

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Pengembangan Metode TDD

Metode pengembangan *Test-Driven Development (TDD)* adalah salah satu metode pengembangan *software Agile* yang dikenalkan oleh *eXtreme Programming (XP)*. Metode TDD ini dapat menghasilkan perangkat lunak yang terbebas dari *bug* atau *error*. Hal ini dikarenakan TDD mengharuskan pengembang untuk berhasil menerapkan satu fitur / fungsi terlebih dahulu, kemudian apabila berhasil melewati berbagai uji coba akan dilanjutkan ke fitur / fungsi selanjutnya.

Pada TDD, pengembang akan melewati tiga tahap utama yang disebut sebagai tahap *red-green-refactoring*. Pada tahap *red*, pengembang akan menuliskan kode untuk meloloskan sebuah pengujian untuk suatu fitur / fungsi hingga berhasil. Apabila berhasil, maka pengembang akan masuk ke tahap *green* yaitu dimana kode yang dituliskan sudah berhasil melewati uji coba suatu fitur / fungsi. Setelahnya, pengembang akan masuk ke tahap *refactoring* dimana pengembang harus membersihkan kode yang telah dikembangkan dengan membuang fungsi atau baris kode yang tidak dipakai sehingga menghasilkan perangkat lunak dengan *clean code*. Setelah tahap *refactoring*, pengembang bisa langsung masuk lagi ke tahap *red* dan memulai siklus baru untuk fitur / fungsi baru pada perangkat lunak yang dikembangkan.

Aplikasi web yang dikembangkan dapat dicapai dengan memenuhi kebutuhan fungsional dan kebutuhan non-fungsional yang didapat setelah mengumpulkan *user story* dari calon pengguna aplikasi yang dikembangkan sebagaimana dijabarkan pada bab sebelumnya. Kemudian dengan ini didapatkan rancangan perangkat lunak berupa *use case diagram*, *activity diagram* dan *mockup* sebagai rancangan antarmuka.

Penelitian ini dilakukan dengan maksud menghasilkan aplikasi yang dapat mempermudah para pengajar dalam menilai jawaban esai pelajar, mengetahui banyaknya jumlah pelajar yang perlu diperiksa oleh satu guru yang dapat memakan

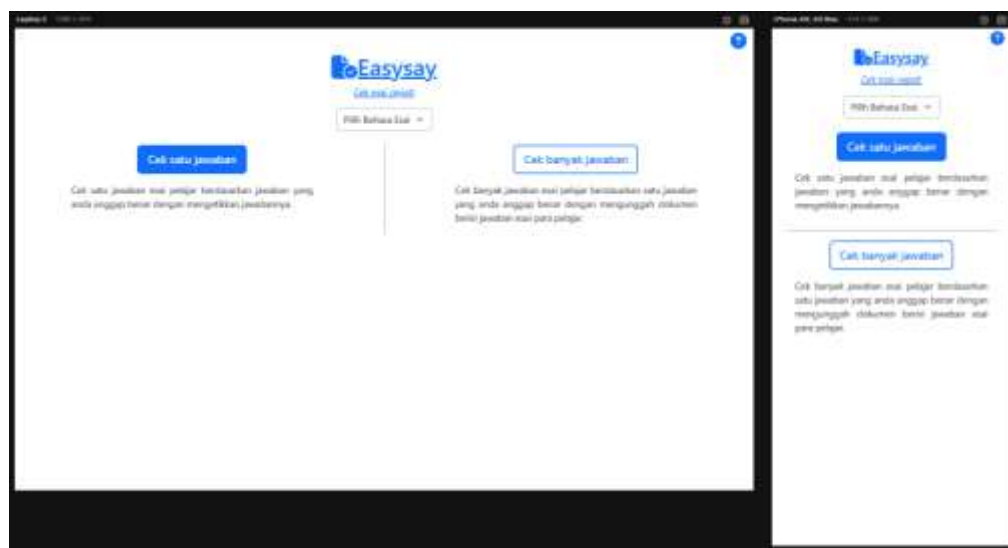
waktu lama. Aplikasi yang dikembangkan penulis memiliki *frontend*, *backend*, dan 2 *model machine learning* yang dipakai untuk melakukan penilaian esai pada *backend*. Dengan ini, pengembangan perangkat lunak yang dilakukan dapat dilihat secara rinci pada sub-bab berikut ini.

4.1.1 Pengembangan Frontend

Frontend yang dikembangkan digunakan sebagai antarmuka agar pengguna bisa memilih bahasa yang hendak digunakan, memilih cara penilaian, memasukkan kunci jawaban dan jawaban yang ingin diperiksa, dan memulai proses penilaian. Setelah pengguna memulai proses tombol penilaian, *frontend* akan mengirimkan data yang diberikan oleh pengguna ke *backend* untuk melakukan proses penilaian. Gambar yang dilampirkan dibawah ini adalah hasil akhir dari *frontend* aplikasi penilai esai otomatis berdasarkan tampilan *desktop* dan *mobile*.

4.1.1.1 Frontend Halaman Home

Pada Gambar 4.1 dapat dilihat *frontend* dari halaman utama aplikasi penilaian esai otomatis. Pada halaman ini, pengguna dapat memilih bahasa yang ingin digunakan untuk memeriksa esai, memilih metode penilaian esai, ataupun mengakses halaman bantuan.



Gambar 4.1 *Frontend* Halaman Home



Gambar 4.3 *Frontend* Halaman Hasil Metode Penilaian Tunggal

4.1.1.4 Frontend Halaman Metode Penilaian Jamak

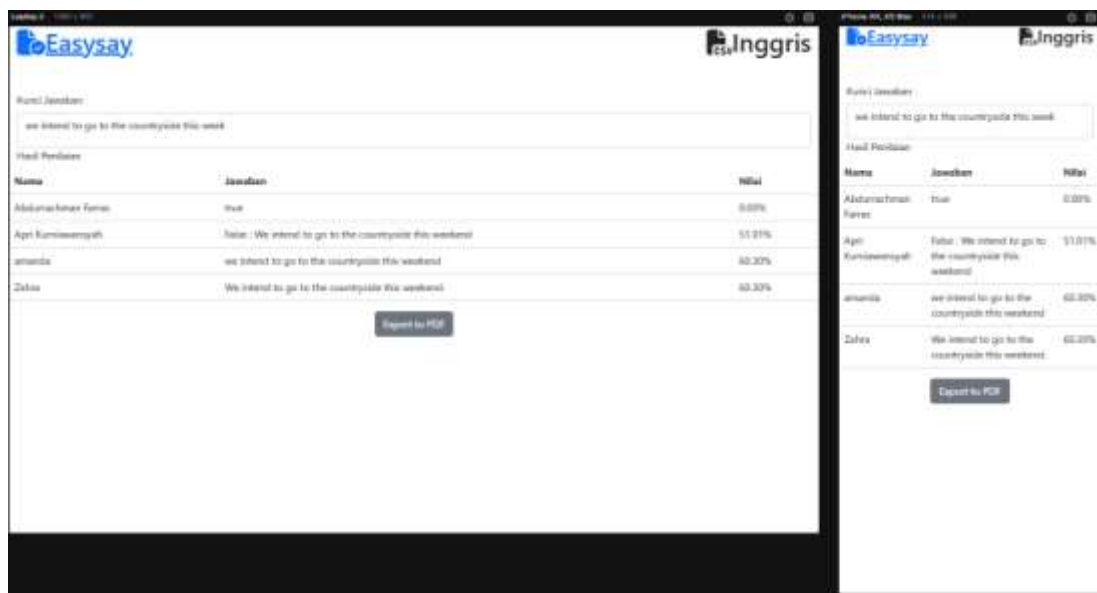
Apabila pengguna ingin melakukan penilaian jamak, pengguna dapat memilih “Cek banyak jawaban” pada halaman *home*, dan pengguna akan diarahkan ke halaman metode penilaian jamak seperti pada Gambar 4.4. Pada halaman ini, pengguna dapat mengisi jawaban pengajar, kemudian mengunggah file CSV yang berisi nama dan jawaban pelajar, kemudian menspesifikasikannya pada kolom yang disediakan. Setelahnya, pengguna dapat menekan tombol “periksa”.



