BAB IV
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi *(Implementation)*

Uji implementasi sistem dilakukan untuk mengukur tingkat efisiensi dan efektifitas penerapan sebuah aplikasi dan atau sistem bagi pengguna. Dilakukan menggunakan suatu metode pengukuran efisiensi dan efektifitas penerapan *system*.

4.1.1 Implementasi Algoritma *Rabin-Karp*

```
// Preprocessing) ---
function normalizeText($text) {
    $text = strtolower($text);              // ubah ke huruf kecil
    $text = preg_replace("/[^a-z]/", "", $text); // hapus semua selain a-z
    return $text;
}

// Parsing K-Gram ---
function rabinKarp($text, $pattern, $d = 256, $q = 101) {
    $n = strlen($text);
    $m = strlen($pattern);

    if ($m == 0 || $n == 0 || $m > $n) return false;

    $h = 1;
    for ($i = 0; $i < $m - 1; $i++) {
        $h = ($h * $d) % $q;
    }

    // Hashing
    $p = 0; // hash pattern
    $t = 0; // hash teks window pertama
    for ($i = 0; $i < $m; $i++) {
        $p = ($d * $p + ord($pattern[$i])) % $q;
        $t = ($d * $t + ord($text[$i])) % $q;
    }

    // Rolling Hash
    for ($i = 0; $i <= $n - $m; $i++) {
        if ($p == $t) {
            // Verifikasi karakter per karakter
            $j = 0;
            for (; $j < $m; $j++) {
                if ($text[$i + $j] !== $pattern[$j]) break;
```

```
            }
            if ($j == $m) {
                return true;
            }
        }
        if ($i < $n - $m) {
            $t = ($d * ($t - ord($text[$i]) * $h) + ord($text[$i + $m])) % $q;
            if ($t < 0) $t += $q;
        }
    }
    return false;
}

// --- Input dari client ---
$siswa_id = isset($_POST['siswa_id']) ? (int)$_POST['siswa_id'] : 0;
$soal_id  = isset($_POST['soal_id']) ? (int)$_POST['soal_id'] : 0;
$jawaban  = isset($_POST['jawaban']) ? $_POST['jawaban'] : '';

if ($siswa_id <= 0 || $soal_id <= 0 || trim($jawaban) === '') {
    echo json_encode(['status'=>'error','message'=>'Parameter tidak lengkap atau tidak valid']);
    exit;
}

// Cek apakah sudah pernah menjawab
$cek = mysqli_prepare($con, "SELECT id_jawaban FROM jawaban_siswa WHERE
siswa_id=? AND soal_id=? LIMIT 1");
mysqli_stmt_bind_param($cek, "ii", $siswa_id, $soal_id);
mysqli_stmt_execute($cek);
$resCek = mysqli_stmt_get_result($cek);
if (mysqli_num_rows($resCek) > 0) {
    echo json_encode(['status'=>'error','message'=>'Soal ini sudah pernah dijawab']);
    exit;
}

// Ambil jawaban benar
$q = mysqli_prepare($con, "SELECT jawaban_benar FROM bank_soal WHERE id_soal=?
LIMIT 1");
mysqli_stmt_bind_param($q, "i", $soal_id);
mysqli_stmt_execute($q);
$res = mysqli_stmt_get_result($q);
if (!$res || mysqli_num_rows($res) === 0) {
    echo json_encode(['status'=>'error','message'=>'Soal tidak ditemukan']);
    exit;
}
$row = mysqli_fetch_assoc($res);
$jawaban_benar_raw = $row['jawaban_benar'];
```

```
// --- Normalisasi untuk perbandingan ---
$jawaban_benar = normalizeText($jawaban_benar_raw);
$jawaban_siswa = normalizeText($jawaban);

// ---Pencocokan ---
$benar = rabinKarp($jawaban_siswa, $jawaban_benar) ? 1 : 0;

// --- Pengukuran Nilai Kemiripan ---
$persentase = 0.0;
similar_text($jawaban_siswa, $jawaban_benar, $persentase);
$persentase = round($persentase, 2);
```

Ringkasan Struktur Rabin-Karp pada implementasi di atas :

- *Preprocessing*: normalisasi teks *(lowercase* + hilangkan *non*-huruf a–z).
- *Parsing K-Gram*: pola (*pattern*) yang akan dicari adalah jawaban benar yang sudah dinormalisasi; teks sumber adalah jawaban siswa yang sudah dinormalisasi; algoritma menggeser *window* sepanjang teks untuk membandingkan setiap k-gram (k = panjang *pattern*).
- *Hashing* awal: hitung *hash pattern* dan *hash window* pertama teks.
- *Rolling hash*: perbarui *hash window* secara efisien saat *window* digeser satu posisi.
- Verifikasi saat *hash* cocok: jika hash sama, lakukan pemeriksaan karakter-per-karakter untuk menghindari *false positive*.
- *Output*: *boolean match* untuk benar/salah ($benar = 1/0) dan pengukuran kemiripan (persentase dengan *similar_text*).

Berikut adalah penjelasan lengkap dan detail dari setiap tahapan dalam kode Rabin-Karp yang Anda berikan. Fungsi/*fragmen* ini dipakai untuk mencocokkan jawaban siswa terhadap jawaban benar (*pattern*) dan mengukur kemiripannya.

1. *Preprocessing*

```
function normalizeText($text) {
    $text = strtolower($text);              // ubah ke huruf kecil
```

```
    $text = preg_replace("/[^a-z]/", "", $text); // hapus semua selain a-z
    return $text;
}
```

- Tujuan: menyamakan format sehingga perbandingan menjadi stabil terhadap perbedaan huruf besar/kecil dan tanda baca/spasi/angka.
- Hasil: string hanya berisi huruf a–z dalam huruf kecil, tanpa spasi atau tanda lain. Ini membuat operasi K-gram mengikuti urutan huruf murni.

2. Persiapan *Input* dan Validasi

```
$cek = mysqli_prepare($con, "SELECT id_jawaban FROM
jawaban_siswa WHERE siswa_id=? AND soal_id=? LIMIT 1");
mysqli_stmt_bind_param($cek, "ii", $siswa_id, $soal_id);
mysqli_stmt_execute($cek);
$resCek = mysqli_stmt_get_result($cek);
if (mysqli_num_rows($resCek) > 0) {
    echo json_encode(['status'=>'error','message'=>'Soal ini sudah pernah
dijawab']);
    exit;
}
```

- Memastikan parameter *valid* dan siswa belum menjawab soal yang sama (menghindari double submission).

3. Ambil *Pattern* (jawaban benar) dari *database*

```
$q = mysqli_prepare($con, "SELECT jawaban_benar FROM bank_soal
WHERE id_soal=? LIMIT 1");
mysqli_stmt_bind_param($q, "i", $soal_id);
mysqli_stmt_execute($q);
$res = mysqli_stmt_get_result($q);
if (!$res || mysqli_num_rows($res) === 0) {
    echo json_encode(['status'=>'error','message'=>'Soal tidak ditemukan']);
```

```
    exit;
}
$row = mysqli_fetch_assoc($res);
$jawaban_benar_raw = $row['jawaban_benar'];
```

- *Pattern* yang dicari adalah jawaban_benar yang sudah dinormalisasi.
- Teks yang dipindai (*text*) adalah jawaban_siswa yang sudah dinormalisasi.

4. *Parsing K-Gram*

Pada implementasi ini tidak ada fungsi eksplisit yang memecah teks menjadi *array k-gram* terlebih dahulu. Sebagai gantinya, *Parsing K-Gram* terjadi konseptual dalam *rabinKarp():*

- *pattern* (jawaban_benar) adalah k-gram yang ingin dicari, dengan panjang m = *strlen(pattern).*
- rabinKarp() kemudian menggeser sebuah window berukuran m sepanjang text (jawaban_siswa). Setiap posisi window mewakili satu k-gram teks yang sedang diuji.

Jadi: *parsing k-gram = memilih ukuran k (*yaitu panjang *pattern) dan* menggeser *window ukuran k* di atas teks untuk memeriksa setiap k-gram. Dalam kode, `loop for ($i = 0; $i <= $n - $m; $i++) { ... }` adalah realisasi *parsing k-gram* secara *sliding-window.*

5. Fungsi *rabinKarp()*

```
function rabinKarp($text, $pattern, $d = 256, $q = 101) {
    $n = strlen($text);
    $m = strlen($pattern);
    if ($m == 0 || $n == 0 || $m > $n) return false;
```

```
$h = 1;
for ($i = 0; $i < $m - 1; $i++) {
    $h = ($h * $d) % $q;
}
```

Variabel:

- *n = strlen($text), m = strlen($pattern).*
- d = basis (biasanya jumlah kemungkinan karakter; di sini 256), q = modul prima untuk menghindari *overflow* dan mengurangi tabrakan *hash.*

Hitung h = (d^(m-1)) % q dipakai untuk mengeluarkan kontribusi karakter paling kiri saat rolling hash.

Hashing awal:

- p = *hash pattern* (diakumulasi menggunakan p = (d*p + ord(pattern[i])) % q).
- t = *hash window* pertama teks (diakumulasi serupa).

Rolling Hash (untuk setiap posisi window):

- Bandingkan p dan t. Jika sama → lakukan verifikasi karakter-per-karakter.
- Jika belum akhir window: perbarui t dengan rumus:
- t = (d*(t - ord(text[i]) * h) + ord(text[i+m])) % q;
- if (t < 0) t += q;

Rumus ini menghapus kontribusi karakter paling kiri, menggeser basis, dan menambahkan karakter baru di kanan — semua dilakukan dalam O(1).

Verifikasi karakter-per-karakter:

- Jika p == t, lakukan loop pengecekan for (; $j < $m; $j++) { if ($text[$i + $j] !== $pattern[$j]) break; }
- Hanya jika seluruh karakter cocok ($j == $m) fungsi mengembalikan *true* — yaitu ditemukan.

6. Keputusan benar/salah

```
$benar = rabinKarp($jawaban_siswa, $jawaban_benar) ? 1 : 0;
```

Logika sederhana:

- Jika Rabin-Karp menemukan setidaknya satu kemunculan *exact-match* dari jawaban_benar (setelah normalisasi) di dalam jawaban_siswa, maka $benar = 1.
- Jika tidak ada kemunculan, $benar = 0.
- Penilaian berbasis kemunculan *pattern*: ini bukan perbandingan keseluruhan *string* secara langsung; melainkan mencari apakah urutan huruf yang identik dengan jawaban benar ada sebagai suatu *substring* (k-gram) di jawaban siswa.
- Jika jawaban benar adalah kata dan jawaban siswa mengandung *substring* kata (misalnya. "...mengandungkata...”), maka dianggap benar meskipun ada karakter lain sebelum/ sesudahnya — karena *match substring* cukup untuk menetapkan $benar = 1.
- Jika jawaban siswa menulis jawaban benar namun diselingi karakter *non*-huruf sebelum normalisasi (misalnya angka/tanda), normalisasi menghapusnya—yang dapat membuat *match* tetap terjadi atau hilang bergantung posisi tanda.
- Algoritme ini sensitif pada urutan huruf: hanya substrings yang identik (setelah normalisasi) akan dianggap match.

