

BAB V. IMPLEMENTASI DAN PENGUJIAN

Setelah melalui tahapan analisis dan perancangan sistem pada bab sebelumnya, pada bab ini akan dibahas mengenai implementasi dari perencanaan dan pengujian sistem. Tahap implementasi adalah tahap mengubah desain menjadi aplikasi.

5.1 Implementasi Source Code

5.1.1 Implementasi Source Code Register User

Berikut merupakan *source code register data users* yang digunakan untuk membuat akun agar dapat akses fitur tambah data yang dapat dilakukan oleh *user*, dapat dilihat pada tabel 5.1 .

Tabel 5. 1 Source code register

<i>Source code register</i> atau membuat akun
<pre>function register() { \$this->form_validation->set_rules('username', 'username', 'required'); \$this->form_validation->set_rules('email', 'email', 'required'); \$this->form_validation->set_rules('password', 'Password', 'min_length[6]', 'required trim'); if (\$this->form_validation->run() == false) { \$this->load->view('header'); \$this->load->view('content/register');; } else { \$username = ucfirst(\$this->input->post('username')); \$email = \$this->input->post('email'); \$password = md5(\$this->input->post('password')); \$data = array('username' => \$username, 'email' => \$email, 'password' => \$password); } }</pre>

```

        $this->Auth_model->register('user', $data);
        $this->session->set_flashdata('pesan', 'daftar akun
berhasil');
        redirect('auth');
    }
}

```

5.1.2 Implementasi Source CRUD Data Hotel

Berikut merupakan *source code* CRUD data hotel yang akan digunakan untuk pengolahan data sistem pendukung keputusan hotel yang dapat dilakukan oleh *admin* pada tabel 5.2 .

Tabel 5. 2 Source code CRUD data

<i>Source code</i> tambah data hotel
<pre> public function tambah() { \$nama_hotel = ucfirst(\$this->input->post('nama_hotel')); \$harga = \$this->input->post('harga'); \$latitude = \$this->input->post('latitude'); \$longitude = \$this->input->post('longitude'); \$bintang = \$this->input->post('bintang'); \$tipe_kamar = \$this->input->post('tipe_kamar'); \$photo = \$this->input->post('foto'); \$image = \$this->UploadImage(\$photo); \$lokasi_hotel = \$this->input->post('lokasi_hotel'); \$data = array('nama_hotel' => \$nama_hotel, 'harga' => \$harga, 'latitude' => \$latitude, 'longitude' => \$longitude, 'bintang' => \$bintang, 'tipe_kamar' => \$tipe_kamar, 'foto_hotel' => \$image,); \$sid = \$this->Malternatif->addData('hotel', \$data); </pre>

```

        $data2 = array('fkHotel' => $id, 'lokasi_hotel' =>
        $lokasi_hotel);
        $this->Malternatif->lokasi($data2);
        $this->session->set_flashdata('pesan', 'Data Berhasil
        Dimasukkan');
        redirect('alternatif');
    }

```

Source code upload foto hotel

```

    public function UploadImage($data_photo)
    {
        $config['upload_path']           = './assets/img/foto-
        hotel/';
        $config['allowed_types']         = 'jpeg|jpg|png';
        $config['max_size']              = 2048;
        $this->upload->initialize($config);
        if (!$this->upload->do_upload('foto')) {
            return $data_photo = "no-image.png";
        } else {
            return $data_photo = $this->upload->file_name;
        }
    }

```

Source code ubah data hotel

```

    public function ubah()
    {
        $id = $this->input->post('id');
        $nama = $this->input->post('nama');
        $harga = $this->input->post('harga');
        $latitude = $this->input->post('latitude');
        $longitude = $this->input->post('longitude');
        $bintang = $this->input->post('bintang');
        $tipe = $this->input->post('tipe');
        $photo = $this->input->post('foto');
        $image = $this->UploadImage($photo);
        $lokasi_hotel = $this->input->post('lokasi_hotel');

        if ($image == "no-image.png") {
            $data = array(
                'nama_hotel' => $nama,

```

```

        'harga' => $harga,
        'latitude' => $latitude,
        'longitude' => $longitude,
        'bintang' => $bintang,
        'tipe_kamar' => $tipe,
    );
} else {
    $data = array(
        'nama_hotel' => $nama,
        'harga' => $harga,
        'latitude' => $latitude,
        'longitude' => $longitude,
        'bintang' => $bintang,
        'tipe_kamar' => $tipe,
        'foto_hotel' => $image,
    );
}

$data2 = array(
    'lokasi_hotel' => $lokasi_hotel
);

$where = array(
    'idhotel' => $id
);

$where2 = array(
    'fkHotel' => $id
);

$this->Malternatif->editDataHotel($data, $where);
$this->Malternatif->editDataLokasi($data2, $where2);
$this->session->set_flashdata('pesan', 'Data Berhasil
Diubah');
    redirect('alternatif');
}

```

Source code hapus data hotel

```
public function delete()
```

```

{
    $id = $this->input->post('id');
    $where = array(
        'idhotel' => $id
    );

    $where2 = array(
        'fkHotel' => $id
    );
    $this->Malternatif->deleteHotel($where, 'hotel');

    $this->Malternatif->deleteLokasi($where2, 'lokasi');
    redirect('alternatif');
}

```

5.1.3 Implementasi Source kirim saran

Berikut merupakan *source code* fitur saran yang dapat dilakukan oleh *user* dapat dilihat pada tabel 5.3 .