Gambar 4.4 *Frontend* Halaman Metode Penilaian Jamak

4.1.1.5 Frontend Halaman Hasil Metode Penilaian Jamak

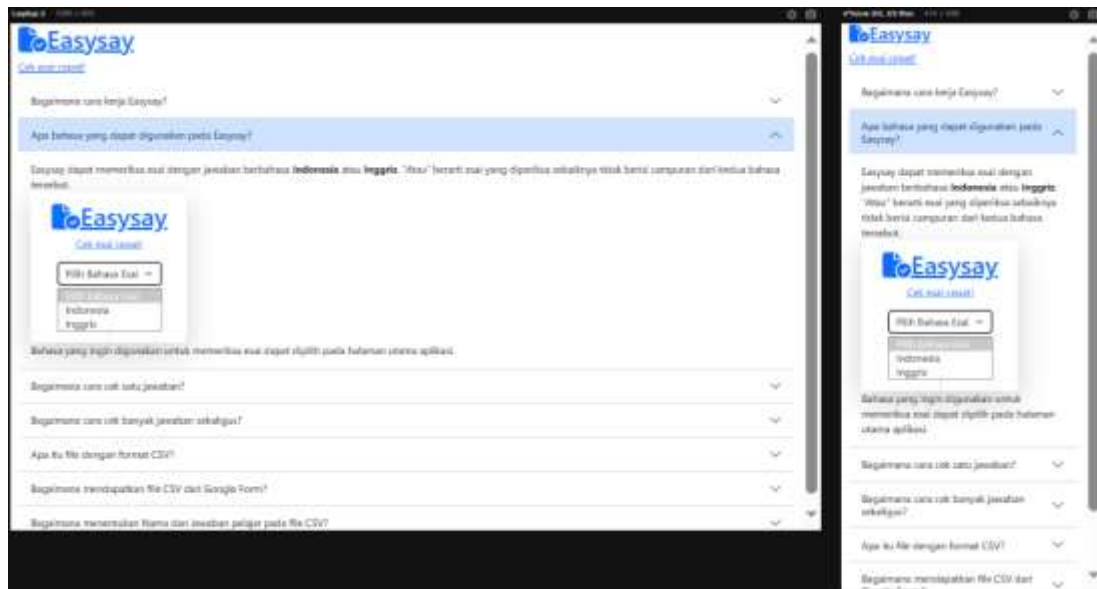
Setelah menekan tombol “periksa”, pengguna akan diarahkan ke halaman hasil metode penilaian jamak seperti pada Gambar 4.5. Di halaman ini, pengguna dapat melihat tabel pasangan nama, jawaban, dan skor esai pelajar serta mengunduhnya dalam format PDF dengan menekan tombol “Export to PDF”.



Gambar 4.5 *Frontend* Halaman Hasil Metode Penilaian Jamak

4.1.1.6 Frontend Halaman Bantuan

Apabila pengguna mengalami kesulitan dalam menggunakan aplikasi yang telah dibangun oleh penulis, pengguna dapat menekan tombol “?” yang ada di pojok kanan atas pada halaman *home* seperti pada Gambar 4.1, dan akan diarahkan ke halaman bantuan seperti pada Gambar 4.6. Pada halaman bantuan ini, pengguna dapat menemukan pertanyaan, jawaban, dan tutorial cara menggunakan aplikasi penilai esai otomatis ini.



Gambar 4.6 *Frontend* Halaman Bantuan

4.1.2 Pengembangan Backend

Backend akan berperan untuk menerima dan memproses permintaan dari *frontend*, baik permintaan rute (*routing*) maupun memproses penilaian. Saat memproses rute, *backend* akan mengarahkan pengguna ke halaman yang dituju oleh rute tersebut baik dari permintaan *get* atau *post*. Khususnya pada saat memeriksa jawaban esai, *backend* akan menggunakan model *machine learning* melalui metode *child process*.

4.1.2.1 Rute Untuk Halaman Home

Pada saat mengakses aplikasi, pengguna akan diarahkan ke halaman utama / *home*. *Backend* disini berperan untuk menerima permintaan *get* ke rute *"/"*. Aksi yang dilakukan rute ini hanyalah menampilkan halaman home kepada pengguna sebagaimana pada Kode 4.1.

```
app.get("/", (req, res) => {
  res.render("home/home");
});
```

Kode 4.1 Kode Program Rute /

4.1.2.2 Rute Untuk Mengakses Halaman Bantuan

Apabila pengguna menekan tombol bantuan, *backend* akan menerima permintaan *get* ke rute “/help” sebagaimana dilihat pada Kode 4.2. Rute ini hanya akan menampilkan halaman bantuan.

```
app.get("/", (req, res) => {
  res.render("home/home");
});
```

Kode 4.2 Kode Program Rute /help

4.1.2.3 Rute Untuk Menentukan Pilihan

Pada halaman *home*, pengguna dapat menentukan bahasa esai yang akan digunakan untuk melakukan penilaian dan metode penilaian esainya melalui permintaan *post* ke rute “/option”. Pada Kode 4.3, *backend* akan memberikan pengguna halaman sesuai dengan permintaannya pada saat memilih di halaman *home*.

```
app.route("/option").post((req, res) => {
  const obj = {
  };
  if (obj.language === "Indonesia") {
  } else if (obj.language === "Inggris") {
  } else {
    res.redirect("/");
  }
});
```

