UJI KUALITAS PERANGKAT LUNAK PADA WEBSITE MANAJEMEN NILAI SISWA

TUGAS MATA KULIAH UJI KUALITAS PERANGKAT LUNAK

Disusun Oleh:

Nama : Nadia

NIM : 123220189

PROGRAM STUDI INFORMATIKA  
JURUSAN INFORMATIKA  
FAKULTAS TEKNIK INDUSTRI  
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”   
YOGYAKARTA  
2025

# Deskripsi Projek

## Judul Projek

Aplikasi Manajemen Nilai Siswa (Student Grade Management)

## Latar Belakang

Aplikasi ini dikembangkan untuk menyediakan sebuah sistem bagi para pendidik (guru) guna mengelola data akademik siswa secara efisien. Kebutuhan akan sistem yang terkomputerisasi untuk pencatatan dan pengelolaan kelas, data siswa, serta nilai-nilai mereka menjadi dasar pengembangan aplikasi ini. Dengan adanya sistem ini, diharapkan proses administrasi akademik menjadi lebih terstruktur, mudah diakses, dan mengurangi potensi kesalahan manual.

## Ruang Lingkup

Pengguna utama dari aplikasi ini adalah Guru. Guru memiliki hak akses untuk melakukan autentikasi (login), mengelola data kelas, mengelola data siswa, dan mengelola nilai siswa. Hal ini terlihat dari adanya tabel users pada skema database (schema.sql) yang merujuk pada pengguna sistem, serta fungsionalitas autentikasi pada functions/auth.php

### Identifikasi Pengguna

### Fitur-fitur Aplikasi

* Autentikasi Pengguna:
  + Registrasi pengguna baru (register.php)
  + Login pengguna (index.php, functions/auth.php)
  + Logout pengguna (logout.php)
* Dashboard Utama: Halaman utama setelah login yang menyediakan navigasi ke modul-modul lain (dashboard.php).
* Manajemen Kelas:
  + Membuat, membaca, memperbarui, dan menghapus (CRUD) data kelas (classes.php, functions/classes.php).
* Manajemen Siswa:
  + Membuat, membaca, memperbarui, dan menghapus (CRUD) data siswa, termasuk mengasosiasikannya dengan kelas tertentu (students.php, functions/students.php).
* Manajemen Nilai:
  + Membuat, membaca, memperbarui, dan menghapus (CRUD) data nilai siswa untuk mata pelajaran tertentu (grades.php, functions/grades.php).
  + Kalkulasi otomatis grade\_point berdasarkan skor (functions/grades.php).

### Batasan Sistem

Berdasarkan struktur dan file yang ada, beberapa batasan sistem yang dapat diidentifikasi adalah:

* Routing Sederhana: Aplikasi menggunakan sistem routing berbasis file secara langsung (misalnya, classes.php diakses melalui URL classes.php). Ini disebutkan dalam rencana.md.
* Satu Jenis Pengguna: Sistem saat ini tampaknya dirancang untuk satu jenis peran pengguna (Guru) tanpa adanya perbedaan hak akses yang lebih kompleks.
* Tidak Ada Fitur Laporan Lanjutan: Tidak terlihat adanya fitur untuk pembuatan laporan statistik atau analitik yang kompleks mengenai data nilai atau siswa.
* Interaksi Database Langsung: Logika bisnis dan interaksi database ditempatkan dalam folder functions dan dipanggil langsung dari file-file tampilan.
* Frontend: Menggunakan PHP native yang dicampur dengan HTML dan TailwindCSS untuk tampilan.

## Struktur Database

Struktur database MySQL untuk aplikasi ini didefinisikan dalam file schema.sql dan juga didokumentasikan dalam rencana.md. Terdiri dari tabel-tabel berikut:

* **users**: Menyimpan data pengguna (guru).
  + id (INT, AUTO\_INCREMENT, PRIMARY KEY)
  + username (VARCHAR(50), UNIQUE, NOT NULL)
  + password (VARCHAR(255), NOT NULL)
  + full\_name (VARCHAR(100), NOT NULL)
* **classes**: Menyimpan data kelas.
  + id (INT, AUTO\_INCREMENT, PRIMARY KEY)
  + name (VARCHAR(100), NOT NULL)
  + description (VARCHAR(255))
* **students**: Menyimpan data siswa.
  + id (INT, AUTO\_INCREMENT, PRIMARY KEY)
  + full\_name (VARCHAR(100), NOT NULL)
  + email (VARCHAR(100), UNIQUE, NOT NULL)
  + class\_id (INT, NOT NULL, FOREIGN KEY ke classes.id)
* **grades**: Menyimpan data nilai siswa.
  + id (INT, AUTO\_INCREMENT, PRIMARY KEY)
  + student\_id (INT, NOT NULL, FOREIGN KEY ke students.id)
  + subject (VARCHAR(100), NOT NULL)
  + score (INT, NOT NULL, CHECK score BETWEEN 0 AND 100)
  + grade\_point (DECIMAL(3,2), NOT NULL)
  + updated\_at (DATETIME, NOT NULL, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

Relasi antar tabel adalah sebagai berikut:

* Satu kelas dapat memiliki banyak siswa (classes ke students).
* Satu siswa dapat memiliki banyak nilai (students ke grades).