Tabel 5. 3 Source code kirim saran

<i>Source code</i> kirim saran
<pre> <script type="text/javascript"> var Tawk_API = Tawk_API {}, Tawk_LoadStart = new Date(); (function() { var s1 = document.createElement("script"), s0 = document.getElementsByTagName("script")[0]; s1.async = true; s1.src = 'https://embed.tawk.to/6039b3a81c1c2a130d62f0a3/1evgmbjjf'; s1.charset = 'UTF-8'; s1.setAttribute('crossorigin', '*'); s0.parentNode.insertBefore(s1, s0); })(); </script> </pre>

5.1.4 Implementasi Source Code Perhitungan SPK

Berikut merupakan *source code* perhitungan sistem pendukung keputusan hotel dapat dilihat pada tabel 5.4 .

Tabel 5. 4 Source code VS-TOPSIS

<i>Source code</i> perhitungan VS-TOPSIS
<pre>function distance(\$lat1, \$lon1, \$lat2, \$lon2, \$unit) { \$theta = \$lon1 - \$lon2; \$dist = sin(deg2rad(\$lat1)) * sin(deg2rad(\$lat2)) + cos(deg2rad(\$lat1)) * cos(deg2rad(\$lat2)) * cos(deg2rad(\$theta)); \$dist = acos(\$dist); \$dist = rad2deg(\$dist); \$miles = \$dist * 60 * 1.1515; \$unit = strtoupper(\$unit); if (\$unit == "K") { return (\$miles * 1.609344); } else if (\$unit == "N") { return (\$miles * 0.8684); } else { return \$miles; } } public function search() { \$array_truefalse = []; // Bobot pilihan kriteria \$data = \$this->input->post('harga_hotel'); \$harga = array_map('floatval', explode(',', \$data)); \$array_truefalse['harga'] = \$harga; \$hasil1 = 1 - \$harga[1]; \$data2 = \$this->input->post('jarak_hotel'); \$jarak = array_map('floatval', explode(',', \$data2)); \$array_truefalse['jarak'] = \$jarak; \$hasil2 = 1 - \$jarak[1]; }</pre>

```

$data3 = $this->input->post('bintang_hotel');
$bintang = array_map('floatval', explode(',', $data3));
$array_truefalse['bintang'] = $bintang;
$hasil3 = 1 - $bintang[1];

$data4 = $this->input->post('jumlah_tipe_kamar');
$tipekamar = array_map('floatval', explode(',', $data4));
$array_truefalse['tipe_kamar'] = $tipekamar;
$hasil4 = 1 - $tipekamar[1];

$data_alter = $this->Malternatif->getAll()->result();

$lat_user = $this->input->post('lat_user');
$long_user = $this->input->post('long_user');
// print($lat_user);

// print($long_user);
//normallisasi
$array_alter = [];
foreach ($data_alter as $a) {
    $lat_hotel = $a->latitude;
    $long_hotel = $a->longitude;

    //                               $response                               =
json_decode(file_get_contents("http://www.mapquestapi.com/direct
ions/v2/route?key=nkMiH0KEbWDCelAqhHlxrGbWYC05wohX&from=$lat_use
r,$long_user&to=$lat_hotel,$long_hotel"));
    // $jarak = $response->route->distance;

    $jarak_m = $this->distance($lat_user, $long_user,
$lat_hotel, $long_hotel, 'K');

    // echo "jarak api = " . $jarak . " dan jarak manual
= " . $jarak_m;
    // break;

    $per = 0;
    $per2 = 0;

```

```

$per3 = 0;
$per4 = 0;
foreach ($data_alter as $b) {
    $b_lat_hotel = $b->latitude;
    $b_long_hotel = $b->longitude;

    //                                $b_response                                =
    json_decode(file_get_contents("http://www.mapquestapi.com/direct
ions/v2/route?key=nkMiH0KEbWDCelAqhHlxrGbWYC05wohX&from=$lat_use
r,$long_user&to=$b_lat_hotel,$b_long_hotel"));
    // $b_jarak = $b_response->route->distance;
    $b_jarak_m      =      $this->distance($lat_user,
$long_user, $b_lat_hotel, $b_long_hotel, 'K');
    $per += pow($b->harga, 2);
    $per2 += pow($b_jarak_m, 2);
    $per3 += pow($b->bintang, 2);
    $per4 += pow($b->tipe_kamar, 2);
}
$per_final = pow($per, 0.5);
$per_final2 = pow($per2, 0.5);
$per_final3 = pow($per3, 0.5);
$per_final4 = pow($per4, 0.5);

$final = $a->harga / $per_final;
$array_alter['harga'][] = $final;

$final2 = $jarak_m / $per_final2;
$array_alter['jarak'][] = $final2;

$final3 = $a->bintang / $per_final3;
$array_alter['bintang'][] = $final3;

$final4 = $a->tipe_kamar / $per_final4;
$array_alter['tipe_kamar'][] = $final4;

$array_alter['nama'][] = $a->nama_hotel;
$array_alter['lokasi'][] = $a->lokasi_hotel;
}