Kode 4.3 Kode Program Rute /option

4.1.2.4 Rute Untuk Mengakses Halaman Penilaian Tunggal Bahasa Inggris

Apabila pengguna menginginkan untuk memeriksa esai dengan bahasa Inggris dan metode penilaian tunggal, maka pengguna akan mengunggah jawaban pengajar dan pelajar. Pada tahap ini, *backend* mendapatkan permintaan *post* pada rute “/ens” dan menjalankan model *machine learning* bahasa Inggris untuk melakukan penilaian tersebut. Setelahnya, *backend* akan menyajikan hasil penilaian kepada pengguna. Rinci dari rute ini dapat dilihat pada Kode 4.4.

```
app.post("/ens", (req, res) => {
  const keyAnswer = req.body.teacherAnswer;
  const answer = req.body.studentAnswer;
  const language = req.body.language;
```

```

const docid = req.body.docid;

let Dataset =
`keyAnswer,studentAnswer\n"${keyAnswer}","${answer}"`;

let scoreLoad;

fs.writeFileSync(docid + ".csv", Dataset, "utf8");

let options = {
};

PythonShell.run("models/english/aprilModel.py", options)
  .then((messages) => {
  })
  .then(() => {
  })
  .then(() => {
  })
  .catch((err) => {
  });
});

```

Kode 4.4 Kode Program Rute /ens

4.1.2.5 Rute Untuk Mengakses Halaman Penilaian Jamak Bahasa Inggris

Apabila pengguna menginginkan untuk memeriksa esai dengan bahasa Inggris dan metode penilaian jamak, maka pengguna akan mengunggah jawaban pengajar, berkas CSV jawaban pelajar, dan nama kolom untuk nama dan jawaban. Pada tahap ini, *backend* mendapatkan permintaan *post* pada rute “/enb” dan menjalankan model *machine learning* bahasa Inggris untuk melakukan penilaian tersebut. Setelahnya, *backend* akan menyajikan hasil penilaian kepada pengguna. Rinci dari rute ini dapat dilihat pada Kode 4.5.

```

app.post("/enb", upload.single("studentAnswers"), (req, res) => {
  const keyAnswer = req.body.teacherAnswer;
  const studentNames = req.body.nameColumns;
  const answerColumns = req.body.answerColumns;
  const language = req.body.language;
  let docid = bulkDocumentID;
  let scoreLoad;

  CSVToJSON()
    .fromFile("./" + docid + ".csv")
    .then((source) => {
    });
});

```

Kode 4.5 Kode Program Rute /enb

4.1.2.6 Rute Untuk Mengakses Halaman Penilaian Tunggal Bahasa Indonesia

Apabila pengguna menginginkan untuk memeriksa esai dengan bahasa Indonesia dan metode penilaian tunggal, maka pengguna akan mengunggah jawaban pengajar dan pelajar. Pada tahap ini, *backend* mendapatkan permintaan *post* pada rute “/ids” dan menjalankan model *machine learning* bahasa Indonesia untuk melakukan penilaian tersebut. Setelahnya, *backend* akan menyajikan hasil penilaian kepada pengguna. Rinci dari rute ini dapat dilihat pada Kode 4.6.

```
app.post("/ids", (req, res) => {
  const { language, teacherAnswer, studentAnswer, docid } =
    req.body;

  let option = {
  };

  PythonShell.run("./models/indonesia/ASAG.py",
    option).then((messages) => {
    });
});
```

Kode 4.6 Kode Program Rute /ids

4.1.2.7 Rute Untuk Mengakses Halaman Penilaian Jamak Bahasa Indonesia

Apabila pengguna menginginkan untuk memeriksa esai dengan bahasa Indonesia dan metode penilaian jamak, maka pengguna akan mengunggah jawaban pengajar, berkas CSV jawaban pelajar, dan nama kolom untuk nama dan jawaban. Pada tahap ini, *backend* mendapatkan permintaan *post* pada rute “/idbulk” dan menjalankan model *machine learning* bahasa Indonesia untuk melakukan penilaian tersebut. Setelahnya, *backend* akan menyajikan hasil penilaian kepada pengguna. Rinci dari rute ini dapat dilihat pada Kode 4.7.

```
app.post("/idbulk", upload.single("studentAnswers"), async (req,
res) => {
  const { teacherAnswer, nameColumns, answerColumns, language } =
    req.body;
  let docid = bulkDocumentID;

  let idbulk1 = fs.readFileSync("public/pages/bulkID1.html", "utf-
8");
  let idbulk2 = fs.readFileSync("public/pages/bulkID2.html", "utf-
8");
  let idbulk3 = fs.readFileSync("public/pages/bulkID3.html", "utf-
8");
```

```

res.write(idbulk1);
res.write("<p>");
res.write(teacherAnswer);
res.write("</p>");
res.write(idbulk2);

const sources = await CSVToJSON().fromFile("./" + docid + ".csv");

if (
  sources[0][nameColumns] == undefined &&
  sources[0][answerColumns] == undefined
) {
} else {
}
});

```

Kode 4.7 Kode Program Rute /idbulk

4.1.2.8 Konversi Hasil Penilaian Menjadi PDF

Khusus untuk fungsi penilaian jamak dimana pengguna dapat melihat banyak hasil penilaian esai pelajar, peneliti membuat fitur untuk dapat menyimpan hasil penilaian dengan bentuk PDF.

```

document.getElementById("print").addEventListener("click", () => {
  document.getElementById("print").style.opacity = 0;
  window.print();
  document.getElementById("print").style.opacity = 1;
});

```

Kode 4.8 Kode Konversi Hasil Penilaian Menjadi PDF

Pada Kode 4.8 tersebut, *JavaScript* dari *frontend* hasil penilaian dapat mendeteksi saat pengguna menekan tombol “Export to PDF” dengan Id “print”, untuk menjalankan perintah cetak halaman. Namun sebelumnya, akan disembunyikan terlebih dahulu tombol “Export to PDF” yang diklik pengguna agar tidak muncul pada hasil penilaian yang telah dicetak, kemudian setelahnya akan ditampilkan lagi.

4.1.2.9 Konversi Berkas CSV Menjadi JSON

Sebelum berkas CSV unggahan pengguna dapat digunakan, berkas CSV tersebut perlu diubah menjadi JSON untuk dapat digunakan isinya. Hal ini dilakukan agar *backend* dapat menambahkan kolom baru pada berkas CSV yang ingin diperiksa, khususnya untuk menentukan kolom jawaban dan nama pelajar.

```

CSVToJSON()
  .fromFile("./" + docid + ".csv")
  .then((source) => {
    for (let i = 0; i < source.length; i++) {
      source[i].keyAnswer = keyAnswer;
      source[i].studentAnswer = source[i][answerColumns];
      source[i].studentName = source[i][studentNames];
    }
    const csv = JSONToCSV(source);
    fs.writeFileSync(docid + ".csv", csv);
  })

```

Kode 4.9 Kode Konversi Berkas CSV ke JSON

Kode program pada Kode 4.9 diatas menunjukkan konversi berkas CSV menjadi JSON, dimana konten akan ditambah dengan informasi jawaban pelajar, jawaban pelajar, dan nama pelajar. Setelah dilakukan hal tersebut, JSON akan dikonversi kembali menjadi berkas CSV dengan informasi tambahan yang cocok untuk dilakukan pemeriksaan pada model *machine learning*.