Keamanan terhadap *false positive*:

- Rabin-Karp menggunakan *hash* untuk menemukan kandidat; karena hash dapat bertabrakan, verifikasi karakter-per-karakter diperlukan untuk memastikan kebenaran *match*. Kode Anda melakukan verifikasi ini — sehingga jika hash sama tetapi karakter berbeda, match ditolak.

Performa dan sifat penilaian:

- Jika *pattern* panjang m jauh lebih pendek dari *text n, algoritme* sangat efisien (O(n+m) rata-rata).
- Namun pendekatan ini hanya mendeteksi keberadaan substring yang identik — bukan kemiripan kontekstual atau parafrase.

7. Pengukuran nilai kemiripan (terpisah dari benar/salah)

```
$persentase = 0.0;
similar_text($jawaban_siswa, $jawaban_benar, $persentase);
$persentase = round($persentase, 2);
```

- *similar_text* menghitung persentase kemiripan dua string secara keseluruhan (bukan substring).

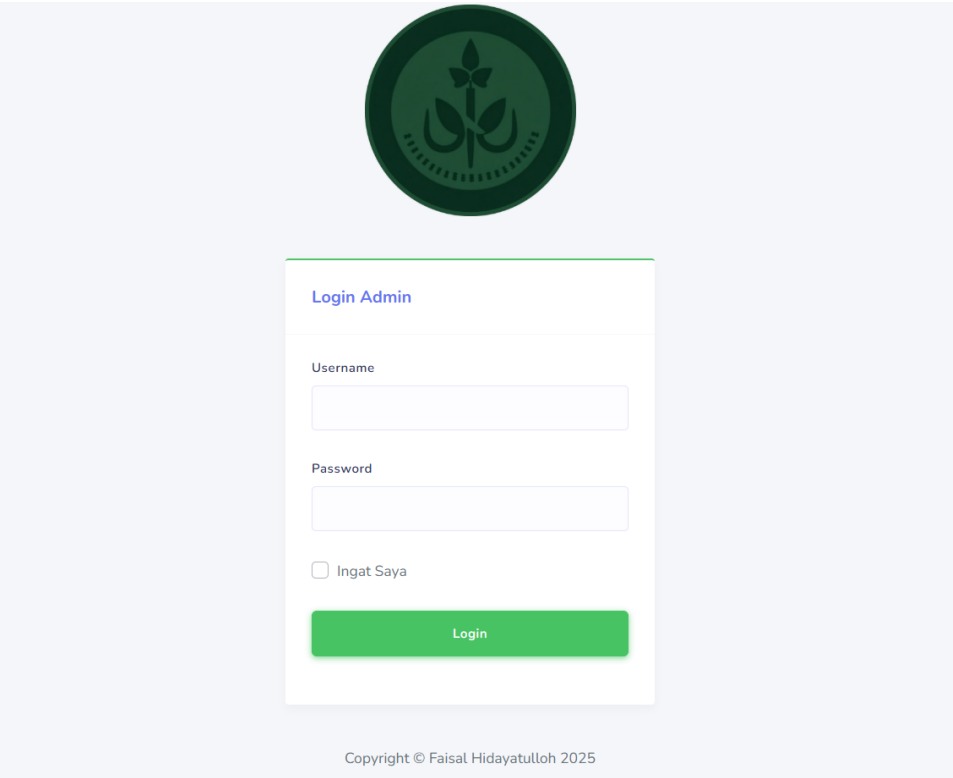
4.2 Implementasi Desain Antar Muka

Implementasi desain antarmuka adalah tahap penerapan rancangan UI/UX ke dalam bentuk tampilan aplikasi yang sebenarnya. Pada proses ini, setiap elemen seperti layout, warna, dan navigasi diwujudkan agar tampilan aplikasi dapat berfungsi dan digunakan dengan baik. Tahap ini memastikan desain yang telah direncanakan dapat tampil secara konsisten, mudah dipahami, dan siap untuk diuji pada langkah berikutnya.

4.2.1 Antar Muka *Web* Admin

1. Antarmuka Halaman *Login Admin*

   Adapun tampilan halaman *Login* Admin, dapat dilihat pada Gambar di Bawah Ini:

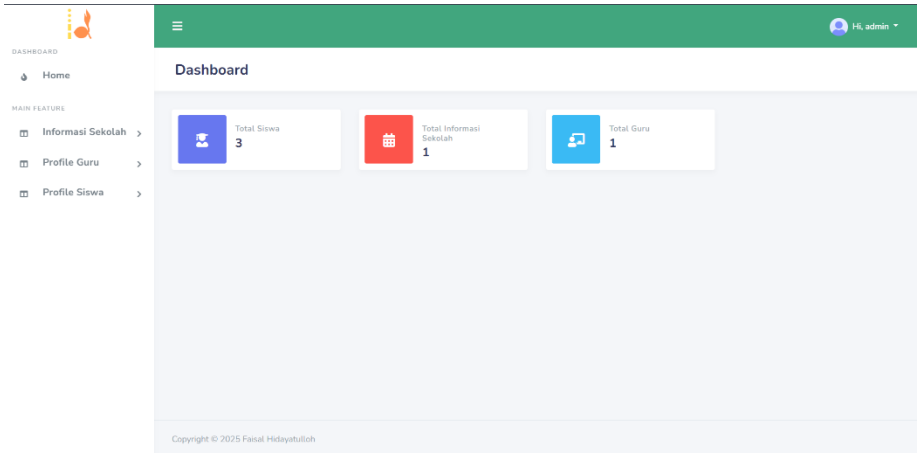


Gambar 4. 1 Antarmuka Halaman Login Admin

Halaman *Login Admin* digunakan untuk mengakses sistem. *Admin* diminta memasukkan *username* dan *password* pada kolom yang disediakan. Terdapat juga opsi "Ingat Saya" agar Admin tidak perlu Login ulang di kemudian hari. Setelah menekan tombol *Login*, sistem akan memverifikasi Data dan mengarahkan ke halaman *Dashboard* jika *username* dan *password* yang dimasukan sudah sesuai.

2. Antarmuka Halaman *Dasboard Admin*

Adapun tampilan halaman *Dashboard* Admin, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 2 Implementasi Antarmuka Halaman *Dasboard Admin*

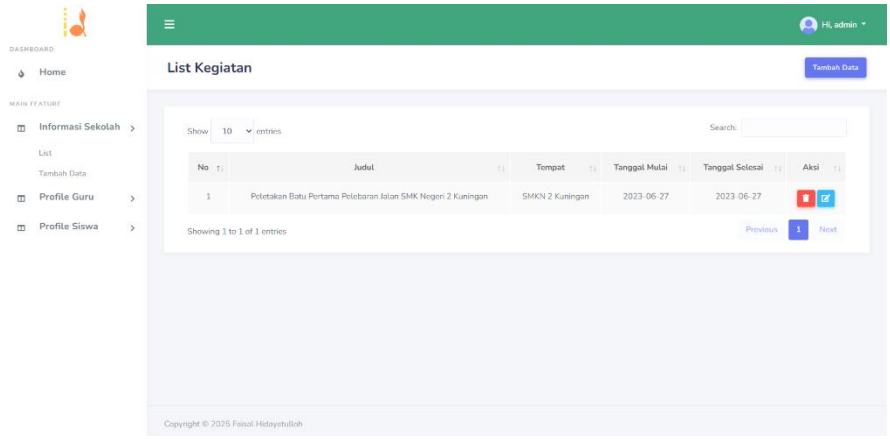
Setelah berhasil *Login*, Admin diarahkan ke halaman *Dashboard*. Pada *Dashboard*, guru dapat melihat Data ringkasan seperti:

1. Total Siswa.
2. Total Informasi Sekolah
3. Total Guru

Tampilan *Dashboard* dilengkapi menu navigasi di sisi kiri untuk mengelola Profile siswa, Profile Guru dan Informasi Sekolah. Antarmuka ini memudahkan Admin dalam mengontrol dan memonitor aktivitas sistem.

3. Antarmuka Halaman *List* Informasi Sekolah

Adapun tampilan halaman *List* Informasi Sekolah, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 3 Antarmuka Halaman List Informasi Sekolah

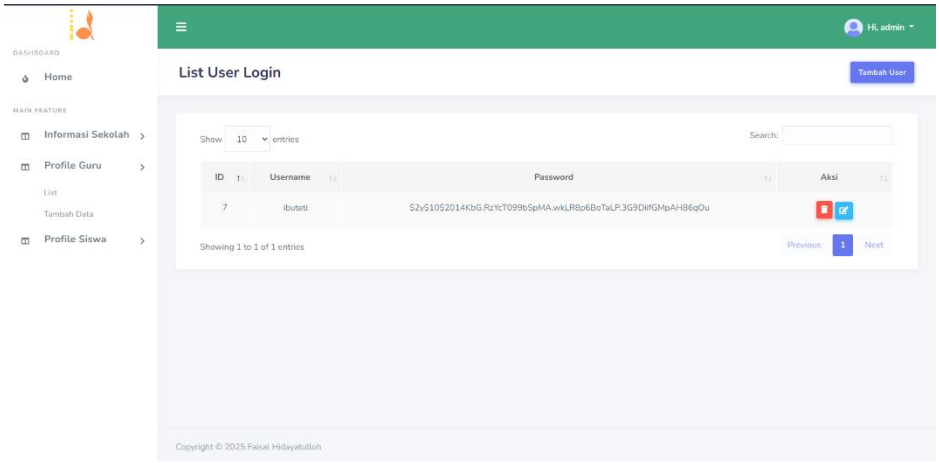
Halaman List Informasi Sekolah menampilkan Data seluruh Informasi Sekolah yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

1. No
2. Judul
3. Tempat
4. Tanggal Mulai
5. Tanggal Selesai

Terdapat fitur pencarian dan opsi jumlah Data yang ditampilkan per halaman. Selain itu, terdapat tombol edit dan hapus pada kolom aksi untuk memudahkan pengelolaan Data Informasi Sekolah.

4. Antarmuka Halaman Kelola Profile Guru

Adapun tampilan halaman Kelola Profile Guru, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 4 Antarmuka Halaman Kelola Profile Guru

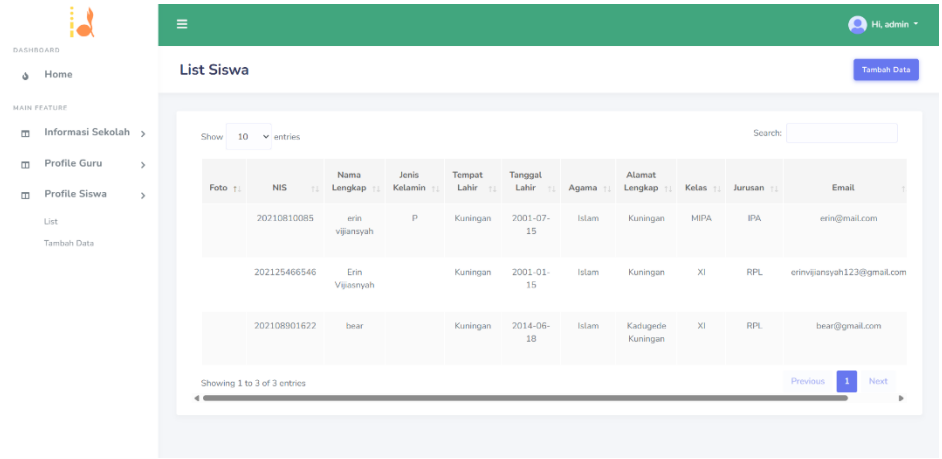
Halaman Kelola Profile Guru menampilkan Data seluruh Guru yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

1. *Username*
2. *Password*

Terdapat fitur pencarian dan *opsi* jumlah Data yang ditampilkan per halaman. Selain itu, terdapat tombol edit dan hapus pada kolom aksi untuk memudahkan pengelolaan Data pada Profile Guru.