```



```

// pis (cost= min, benefit= max)
foreach ($array_alter as $key => $s) {
    $key_pis = 'pis_' . $key;
    $key_nis = 'nis_' . $key;
    if ($key == 'harga' || $key == 'jarak') {
        $array_alter[$key_pis][] = min($s);
        $array_alter[$key_nis][] = max($s);
        // print("nilai " . $key . min($s));
        // print("nilai minnya " . min($s));
    } else if ($key == 'bintang' || $key == 'tipe_kamar')
{
        $array_alter[$key_pis][] = max($s);
        $array_alter[$key_nis][] = min($s);
        // print("nilai " . $key . max($s));
        // print("nilai maxnya " . max($s));
    }
}

//ukuran pemisahan menggunakan jarak Euclidean dimensi-N
//D+
//alpa+
for ($z = 0; $z < count($data_alter); $z++) {
    $h1 = 0;
    foreach ($array_truefalse as $key => $w) {
        $h1 += (pow($w[0], 2) *
pow(($array_alter[$key][$z] - $array_alter['pis_' . $key][0]),
2));
    }
    //TRUE / alpha+
    // print("hasil:" . pow($h1, 0.5));
    $array_alter['trueAlpha'][] = pow($h1, 0.5);
    $array_alter['alphaPlus'][] = 1 - pow($h1, 0.5);
    // print("hasil:" . $alphaPlus);
}

for ($z = 0; $z < count($data_alter); $z++) {
    $h1 = 0;

```

```

        foreach ($array_truefalse as $key => $w) {
            $h1 += (pow($w[1], 2) *
pow(($array_alter[$key][$z] - $array_alter['pis_' . $key][0]),
2));
        }
        // print("hasil:" . pow($h1, 0.5));
        $array_alter['betaPlus'][] = pow($h1, 0.5);
        $hasilFals = 1 - pow($h1, 0.5);
        $array_alter['hasilDP'][] = 1 - pow($h1, 0.5);
        //print("hasil:" . $hasilFals);
    }

//D-
//D-FALSE
for ($z = 0; $z < count($data_alter); $z++) {
    $h1 = 0;
    foreach ($array_truefalse as $key => $w) {
        $h1 += (pow($w[0], 2) *
pow(($array_alter[$key][$z] - $array_alter['nis_' . $key][0]),
2));
    }
    $array_alter['dTrue'][] = pow($h1, 0.5);
    //print("hasil:" . pow($h1, 0.5));
}

for ($z = 0; $z < count($data_alter); $z++) {
    $h1 = 0;
    foreach ($array_truefalse as $key => $w) {
        $h1 += (pow($w[1], 2) *
pow(($array_alter[$key][$z] - $array_alter['nis_' . $key][0]),
2));
    }
    $array_alter['dFalse'][] = pow($h1, 0.5);
    //print("hasil:" . pow($h1, 0.5));
    $hasilFals = 1 - pow($h1, 0.5);
    $array_alter['hasilDM'][] = 1 - pow($h1, 0.5);
    // print("hasil:" . $hasilFals);
}

```

```

//max(alpha-, alpha $\ominus$ )
for ($z = 0; $z < count($array_alter['dTrue']); $z++) {
    // $alpha[] = max($alphaPlus[$z], $dFalse[$z]);
    //print(max($dTrue[$z], $hasilDP[$z]));
    $array_alter['alpha'][] =
max($array_alter['dTrue'][$z], $array_alter['hasilDP'][$z]);
}

for ($z = 0; $z < count($array_alter['hasilDM']); $z++)
{
    $array_alter['beta'][] =
max($array_alter['hasilDM'][$z], $array_alter['dFalse'][$z]);
    // print(max($alphaPlus[$z], $dFalse[$z]));
}

for ($z = 0; $z < count($array_alter['alpha']); $z++) {

    $array_alter['yi'][] = 1 - $array_alter['beta'][$z] -
$array_alter['alpha'][$z];
    // print(max($alphaPlus[$z], $dFalse[$z]));
}

$data5 = $this->input->post('confirmed');
$data6 = $this->input->post('uncertain');
// print($data5);
// print($data6);
for ($z = 0; $z < count($array_alter['yi']); $z++) {
    $array_alter['confirmed'][] = $data5 *
($array_alter['alpha'][$z] - $array_alter['beta'][$z]) - ($data6
* $array_alter['yi'][$z]);
}

for ($z = 0; $z < count($array_alter['yi']); $z++) {
    // $array_alter['unconfirm'][] = $data5 *
(($array_alter['alpha'][$z] - $array_alter['beta'][$z]) - (1 *
$array_alter['yi'][$z]));
    $array_alter['un_result'][] = $data5 *
($array_alter['alpha'][$z] - $array_alter['beta'][$z]) - (1 *
$array_alter['yi'][$z]);
}