4.1.3 Child-Process Model Machine Learning

Setelah *backend* menerima jawaban pengajar dan jawaban pelajar untuk diperiksa, *backend* akan mengirimkan jawaban pengajar dan jawaban pelajar tersebut ke model *machine learning* untuk diperiksa melalui metode *child process* sesuai dengan bahasa yang pengguna pilih sebelumnya pada *frontend*. Model *machine learning* kemudian akan melakukan penilaian esai otomatis dan akan menghasilkan *output* yang akan dikembalikan kepada *backend*. Setelah *backend* menerima hasil penilaian dari model *machine learning*, maka akan dikirimkan *frontend* dengan hasil penilaian untuk dapat dilihat oleh pengguna.

4.1.3.1 Model Machine Learning Bahasa Inggris

Model *machine learning* bahasa Inggris menerima nama CSV berisi jawaban pengajar dan jawaban pelajar dari *backend* melalui argumen sistem. Model bahasa Inggris memeriksa dokumen CSV yang dibuat oleh *backend* untuk diproses seperti pada Kode 4.10.

```

full_df = pd.read_csv(f"{sys.argv[1]}.csv")
full_df

```

Kode 4.10 Model Bahasa Inggris Menerima Nama Berkas CSV

Apabila model telah selesai melakukan penilaian, maka model akan mengirimkan kembali hasil penilaian dalam bentuk JSON ke backend seperti pada Kode 4.11.

```
print(full_df.to_json())
```

Kode 4.11 Model Bahasa Inggris Mengirim Hasil Penilaian

4.1.3.2 Model Machine Learning Bahasa Indonesia

Model *machine learning* bahasa Indonesia menerima jawaban pengajar dan jawaban pelajar berupa teks langsung dari *backend* melalui argumen sistem. Hal ini dapat dilihat pada Kode 4.12.

```
# print("Masukkan Kunci Jawaban:")
kj = sys.argv[1]
# print("Masukkan Jawaban Siswa:")
jawaban = sys.argv[2]
```

Kode 4.12 Model Bahasa Indonesia Menerima Teks Jawaban

Setelah melakukan penilaian, model akan menuliskan berkas teks berisi hasil penilaian dan sebuah *output* “done” sebagai penanda bahwa proses penilaian telah selesai untuk *backend*. Hal ini dapat dilihat pada Kode 4.13 berikut ini.

```
newfile.write(f"{sctScore}\n")
newfile.close()
print("done")
```

Kode 4.13 Model Bahasa Indonesia Selesai Melakukan Penilaian

4.1.4 Hasil Scenario Testing

Scenario testing adalah uji coba yang dilakukan yang termasuk dalam cakupan metode pengembangan yang digunakan, yakni *Test-Driven Development (TDD)*. *Scenario testing* ini dilakukan untuk memastikan bahwa fitur yang dikembangkan bekerja sesuai dengan yang diinginkan. Pada kasus ini, *scenario testing* dilakukan untuk memastikan bahwa hasil keluaran pada saat melakukan penilaian esai otomatis pada model *machine learning* sama dengan hasil keluaran pada aplikasi pada masukkan spesifik. Hal ini membuktikan bahwa selama proses implementasi model *machine learning*, keaslian model tetap terjaga dan tidak terjadi adanya perubahan atas pengaruh implementasi pada aplikasi web yang dikembangkan pada penelitian

ini. Dengan itu, pada Tabel 4.1 berikut dapat diperhatikan hasil *scenario testing* yang dilakukan pada pengembangan aplikasi terkait.

Tabel 4.1 Hasil Scenario Testing

Skenario	Masukkan Jawaban Pengajar	Masukkan Jawaban Pelajar	Ekspektasi Keluaran	Keluaran	Sukses
SC1	Teks untuk uji coba jawaban tunggal “The testing stage can influence both the coding stage (phase 5) and the solution refinement stage (phase 7)”	Teks untuk uji coba tunggal “The implementation phase and the maintenance phase are effected”	1.128299 dari 5.0	22%	Ya
SC2	Teks untuk uji coba masal “To sumulate the behaviour of portions of the desired software product.”	Dokumen CSV berisi jawaban pelajar (“dataset.csv”)	Hasil Penilaian dari 0.0 sampai dengan 5.0	Persentase Hasil Penilaian	Ya
SC3	Teks untuk uji coba tunggal “Fungsi karbohidrat adalah sebagai pemasok energi, dapat memperlancar proses pada pencernaan, memberikan efek kenyang dengan kandungan selulosa-nya dan penyeimbang	Teks untuk uji coba tunggal “Karbohidrat mempunyai peran penting untuk proses metabolisme atau proses yang berfungsi sebagai penyeimbang asam dan basa dalam tubuh. Karbohidrat dapat mencegah terjadinya ketidaksempurnaan proses oksidasi lemak. Fungsi karbohidrat yang utama adalah	55 dari 100	55%	Ya

Skenario	Masukkan Jawaban Pengajar	Masukkan Jawaban Pelajar	Ekspektasi Keluaran	Keluaran	Sukses
	asam dan basa dalam tubuh “	sebagai pemasok energi. Makanan berkarbohidrat dan serat tinggi dapat membantu memperlancar proses pada pencernaan.”			

Pada Tabel 4.1, dapat dilihat bahwa uji coba skenario berhasil untuk memenuhi ekspektasi awal. Hasil ekspektasi keluaran adalah hasil penilaian yang dilakukan pada model *machine learning* langsung, sedangkan keluaran adalah hasil penilaian yang dilakukan melalui implementasi aplikasi saat tahap pembuatan fungsi. Ekspektasi yang dipenuhi merupakan sinkronnya hasil keluaran penilaian esai melalui aplikasi maupun langsung melalui model *machine learning* yang digunakan.

Pada SC1, peneliti memberikan masukkan dua teks jawaban pengajar dan jawaban sebagaimana tertera pada Tabel 4.1. Model *machine learning* bahasa Inggris kemudian mengaplikasikan *Term Frequency – Inverse Document Frequency* (TF-IDF) untuk menghasilkan bobot nilai dari setiap kata pada kedua teks tersebut. Setelahnya, model akan mengaplikasikan *Cosine Similarity* dengan nilai dikalikan dengan 5 pada kedua teks yang telah diterapkan TF-IDF sebelumnya, yang berarti hasil penilaian berupa angka dari 0.0 hingga 5.0. Dengan ini, hasil menunjukkan 1.128299 pada model dan 22% pada aplikasi, dengan mengetahui bahwa model *machine learning* melakukan penilaian dari rentang angka 0.0 s/d 5.0, maka keluaran hasil penilaian identik dikarenakan 1.128299 merupakan 22% dari 5.0. Dengan ini dapat dinyatakan bahwa tidak ada keaslian yang berubah saat mengimplemetasikan model *machine learning* bahasa Inggris pada pengembangan aplikasi untuk metode penilaian tunggal.