5. Antarmuka Halaman Kelola Profile Siswa

Adapun Tampilan halaman Kelola Profile Siswa, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 5 Antarmuka Halaman Kelola Profile Siswa

Halaman Kelola Profile Siswa menampilkan Data seluruh Siswa yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

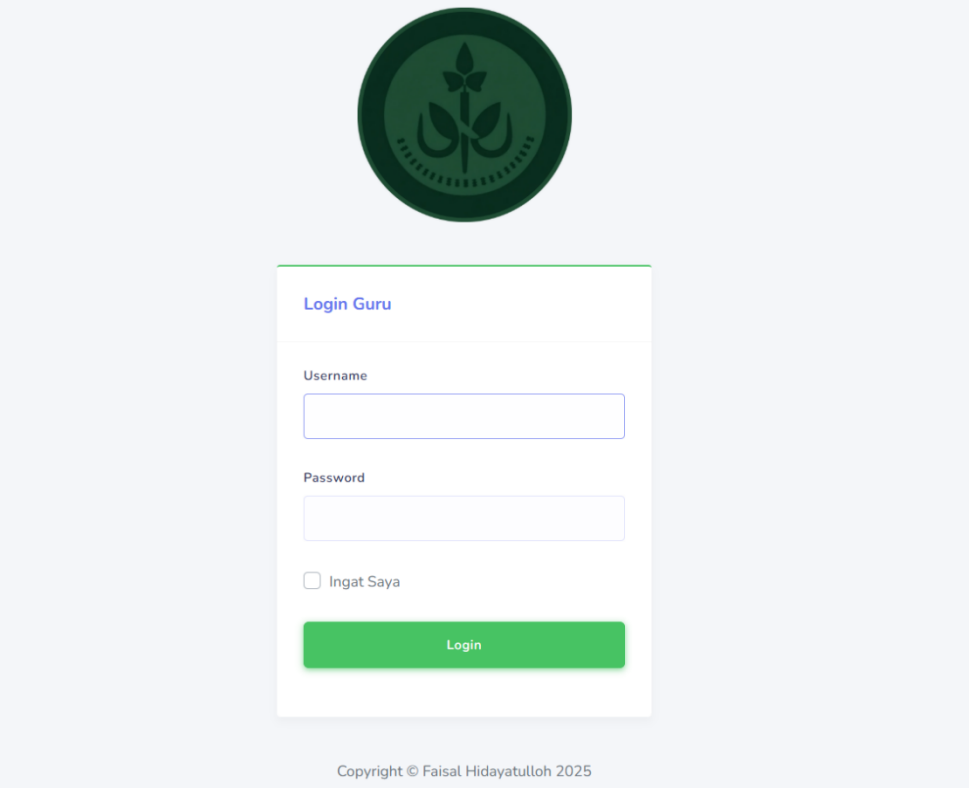
1. Foto
2. Nis
3. Nama Lengkap
4. Jenis Kelamin
5. Tempat Lahir
6. Tanggal Lhir
7. Agama
8. Alamat Lengkap
9. Kelas
10. Jurusan
11. E-Mail

Terdapat fitur pencarian dan opsi jumlah Data yang ditampilkan per halaman. Selain itu, terdapat tombol edit dan hapus pada kolom aksi untuk memudahkan pengelolaan Data pada Profile Siswa.

4.2.2 Antar Muka *Web* Guru

1. Antarmuka Halaman *Login* Guru

   Adapun Tampilan halaman *Login* Guru, dapat dilihat pada Gambar di Bawah Ini:

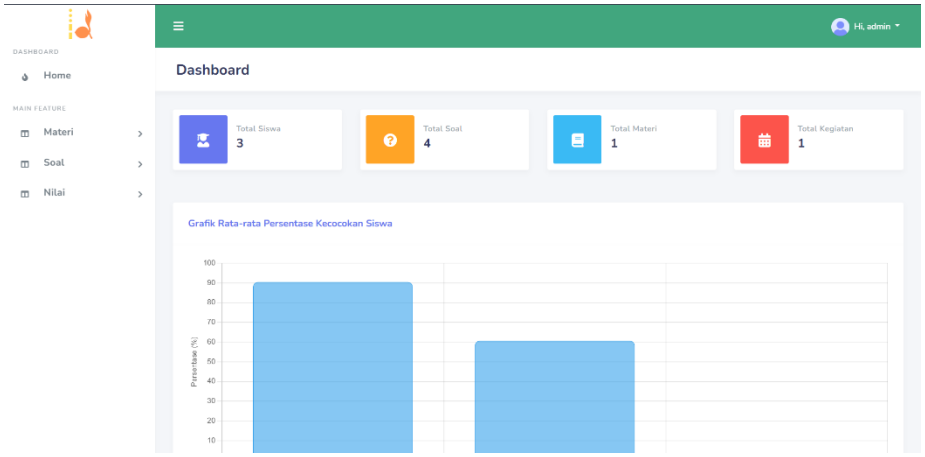


Gambar 4. 6 Antarmuka Halaman Login Guru

Halaman *Login* Guru digunakan untuk mengakses sistem. Guru diminta memasukkan *username* dan *password* pada kolom yang disediakan. Terdapat juga opsi "Ingat Saya" agar Guru tidak perlu *Login* ulang di kemudian hari. Setelah menekan tombol *Login*, sistem akan memverifikasi Data dan mengarahkan ke halaman *Dashboard* jika *username* dan *password* yang dimasukan sudah sesuai.

2. Antarmuka Halaman *Dashboard Guru*

Adapun *interface* halaman *Dashboard* Guru, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 7 Antarmuka Halaman *Dasboard* Guru

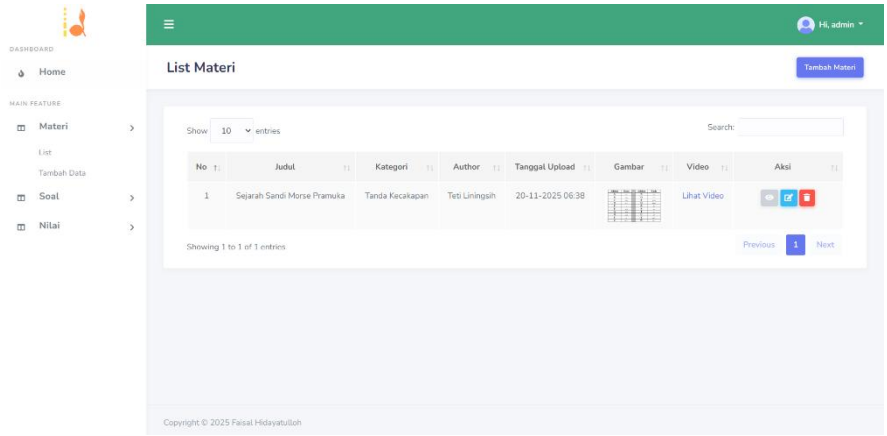
Setelah berhasil *Login*, Guru diarahkan ke halaman *Dashboard*. Pada *Dashboard*, guru dapat melihat Data ringkasan seperti:

1. Total Siswa
2. Total Soal
3. Total Materi
4. Total Informasi Sekolah
5. Grafik Nilai Siswa

Tampilan *Dashboard* dilengkapi menu navigasi di sisi kiri untuk mengelola Materi, Soal dan melihat detail nilai siswa. Antarmuka ini memudahkan Guru dalam mengontrol dan memonitor aktivitas sistem.

3. Antarmuka Halaman Kelola Materi

Adapun tampilan halaman Kelola Materi, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 8 Antarmuka Halaman Kelola Materi

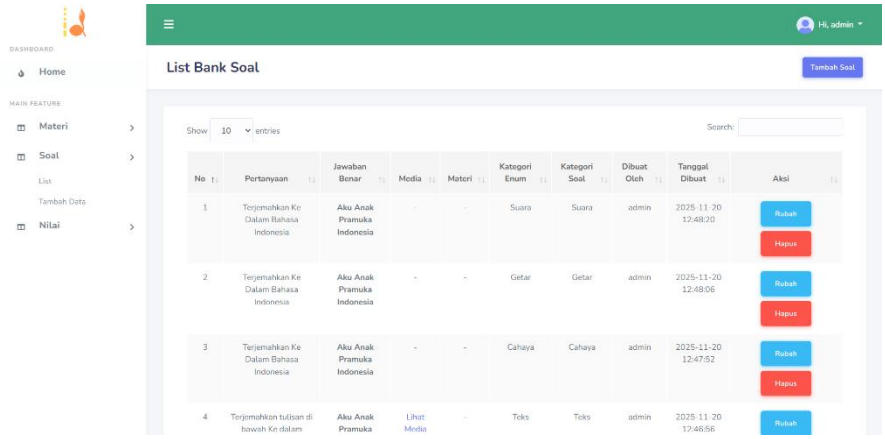
Halaman *List* Kelola Materi menampilkan Data seluruh Materi yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

1. No
2. Judul
3. Kategori
4. Author
5. Tanggal Upload
6. Gambar
7. Vidio

Terdapat fitur pencarian dan opsi jumlah Data yang ditampilkan per halaman. Selain itu terdapat tombol *detail, edit* dan hapus pada kolom aksi untuk memudahkan pengelolaan Data Informasi Sekolah.