```

```

        $array_alter['un_nama'][] =
$array_alter['nama'][$z];
        $array_alter['un_map'][] =
$array_alter['lokasi'][$z];
        //          print($array_alter['nama'][] =
$array_alter['nama'][$z]);
    }

    for ($i = 0; $i < count($data_alter); $i++) {
        for ($z = $i; $z < count($data_alter); $z++) {
            if ($array_alter['un_result'][$i] <
$array_alter['un_result'][$z]) {
                $temp1 = $array_alter['un_result'][$i];
                $array_alter['un_result'][$i] =
$array_alter['un_result'][$z];
                $array_alter['un_result'][$z] = $temp1;

                $temp2 = $array_alter['un_nama'][$i];
                $array_alter['un_nama'][$i] =
$array_alter['un_nama'][$z];
                $array_alter['un_nama'][$z] = $temp2;

                $temp3 = $array_alter['un_map'][$i];
                $array_alter['un_map'][$i] =
$array_alter['un_map'][$z];
                $array_alter['un_map'][$z] = $temp3;
            }

            // else if ($array_alter['un_result'][$i] ==
$array_alter['un_result'][$z]) {
                //          $array_alter['ok'][$z] = 1;
                // }
            }
        }

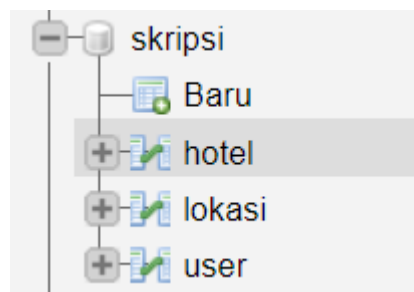
        // for ($z = 0; $z < count($array_alter['un_result']);
$z++) {
            //          if()

```

```
// }  
  
//print_r($unconfirmed);  
  
// // Rank Hotel  
$this->load->view('content/hasil_perhitungan',  
$array_alter);  
  
//          $this->load->view('content/hasil_perhitungan',  
$unconfirmed);  
}
```

5.2 Implementasi Database

Pada sub bab implementasi database sistem ini menggunakan database dari mysql. Dalam pembuatan sistem diperlukan database untuk menyimpan data yang telah ditambahkan, diubah atau ditampilkan oleh sistem. Berikut ini database system yang telah dibuat :




Gambar 5. 1 Database

Pada sistem pendukung keputusan hotel pada penelitian ini mempunyai nama database skripsi dan berisi 3 tabel diantaranya tabel hotel, tabel lokasi dan tabel user sesuai dengan yang ada pada gambar 5.1.

5.2.1 Tabel hotel



Tabel hotel ini digunakan untuk menyimpan, mengubah dan hapus data hotel.

#	Nama	Jenis	Penyortiran
<input type="checkbox"/> 1	idhotel 	int(11)	
<input type="checkbox"/> 2	nama_hotel	varchar(255)	utf8mb4_general_ci
<input type="checkbox"/> 3	harga	float	
<input type="checkbox"/> 4	latitude	float	
<input type="checkbox"/> 5	longitude	float	
<input type="checkbox"/> 6	bintang	float	
<input type="checkbox"/> 7	tipe_kamar	float	
<input type="checkbox"/> 8	foto_hotel	varchar(255)	utf8mb4_general_ci

Gambar 5. 2 Tabel hotel

5.2.2 Tabel lokasi


Tabel lokasi ini digunakan untuk menyimpan data link map hotel, dimana tabel ini memiliki *foreign key* yaitu suatu tabel yang digunakan untuk menciptakan hubungan (relasi) antara dua tabel (tabel hotel dan tabel lokasi).

#	Nama	Jenis	Penyortiran
<input type="checkbox"/> 1	idLokasi 	int(11)	
<input type="checkbox"/> 2	fkHotel 	int(11)	
<input type="checkbox"/> 3	lokasi_hotel	varchar(255)	utf8mb4_general_ci

Gambar 5. 3 Tabel lokasi

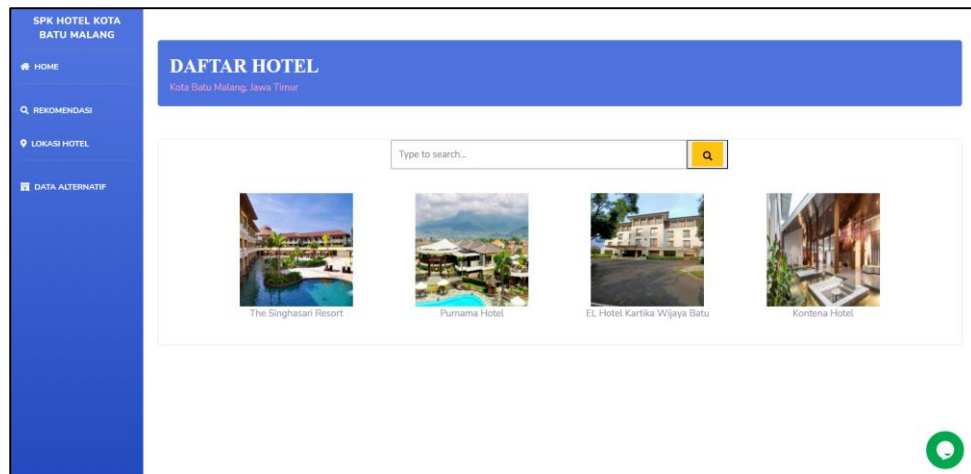
5.2.3 Tabel user

Tabel user ini digunakan untuk menyimpan data login pada sistem dan yang nantinya berfungsi pengecekan level apakah pengguna tersebut wisatawan atau admin.