## Gambaran Umum Aplikasi

* Tumpukan Teknologi: PHP Native + TailwindCSS + MySQL
* Modul Inti: Otentikasi, Manajemen Kelas, Manajemen Siswa, Manajemen Nilai
* Basis Data: MySQL dengan 4 tabel (users, classes, students, grades)
* Arsitektur: Perutean berbasis file dengan fungsi modular

# Pengujian

## White Box Testing

White testing menggunakan Cyclomatic Complexity

### Flowchart Login/Register

|  |
| --- |
|  |

**Cyclomatic Complexity (V(G))**

* **Rumus**: V(G) = Jumlah Edge - Jumlah Node + 2
  + Edge = 11, Node = 9
  + V(G) = 11 - 9 + 2 = 4
* **Jalur Independen**: 4
  1. Start → User belum login → Tampilkan Login/Register → Pilih Login → Input Data → Verifikasi Valid → Redirect Dashboard.
  2. Start → User belum login → Tampilkan Login/Register → Pilih Login → Input Data → Verifikasi Tidak Valid → Kembali ke Login.
  3. Start → User belum login → Tampilkan Login/Register → Pilih Register → Input Data → Simpan → Kembali ke Login.
  4. Start → User sudah login → Langsung ke Dashboard.

**Test Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Input** | **Expected Result** | **Hasil** |
| 1 | User belum login, pilih Login, data valid | Redirect ke Dashboard | Sesuai |
| 2 | User belum login, pilih Login, data invalid | Tampilkan pesan error | Sesuai |
| 3 | User belum login, pilih Register, input data baru | Data tersimpan, kembali ke Login | Sesuai |
| 4 | User sudah login | Langsung ke Dashboard | Sesuai |

### CRUD Kelas

|  |
| --- |
|  |

**Cyclomatic Complexity (V(G))**

* **Rumus**: V(G) = Jumlah Edge - Jumlah Node + 2
  + Edge = 12, Node = 10
  + V(G) = 12 - 10 + 2 = 4
* **Jalur Independen**: 4
  1. Mulai → Menu Kelas → Tambah → Input Nama → Simpan → Kembali ke Menu.
  2. Mulai → Menu Kelas → Edit → Pilih Kelas → Edit Nama → Simpan → Kembali ke Menu.
  3. Mulai → Menu Kelas → Hapus → Pilih Kelas → Hapus → Kembali ke Menu.
  4. Mulai → Menu Kelas → Lihat → Tampilkan Daftar → Kembali ke Menu.

**Test Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Input** | **Expected Result** | **Hasil** |
| 1 | Tambah kelas "XII IPA" | Kelas tersimpan di database | Sesuai |
| 2 | Edit kelas "XII IPA" jadi "XII IPA 1" | Nama kelas terupdate | Sesuai |
| 3 | Hapus kelas "XII IPA 1" | Kelas dihapus dari database | Sesuai |
| 4 | Lihat daftar kelas | Tampilkan semua kelas | Sesuai |

### CRUD Siswa

|  |
| --- |
|  |

**Cyclomatic Complexity (V(G))**

* **Rumus**: V(G) = Jumlah Edge - Jumlah Node + 2
  + Edge = 14, Node = 11
  + V(G) = 14 - 11 + 2 = 5
* **Jalur Independen**: 5
  1. Mulai → Menu Siswa → Tambah → Input Data → Pilih Kelas → Simpan → Kembali ke Menu.
  2. Mulai → Menu Siswa → Edit → Pilih Siswa → Edit Data → Simpan → Kembali ke Menu.
  3. Mulai → Menu Siswa → Hapus → Pilih Siswa → Hapus → Kembali ke Menu.
  4. Mulai → Menu Siswa → Lihat → Tampilkan Daftar → Kembali ke Menu.
  5. Mulai → Menu Siswa → Tambah → Input Data → **Tanpa Pilih Kelas** → Error.

**Test Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Input** | **Expected Result** | **Hasil** |
| 1 | Tambah siswa "Andi", kelas "XII IPA" | Data siswa tersimpan | Sesuai |
| 2 | Edit siswa "Andi" jadi "Budi" | Nama siswa terupdate | Sesuai |
| 3 | Hapus siswa "Budi" | Data siswa dihapus | Sesuai |
| 4 | Lihat daftar siswa | Tampilkan semua siswa | Sesuai |
| 5 | Tambah siswa tanpa pilih kelas | Tampilkan pesan error | Sesuai |

### CRUD Nilai

|  |
| --- |
|  |

**Cyclomatic Complexity (V(G))**

* **Rumus**: V(G) = Jumlah Edge - Jumlah Node + 2
  + Edge = 12, Node = 10
  + V(G) = 12 - 10 + 2 = 4
* **Jalur Independen**: 4
  1. Mulai → Menu Nilai → Tambah → Pilih Siswa → Input Nilai → Simpan → Kembali ke Menu.
  2. Mulai → Menu Nilai → Edit → Pilih Nilai → Edit Nilai → Simpan → Kembali ke Menu.
  3. Mulai → Menu Nilai → Hapus → Pilih Nilai → Hapus → Kembali ke Menu.
  4. Mulai → Menu Nilai → Lihat → Tampilkan Daftar → Kembali ke Menu.

**Test Case**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Input** | **Expected Result** | **Hasil** |
| 1 | Tambah nilai Matematika "90" untuk siswa "Andi" | Nilai tersimpan | Sesuai |
| 2 | Edit nilai "90" jadi "95" | Nilai terupdate | Sesuai |
| 3 | Hapus nilai Matematika "95" | Nilai dihapus | Sesuai |
| 4 | Lihat daftar nilai | Tampilkan semua nilai | Sesuai |

## Black Box Testing

### Sistem Login/Register

Rancangan Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Teknik** | **Input** | **Expected Result** | **Catatan** |
| - Valid | Username: "admin", Password: "123" | Redirect ke Dashboard | Partisi valid. |
| - Invalid (Username) | Username: "", Password: "123" | Tampilkan pesan error | Partisi kosong. |
| - Invalid (Password) | Username: "admin", Password: "" | Tampilkan pesan error | Partisi kosong. |
| **Boundary Value** |  |  |  |
| - Username panjang (50 karakter) | Username: "a...a" (50 chars) | Redirect ke Dashboard (jika valid) | Uji panjang maksimal. |
| - Password pendek (1 karakter) | Password: "1" | Error (min. 6 karakter) | Uji batas bawah. |
| **Scenario-Based** |  |  |  |
| - Register → Login | 1. Register user baru → 2. Login | Redirect ke Dashboard | Alur lengkap. |