4. Antarmuka Halaman Kelola Soal

Adapun tampilan halaman Kelola Soal, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 9 Antarmuka Halaman Kelola Soal

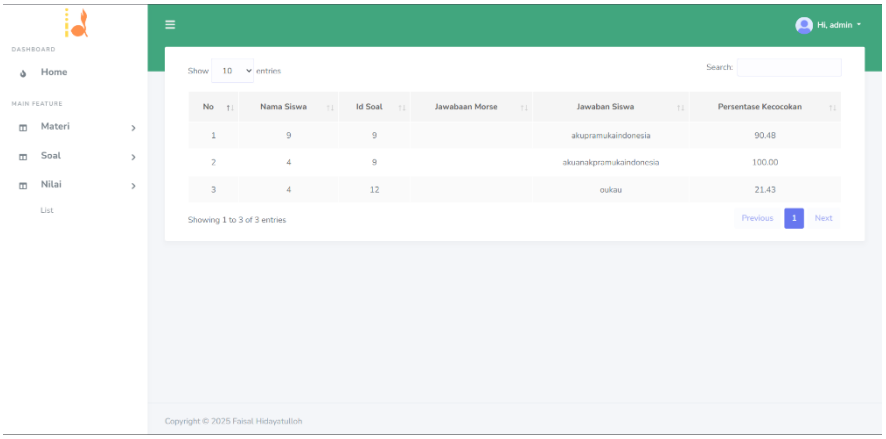
Halaman *List* Kelola Soal menampilkan Data seluruh Soal yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

1. No
2. Pertanyaan
3. Jawaban Benar
4. Media
5. Materi
6. Kategori Umum
7. Kategori Soal
8. Dibuat Oleh
9. Tanggal Dibuat

Terdapat fitur pencarian dan opsi jumlah Data yang ditampilkan per halaman. Selain itu terdapat tombol edit dan hapus pada kolom aksi untuk memudahkan pengelolaan Data Informasi Sekolah.

5. Antarmuka Halaman Kelola Soal

Adapun tampilan halaman Nilai, dapat dilihat pada Gambar di Bawah Ini:



Gambar 4. 10 Antarmuka Halaman Nilai

Halaman List Nilai menampilkan Data seluruh Nilai yang terdaftar dalam sistem. Data yang ditampilkan meliputi:

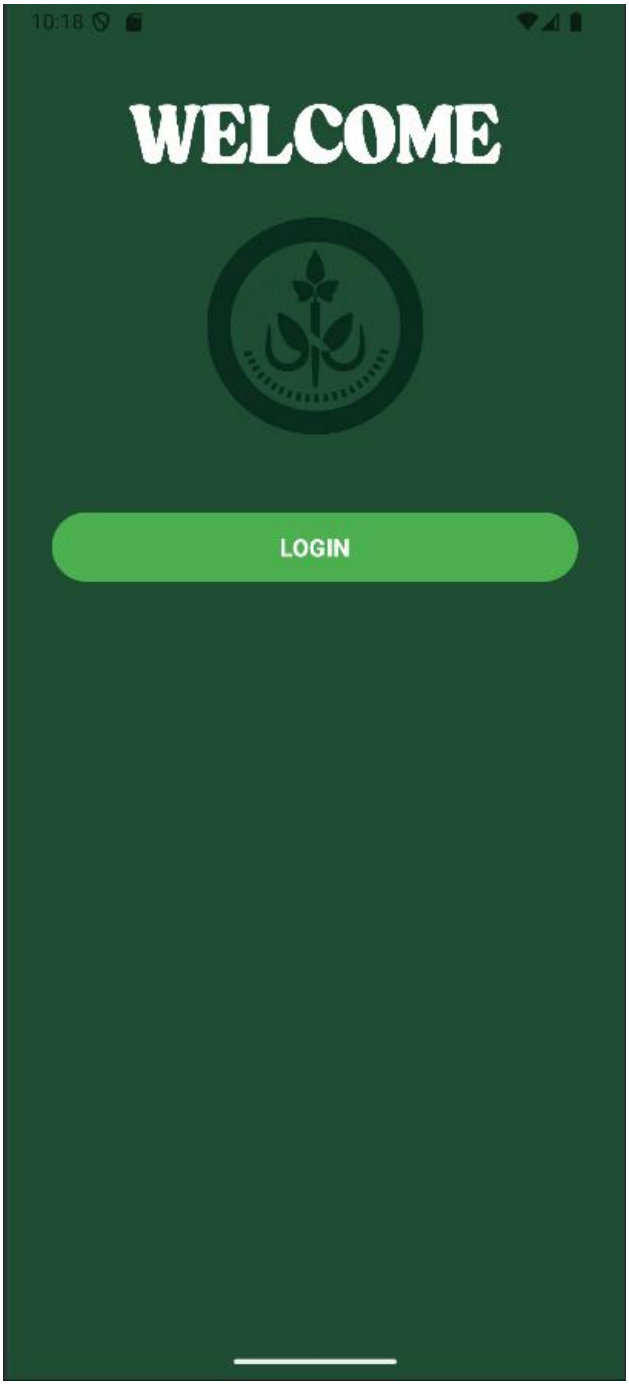
1. No
2. Nama Siswa
3. Id Soal
4. Jawaban Morse
5. Jawaban Siswa
6. Persentase Kecocokan

Terdapat fitur pencarian dan *opsi* jumlah Data yang ditampilkan per halaman agar memudahkan guru dalam melihat perkembangan setiap siswanya.

4.2.3 Antar Muka *Android* Siswa

1. Halaman Awal Aplikasi Android Siswa

Adapun tampilan halaman awal Ketika pertama kali aplikasi di buka dapat dilihat pada dambar di bawah ini:

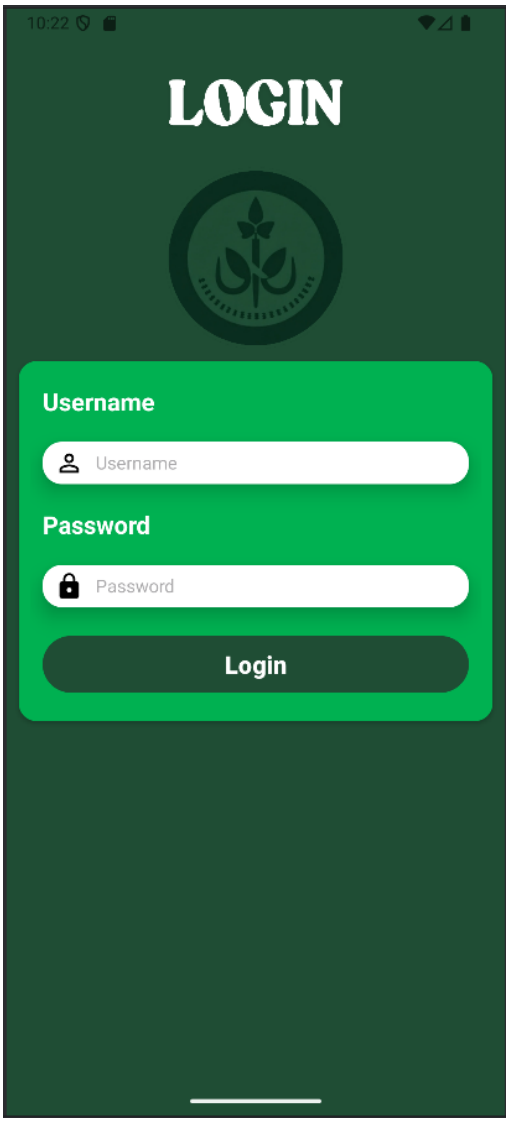


Gambar 4. 11 Halaman Awal Aplikasi Android Siswa

Halaman ini adalah halaman yang muncul pertama kali ketika siswa membuka aplikasi, siswa dapat menekan tombol *login* untuk masuk ke halaman *login.*

2. Halaman *Login* Aplikasi Android Siswa

Adapun tampilan halaman *Login* dapat dilihat pada dambar di bawah ini:

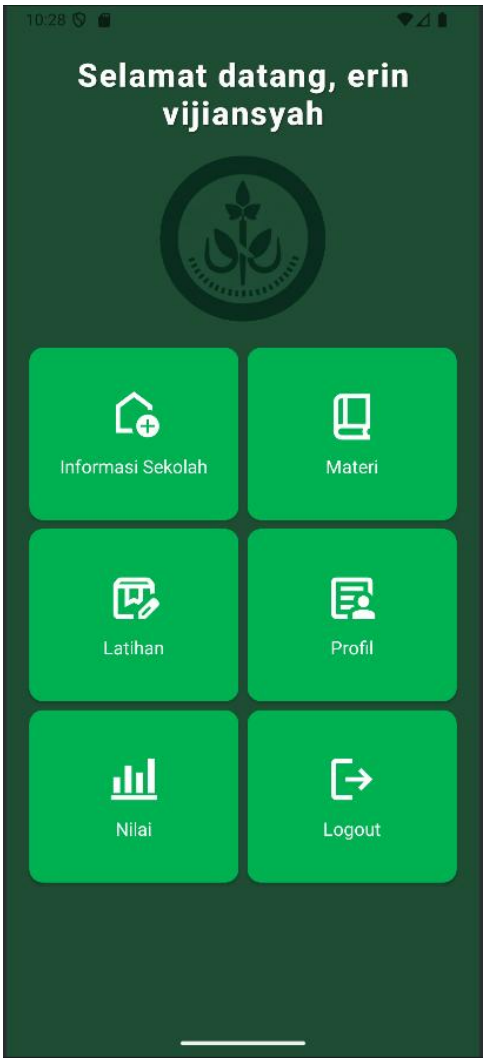


Gambar 4. 12 Halaman *Login* Aplikasi Android Siswa

Pada gambar halaman *Login* diatas, siswa diminta untuk memasukan *username* dan *password* mereka ke dalam kolom yang tersedia. Setelah diisi, pengguna menekan tombol "*Login*" untuk masuk ke aplikasi. Data *Login* ini akan digunakan untuk mengidentifikasi pengguna dan menyimpan nilai latihan.

3. Halaman Menu Utama Aplikasi Android Siswa

Adapun tampilan halaman menu utama aplikasi android siswa dapat dilihat pada dambar di bawah ini:



Gambar 4. 13 Halaman Menu Utama Aplikasi Android Siswa

Pada halaman Menu Utama di atas terdapat beberapa tombol yang digunakan untuk masuk ke masing-masing menu yang tersedia, di antaranya menu Informasi Sekolah, menu Materi, menu Latihan, menu Profil, menu Nilai, dan tombol *Logout*. Di bagian atas juga ditampilkan nama *user* sebagai penanda bahwa aplikasi tersebut sedang digunakan oleh pengguna sesuai data *login*.

4. Halaman Daftar Informasi Sekolah Aplikasi Android Siswa

Adapun tampilan halaman Daftar Informasi Sekolah aplikasi android siswa dapat dilihat pada dambar di bawah ini:



Gambar 4. 14 Halaman Daftar Informasi Sekolah Aplikasi Android Siswa

Pada halaman Daftar Informasi Sekolah di atas terdapat tombol Kembali untuk menuju ke halaman sebelumnya. Selain itu, terdapat juga daftar informasi sekolah yang dapat diklik untuk melihat detail setiap informasi.

5. Halaman Detail Informasi Sekolah Aplikasi Android Siswa

Adapun tampilan halaman Detail Informasi Sekolah aplikasi android siswa dapat dilihat pada dambar di bawah ini:



Gambar 4. 15 Halaman Detail Informasi Sekolah Aplikasi Android Siswa

Pada halaman Detail Informasi Sekolah di atas terdapat tombol Kembali untuk menuju ke halaman sebelumnya. Di bawahnya terdapat detail informasi sekolah yang di tampilkan secara lengkap.