#	Nama	Jenis	Penyortiran
<input type="checkbox"/> 1	iduser 	int(10)	
<input type="checkbox"/> 2	username	varchar(255)	utf8mb4_general_ci
<input type="checkbox"/> 3	email	varchar(255)	utf8mb4_general_ci
<input type="checkbox"/> 4	password	varchar(255)	utf8mb4_general_ci
<input type="checkbox"/> 5	level	int(11)	

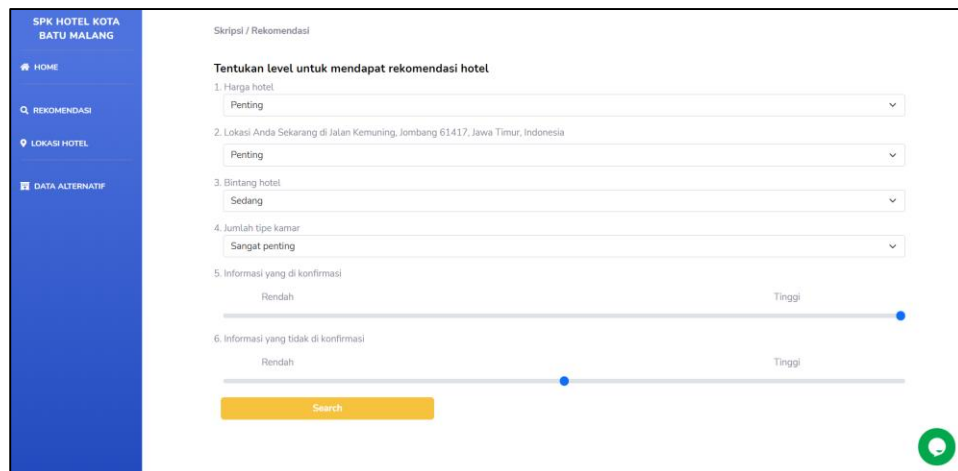
Gambar 5. 4 Tabel user

5.3 Implementasi Sistem



Gambar 5. 5 Tampilan halaman home

Gambar 5.5 tampilan halaman home yaitu tampilan pertama ketika user atau admin melakukan akses sistem, pada halaman ini digunakan untuk melihat hotel dan mencari hotel yang tersedia.



Gambar 5. 6 Tampilan halaman rekomendasi

Gambar 5.6 tampilan halaman rekomendasi yaitu tampilan form pilihan level setiap kriteria, sistem akan otomatis mendeteksi lokasi user pada saat itu. Halaman ini digunakan untuk mendapatkan hasil rekomendasi hotel, misal user menentukan level pada setiap kriteria sesuai pada gambar 5.6 maka ketika ditekan tombol search akan menghasilkan rekomendasi hotel seperti gambar 5.7 .

The screenshot shows a web interface for 'SPK HOTEL KOTA BATU MALANG'. The main content area is titled 'REKOMENDASI HOTEL' and lists four hotel recommendations. Below this, it shows the 'Proses perhitungan VS-TOPSIS' with a table of normalized values for three hotels: Golden Tulip Holland, Aston Inn, and Batu Wonderland Resort. The table includes columns for 'Data Alternatif', 'Harga', 'Jarak', 'Bintang', and 'Tipe'.

No	Nama Hotel	Nilai	Map
1	Golden Tulip Holland	0.47004840733137	
2	Aston Inn	0.22677301674966	
3	Batu Wonderland Resort	-0.3639720054164	
4	Hotel Asida	-0.50239406963776	

Data Alternatif	Harga	Jarak	Bintang	Tipe
Golden Tulip Holland	0.72461024072687	0.51165343214339	0.700140042014	0.88345220859877
Aston Inn	0.39455106615016	0.50206751688887	0.5601120336112	0.44172610429939
Batu Wonderland Resort	0.40054822754784	0.50192732277774	0.4200840252084	0.11043152607485

Gambar 5. 7 hasil rekomendasi

Gambar 5.7 berisi tampilan tabel-tabel dari hasil form pilihan level dan data alternatif yang dihitung dengan metode *VS-TOPSIS*. Pada tabel pertama yaitu hasil rekomendasi yang didapat. Untuk tabel-tabel berikutnya adalah proses dari perhitungan *VS-TOPSIS*. Proses perhitungan *VS-TOPSIS* ini hanya dapat dilihat oleh admin. Merumuskan matriks keputusan yang dinormalisasi R dapat dilihat pada tabel 2 gambar 5.7 .

The screenshot shows the 'Tabel PIS(Solusi Ideal Positif)' and 'Tabel NIS(Solusi Ideal Negatif)' for the VS-TOPSIS calculation. It also includes tables for calculating separation measures using Euclidean distance in dimension-N, labeled as D+ and D-.