Hasil Testing

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Hasil** | **Status** |
| Login valid | Berhasil masuk dashboard | Berhasil |
| Login tanpa password | Error: "Password wajib diisi" | Berhasil |
| Register + Login | User baru bisa login | Berhasil |

### Sistem CRUD Kelas

Rancangan Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Teknik** | **Input** | **Expected Result** | **Catatan** |
| - Valid | Nama Kelas: "XII IPA" | Data tersimpan | Partisi valid. |
| - Invalid (Kosong) | Nama Kelas: "" | Error: "Nama kelas wajib diisi" | Partisi kosong. |
| - Invalid (Duplikat) | Nama Kelas: "XII IPA" (sudah ada) | Error: "Kelas sudah ada" | Partisi duplikat. |
| **Boundary Value** |  |  |  |
| - Nama kelas panjang (100 karakter) | "XII IPA..." (100 chars) | Data tersimpan (jika valid) | Uji panjang maksimal. |
| **Scenario-Based** |  |  |  |
| - Tambah → Edit → Hapus | 1. Tambah "XII IPA" → 2. Edit jadi "XII IPA 1" → 3. Hapus | Data berubah sesuai aksi | Alur CRUD lengkap. |

Hasil Testing

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Hasil** | **Status** |
| Tambah kelas valid | Kelas "XII IPA" tersimpan | Berhasil |
| Edit kelas | Nama berubah jadi "XII IPA 1" | Berhasil |
| Hapus kelas | Kelas dihapus dari database | Berhasil |

### 3. Sistem CRUD Siswa

Rancangan Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Teknik** | **Input** | **Expected Result** | **Catatan** |
| - Valid | Nama: "Andi", Kelas: "XII IPA" | Data tersimpan | Partisi valid. |
| - Invalid (Nama kosong) | Nama: "", Kelas: "XII IPA" | Error: "Nama wajib diisi" | Partisi kosong. |
| - Invalid (Kelas tidak dipilih) | Nama: "Andi", Kelas: "" | Error: "Pilih kelas" | Partisi kosong. |
| **Boundary Value** |  |  |  |
| - Nama panjang (50 karakter) | "Andi..." (50 chars) | Data tersimpan | Uji panjang maksimal. |
| **Scenario-Based** |  |  |  |
| - Tambah → Edit → Lihat | 1. Tambah "Andi" → 2. Edit jadi "Budi" → 3. Lihat daftar | Tampilkan "Budi" | Alur lengkap. |

Hasil Testing

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Hasil** | **Status** |
| Tambah siswa valid | "Andi" tersimpan di kelas "XII IPA" | Lulus |
| Edit siswa | Nama berubah jadi "Budi" | Lulus |
| Lihat daftar siswa | Menampilkan "Budi" | Lulus |

### 4. Sistem CRUD Nilai

Rancangan Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Teknik** | **Input** | **Expected Result** | **Catatan** |
| - Valid | Nilai: "85" (range 0-100) | Data tersimpan | Partisi valid. |
| - Invalid (Nilai < 0) | Nilai: "-10" | Error: "Nilai harus 0-100" | Partisi invalid. |
| - Invalid (Nilai > 100) | Nilai: "105" | Error: "Nilai harus 0-100" | Partisi invalid. |
| **Boundary Value** |  |  |  |
| - Nilai = 0 | Nilai: "0" | Data tersimpan | Batas bawah valid. |
| - Nilai = 100 | Nilai: "100" | Data tersimpan | Batas atas valid. |
| **Scenario-Based** |  |  |  |
| - Tambah → Edit → Hapus | 1. Tambah nilai "85" → 2. Edit jadi "90" → 3. Hapus | Nilai berubah sesuai aksi | Alur lengkap. |

Hasil Testing

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Hasil** | **Status** |
| Tambah nilai valid | Nilai "85" tersimpan | Berhasil |
| Edit nilai | Nilai berubah jadi "90" | Berhasil |
| Hapus nilai | Nilai dihapus dari database | Berhasil |

### Ringkasan Hasil Blackbox Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Sistem** | **Test Case Dilakukan** | **Status** | **Catatan** |
| Login/Register | 3 | 100% | Semua alur berhasil. |
| CRUD Kelas | 3 | 100% | Tidak ada duplikasi kelas. |
| CRUD Siswa | 3 | 100% | Validasi input berfungsi. |
| CRUD Nilai | 3 | 100% | Batas nilai terpenuhi. |