6. Halaman Daftar Materi Aplikasi Android Siswa

Adapun tampilan halaman Daftar Materi aplikasi android siswa dapat dilihat pada dambar di bawah ini:



Gambar 4. 16 Halaman Daftar Materi Aplikasi Android Siswa

Pada halaman Daftar Materi di atas terdapat tombol Kembali untuk menuju ke halaman sebelumnya. Selain itu, terdapat juga daftar Materi yang dapat diklik untuk melihat detail setiap materi.

7. Halaman Detail Materi Aplikasi Android Siswa

Adapun tampilan halaman Detail Materi aplikasi android siswa dapat dilihat pada dambar di bawah ini:

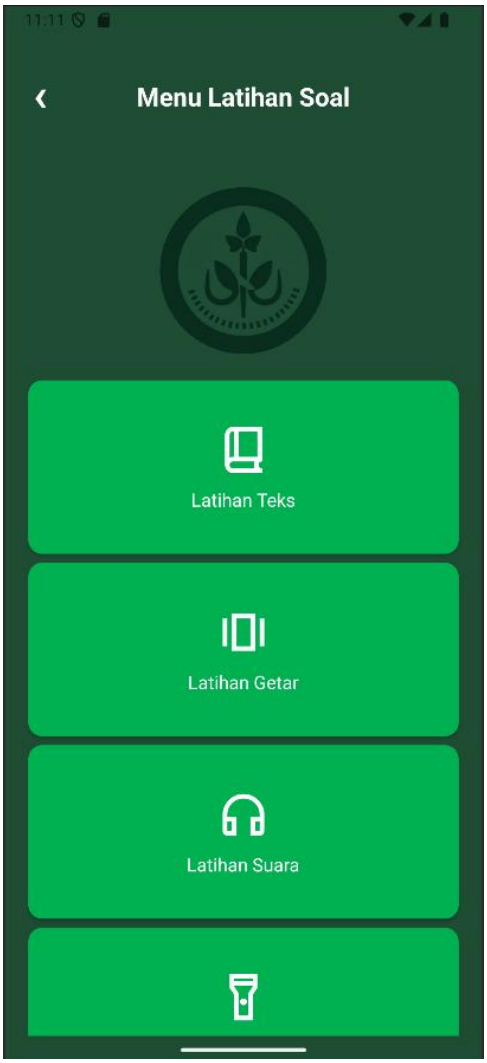


Gambar 4. 17 Halaman Detail Materi Aplikasi Android Siswa

Pada halaman Materi di atas terdapat tombol Kembali untuk menuju ke halaman sebelumnya. Di bawahnya terdapat Materi yang di tampilkan secara lengkap.

8. Halaman Menu Latihan Soal Aplikasi Android Siswa

Adapun tampilan halaman menu Latihan Soal aplikasi android siswa dapat dilihat pada dambar di bawah ini:

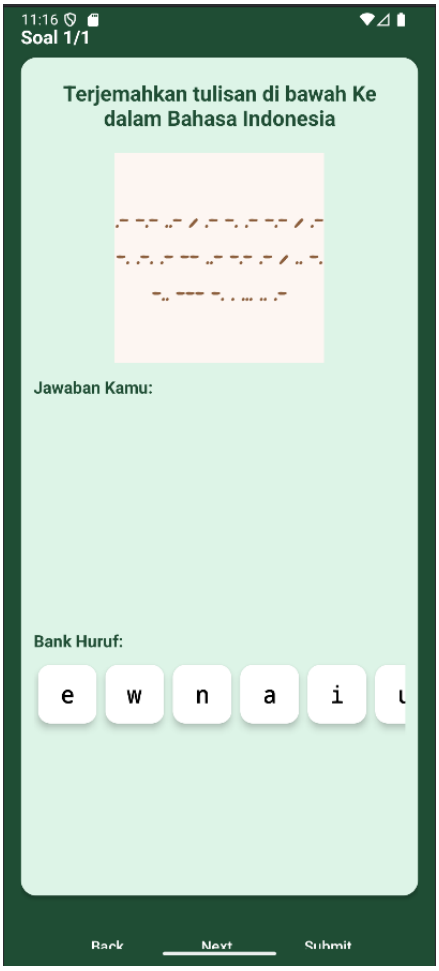


Gambar 4. 18 Halaman Menu Latihan Soal Aplikasi Android Siswa

Pada halaman Menu Utama di atas terdapat beberapa tombol yang digunakan untuk masuk ke masing-masing menu latihan yang tersedia, di antaranya Latihan Teks, Latihan Getar, Latihan Suara, dan Latihan Cahaya. Di bagian atas juga terdapat tombol Kembali untuk kembali ke menu sebelumnya.

9. Halaman Latihan Soal Teks Aplikasi Android Siswa

Adapun tampilan halaman Latihan Soal Teks aplikasi android siswa dapat dilihat pada dambar di bawah ini:

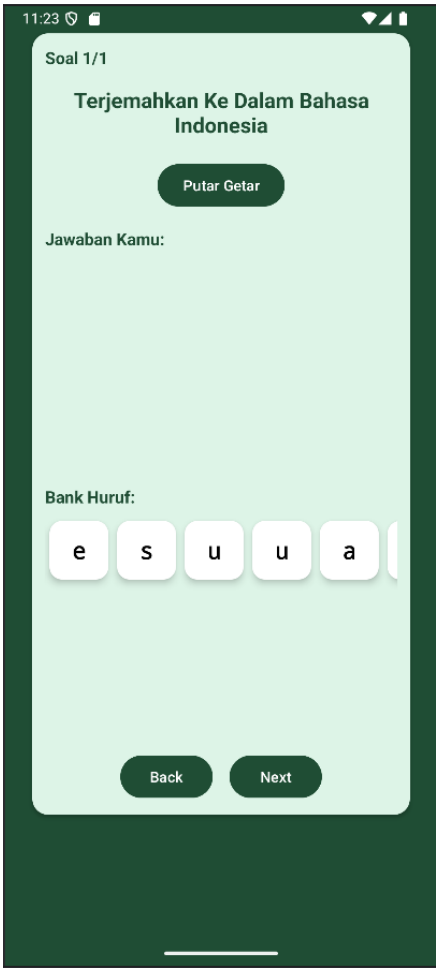


Gambar 4. 19 Halaman Latihan Soal Teks Aplikasi Android Siswa

Pada halaman Latihan Soal Teks di atas terdapat soal, kolom jawaban, serta kolom teks. Di dalam kolom teks terdapat tombol-tombol huruf yang nantinya dapat digeser oleh siswa ke dalam kolom jawaban untuk menyusun kata sebagai jawaban. Selain itu, terdapat beberapa tombol navigasi, yaitu tombol *Back* untuk kembali ke soal sebelumnya, tombol *Next* untuk melanjutkan ke soal berikutnya, dan tombol *Submit* untuk menyimpan jawaban.

10. Halaman Latihan Soal Getar Aplikasi Android Siswa

Adapun tampilan halaman Latihan Soal Getar aplikasi android siswa dapat dilihat pada dambar di bawah ini:

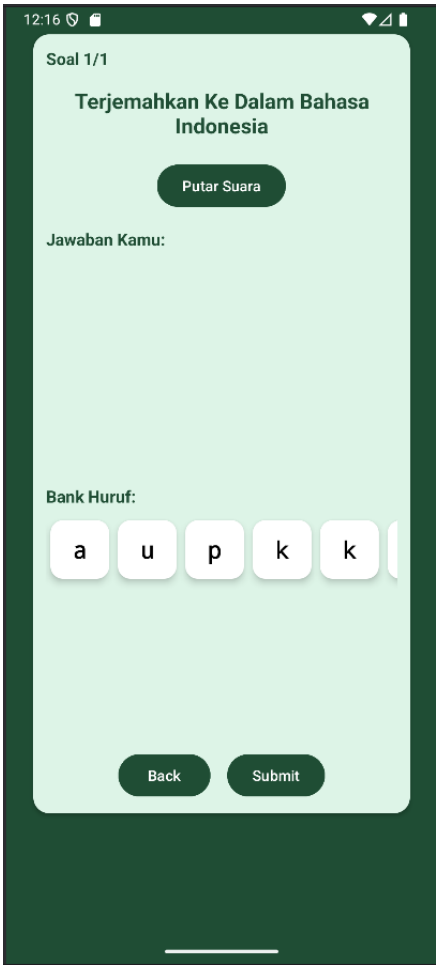


Gambar 4. 20 Halaman Latihan Soal Getar Aplikasi Android Siswa

Pada halaman Latihan Soal Getar di atas terdapat soal, kolom jawaban, serta kolom teks. Di dalam kolom teks terdapat tombol-tombol huruf yang nantinya dapat digeser oleh siswa ke dalam kolom jawaban untuk menyusun kata sebagai jawaban. Selain itu, terdapat beberapa tombol navigasi, yaitu tombol *Back* untuk kembali ke soal sebelumnya, tombol *Next* untuk melanjutkan ke soal berikutnya, dan tombol Putar Getar untuk memutar soal berbasis getaran sandi morse.

11. Halaman Latihan Soal Suara Aplikasi Android Siswa

Adapun tampilan halaman Latihan Soal Suara aplikasi android siswa dapat dilihat pada dambar di bawah ini:

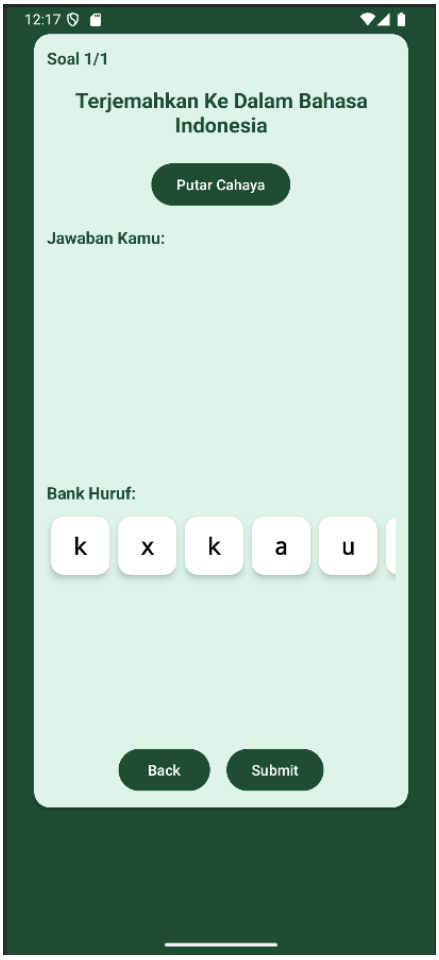


Gambar 4. 21 Halaman Latihan Soal Suara Aplikasi Android Siswa

Pada halaman Latihan Soal Getar di atas terdapat soal, kolom jawaban, serta kolom teks. Di dalam kolom teks terdapat tombol-tombol huruf yang nantinya dapat digeser oleh siswa ke dalam kolom jawaban untuk menyusun kata sebagai jawaban. Selain itu, terdapat beberapa tombol navigasi, yaitu tombol *Back* untuk kembali ke soal sebelumnya, tombol *Nex*t untuk melanjutkan ke soal berikutnya, dan tombol Putar Suara untuk memutar soal berbasis suara dari sandi morse.