Tabel	Harga	Jarak	Bintang	Tipe
PIS+	0.39455106615016	0.48395034508729	0.700140042014	0.88345220859877

Tabel	Harga	Jarak	Bintang	Tipe
NIS+	0.72461024072687	0.51165343214339	0.1400280084028	0.11043152607485

Menghitung ukuran pemisahan menggunakan jarak Euclidean dimensi-N

D+

Alternatif	TRUE / alpha+	1 - FALSE / 1 - beta+	FALSE / beta+
Golden Tulip Holland	0.24841480906342	0.26497579633432	0.73502420366568
Aston Inn	0.38218084854174	0.38661349162517	0.61338650837483
Batu Wonderland Resort	0.67197296296476	0.6819860027082	0.3180139972918

D-

Alternatif	TRUE / alpha-	1 - FALSE / 1 - beta-	FALSE / beta-
Golden Tulip Holland	0.71426128092391	0.75119027182591	0.24880972817409
Aston Inn	0.42982128387766	0.47290317302073	0.52709682697927
Batu Wonderland Resort	0.28059340696298	0.31687246279686	0.68312753720314

Gambar 5. 8 Perhitungan VS-TOPSIS

Gambar 5.8 perhitungan dari proses *VS-TOPSIS*, pada tabel 1 dan 2 merupakan Identifikasi PIS ($A^+ = \{v_1^+, v_2^+, \dots, v_n^+\}$) dan NIS ($A^- = \{v_1^-, v_2^-, \dots, v_n^-\}$). Kemudian tabel 3 dan 4 merupakan tabel menghitung ukuran pemisahan menggunakan jarak Euclidean dimensi-N.

Alternatif	Beta+	1 - alpha+
Golden Tulip Holland	0.73502420366568	0.24880972817409
Aston Inn	0.61338650837483	0.52709682697927
Batu Wonderland Resort	0.3180139972918	0.68312753720314

Alternatif	MAX[alpha-alpha θ]	MAX[1- beta-, 1 - beta θ]	γ_i
Golden Tulip Holland	0.73502420366568	0.75119027182591	-0.4862144754916
Aston Inn	0.61338650837483	0.52709682697927	-0.1404833353541
Batu Wonderland Resort	0.3180139972918	0.68312753720314	-0.0011415344949445

Alternatif	Astj	Jika nilai $\theta = 1$
Golden Tulip Holland	0.22694118958557	0.47004840733137
Aston Inn	0.15653134907261	0.22677301674966
Batu Wonderland Resort	-0.36454277266387	-0.3639720054164

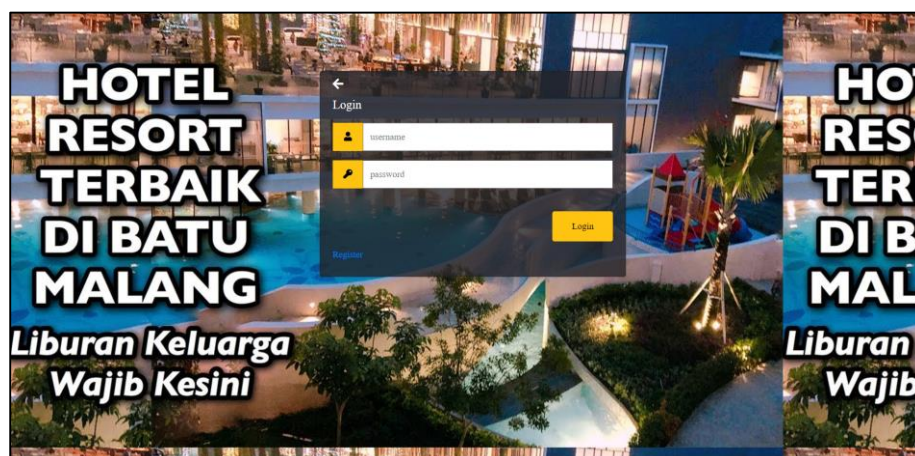
Gambar 5. 9 Lanjutan VS-TOPSIS

Gambar 5.9 lanjutan dari proses VS-TOPSIS, pada tabel 1 merupakan convert nilai dari D+ dan D-. Kemudian 2 adalah aggregate dan terakhir adalah kalkulasi M.

No	Nama Hotel	Lokasi
1	The Singhasari Resort	
2	Purnama Hotel	
3	EL Hotel Kartika Wijaya Batu	
4	Kontena Hotel	

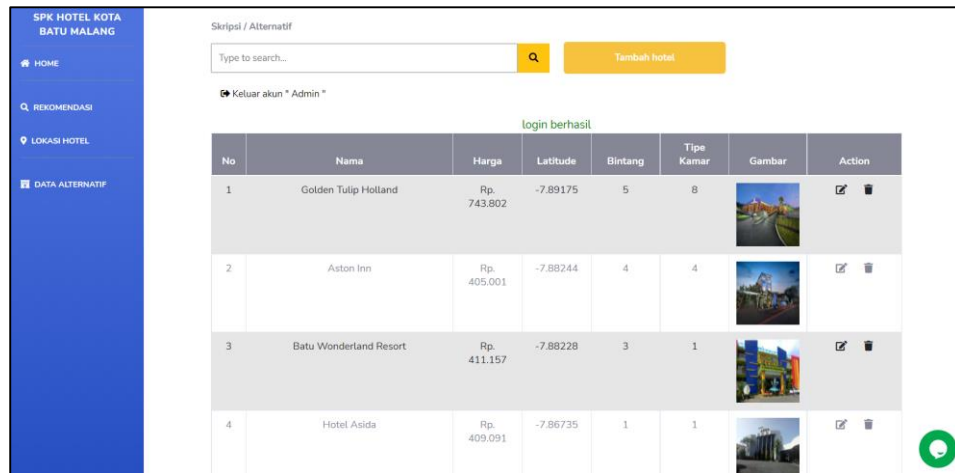
Gambar 5. 10 Tampilan halaman lokasi hotel

Gambar 5.10 tampilan halaman lokasi sistem.