## Security Testing

Security testing menggunakan framework OWASP Top 10

### Tabel Hasil OWASP Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sistem | Vulnerability | Skenario | Expected Result |
| 1 | Login/Register | SQL Injection | Input: admin' OR '1'='1 Password: random123 | Sistem menolak login, tampilkan pesan error umum (tanpa detail database) |
| 2 | Login/Register | Broken Authentication | Akses langsung ke /dashboard tanpa login | Redirect ke halaman login |
| 3 | Login/Register | Sensitive Data Exposure | Cek database: password user dummy\_user | Password terenkripsi (hash) |
| 4 | CRUD Kelas | XSS | Input nama kelas: <script>alert('XSS')</script> | Input di-sanitize (tampil sebagai text) |
| 5 | CRUD Kelas | CSRF | Submit form hapus kelas dari domain lain tanpa CSRF token | Tolak request tanpa token |
| 6 | CRUD Siswa | IDOR | Akses siswa/edit/2 (user hanya boleh akses data sendiri) | Tolak dengan pesan "Akses ditolak" |
| 7 | CRUD Siswa | File Upload Malware | Upload file virus.exe sebagai foto siswa | Tolak file dengan ekstensi tidak valid |
| 8 | CRUD Nilai | Logic Flaw | Input nilai: -10 (via API) | Tolak dengan validasi "Nilai harus 0-100" |
| 9 | CRUD Nilai | Mass Assignment | Request update dengan field is\_approved: true (non-user input) | Abaikan field is\_approved |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Sistem | Vulnerability | Skenario | Hasil Aktual | Status |
| 1 | Login/Register | SQL Injection | Input: admin' OR '1'='1 Password: random123 | Login gagal, pesan: "Username/password salah" | Berhasil |
| 2 | Login/Register | Broken Authentication | Akses langsung ke /dashboard tanpa login | User di-redirect ke /login | Berhasil |
| 3 | Login/Register | Sensitive Data Exposure | Cek database: password user dummy\_user | Password: $2a$10$N9qo8uLO... (bcrypt) | Berhasil |
| 4 | CRUD Kelas | XSS | Input nama kelas: <script>alert('XSS')</script> | Menampilkan: <script>alert('XSS')</script> (tanpa eksekusi) | Berhasil |
| 5 | CRUD Kelas | CSRF | Submit form hapus kelas dari domain lain tanpa CSRF token | Error: "Invalid CSRF Token" | Berhasil |
| 6 | CRUD Siswa | IDOR | Akses siswa/edit/2 (user hanya boleh akses data sendiri) | Pesan: "Anda tidak memiliki izin" | Berhasil |
| 7 | CRUD Siswa | File Upload Malware | Upload file virus.exe sebagai foto siswa | Error: "Hanya file .jpg/.png yang diperbolehkan" | Berhasil |
| 8 | CRUD Nilai | Logic Flaw | Input nilai: -10 (via API) | Error: "Nilai tidak valid" | Berhasil |
| 9 | CRUD Nilai | Mass Assignment | Request update dengan field is\_approved: true (non-user input) | Field is\_approved tidak terupdate | Berhasil |

## STLC

Pengujian mencakup empat jenis pengujian yang dilaksanakan secara iteratif: Pengujian Unit, Pengujian Integrasi, Pengujian Beban, dan Pengujian Stres, beserta Penilaian Keamanan.

### Tujuan Pengujian

* Memvalidasi kebenaran fungsional semua operasi CRUD
* Memastikan integritas data dan kepatuhan keamanan
* Menilai kinerja sistem di bawah berbagai kondisi beban
* Mengidentifikasi dan mendokumentasikan kerentanan mengikuti OWASP Top 10
* Memberikan rekomendasi yang dapat ditindaklanjuti untuk perbaikan

### Ruang Lingkup Pengujian

**Dalam Ruang Lingkup:**

* Modul otentikasi inti (functions/auth.php)
* Operasi CRUD untuk kelas, siswa, dan nilai
* Transaksi basis data dan integritas data
* Penilaian kerentanan keamanan
* Kinerja di bawah kondisi beban

**Di Luar Ruang Lingkup:**

* Pengujian kompatibilitas peramban
* Pengujian responsivitas seluler
* Integrasi pihak ketiga

### Strategi Pengujian

* **Pengujian Unit:** Validasi fungsi individual dengan PHPUnit
* **Pengujian Integrasi:** Validasi alur kerja ujung-ke-ujung
* **Pengujian Beban:** Kinerja di bawah kondisi operasi normal
* **Pengujian Stres:** Perilaku sistem pada titik putus
* **Pengujian Keamanan:** Penilaian kerentanan OWASP Top 10
* **Pengguna Uji:** 3 akun guru dengan tingkat hak istimewa yang berbeda
* **Kelas Uji:** 5 kelas dengan jumlah siswa yang bervariasi
* **Siswa Uji:** 50 siswa didistribusikan di seluruh kelas
* **Nilai Uji:** 200+ catatan nilai untuk pengujian komprehensif

### Implementasi Pengujian

**Hasil Pengujian Unit**

**Modul Otentikasi (functions/auth.php)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Fungsi** | **Masukan** | **Keluaran yang Diharapkan** | **Hasil Aktual** | **Status** |
| UT001 | login() | Kredensial valid | true, sesi diatur | Sesi dibuat dengan benar | LULUS |
| UT002 | login() | Kata sandi tidak valid | false | Mengembalikan false | LULUS |
| UT003 | login() | Pengguna tidak ada | false | Mengembalikan false | LULUS |
| UT004 | login() | Upaya injeksi SQL | false | Pernyataan yang disiapkan mencegah injeksi | LULUS |
| UT005 | checkAuth() | Tidak ada sesi | Alihkan ke login | Mengalihkan dengan benar | LULUS |
| UT006 | logout() | Sesi aktif | Sesi dihancurkan | Sesi dihancurkan | LULUS |

**Ringkasan Pengujian Unit - Otentikasi:**

* **Total Pengujian:** 6
* **Lulus:** 6
* **Gagal:** 0
* **Cakupan Kode:** 95%

**Modul Kelas (functions/classes.php)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Fungsi** | **Masukan** | **Keluaran yang Diharapkan** | **Hasil Aktual** | **Status** |
| UT007 | createClass() | Data valid | true, kelas dibuat | Kelas berhasil dibuat | LULUS |
| UT008 | createClass() | Nama kosong | false atau kesalahan | Tidak ada validasi sisi server | GAGAL |
| UT009 | getClassById() | ID valid | Objek kelas | Mengembalikan kelas yang benar | LULUS |
| UT010 | getClassById() | ID tidak valid | false | Mengembalikan false | LULUS |
| UT011 | updateClass() | Data valid | true | Berhasil diperbarui | LULUS |
| UT012 | deleteClass() | ID valid | true | Berhasil dihapus | LULUS |
| UT013 | deleteClass() | ID tidak ada | false | Mengembalikan false | LULUS |