Kemudian pada SC2 peneliti juga melakukan uji coba pada model *machine learning* berbahasa Inggris, namun kali ini dengan metode penilaian jamak. Ekspektasi bukanlah nilai satu jawaban, namun nilai banyak jawaban hasil penilaian model

tersebut berupa JSON. Dan pada hasil pegujian skenario yang didapat, aplikasi berhasil melakukan penilaian dengan metode jamak menggunakan berkas CSV bawaan dari model *machine learning* bahasa Inggris yang digunakan.

Selanjutnya pada SC3 juga dilakuka uji coba terhadap model *machine learning* penilai esai otomatis berbahasa Indonesia. Peneliti memberi masukan jawaban pengajar dan jawaban pelajar untuk diperiksa, kemudian membandingkan keluaran hasil jawabannya untuk dibandingkan. Hasil percobaan ini berhasil karena keluaran dari pengecekan kedua paragraf tersebut identik. Nilai 55 dari model *machine learning* didapat dengan mengaplikasikan teknik TF-IDF dan *Cosine Similarity*, namun dengan skor dikalikan dengan 100 yang artinya hasil penilaian berada dari rentang 0 hingga 100. Dengan hasil 55% dari aplikasi dan 55 dari model *machine learning*, dapat dinyatakan bahwa hasil implementasi model pada aplikasi tidak memiliki perubahan yang dapat mempengaruhi hasil penilaian esai singkat.

Untuk model *machine learning* bahasa Indonesia, tidak dilakukan percobaan skenario penilaian jamak karena pada dasarnya model yang digunakan hanya dapat menilai secara tunggal. Namun pada aplikasi yang dikembangkan, dapat dilakukan penilaian secara jamak dengan melakukan penilaian tunggal secara berulang.

4.2 Hasil Pre-Deployment Testing (Integration Testing)

Untuk memastikan bahwa aplikasi yang dikembangkan oleh penelitian ini berfungsi semestinya sebelum diluncurkan untuk pemakaian umum, perlu adanya tolak ukur untuk memastikan bahwa aplikasi akan bekerja sesuai dengan ekspektasi awal. Dengan itu, pada bab 3 telah dibuat uji coba yang harus diloloskan saat mengembangkan aplikasi ini. Berikut merupakan hasil uji coba tersebut.

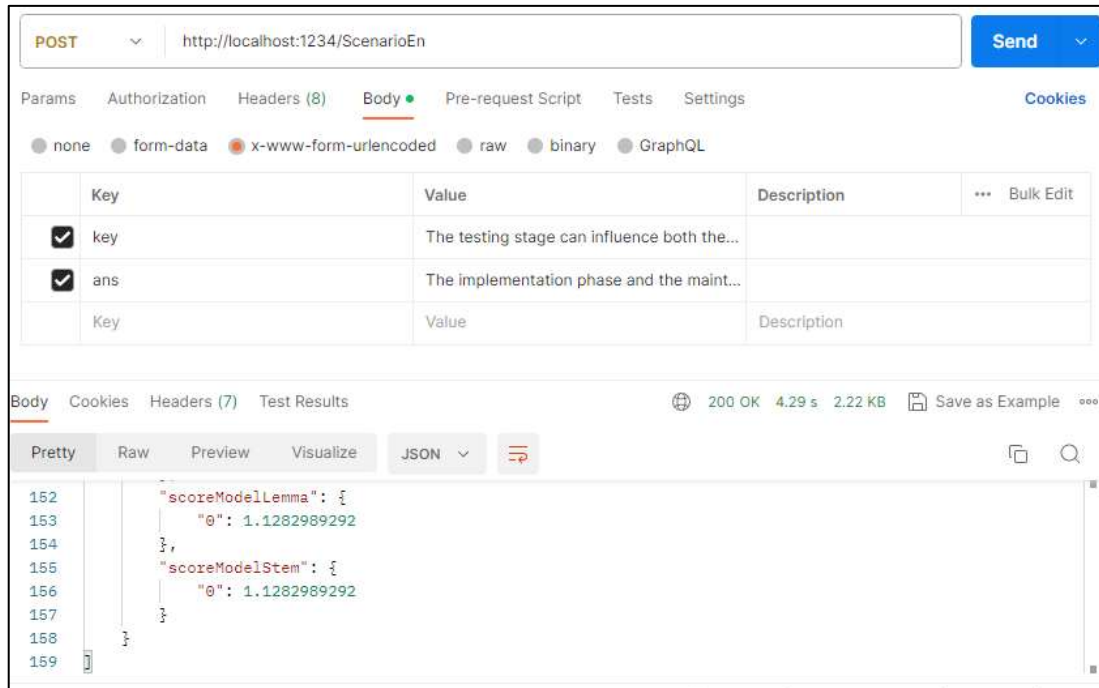
Diluar cakupan metode pengembangan *Test-Driven Development (TDD)*, dilakukan juga *Integration Testing* untuk memastikan aplikasi dapat berjalan dalam kondisi umum dengan hasil keluaran yang mirip seperti *scenario testing* namun tidak memiliki masukan dan keluaran spesifik. Pada Tabel 4.2 dibawah ini dapat dilihat hasil pengujian unit yang telah dilakukan beserta hasilnya.

Tabel 4.2 Hasil Integration Testing

ID	Metode	Input 1	Input 2	Output	Sukses
IT1	ScenarioEn(req, res)	Teks jawaban pengajar berbahasa Inggris.	Teks jawaban pelajar berbahasa Inggris.	JSON hasil penilaian tunggal berbahasa Inggris	Ya
IT2	ScenarioEnTab(req, res)	To sumulate the behaviour of portions of the desired software product.	dataset.csv	JSON hasil penilaian jamak berbahasa Inggris.	Ya
IT3	ScenarioId(req, res)	Teks jawaban pengajar berbahasa Indonesia.	Teks jawaban pelajar berbahasa Indonesia.	Hasil penilaian tunggal berbahasa Indonesia	Ya