Gambar 5. 11 Tampilan halaman login

Gambar 5.11 tampilan halaman login digunakan untuk masuk ke dalam sistem agar dapat melakukan akses pada fitur data alternatif.

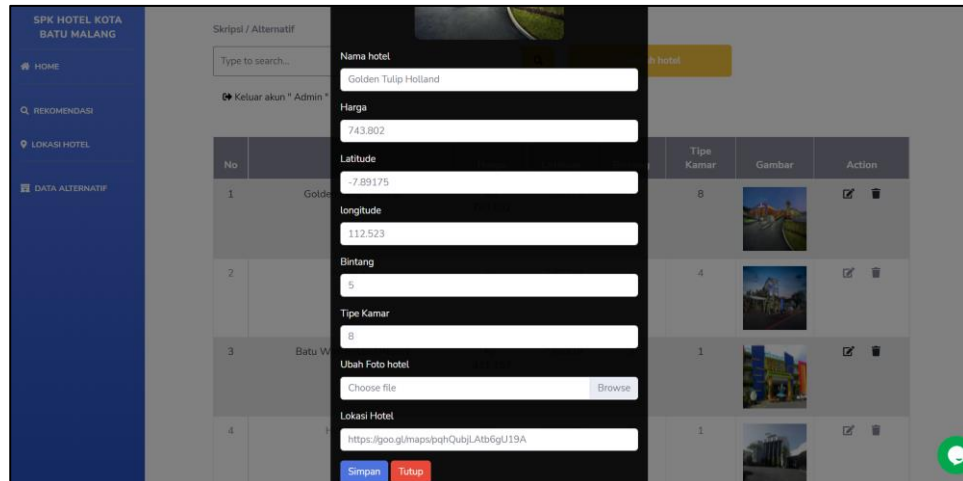


Gambar 5. 12 Tampilan halaman data alternatif

Gambar 5.12 tampilan halaman data alternatif digunakan untuk melakukan CRUD data hotel.

Gambar 5. 13 Tampilan halaman tambah data

Gambar 5.13 tampilan halaman untuk menambah data hotel.



Gambar 5. 14 Tampilan form ubah data

Gambar 5.14 tampilan modal form untuk mengubah data hotel.

5.4 Pengujian

Setelah melakukan implementasi database dan sistem, selanjutnya adalah melakukan tahap uji coba. Pengujian sistem ini menggunakan *black box testing*. *Black box testing* adalah teknik pengujian yang berfokus pada keluaran hasil dari respon masukan atau secara sederhana, *black box* proses menjalankan aplikasi untuk mengetahui apakah ada *error* atau fungsi yang tidak berjalan sesuai dengan harapan. Pengujian ini dimaksudkan untuk mengetahui apakah hasil keluaran konten dari sistem berjalan sesuai dengan fungsinya dengan data yang valid. Sedangkan pengujian metode digunakan untuk mengetahui kesesuaian nilai perhitungan pada sistem dengan landasan teori yang ada.

5.4.1 Pengujian Register

Pada pengujian register ini memasukkan username, email, dan password untuk melakukan pendaftaran akun pada tabel 5.5 .

Tabel 5. 5 Pengujian register

Deskripsi	
Nama Pengujian	Register
Tujuan	Mendaftarkan akun agar bisa melakukan login
Tampilan Awal	Halaman register sistem
Skenario Utama	
Aksi Penguji	Reaksi Sistem
- Mengisi username, email, dan Password lalu mengklik 'Login'	- Menampilkan peringatan daftar akun berhasil
Terjadi	- Keluar peringatan ketika disubmit tetapi form kosong - Keluar peringatan jika ada salah satu kolom kosong - Keluar peringatan dapat masuk system
Kondisi Yang Diharapkan	Menampilkan halaman form login
Keterangan	Sesuai

5.4.2 Pengujian Login

Pada pengujian login ini memasukkan username dan password untuk masuk kedalam sistem untuk mendapat hak akses fitur alternatif pada tabel 5.6.

Tabel 5. 6 Pengujian Login

Deskripsi	
Nama Pengujian	Login
Tujuan	Validasi data untuk masuk system
Tampilan Awal	Halaman awal system
Skenario Utama	
Aksi Penguji	Reaksi Sistem
Mengisi nama dan Password lalu mengklik 'Login'	Menampilkan peringatan login berhasil kedalam sistem
Terjadi	<ul style="list-style-type: none"> - Keluar peringatan ketika disubmit tetapi form kosong - Keluar peringatan jika ada salah satu kolom kosong - Keluar peringatan dapat masuk system
Kondisi Yang Diharapkan	Menampilkan halaman data alternatif
Keterangan	Sesuai

5.4.3 Pengujian Data Alternatif

Pada pengujian data alternatif ini dapat memasukkan data, melakukan edit, dan menghapus data hotel pada tabel 5.7 .