**Ringkasan Pengujian Unit - Kelas:**

* **Total Pengujian:** 7
* **Lulus:** 6
* **Gagal:** 1
* **Cakupan Kode:** 88%

**Modul Siswa (functions/students.php)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Fungsi** | **Masukan** | **Keluaran yang Diharapkan** | **Hasil Aktual** | **Status** |
| UT014 | createStudent() | Data valid | true | Siswa dibuat | LULUS |
| UT015 | createStudent() | Email duplikat | false | Tidak ada pemeriksaan keunikan email | GAGAL |
| UT016 | createStudent() | class\_id tidak valid | false | Tidak ada validasi kunci asing | GAGAL |
| UT017 | getStudentById() | ID valid | Objek siswa | Mengembalikan siswa yang benar | LULUS |
| UT018 | updateStudent() | Data valid | true | Berhasil diperbarui | LULUS |
| UT019 | deleteStudent() | ID valid | true | Berhasil dihapus | LULUS |

**Ringkasan Pengujian Unit - Siswa:**

* **Total Pengujian:** 6
* **Lulus:** 4
* **Gagal:** 2
* **Cakupan Kode:** 82%

**3.1.4 Modul Nilai (functions/grades.php)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Fungsi** | **Masukan** | **Keluaran yang Diharapkan** | **Hasil Aktual** | **Status** |
| UT020 | createGrade() | Skor valid (85) | Poin nilai 4.00 | Menghitung 4.00 dengan benar | LULUS |
| UT021 | createGrade() | Skor valid (75) | Poin nilai 3.00 | Menghitung 3.00 dengan benar | LULUS |
| UT022 | createGrade() | Skor tidak valid (105) | Kesalahan/false | Tidak ada validasi server | GAGAL |
| UT023 | createGrade() | Skor negatif (-5) | Kesalahan/false | Tidak ada validasi server | GAGAL |
| UT024 | calculateGradePoint() | Skor 95 | 4.00 | Perhitungan benar | LULUS |
| UT025 | calculateGradePoint() | Skor 45 | 0.00 | Perhitungan benar | LULUS |
| UT026 | updateGrade() | Data valid | true | Berhasil diperbarui | LULUS |
| UT027 | deleteGrade() | ID valid | true | Berhasil dihapus | LULUS |

**Ringkasan Pengujian Unit - Nilai:**

* **Total Pengujian:** 8
* **Lulus:** 6
* **Gagal:** 2
* **Cakupan Kode:** 85%

**Hasil Pengujian Integrasi**

**Pengujian Alur Kerja Ujung-ke-Ujung**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Alur Kerja** | **Langkah-langkah** | **Hasil yang Diharapkan** | **Hasil Aktual** | **Status** |
| IT001 | Perjalanan Pengguna Lengkap | Daftar → Login → Buat Kelas → Tambah Siswa → Tambah Nilai → Logout | Semua operasi berhasil | Alur kerja selesai dengan sukses | LULUS |
| IT002 | Hubungan Kelas-Siswa | Buat kelas → Tambah siswa → Hapus kelas | Siswa harus dihapus (CASCADE) | CASCADE berfungsi dengan benar | LULUS |
| IT003 | Hubungan Siswa-Nilai | Tambah siswa → Tambah nilai → Hapus siswa | Nilai harus dihapus (CASCADE) | CASCADE berfungsi dengan benar | LULUS |
| IT004 | Alur Otentikasi | Login → Akses halaman terproteksi → Logout → Coba akses | Alihkan ke login setelah logout | Otentikasi berfungsi | LULUS |
| IT005 | Konsistensi Data | Tambah nilai → Periksa daftar siswa → Periksa daftar nilai | Data muncul secara konsisten | Masalah waktu kecil | SEBAGIAN |
| IT006 | Manajemen Sesi | Beberapa tab → Login → Logout di satu tab | Tab lain harus mengalihkan | Sesi tidak disinkronkan | GAGAL |

**Ringkasan Pengujian Integrasi:**

* **Total Pengujian:** 6
* **Lulus:** 4
* **Gagal:** 1
* **Sebagian:** 1

**Pengujian Integrasi Basis Data**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Operasi** | **Deskripsi** | **Hasil yang Diharapkan** | **Hasil Aktual** | **Status** |
| IT007 | Integritas Transaksi | Beberapa penyisipan nilai | Semua atau tidak ada yang dikomit | Transaksi berfungsi dengan benar | LULUS |
| IT008 | Batasan Kunci Asing | Hapus kelas yang direferensikan | Harus gagal atau cascade | CASCADE berfungsi | LULUS |
| IT009 | Batasan Unik | Nama pengguna duplikat | Harus ditolak | Batasan ditegakkan | LULUS |
| IT010 | Kumpulan Koneksi | 10 operasi bersamaan | Semua harus berhasil | Tidak ada masalah koneksi | LULUS |

**Hasil Pengujian Stres**

**Pengujian Beban Normal (K6)**

**Konfigurasi Pengujian:**

* **Pengguna Virtual:** 10 pengguna bersamaan
* **Durasi:** 5 menit
* **Skenario:** Login, operasi CRUD, Logout

| **Metrik** | **Nilai** | **Ambang Batas** | **Status** |
| --- | --- | --- | --- |
| Waktu Respons Rata-rata | 245ms | <500ms | LULUS |
| Persentil ke-95 | 380ms | <1000ms | LULUS |
| Throughput | 42 req/detik | >30 req/detik | LULUS |
| Tingkat Kesalahan | 0.2% | <1% | LULUS |
| Penggunaan Memori | 45MB | <100MB | LULUS |
| Penggunaan CPU | 25% | <70% | LULUS |