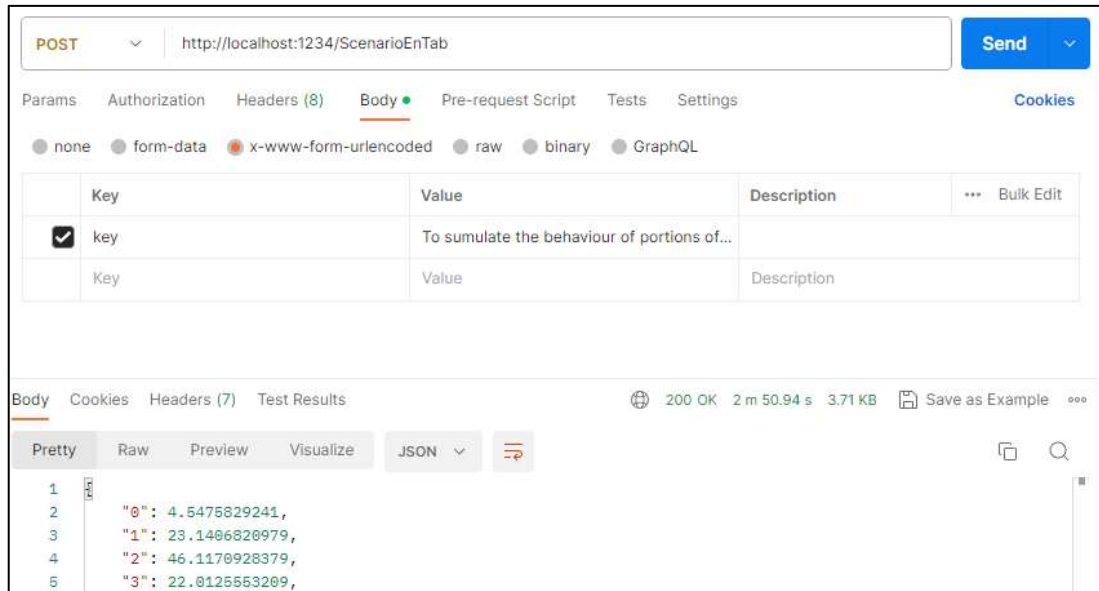
Integration testing pada Tabel 4.2 diatas dilakukan menggunakan aplikasi *Postman*, yaitu aplikasi yang membantu untuk melakukan permintaan API. Pada tahap pengujian, *integration testing* dilakukan dengan melakukan permintaan *post* pada API sebagaimana dispesifikasikan pada Tabel 4.2.

Pada IT1, aplikasi *Postman* melakukan permintaan *post* ke API “/ScenarioEn” dengan *body* menggunakan format *x-www-form-urlencoded* dengan *key* “*key*” dan “*ans*” merupakan teks pada IT1 *scenario testing* sebelumnya. API kemudian akan mencari “*key*” dan “*ans*” dan mendapatkan isi dari kedua *key* tersebut untuk melakukan penilaian dan mengirimkannya kembali kepada aplikasi *Postman* seperti pada Gambar 4.7 berikut ini.



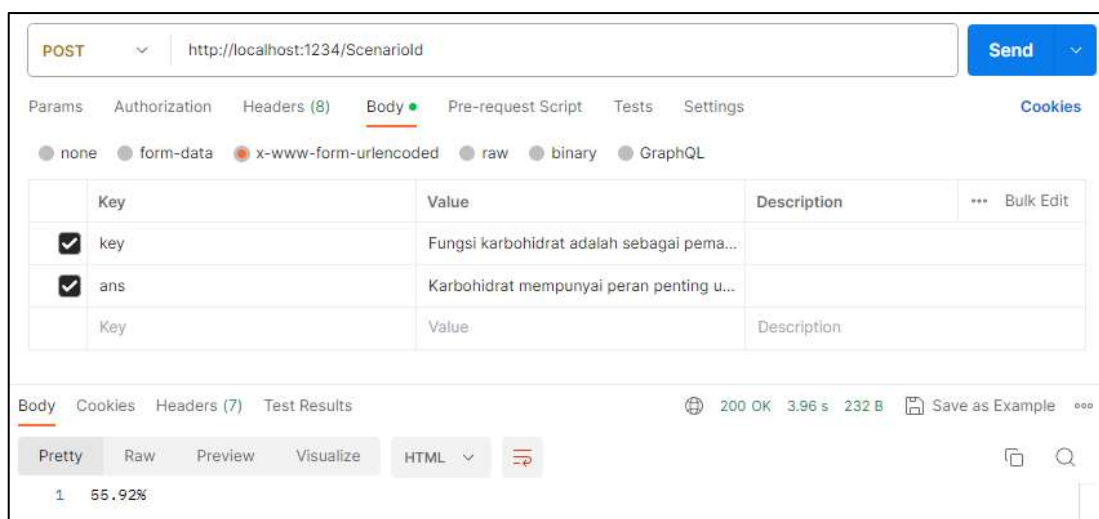
Gambar 4.7 Hasil IT1

Pada IT2, aplikasi *Postman* melakukan permintaan *post* ke API `"/ScenarioEnTab"` dengan *body* menggunakan format *x-www-form-urlencoded* dengan *key* `"key"` berisi teks berdasarkan IT2 *scenario testing* sebelumnya. Pada *integration test* ini, API melakukan penilaian terhadap teks yang didapat dari `"key"` untuk melakukan penilaian dengan berkas CSV bawaan dari model *machine learning* seperti yang telah disebut sebelumnya pada *scenario testing*. Setelah berhasil, API akan mengembalikan hasil penilaian esai ke aplikasi *Postman* sebagai respon sebagaimana ditunjukkan pada Gambar 4.8.



Gambar 4.8 Hasil IT2

Kemudian pada IT3, aplikasi *Postman* melakukan permintaan *post* ke API `"/ScenarioId"` dengan *body* menggunakan format *x-www-form-urlencoded* dengan *key* `"key"` dan `"ans"` merupakan teks pada IT3 *scenario testing* sebelumnya. API kemudian akan mencari `"key"` dan `"ans"` dan mendapatkan isi dari kedua *key* tersebut untuk melakukan penilaian dan mengirimkannya kembali kepada aplikasi *Postman*. Gambar 4.9 berikut merupakan contoh hasil pengujian IT3 yang dimaksud.



Gambar 4.9 Hasil IT3

Dengan berhasilnya *integration testing*, maka *scenario testing* juga berhasil dilakukan karena keduanya merupakan uji coba yang sinkron untuk suatu kasus umum dan spesifik. Maka dari itu, pengujian aplikasi penilaian esai otomatis ini dinyatakan berhasil memenuhi syarat dan ekspektasi awal.

4.3 Hasil Peluncuran

Setelah melakukan pengembangan aplikasi untuk penelitian ini, aplikasi yang telah dikembangkan diluncurkan untuk dapat digunakan oleh banyak pengguna. Aplikasi diluncurkan dengan nama “Easysay”.

Aplikasi ini diluncurkan pada server IdCloudHost menggunakan *virtual machine* dengan *Operating System* Ubuntu, 2 VCPU, 2048 MB RAM, dan memori sebesar 20 GB. *Virtual Machine* pada IdCloudHost juga diatur untuk dapat menjalankan *Node.js* dan *Python* agar aplikasi bisa dijalankan sebagaimana di komputer pengembang.

Apabila pada pengujian setelah peluncuran yang dilakukan, didapatkan *bug* atau *error* yang perlu diperbaiki, akan dilakukan peluncuran ulang terhadap aplikasi Easysay dengan spesifikasi server yang sama.