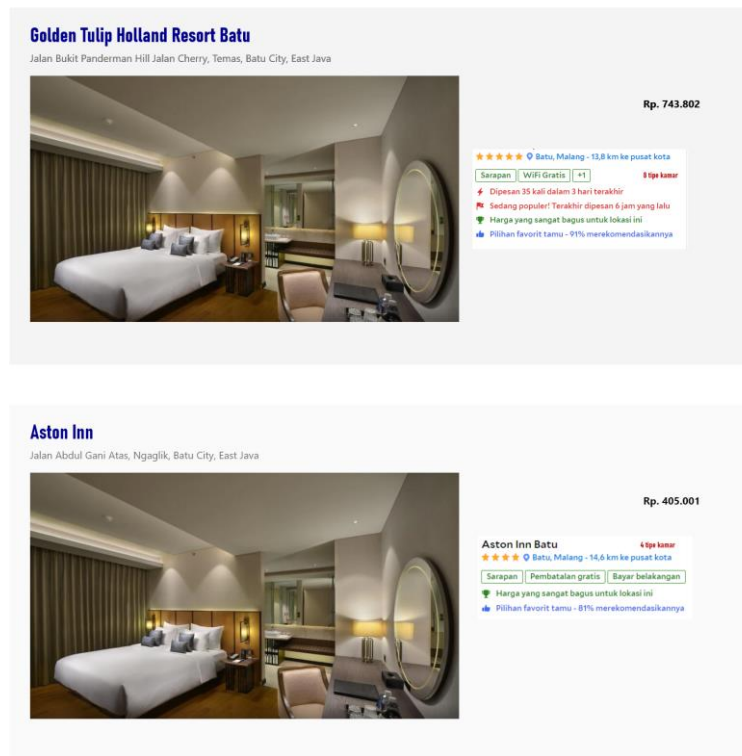
Tabel 5. 7 Pengujian Alternatif

Deskripsi	
Nama Pengujian	Data Alternatif
Tujuan	Menambah, menampilkan, mengedit, dan menghapus data hotel
Kondisi Awal	Halaman Data Alternatif
Skenario Utama	
Aksi Penguji	Reaksi Sistem
<ul style="list-style-type: none"> - Mengisi (nama, harga, jarak, bintang, tipe kamar, foto hotel, dan lokasi) - Menyimpan data hotel - Mengedit data hotel - Menghapus data hotel 	<ul style="list-style-type: none"> - Menambah data hotel - Menyimpan data hotel - Mengubah data hotel - Menghapus data hotel
Terjadi	Data tersimpan
Kondisi Yang Diharapkan	Menampilkan halaman data alternatif
Keterangan	Sesuai

5.4.4 Pengujian Metode SPK

Pada tahap ini merupakan tahap pengujian metode VS-TOPSIS yang digunakan untuk sistem pendukung keputusan hotel. Membuat scenario seperti gambar 5.15 ,berikut merupakan hasil pengujian yang dilakukan dalam sistem pendukung keputusan menggunakan perhitungan sistem pada data hotel dari bulan januari 2021 yang diambil dari gabungan data website Dinas Pariwisata Kota Batu dan Agoda sehingga mendapat 44 data hotel kemudian dihitung menggunakan langkah dari metode *VS-TOPSIS* pada excel dan sistem untuk mengetahui hasil

rekomendasi hotel. Pengujian dilakukan dari hasil salah satu anggota pegawai dinas pariwisata sehingga didapatkan hasil berikut :



Gambar 5. 15 Skenario

Tabel 5. 8 Hasil Kuesioner Dinas Pariwisata

Pegawai dinas pariwisata	
Timestamp	11/07/2021 17:59:34
Email anda	megatacik@gmail.com
Harga Hotel	Sangat Penting
Jarak user ke hotel	Sangat Penting
Bintang	Sedang
Jumlah tipe kamar	Sedang
Informasi yang di konfirmasi	1
Informasi yang tidak di konfirmasi	0,5

Sehingga didapatkan hasil rekomendasi hotel pada gambar 5.16:

Rekomendasi Hotel			
No	Nama Hotel	Nilai	Map
1	THE BATU Hotel & Villas	0.84978999250074	
2	Hotel Gradia 2	0.68746867813306	
3	Jambu Luwuk Convention Hall & Resort	0.66309902766522	
4	Hotel Nirwana	0.65541561547929	
5	Royal Orchid Garden Hotel	0.58103090895446	
6	Ciptaningati hotel	0.56955116314056	
7	Hotel Gradia 1	0.56288978495074	

Gambar 5. 16 Hasil Rekomendasi Dinas

5.4.5 Pengujian User

Pada pengujian user dilakukan presentasi secara online dengan menggunakan aplikasi Zoom Meeting bersama pengguna masyarakat umum dan memberikan kuesioner sebagai feedback dari sistem. Presentasi terhadap pengguna yaitu penulis menjalankan aplikasi sistem pendukung keputusan dan menjalankan setiap fitur yang ada dalam sistem pendukung keputusan dan memperhatikan kesesuaian hasil dari sistem. Dari presentasi yang dilakukan maka mendapatkan rangkuman hasil pengujian sebagai berikut:

1. Setiap fitur yang telah dijalankan sudah sesuai dengan fungsi dari fitur-fitur tersebut.
2. Pengguna mengatakan fitur rekomendasi ini dapat membantu dalam memberikan rekomendasi hotel.
3. Pengguna bisa menginputkan data alternatif hotel yang masih belum terdapat di sistem.