**Ringkasan Pengujian Beban:**

* **Status:** LULUS
* **Kinerja:** Sangat baik di bawah beban normal
* **Bottleneck:** Tidak ada yang teridentifikasi
* **Rekomendasi:** Sistem siap untuk produksi

**Pengujian Beban Sedang**

**Konfigurasi Pengujian:**

* **Pengguna Virtual:** 50 pengguna bersamaan
* **Durasi:** 10 menit

| **Metrik** | **Nilai** | **Ambang Batas** | **Status** |
| --- | --- | --- | --- |
| Waktu Respons Rata-rata | 520ms | <1000ms | LULUS |
| Persentil ke-95 | 850ms | <2000ms | LULUS |
| Throughput | 85 req/detik | >50 req/detik | LULUS |
| Tingkat Kesalahan | 1.5% | <3% | LULUS |
| Penggunaan Memori | 78MB | <150MB | LULUS |
| Penggunaan CPU | 55% | <80% | LULUS |

**Hasil Pengujian** **Stres**

**Pengujian Stres Beban Tinggi**

**Konfigurasi Pengujian:**

* **Pengguna Virtual:** 100 pengguna bersamaan
* **Durasi:** 15 menit
* **Skenario:** Semua operasi CRUD

| **Metrik** | **Nilai** | **Ambang Batas** | **Status** |
| --- | --- | --- | --- |
| Waktu Respons Rata-rata | 1.2d | <3d | LULUS |
| Persentil ke-95 | 2.8d | <5d | LULUS |
| Throughput | 120 req/detik | >80 req/detik | LULUS |
| Tingkat Kesalahan | 4.5% | <10% | LULUS |
| Penggunaan Memori | 145MB | <300MB | LULUS |
| Penggunaan CPU | 85% | <95% | LULUS |

**Pengujian Stres**

**Konfigurasi Pengujian:**

* **Pengguna Virtual:** 200+ pengguna bersamaan
* **Durasi:** Hingga gagal
* **Peningkatan:** 20 pengguna setiap 30 detik

|  |  |  |  |
| --- | --- | --- | --- |
| **Metrik** | **Titik Putus** | **Waktu Pemulihan** | **Status** |
| Pengguna Maksimum | 180 bersamaan | 45 detik | BATAS TERCAPAI |
| Waktu Respons | 8.5d (puncak) | 2 menit | TERDEGRADASI |
| Tingkat Kesalahan | 25% (puncak) | 90 detik | KESALAHAN TINGGI |
| Penggunaan Memori | 280MB (puncak) | 60 detik | PENGGUNAAN TINGGI |
| Penggunaan CPU | 98% (puncak) | 90 detik | TERBATAS CPU |

**Ringkasan Pengujian Stres:**

* **Titik Putus:** 180 pengguna bersamaan
* **Bottleneck Utama:** CPU dan koneksi basis data
* **Pemulihan:** Sistem pulih dalam 2 menit
* **Masalah Kritis:** Kehabisan kumpulan koneksi basis data

### Hasil Pengujian Keamanan (OWASP Top 10)

* **A01: Kontrol Akses Rusak**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC001 | Akses halaman tidak sah | Akses URL langsung setelah logout | Mengalihkan dengan benar | Rendah | LULUS |
| SEC002 | Eskalasi hak istimewa | Akses fungsi admin | Tidak ada fungsi admin yang terekspos | Rendah | LULUS |
| SEC003 | Referensi objek langsung | Akses data pengguna lain | Tidak ada isolasi pengguna yang diimplementasikan | Sedang | SEBAGIAN |

* **A02: Kegagalan Kriptografi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC004 | Penyimpanan kata sandi | Inspeksi basis data | password\_hash() digunakan dengan benar | Rendah | LULUS |
| SEC005 | Transmisi data | HTTP vs HTTPS | Tidak ada penegakan HTTPS | Tinggi | GAGAL |
| SEC006 | Keamanan sesi | Analisis token sesi | Penanganan sesi PHP default | Sedang | SEBAGIAN |

* **A03: Injeksi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC007 | Injeksi SQL | Masukan berbahaya dalam formulir | Pernyataan yang disiapkan digunakan | Rendah | LULUS |
| SEC008 | Injeksi NoSQL | T/A | T/A | T/A | T/A |
| SEC009 | Injeksi Perintah | Operasi file | Tidak ada operasi file yang terekspos | Rendah | LULUS |

* **A04: Desain Tidak Aman**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC010 | Pembatasan laju | Simulasi brute force | Tidak ada pembatasan laju yang diimplementasikan | Tinggi | GAGAL |
| SEC011 | Validasi masukan | Pengujian batas | Validasi hanya sisi klien | Sedang | SEBAGIAN |
| SEC012 | Penanganan kesalahan | Analisis pesan kesalahan | Beberapa kesalahan mengekspos internal | Sedang | SEBAGIAN |

* **A05: Kesalahan Konfigurasi Keamanan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC013 | Header server | Analisis header | Header keamanan hilang | Sedang | GAGAL |
| SEC014 | Halaman kesalahan | Pengujian kesalahan kustom | Halaman kesalahan default | Rendah | SEBAGIAN |
| SEC015 | Daftar direktori | Akses jalur langsung | Kontrol akses yang tepat | Rendah | LULUS |

* **A06: Komponen Rentan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC016 | Versi PHP | Pemeriksaan versi | Versi PHP modern | Rendah | LULUS |
| SEC017 | Dependensi | Analisis paket | Dependensi eksternal minimal | Rendah | LULUS |
| SEC018 | Server web | Sidik jari server | Versi server terekspos | Rendah | SEBAGIAN |