12. Halaman Latihan Soal Cahaya Aplikasi Android Siswa

Adapun tampilan halaman Latihan Soal Cahaya aplikasi android siswa dapat dilihat pada dambar di bawah ini:

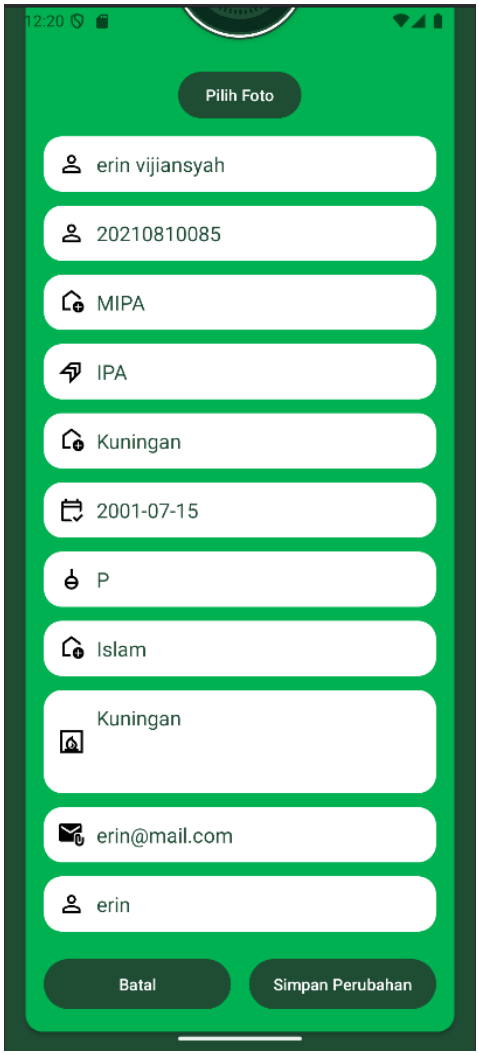


Gambar 4. 22 Halaman Latihan Soal Cahaya Aplikasi Android Siswa

Pada halaman Latihan Soal Cahaya di atas terdapat soal, kolom jawaban, serta kolom teks. Di dalam kolom teks terdapat tombol-tombol huruf yang nantinya dapat digeser oleh siswa ke dalam kolom jawaban untuk menyusun kata sebagai jawaban. Selain itu, terdapat beberapa tombol navigasi, yaitu tombol *Back* untuk kembali ke soal sebelumnya, tombol *Next* untuk melanjutkan ke soal berikutnya, dan tombol Putar Suara untuk memutar soal berbasis cahaya dari sandi morse.

13. Halaman Profile Aplikasi Android Siswa

Adapun tampilan halaman Profile aplikasi android siswa dapat dilihat pada dambar di bawah ini:

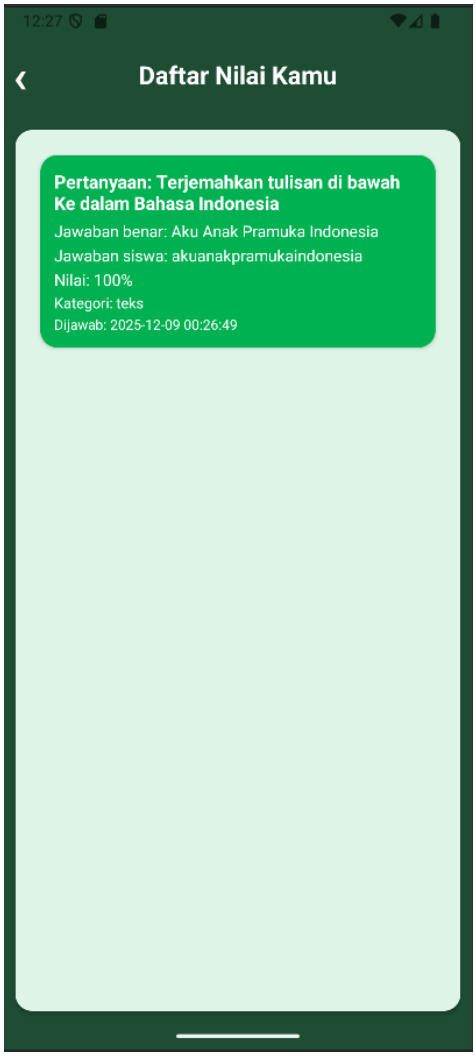


Gambar 4. 23 Halaman *Profile* Aplikasi Android Siswa

Pada halaman Profile di atas terdapat kolom untuk mengubah Nama, Nis, Kelas, Jurusan, Tempat Lahir, Tanggal Lahir, Jenis Kelamin, Agama, Alamat *Email* dan *Password*. Selain itu, terdapat beberapa tombol navigasi, yaitu tombol Batal untuk membatalkan perubahan dan kembali ke halaman sebelumnya, tombol Simpan Perubahan untuk melanjutkan perubahan data, dan tombol Pilih Foto untuk upload foto profile.

14. Halaman Nilai Aplikasi Android Siswa

Adapun tampilan halaman Nilai aplikasi android siswa dapat dilihat pada dambar di bawah ini:



Gambar 4. 24 Halaman Nilai Aplikasi Android Siswa

Pada halaman Nilai di atas terdapat *List* Nilai yang di tampilkan sesuai dengan nilai persentase kecocokan jawaban siswa, disana juga terdapat tombol kembali untuk kembali ke haalaman sebelumnya.

4.3 Pengujian System (System Testing)

Pengujian sistem dilakukan untuk mengukur keselarasan fungsi logika, mekanisme operasional sebuah aplikasi dan atau sistem. Dilakukan menggunakan suatu metode pengujian aplikasi dan atau sistem

4.3.1 Black Box Testing

Black box testing merupakan metode pengujian yang dilakukan dengan cara mengamati output berdasarkan data uji tanpa mengetahui struktur internal perangkat lunak. Tujuan dari pengujian ini adalah menyusun kumpulan kondisi input yang dapat menguji semua persyaratan fungsional dari suatu program.
Adapun *black box testing* dapat dilihat pada di bawah ini :

Tabel 4. 1 *Black Box Testing* Aplikasi Web Admin

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
Aplikasi Web Admin					
1.	Login	Admin mengisi *username* yang salah, lalu menekan tombol login.	Sistem menampilkan notifikasi "Username Tidak di Temukan!"	Sesuai harapan	*Valid*
		Admin mengisi *password* yang salah, lalu menekan tombol login.	Sistem menampilkan notifikasi "Password salah!"	Sesuai harapan	*Valid*
		Admin tidak mengisi	Sistem Menampilkan	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		username dan *password* lalu menekan tombol login.	Notifikasi ”Harap Isi Username” dan ”Harap Isi Kata Sandi”		
		Admin mengisi *username* dan *password* yang benar, dan menekan tombol login.	Sistem menampilkan halaman dashboard	Sesuai harapan	*Valid*
2.	Menuju Halaman Informasi Sekolah	Admin Menekan Tombol Informasi sekolah di navigasi samping dan melanjutkan menekan tombol list.	Sistem menampilkan halaman list informasi sekolah	Sesuai harapan	*Valid*
3.	Kelola Informasi Sekolah	Admin menekan tombol tambah data	Sistem menampilkan form input untuk	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
			menambahkan informasi sekolah		
		Admin mengisi form tambah informasi sekolah lalu tekan tombol simpan	Sistem menampilkan notifikasi ”Data Berhasil Di Simpan” dan data tersimpan kedalam database	Sesuai harapan	*Valid*
		Admin menekan edit informasi sekolah mengisi form lalu menekan tombol ubah	Sistem menampilkan notifikasi ”Data Berhasil Di Ubah”	Sesuai harapan	*Valid*
		Admin menekan tombol hapus untuk menghapus data informasi sekolah	Sistem menampilkan notifikasi “Data Berhasil Dihapus” dan data terhapus	Sesuai harapan	*Valid*
4.	Menuju Halaman	Admin Menekan Tombol	Sistem menampilkan	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
	Profile Guru	Profile Guru di navigasi samping dan melanjutkan menekan tombol list.	halaman list Profile Guru		
5.	Kelola Profile Guru	Admin menekan tombol tambah data	Sistem menampilkan form input untuk menambahkan Profile Guru	Sesuai harapan	*Valid*
		Admin mengisi form tambah Profile Guru lalu tekan tombol simpan	Sistem menampilkan notifikasi ”Data Berhasil Di Simpan” dan data tersimpan kedalam database.	Sesuai harapan	*Valid*
		Admin menekan edit Profile Guru mengisi form lalu menekan tombol ubah	Sistem menampilkan notifikasi ”Data Berhasil Di Ubah”	Sesuai harapan	*Valid*
		Admin menekan tombol	Sistem menampilkan notifikasi “Data	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		hapus untuk menghapus data Profile Guru	Berhasil Dihapus" dan data terhapus		
6.	Menuju Halaman Profile Siswa	Admin Menekan Tombol Profile Siswa di navigasi samping dan melanjutkan menekan tombol list.	Sistem menampilkan halaman list Profile Siswa	Sesuai harapan	*Valid*
7.	Kelola Profile Siswa	Admin menekan tombol tambah data	Sistem menampilkan form input untuk menambahkan Profile Siswa	Sesuai harapan	*Valid*
		Admin mengisi form tambah Profile Siswa lalu tekan tombol simpan	Sistem menampilkan notifikasi "Data Berhasil Di Simpan" dan data tersimpan kedalam database.	Sesuai harapan	*Valid*
		Admin menekan edit	Sistem menampilkan	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		Profile Siswa mengisi form lalu menekan tombol ubah	notifikasi ”Data Berhasil Di Ubah”		
		Admin menekan tombol hapus untuk menghapus data Profile Siswa	Sistem menampilkan notifikasi “Data Berhasil Dihapus” dan data terhapus	Sesuai harapan	*Valid*
8.	Tombol *Logout*	Admin menekan tombol *Logout* untuk kembali ke halaman *login* dan mengakhiri *session login*	Sistem menampilkan halaman *login* dan mengakhiri *session login*	Sesuai harapan	*Valid*