4.4 Hasil Post-Deployment Testing

Setelah melakukan peluncuran aplikasi, peneliti melakukan pengujian lebih lanjut kepada narasumber awal dan pengguna lainnya untuk memastikan bahwa aplikasi yang dikembangkan memenuhi kebutuhan pengguna dan dapat berjalan dengan baik. Berikut merupakan pengujian yang dilakukan.

4.4.1 Hasil Pengujian System Usability Scale

Setelah aplikasi Easysay diluncurkan, peneliti menyebarkan aplikasi agar dapat digunakan dan dinilai oleh pengguna, khususnya narasumber yang terlibat pada Tabel 3.1 sebelumnya. Peneliti juga meminta narasumber terkait untuk menyebarkan aplikasi tersebut dan kuesioner SUS pada pengajar lainnya. Pada Gambar berikut ini merupakan sampel dari kuesioner SUS yang diisi responden sebelumnya.

Section 2 of 2

System Usability Scale

Mohon berikan pendapat anda pada ratio yang diberikan.

Skor dari 1 sampai 5, dengan 1 berarti **sangat tidak setuju**, dan 5 berarti **sangat setuju**.

Saya berpikir akan menggunakan aplikasi penilaian esai otomatis berbasis web ini lagi

1 2 3 4 5

Sangat Tidak Setuju ☐ ☐ ☐ ☐ ☐ Sangat Setuju

Saya merasa aplikasi penilaian esai otomatis berbasis web ini sulit digunakan

1 2 3 4 5

Sangat Tidak Setuju ☐ ☐ ☐ ☐ ☐ Sangat Setuju

Gambar 4.10 Sampel Google Form SUS

Semenjak peluncuran dilakukan dari tanggal 26 Agustus 2023, telah didapatkan 16 responden SUS dengan hasil sebagai pada Tabel 4.3 berikut.

Tabel 4.3 Responden SUS

No	Nama	Peran	Skor Untuk Pertanyaan									
			1	2	3	4	5	6	7	8	9	10
1	Muhammad Lutfi, S.Pd	Guru	5	1	4	1	5	1	4	2	3	2
2	Ajeng Yeni Ratnasari, S.Pd	Guru	5	3	4	2	5	3	5	2	4	1
3	Dita Putri Bestari S.Pd	Guru	5	1	3	1	4	2	5	1	5	2
4	Ayu Nulantika S.Pd	Guru	4	2	4	1	5	3	4	1	4	2
5	Ahmad Arfandi, S.Kom	Guru	4	1	5	1	4	2	5	2	5	3
6	Virgawaty Dewy, S.Pd	Guru	5	2	5	1	4	3	3	1	4	1
7	Isma Nurfitriah, S.Pd	Guru	4	3	5	2	5	2	4	1	3	2
8	Renny Noviani Riandy, S.S	Guru	5	1	4	1	5	1	5	2	5	1

No	Nama	Peran	Skor Untuk Pertanyaan									
			1	2	3	4	5	6	7	8	9	10
9	Sukma Erawan, S.Kom	Guru	3	2	3	1	5	2	5	1	4	2
10	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	4	1	5	2	4	3	4	3	4	1
11	Zamroni Riza S.E	Guru	5	2	5	1	5	2	5	2	3	3
12	Alan Syahwandi, S.Pd.I	Guru	5	2	5	1	3	1	4	1	5	2
13	Sahlul Efendi, S.pd	Guru	4	2	5	1	4	1	3	2	4	2
14	Yayah Shulhiyyah, S.Pd	Guru	5	3	4	3	3	3	4	1	5	3
15	Iyan Maulana, M.pd.i	Guru	4	3	3	1	4	1	5	2	5	3
16	Retno Dewi Kusumaningsih, S.Pd	Guru	4	1	4	3	5	1	4	1	4	2

Berdasarkan hasil responden SUS tersebut, didapatkan rata-rata hasil skor tiap pertanyaan pada Tabel 4.4 sebagai berikut.

Tabel 4.4 Rata-Rata Skor Tiap Pertanyaan SUS

No	Pertanyaan	Skor Rata-Rata
1	Saya berpikir akan menggunakan aplikasi penilaian esai otomatis berbasis web ini lagi	4.4375
2	Saya merasa aplikasi penilaian esai otomatis berbasis web ini sulit digunakan	1.875
3	Saya merasa aplikasi penilaian esai otomatis berbasis web ini mudah digunakan	4.25
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini	1.4375
5	Saya merasa fitur-fitur aplikasi penilaian esai otomatis berbasis web ini berjalan dengan semestinya	4.375
6	Saya merasa banyak hal yang tidak konsisten / tidak serasi pada aplikasi penilaian esai otomatis berbasis web ini	1.9375
7	Saya merasa orang lain akan memahami cara menggunakan	4.3125

No	Pertanyaan	Skor Rata-Rata
	aplikasi penilaian esai otomatis berbasis web ini dengan cepat	
8	Saya merasa aplikasi penilaian esai otomatis berbasis web ini membingungkan	1.5625
9	Saya merasa tidak ada hambatan dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini	4.1875
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan aplikasi penilaian esai otomatis berbasis web ini	2

Berdasarkan skor rata-rata penilaian SUS pada Tabel 4.4 diatas, didapatkan bahwa pertanyaan nomor 9 merupakan pertanyaan positif dengan nilai paling rendah. Hal ini kemungkinan dikarenakan batasan aplikasi, khususnya pada fitur penilaian jamak yang hanya bisa digunakan apabila pengguna memiliki jawaban yang ingin diperiksa dalam bentuk berkas CSV.

Sedangkan pada pertanyaan nomor 10 merupakan pertanyaan negatif dengan skor paling tinggi, hal ini dapat terjadi dikarenakan aplikasi yang dikembangkan merupakan sebuah teknologi baru yang belum pernah ada sebelumnya. Dengan ini, pengguna perlu membiasakan diri terlebih dahulu untuk dapat memahami cara kerja aplikasi ini dari perspektif pengguna sepenuhnya.

Diketahui dengan rumus yang telah dijelaskan pada sub bab 2.2.11 tentang *System Usability Scale*, setiap skor pertanyaan bernomor ganjil merupakan skor dikurang 1, sedangkan setiap skor pertanyaan bernomor genap merupakan 5 dikurangi skor. Maka dari itu, didapat hasil penilaian sebagaimana Tabel 4.5 berikut.