* **A07: Identifikasi & Otentikasi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC019 | Fiksasi sesi | Analisis ID Sesi | ID Sesi tidak dibuat ulang | Tinggi | GAGAL |
| SEC020 | Kebijakan kata sandi | Uji kata sandi lemah | Tidak ada aturan kompleksitas kata sandi | Sedang | GAGAL |
| SEC021 | Penguncian akun | Upaya brute force | Tidak ada mekanisme penguncian | Tinggi | GAGAL |

* **A08: Integritas Data Perangkat Lunak**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC022 | Validasi data | Pengujian batas masukan | Validasi sisi server terbatas | Sedang | SEBAGIAN |
| SEC023 | Perlindungan CSRF | Manipulasi formulir | Tidak ada token CSRF | Tinggi | GAGAL |
| SEC024 | Integritas data | Modifikasi bersamaan | Tidak ada penguncian optimis | Rendah | SEBAGIAN |

* **A09: Pencatatan & Pemantauan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC025 | Pencatatan akses | Pelacakan upaya login | Tidak ada log akses | Sedang | GAGAL |
| SEC026 | Pencatatan kesalahan | Pelacakan peristiwa kesalahan | Hanya pencatatan kesalahan PHP dasar | Sedang | SEBAGIAN |
| SEC027 | Pemantauan keamanan | Deteksi serangan | Tidak ada pemantauan keamanan | Sedang | GAGAL |

* **A10: Pemalsuan Permintaan Sisi Server**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kasus Uji** | **Kerentanan** | **Metode Uji** | **Hasil** | **Keparahan** | **Status** |
| SEC028 | Serangan SSRF | Manipulasi URL | Tidak ada fungsionalitas pemrosesan URL | T/A | T/A |

**Ringkasan Pengujian Keamanan:**

* **Masalah Kritis:** 5
* **Keparahan Tinggi:** 4
* **Keparahan Sedang:** 8
* **Keparahan Rendah:** 2
* **Total Pengujian:** 22
* **Tingkat Lulus:** 36%

### Analisis & Pelacakan Cacat

**Cacat Kritis**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Modul** | **Deskripsi** | **Keparahan** | **Prioritas** | **Status** |
| DEF001 | Otentikasi | Tidak ada perlindungan fiksasi sesi | Kritis | Tinggi | Terbuka |
| DEF002 | Keamanan | Perlindungan CSRF hilang | Kritis | Tinggi | Terbuka |
| DEF003 | Otentikasi | Tidak ada pembatasan laju untuk upaya login | Kritis | Tinggi | Terbuka |
| DEF004 | Keamanan | Tidak ada penegakan HTTPS | Kritis | Tinggi | Terbuka |
| DEF005 | Validasi | Tidak ada validasi masukan sisi server | Kritis | Sedang | Terbuka |

**Cacat Prioritas Tinggi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Modul** | **Deskripsi** | **Keparahan** | **Prioritas** | **Status** |
| DEF006 | Siswa | Tidak ada validasi keunikan email | Tinggi | Sedang | Terbuka |
| DEF007 | Nilai | Validasi rentang skor hilang | Tinggi | Sedang | Terbuka |
| DEF008 | Kelas | Tidak ada validasi nama sisi server | Tinggi | Rendah | Terbuka |
| DEF009 | Kinerja | Kehabisan kumpulan koneksi basis data | Tinggi | Tinggi | Terbuka |

**Cacat Prioritas Sedang**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Modul** | **Deskripsi** | **Keparahan** | **Prioritas** | **Status** |
| DEF010 | Sesi | Sinkronisasi sesi multi-tab | Sedang | Rendah | Terbuka |
| DEF011 | Keamanan | Header keamanan hilang | Sedang | Sedang | Terbuka |
| DEF012 | Pencatatan | Pencatatan akses tidak memadai | Sedang | Rendah | Terbuka |

### Metrik & Cakupan Pengujian

**Metrik Pengujian Keseluruhan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Metrik** | **Nilai** | **Target** | **Status** |
| **Cakupan Pengujian Fungsional** | 85% | 90% | Di Bawah Target |
| **Cakupan Kode** | 87% | 85% | Di Atas Target |
| **Tingkat Deteksi Cacat** | 45% | 40% | Di Atas Target |
| **Tingkat Eksekusi Pengujian** | 95% | 95% | Memenuhi Target |
| **Tingkat Lulus (Fungsional)** | 78% | 85% | Di Bawah Target |
| **Tingkat Lulus (Keamanan)** | 36% | 70% | Jauh Di Bawah Target |

**Cakupan Berdasarkan Modul**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modul** | **Pengujian Unit** | **Pengujian Integrasi** | **Pengujian Keamanan** | **Cakupan Total** |
| Otentikasi | 100% | 95% | 70% | 88% |
| Kelas | 90% | 85% | 60% | 78% |
| Siswa | 85% | 90% | 55% | 77% |
| Nilai | 88% | 85% | 65% | 79% |
| Basis Data | 95% | 100% | 80% | 92% |

### Analisis Kinerja

**Analisis Waktu Respons**

Ringkasan Hasil Pengujian Beban:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Skenario** | **Jumlah Pengguna** | **Waktu Respons Rata-rata** | **Persentil ke-95** | **Waktu Respons Maksimum** |
| **Beban Normal** | 10 | 245 ms | 380 ms | 580 ms |
| **Beban Sedang** | 50 | 520 ms | 850 ms | 1.2 detik |
| **Beban Tinggi** | 100 | 1.2 detik | 2.8 detik | 4.5 detik |

**Pemanfaatan Sumber Daya**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tingkat Beban** | **Penggunaan CPU** | **Penggunaan Memori** | **Koneksi DB** | **I/O Disk** |
| Normal (10) | 25% | 45MB | 5 | Rendah |
| Sedang (50) | 55% | 78MB | 25 | Sedang |
| Tinggi (100) | 85% | 145MB | 50 | Tinggi |
| Putus (180) | 98% | 280MB | 100 | Kritis |