Tabel 4. 2 *Black Box Testing* Aplikasi Web Guru

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
Aplikasi Web Guru					
1	Login	Guru mengisi *username* yang salah,	Sistem menampilkan notifikasi “Username	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		lalu menekan tombol login.	Tidak di Temukan!"		
		Guru mengisi *password* yang salah, lalu menekan tombol login.	Sistem menampilkan notifikasi "Password salah!"	Sesuai harapan	*Valid*
		Guru tidak mengisi *username* dan *password* lalu menekan tombol login.	Sistem Menampilkan Notifikasi "Harap Isi Username" dan "Harap Isi Kata Sandi"	Sesuai harapan	*Valid*
		Guru mengisi *username* dan *password* yang benar, dan menekan tombol login.	Sistem menampilkan halaman dashboard	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
2	Menuju Halaman Materi	Guru Menekan Tombol Materi di navigasi samping dan melanjutkan menekan tombol list.	Sistem menampilkan halaman list Materi	Sesuai harapan	*Valid*
3.	Kelola Materi	Guru menekan tombol tambah data	Sistem menampilkan form input untuk menambahkan Materi	Sesuai harapan	*Valid*
		Guru mengisi form Materi lalu tekan tombol simpan	Sistem menampilkan notifikasi ”Data Berhasil Di Simpan” dan data tersimpan kedalam database	Sesuai harapan	*Valid*
		Guru menekan Materi lalu mengisi form dan menekan tombol ubah	Sistem menampilkan notifikasi ”Data Berhasil Di Ubah”	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		Guru menekan tombol hapus untuk menghapus data Materi	Sistem menampilkan notifikasi “Data Berhasil Dihapus” dan data terhapus	Sesuai harapan	*Valid*
4.	Menuju Halaman Soal	Guru Menekan Tombol Soal di navigasi samping dan melanjutkan menekan tombol list.	Sistem menampilkan halaman list Soal	Sesuai harapan	*Valid*
5.	Kelola Soal	Guru menekan tombol tambah data	Sistem menampilkan form input untuk menambahkan Soal	Sesuai harapan	*Valid*
		Guru mengisi form tambah Soal lalu tekan tombol simpan	Sistem menampilkan notifikasi ”Data Berhasil Di Simpan” dan data tersimpan	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
			kedalam database.		
		Guru menekan edit Soal mengisi form lalu menekan tombol ubah	Sistem menampilkan notifikasi ”Data Berhasil Di Ubah”	Sesuai harapan	*Valid*
		Guru menekan tombol hapus untuk menghapus data Soal	Sistem menampilkan notifikasi “Data Berhasil Dihapus” dan data terhapus	Sesuai harapan	*Valid*
6.	Menuju Halaman Nilai	Admin Menekan Tombol Nilai di navigasi samping dan melanjutkan menekan tombol list.	Sistem menampilkan halaman list Nilai	Sesuai harapan	*Valid*
7.	Tombol *Logout*	Guru menekan tombol *Logout* untuk kembali ke halaman	Sistem menampilkan halaman *login* dan mengakhiri *session login*	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		login dan mengakhiri *session login*			

Tabel 4. 3 *Black Box Testing* Aplikasi Android Siswa

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
Aplikasi Android Siswa					
1.	Login	Siswa mengisi *username* yang salah, lalu menekan tombol login.	Sistem menampilkan notifikasi "Username Tidak di Temukan!"	Sesuai harapan	*Valid*
		Siswa mengisi *password* yang salah, lalu menekan tombol login.	Sistem menampilkan notifikasi "Password salah!"	Sesuai harapan	*Valid*
		Siswa tidak mengisi *username* dan *password* lalu menekan	Sistem Menampilkan Notifikasi "Username dan Password wajib di isi"	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		tombol login.			
		Siswa mengisi *username* dan *password* yang benar, dan menekan tombol login.	Sistem menampilkan halaman dashboard Siswa	Sesuai harapan	*Valid*
2.	Menuju Halaman List Informasi Sekolah	Siswa Menekan Tombol Informasi sekolah.	Sistem menampilkan halaman list informasi sekolah.	Sesuai harapan	*Valid*
2.	Menuju Halaman detail Informasi Sekolah	Siswa Menekan salah satu list Informasi sekolah.	Sistem menampilkan halaman detail informasi sekolah	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
3.	Menuju Halaman List Materi	Siswa Menekan Tombol Materi	Sistem menampilkan halaman list Materi	Sesuai harapan	*Valid*
4.	Menuju Halaman detail Materi	Siswa Menekan salah satu list Materi	Sistem menampilkan halaman detail Materi	Sesuai harapan	*Valid*
5.	Menuju Halaman Latihan	Siswa Menekan latihan	Sistem menampilkan halaman Latihan	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
6.	Latihan Berbasis Teks	Siswa Menekan latihan Teks untuk memulai latihan berbasis teks dan mengisi Latihan.	Sistem menampilkan Notifikasi ”Semua Jawaban Sudah disimpan” dan Kembali ke Halaman Awal.	Sesuai harapan	*Valid*
		Siswa Menekan latihan Teks yang sudah di kerjakan untuk memulai latihan berbasis teks dan mengisi Latihan.	Sistem menampilkan Notifikasi ”Soal ini sudah pernah dijawab”	Sesuai harapan	*Valid*
7.	Latihan Berbasis Getar	Siswa Menekan latihan Getar untuk memulai latihan berbasis Getar dan	Sistem menampilkan Notifikasi ”Semua Jawaban Sudah disimpan” dan Kembali ke Halaman Awal.	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		mengisi Latihan.			
		Siswa Menekan latihan Getar yang sudah di kerjakan untuk memulai latihan berbasis Getar dan mengisi Latihan.	Sistem menampilkan Notifikasi ”Soal ini sudah pernah dijawab”	Sesuai harapan	*Valid*
8.	Latihan Berbasis Suara	Siswa Menekan latihan Suara untuk memulai latihan berbasis Suara dan	Sistem menampilkan Notifikasi ”Semua Jawaban Sudah disimpan” dan Kembali ke Halaman Awal.	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		mengisi Latihan.			
		Siswa Menekan latihan Suara yang sudah di kerjakan untuk memulai latihan berbasis Suara dan mengisi Latihan.	Sistem menampilkan Notifikasi ”Soal ini sudah pernah dijawab”	Sesuai harapan	*Valid*
9.	Latihan Berbasis Cahaya	Siswa Menekan latihan Cahaya untuk memulai latihan berbasis Cahaya dan	Sistem menampilkan Notifikasi ”Semua Jawaban Sudah disimpan” dan Kembali ke Halaman Awal.	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
		mengisi Latihan.			
		Siswa Menekan latihan Cahaya yang sudah di kerjakan untuk memulai latihan berbasis Cahaya dan mengisi Latihan.	Sistem menampilkan Notifikasi ”Soal ini sudah pernah dijawab”	Sesuai harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
10.	Menuju Halaman Profile dan merubah Profile Siswa	Siswa Menekan salah satu Tombol Profile	Sistem menampilkan halaman Form Profile Siswa	Sesuai harapan	*Valid*
		Siswa merubah data profile siswa dan menekan tombol “Simpan Perubahan	Sistem menampilkan Notifikasi ”Profile Berhasil di Perbaharui”	Sesuai Harapan	*Valid*
		Siswa menekan tombol batal untuk membatalkan perubahan pada profile siswa	Sistem menampilkan halaman awal aplikasi	Sesuai Harapan	*Valid*

No	Fungsi Yang Diuji	Cara Menguji	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
11.	Menuju Halaman Nilai	Siswa Menekan Tombol Nilai	Sistem menampilkan halaman Nilai	Sesuai harapan	*Valid*
7.	Tombol *Logout*	Admin menekan tombol *Logout* untuk kembali ke halaman *login* dan mengakhiri *session login*	Sistem menampilkan halaman *login* dan mengakhiri *session login*	Sesuai harapan	*Valid*

4.3.2 White Box Testing

Pengujian white box dilakukan untuk memastikan bahwa proses normalisasi teks jawaban serta pencocokan string antara Jawaban A (kunci) dan Jawaban B (jawaban siswa) berjalan sesuai dengan alur logika yang telah dirancang pada program. Melalui pengujian ini, seluruh struktur kontrol dalam fungsi dapat dievaluasi, sehingga dapat dipastikan bahwa algoritma bekerja tepat pada semua kondisi yang mungkin terjadi.

Pada tahap ini dilakukan pembuatan flowgraph, perhitungan cyclomatic complexity, dan penentuan jalur independen (basis path) sebagai dasar pengujian.

1. Pengujian Fungsi Pencocokan Jawaban (Rabin–Karp)

Algoritma Rabin–Karp digunakan untuk menentukan apakah Jawaban B (siswa) mengandung string yang sama persis dengan Jawaban A (kunci) setelah dinormalisasi.

Kode Yang di Uji

```
function rabinKarp($text, $pattern, $d = 256, $q = 101) {
    $n = strlen($text);
    $m = strlen($pattern);

    if ($m == 0 || $n == 0 || $m > $n) return false;

    $h = 1;
    for ($i = 0; $i < $m - 1; $i++) {
        $h = ($h * $d) % $q;
    }



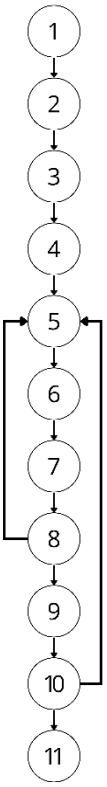
    $p = 0;
    $t = 0;
    for ($i = 0; $i < $m; $i++) {
        $p = ($d * $p + ord($pattern[$i])) % $q;
        $t = ($d * $t + ord($text[$i])) % $q;
    }

    for ($i = 0; $i <= $n - $m; $i++) {
        if ($p == $t) {
            $j = 0;
            for (; $j < $m; $j++) {
                if ($text[$i + $j] !== $pattern[$j]) break;
            }
            if ($j == $m) {
                return true;
            }
        }
        if ($i < $n - $m) {
            $t = ($d * ($t - ord($text[$i]) * $h) + ord($text[$i + $m])) % $q;
```

```
            if ($t < 0) $t += $q;
        }
    }
    return false;
}
```

Daftar Node (Untuk Jawaban A & B)

Node	Source Code
1	$n = strlen($text); dan $m = strlen($pattern);
2	`if ($m == 0
3	$h = 1; dan loop for ($i = 0; $i < $m - 1; $i++) { ... }
4	Inisialisasi hash: $p = 0; $t = 0; dan loop for ($i = 0; $i < $m; $i++)
5	Loop utama: for ($i = 0; $i <= $n - $m; $i++)
6	if ($p == $t)
7	Loop pembanding karakter: for (; $j < $m; $j++) { ... }
8	if ($j == $m) return true;
9	Rolling hash: $t = ($d * ($t - ord(...) * $h) + ord(...)) % $q;
10	if ($t < 0) $t += $q;
11	return false;



Gambar 4. 25 *Flowgraph* Pencocokan Jawaban

Perhitungan Cyclomatic Complexity

Dari struktur di atas:

- N = 11
- E = 13

V(G) = (E – N) + 2
V(G) = (13 – 11) + 2
V(G) = 4

Artinya ada 4 jalur independen yang wajib diuji.

Path 1 — Jawaban cocok sempurna pada pergeseran pertama

1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11

Contoh kasus:

• A = “adatambalan”

• B = “adatambalanpramuka…”

Path 2 — Hash cocok, tetapi karakter tidak cocok

1 → 2 → 3 → 4 → 5 → 6 → 7 →8→ 9 → 10 → kembali ke 5 → 6 → 7 →8→ 9 → 10 → 11

Contoh kasus:

• A = “sandi”

• B = “sando” (hash sama, karakter berbeda)

Path 3 — Hash tidak cocok sejak langkah awal

1 → 2 → 3 → 4 → 5 → 6 →7→ 8 → kembali ke 5 → 6 → 7 →8→ 9 → 10 → 11

Contoh kasus:

• A = “sandi”

• B = “panah”

Path 4 — Pola tidak ditemukan pada seluruh teks

Seluruh iterasi dilalui tanpa ditemukan kecocokan sempurna hingga mencapai Node 11.

