi kemudahan dalam melakukan <i>rating</i> pada tempat wisata.					
4.	Sistem memiliki tampilan yang baik dan menarik					
5.	Sistem dapat memberi informasi yang tepat tentang tempat-tempat wisata di DIY.					
6.	Sistem dapat memberikan pesan apabila anda belum <i>login</i> saat akan memberi <i>rating</i> .					
7.	Sistem memberikan kemudahan pada anda dalam					

	memperoleh rekomendasi tempat wisata di DIY					
8.	Sistem dapat memberikan rekomendasi yang sesuai dengan selera anda					
9.	Fitur rekomendasi pada sistem bermanfaat bagi anda saat ingin mengambil keputusan dalam memilih tempat wisata di DIY					

3.4.3 Mean Absolute Error

Mean absolute error (MAE) adalah pengujian yang dilakukan untuk mengetahui akurasi dari rekomendasi yang dihasilkan dengan cara membandingkan nilai prediksi yang dihasilkan sistem dengan nilai sebenarnya yang diberikan oleh pengguna (Sarwar, Karypis, Konstan, & Riedl, 2001). Semakin kecil nilai MAE yang dihasilkan maka prediksi atau rekomendasi yang dihasilkan semakin baik.

$$MAE = \frac{\sum_{i=1}^N |P_i - R_i|}{N} \quad (3.3)$$

Dengan :

MAE : nilai rata-rata kesalahan hitung

P_i : nilai prediksi pada produk i

R_i : nilai *rate* sebenarnya pada produk i

N : jumlah produk yang dihitung

Contoh perhitungan MAE adalah sebagai berikut. Pengguna U1 diasumsikan belum pernah memberi *rating* pada I2 dan U2 diasumsikan belum pernah memberi *rating* pada I2 dan I3, seperti yang terlihat pada Tabel 3.1. Kemudian setelah dihitung dengan menggunakan *adjusted cosine similarity* dan *weighted sum* didapatkan hasil prediksi *rating* dari U1 untuk I2 adalah 4, serta prediksi *rating* dari U2 untuk I2 dan I3 masing-masing adalah 3. Selanjutnya diasumsikan U1 memberi *rating* pada I2 sebesar 4 dan U2 diasumsikan memberi *rating* pada I3 sebesar 4. Maka didapatkan perhitungan nilai MAE :

$$\begin{aligned} MAE &= \frac{|4 - 4| + |3 - 4|}{2} \\ &= 0,5 \end{aligned}$$

Selisih antara *rating* prediksi dengan *rating* sebenarnya itulah yang akan dihitung kesalahannya menggunakan MAE. Sehingga dari perhitungan diatas diketahui bahwa nilai MAE yang didapatkan adalah 0,5.