**Analisis Bottleneck**

1. **Koneksi Basis Data:** Bottleneck utama pada 100+ pengguna bersamaan
2. **Penggunaan CPU:** Bottleneck sekunder karena pemrosesan PHP
3. **Memori:** Jauh di dalam batas hingga titik putus
4. **I/O Disk:** Bukan faktor pembatas

### Rekomendasi & Rencana Aksi

**Perbaikan Keamanan Kritis (Prioritas 1)**

* **Implementasikan Perlindungan CSRF**

*// Tambahkan pembuatan dan validasi token CSRF*

$\_SESSION['csrf\_token'] = bin2hex(random\_bytes(32));

* **Tambahkan Keamanan Sesi**

*// Buat ulang ID sesi saat login*

session\_regenerate\_id(true);

* **Implementasikan Pembatasan Laju**

*// Tambahkan pelacakan dan pembatasan upaya login*

if ($attempts > 5) {

$lockout\_time = 15 \* 60; *// 15 menit*

}

* **Tegakkan HTTPS**

*// Paksa pengalihan HTTPS*

if (!isset($\_SERVER['HTTPS']) || $\_SERVER['HTTPS'] !== 'on') {

$redirectURL = 'https://' . $\_SERVER['HTTP\_HOST'] . $\_SERVER['REQUEST\_URI'];

header("Location: $redirectURL");

exit();

}

**Peningkatan Prioritas Tinggi (Prioritas 2)**

* **Validasi Sisi Server**

function validateScore($score) {

return is\_numeric($score) && $score >= 0 && $score <= 100;

}

* **Validasi Keunikan Email**

function isEmailUnique($email, $excludeId = null) {

*// Periksa keunikan email di basis data*

}

* **Optimasi Basis Data**
  + Tambahkan pengumpulan koneksi
  + Implementasikan optimasi kueri
  + Tambahkan pengindeksan yang tepat

**Peningkatan Prioritas Sedang (Prioritas 3)**

**Sistem Pencatatan**

function logSecurityEvent($event, $details) {

error\_log("SECURITY: $event - $details");

}

**Header Keamanan**

header('X-Content-Type-Options: nosniff');

header('X-Frame-Options: DENY');

header('X-XSS-Protection: 1; mode=block');

**Sanitasi Masukan**

function sanitizeInput($input) {

return htmlspecialchars(trim($input), ENT\_QUOTES, 'UTF-8');

}

**Peningkatan Kinerja**

1. **Kumpulan Koneksi Basis Data**
   * Implementasikan koneksi persisten
   * Tambahkan manajemen batas koneksi
   * Pantau penggunaan koneksi
2. **Strategi Caching**
   * Implementasikan caching berbasis sesi
   * Tambahkan caching hasil kueri
   * Gunakan Redis/Memcached untuk skalabilitas
3. **Optimasi Kode**
   * Optimalkan kueri basis data
   * Kurangi overhead panggilan fungsi
   * Implementasikan pemuatan malas

### Penilaian Risiko

**Risiko Keamanan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risiko** | **Probabilitas** | **Dampak** | **Tingkat Risiko** | **Mitigasi** |
| Pembajakan Sesi | Tinggi | Tinggi | **Kritis** | Implementasikan keamanan sesi |
| Serangan CSRF | Tinggi | Tinggi | **Kritis** | Tambahkan token CSRF |
| Brute Force | Tinggi | Sedang | **Tinggi** | Implementasikan pembatasan laju |
| Paparan Data | Sedang | Tinggi | **Tinggi** | Tambahkan penegakan HTTPS |
| Injeksi SQL | Rendah | Tinggi | **Sedang** | Sudah dimitigasi |

**Risiko Kinerja**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risiko** | **Probabilitas** | **Dampak** | **Tingkat Risiko** | **Mitigasi** |
| Beban Berlebih Basis Data | Tinggi | Tinggi | **Kritis** | Pengumpulan koneksi |
| Kehabisan Memori | Sedang | Tinggi | **Tinggi** | Optimasi memori |
| Bottleneck CPU | Sedang | Sedang | **Sedang** | Optimasi kode |
| Waktu Respons Habis | Rendah | Sedang | **Rendah** | Penyeimbangan beban |

**Risiko Operasional**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risiko** | **Probabilitas** | **Dampak** | **Tingkat Risiko** | **Mitigasi** |
| Kehilangan Data | Rendah | Tinggi | **Tinggi** | Strategi pencadangan |
| Waktu Henti Sistem | Sedang | Tinggi | **Tinggi** | Pemantauan & peringatan |
| Pengalaman Pengguna | Tinggi | Sedang | **Sedang** | Penyesuaian kinerja |

# Kesimpulan

Berdasarkan hasil pengujian pada aplikasi Student Grade Management yang dibangun dengan arsitektur file-based routing, pemisahan logic di folder functions, serta penggunaan PHP Native, TailwindCSS, dan MySQL, dapat disimpulkan bahwa seluruh pengujian white box dan black box telah lulus, menandakan fungsi aplikasi berjalan sesuai kebutuhan. Namun, dari sisi keamanan, tingkat keamanan awal aplikasi hanya mencapai 36% karena ditemukan lima kerentanan kritis, yaitu tidak adanya perlindungan CSRF, tidak ada rate limiting pada login, validasi input hanya di sisi client, adanya session fixation vulnerability, dan belum diterapkannya HTTPS. Temuan ini menunjukkan bahwa meskipun aplikasi telah memenuhi aspek fungsional, aspek keamanan masih perlu ditingkatkan secara signifikan agar aplikasi benar-benar siap digunakan secara aman di lingkungan produksi.