2. Integrasi Normalisasi + Pencocokan Jawaban

Kode akhir aplikasi

```
$jawabanA = normalizeText($jawaban_benar);

$jawabanB = normalizeText($jawaban_siswa);

$benar = rabinKarp($jawabanB, $jawabanA) ? 1 : 0;
```

Alur logika

Normalisasi A → Normalisasi B → Pencocokan Rabin–Karp → Output 1 atau 0

Normalisasi hanya 1 path

Pencocokan memiliki 3 path

Total skenario uji: 1 × 3 = 3 skenario

4.3.3 User Acceptance Test (UAT)

User Acceptance Test (UAT) adalah jenis pengujian yang bertujuan untuk memastikan kepada pengguna bahwa sistem yang dikembangkan telah sesuai dengan kebutuhan dan pemahaman mereka. Dalam penelitian ini, pengujian dilakukan dengan melibatkan 30.

Nilai bobot jawaban kuesioner dapat dilihat pada Tabel di bawah ini

Tabel 4. 4 Nilai Bobot Kuesioner

Kode	Keterangan Jawaban	Bobot
A	Sangat Setuju	5
B	Setuju	4
C	Netral	3
D	Tidak Setuju	2
E	Sangat Tidak Setuju	1

Adapun hasil jawaban dari pertanyaan quisioner yang saya ajukan bisa dilihat pada table di bawah ini:

Tabel 4. 5 Pertanyaan dan jawaban Responden UAT

No	Pernyataan	Jawaban					Persentase				
		A	B	C	D	E	A	B	C	D	E
1.	Aplikasi ini membantu saya memahami dasar-dasar sandi Morse dengan lebih mudah.	20	8	2	0	0	66,7	26,7	6,7	0	0
2.	Fitur latihan dan evaluasi pada	18	11	1	**0**	**0**	**60**	**36,7**	**3,3**	**0**	**0**

No	Pernyataan	Jawaban					Persentase				
		A	B	C	D	E	A	B	C	D	E
	aplikasi membantu saya meningkatkan kemampuan sandi Morse.										
3.	Tampilan antarmuka aplikasi terlihat menarik dan interaktif.	**12**	17	1	**0**	**0**	**40**	**56,7**	**3,3**	**0**	**0**
4.	Aplikasi ini sesuai dengan kebutuhan siswa dalam mempelajari sandi Morse di lingkungan kepramukaan.	**17**	11	2	**0**	**0**	**56,7**	**36,7**	**6,7**	**0**	**0**
5.	Aplikasi ini membuat proses belajar sandi Morse menjadi lebih menyenangkan.	**16**	12	**2**	**0**	**0**	**53,3**	**40**	**6,7**	**0**	**0**
6.	Aplikasi ini berjalan dengan lancar tanpa gangguan (error)	**11**	18	1	**0**	**0**	**36,7**	**60**	**3,3**	**0**	**0**

No	Pernyataan	Jawaban					Persentase				
		A	B	C	D	E	A	B	C	D	E
	selama saya gunakan.										
7.	Fitur drag-and-drop (fungsi geser jawaban) berfungsi dengan baik dan mudah digunakan.	**8**	22	0	**0**	**0**	**26,7**	**73,3**	**0**	**0**	**0**
8.	Saya merasa aplikasi ini membantu saya mengukur perkembangan pemahaman sandi Morse.	**15**	12	3	**0**	**0**	**50**	**40**	**10**	**0**	**0**
9.	Aplikasi ini bermanfaat untuk pembelajaran mandiri di luar kegiatan sekolah/pramuka.	**25**	5	0	**0**	**0**	**83,3**	**16,7**	**0**	**0**	**0**
10	Saya tertarik untuk terus menggunakan aplikasi ini di waktu mendatang	**19**	11	0	**0**	**0**	**63,3**	**36,7**	**0**	**0**	**0**

Data yang diperoleh kemudian diolah dengan mengalikan setiap jawaban responden dengan bobot sesuai yang tercantum pada Tabel 4.4. Hasil pengolahan data kuesioner dari konsumen dapat dilihat pada Tabel di bawah ini.

Tabel 4. 6 Hasil Pengelolaan Data Kuesioner

No	Pernyataan	Jawaban				
		Ax5	**Bx4**	**Cx3**	**Dx2**	**Ex1**
1.	Aplikasi ini membantu saya memahami dasar-dasar sandi Morse dengan lebih mudah.	100	32	6	0	0
2.	Fitur latihan dan evaluasi pada aplikasi membantu saya meningkatkan kemampuan sandi Morse.	90	44	3	**0**	**0**
3.	Tampilan antarmuka aplikasi terlihat menarik dan interaktif.	**60**	68	3	**0**	**0**
4.	Aplikasi ini sesuai dengan kebutuhan siswa dalam mempelajari	**85**	44	6	**0**	**0**

No	Pernyataan	Jawaban				
		Ax5	**Bx4**	**Cx3**	**Dx2**	**Ex1**
	sandi Morse di lingkungan kepramukaan.					
5.	Aplikasi ini membuat proses belajar sandi Morse menjadi lebih menyenangkan.	**80**	48	**6**	**0**	**0**
6.	Aplikasi ini berjalan dengan lancar tanpa gangguan (error) selama saya gunakan.	**55**	72	3	**0**	**0**
7.	Fitur drag-and-drop (fungsi geser jawaban) berfungsi dengan baik dan mudah digunakan.	**40**	88	0	**0**	**0**
8.	Saya merasa aplikasi ini membantu saya mengukur perkembangan pemahaman sandi Morse.	**75**	48	9	**0**	**0**

No	Pernyataan	Jawaban				
		Ax5	Bx4	Cx3	Dx2	Ex1
9.	Aplikasi ini bermanfaat untuk pembelajaran mandiri di luar kegiatan sekolah/pramuka.	**125**	20	0	**0**	**0**
10	Saya tertarik untuk terus menggunakan aplikasi ini di waktu mendatang	**95**	44	0	**0**	**0**
Jumlah		**1349**				

Hasil analisis terhadap jawaban kuesioner disajikan pada Tabel di bawah ini

Tabel 4. 7 Hasil analisis terhadap jawaban kuesioner

No	Pernyataan	Jawaban	
		(Jumlah/30)/5	Persentase (x100)
1.	Aplikasi ini membantu saya memahami dasar-dasar sandi Morse dengan lebih mudah.	0,92	92%
2.	Fitur latihan dan evaluasi pada aplikasi membantu saya meningkatkan	0,91	**91**%

No	Pernyataan	Jawaban	
		(Jumlah/30)/5	Persentase (x100)
	kemampuan sandi Morse.		
3.	Tampilan antarmuka aplikasi terlihat menarik dan interaktif.	**0,87**	**87**%
4.	Aplikasi ini sesuai dengan kebutuhan siswa dalam mempelajari sandi Morse di lingkungan kepramukaan.	**0,9**	**90**%
5.	Aplikasi ini membuat proses belajar sandi Morse menjadi lebih menyenangkan.	**0,89**	**89**%
6.	Aplikasi ini berjalan dengan lancar tanpa gangguan (error) selama saya gunakan.	**0,86**	**86**%
7.	Fitur drag-and-drop (fungsi	**0,85**	**85**%

No	Pernyataan	Jawaban	
		(Jumlah/30)/5	Persentase (x100)
	geser jawaban) berfungsi dengan baik dan mudah digunakan.		
8.	Saya merasa aplikasi ini membantu saya mengukur perkembangan pemahaman sandi Morse.	**0,88**	**88**%
9.	Aplikasi ini bermanfaat untuk pembelajaran mandiri di luar kegiatan sekolah/pramuka.	**0,96**	**96**%
10	Saya tertarik untuk terus menggunakan aplikasi ini di waktu mendatang	**0,92**	**92**%

Berdasarkan Tabel 4.6, diperoleh skor hasil pengujian sebesar 1349. Selanjutnya, untuk mengetahui persentase kelayakan, dilakukan perhitungan sebagai berikut:

$$\textit{Persetase Kelayakan (\%)} = \frac{Skor\ hasil\ pengujian}{skor\ yang\ diharapkan} \text{X } 100\%$$

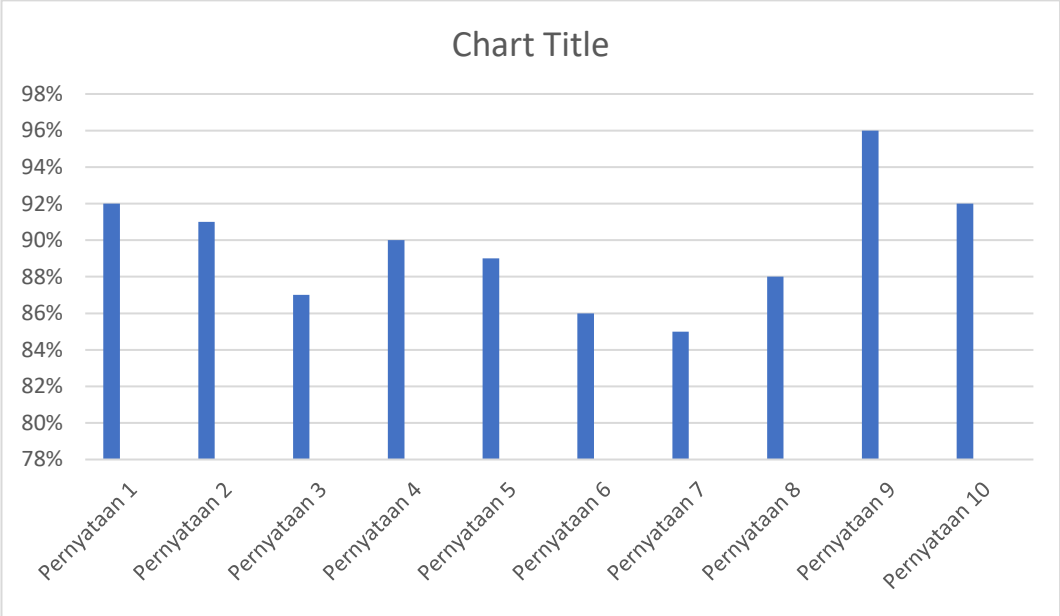
$$= \frac{1349}{1500} x\ 100\%$$

$$= 89\%$$

Berikut ini keterangan rumus diatas :

- Skor hasil pengujian diperoleh dari hasil pengelolaan data kuesioner admin bisa dilihat pada Tabel 4.6.
- $Skor\ yang\ diharapkan = Jumlah\ Pertayaan \times Bobot\ Tertinggi \times Jumlah\ Responden$
- $Skor\ yang\ diharapkan = 6 \times 5 \times 30 = 900$

Dari hasil pengujian tersebut, dapat disimpulkan bahwa aplikasi Media Pembelajaran Sandi Morse dapat diterima dengan baik oleh konsumen dengan nilai persentase kelayakan sebesar 89% .



Grafik 4. 1 Grafik Pers*entase User Acceptance Test*

.