Tabel 4.5 Skor SUS Ganjil dan Genap

No	Nama	Peran	Skor Untuk Pertanyaan									
			1	2	3	4	5	6	7	8	9	10
1	Muhammad Lutfi, S.Pd	Guru	4	4	3	4	4	4	3	3	2	3
2	Ajeng Yeni Ratnasari, S.Pd	Guru	4	2	3	3	4	2	4	3	3	4
3	Dita Putri Bestari S.Pd	Guru	4	4	2	4	3	3	4	4	4	3

No	Nama	Peran	Skor Untuk Pertanyaan									
			1	2	3	4	5	6	7	8	9	10
4	Ayu Nulantika S.Pd	Guru	3	3	3	4	4	2	3	4	3	3
5	Ahmad Arfandi, S.Kom	Guru	3	4	4	4	3	3	4	3	4	2
6	Virgawaty Dewy, S.Pd	Guru	4	3	4	4	3	2	2	4	3	4
7	Isma Nurfitriah, S.Pd	Guru	3	2	4	3	4	3	3	4	2	3
8	Renny Noviani Riandy, S.S	Guru	4	4	3	4	4	4	4	3	4	4
9	Sukma Erawan, S.Kom	Guru	2	3	2	4	4	3	4	4	3	3
10	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	3	4	4	3	3	2	3	2	3	4
11	Zamroni Riza S.E	Guru	4	3	4	4	4	3	4	3	2	2
12	Alan Syahwandi, S.Pd.I	Guru	4	3	4	4	2	4	3	4	4	3
13	Sahlul Efendi, S.pd	Guru	3	3	4	4	3	4	2	3	3	3
14	Yayah Shulhiyyah, S.Pd	Guru	4	2	3	2	2	2	3	4	4	2
15	Iyan Maulana, M.pd.i	Guru	3	2	2	4	3	4	4	3	4	2
16	Retno Dewi Kusumaningsih, S.Pd	Guru	3	4	3	2	4	4	3	4	3	3

Setelah dilakukan hal tersebut, akan dijumlahkan hasil skor setiap responden dan dikalikan dengan 2.5. Dengan ini, didapatkan skor SUS lanjutan sebagai berikut.

Tabel 4.6 Total Skor SUS Setiap Responden Dikali 2.5

No	Nama	Peran	Total Skor x2.5
----	------	-------	-----------------

No	Nama	Peran	Total Skor x2.5
1	Muhammad Lutfi, S.Pd	Guru	85
2	Ajeng Yeni Ratnasari, S.Pd	Guru	80
3	Dita Putri Bestari S.Pd	Guru	87.5
4	Ayu Nulantika S.Pd	Guru	80
5	Ahmad Arfandi, S.Kom	Guru	85
6	Virgawaty Dewy, S.Pd	Guru	82.5
7	Isma Nurfitriah, S.Pd	Guru	77.5
8	Renny Noviani Riandy, S.S	Guru	95
9	Sukma Erawan, S.Kom	Guru	80
10	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	77.5
11	Zamroni Riza S.E	Guru	82.5
12	Alan Syahwandi, S.Pd.I	Guru	87.5
13	Sahlul Efendi, S.pd	Guru	80
14	Yayah Shulhiyyah, S.Pd	Guru	70
15	Iyan Maulana, M.pd.i	Guru	77.5
16	Retno Dewi Kusumaningsih, S.Pd	Guru	82.5
Rata-Rata			81.875

Dengan kalkulasi tersebut, maka skor SUS aplikasi yang dikembangkan dari penelitian ini mendapat skor dengan nilai 81.875 dengan rentang nilai A yaitu sangat baik.

4.4.2 Hasil Pengujian Black Box

Pada tahap ini, peneliti membandingkan ekspektasi hasil pengujian yang diharapkan pada Tabel 3.9 sebelumnya dengan hasil yang didapat setelah peluncuran aplikasi untuk memastikan bahwa aplikasi bekerja sesuai kebutuhan dan semestinya. Pada Tabel 4.7 dibawah ini dapat diperhatikan hasil pengujian *black box* yang dilakukan.

Tabel 4.7 Hasil Pengujian Fungsional *Black Box*

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
F1A	F1A-1	Aplikasi web dapat menerima masukkan teks sebagai jawaban pengajar	Aplikasi web berhasil menyimpan teks sebagai jawaban pengajar	Aplikasi web telah berhasil menyimpan teks sebagai jawaban pengajar
	F1A-2	Aplikasi web dapat menerima masukkan teks sebagai jawaban pelajar	Aplikasi web berhasil menerima teks sebagai jawaban pelajar	Aplikasi web telah berhasil menerima teks sebagai jawaban pelajar
F1B	F1B-1	Aplikasi web dapat menerima masukkan dari fungsi F1A untuk penilaian otomatis tunggal bahasa Inggris	Aplikasi web berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa Inggris	Aplikasi web telah berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa Inggris
	F1B-2	Aplikasi web dapat menerima masukka dari fungsi F1A untuk penilaian otomatis tunggal bahasa Indonesia	Aplikasi web berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa Indonesia	Aplikasi web telah berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa Indonesia
F0	F0	Aplikasi dapat menggunakan model yang tepat untuk melakukan penilaian berdasarkan bahasa yang dipilih pengguna	Aplikasi web berhasil menerima permintaan bahasa yang digunakan untuk penilaian dan menggunakan bahasa tersebut untuk penilaian.	Aplikasi web telah berhasil menerima permintaan bahasa yang digunakan untuk penilaian dan menggunakan bahasa tersebut untuk penilaian.
F2A	F2A-1	Aplikasi web dapat menerima masukkan berupa berkas berbentuk CSV sebagai kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan berkas CSV sebagai kumpulan jawaban pelajar	Aplikasi web telah berhasil menyimpan berkas CSV sebagai kumpulan jawaban pelajar
	F2A-2	Aplikasi web dapat menerima masukkan teks sebagai kolom nama pada	Aplikasi web berhasil menyimpan nama kolom yang berisi nama pelajar pada	Aplikasi web telah berhasil menyimpan nama kolom yang berisi nama pelajar pada

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
		kumpulan jawaban pelajar	berkas CSV	berkas CSV
	F2A-3	Aplikasi web dapat menerima masukkan teks sebagai kolom jawaban pada kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan nama kolom yang berisi jawaban pelajar pada berkas CSV	Aplikasi web telah berhasil menyimpan nama kolom yang berisi jawaban pelajar pada berkas CSV
	F2A-4	Aplikasi web dapat mendapatkan pasangan nama dan jawaban pada kumpulan jawaban pelajar sebagaimana ditentukan pada pengujian F2A-2 dan F2A-3	Aplikasi web berhasil menentukan pasangan nama dan jawaban pada berkas CSV untuk proses penilaian masal	Aplikasi web telah berhasil menentukan pasangan nama dan jawaban pada berkas CSV untuk proses penilaian masal
F2B	F2B-1	Aplikasi web dapat membuat dokumen CSV berdasarkan pengujian F1A-1, dan F1A-2 atau F2A-1 agar cocok dengan syarat penilaian pada algoritma penilaian otomatis bahasa Inggris	Aplikasi web berhasil membuat berkas CSV dari berkas CSV, nama, dan jawaban pelajar untuk diproses pada model penilaian otomatis bahasa Inggris	Aplikasi web telah berhasil membuat berkas CSV dari berkas CSV, nama, dan jawaban pelajar untuk diproses pada model penilaian otomatis bahasa Inggris
	F2B-3	Algoritma penilaian otomatis bahasa Inggris dapat membuat dokumen .json setelah melakukan penilaian otomatis	Algoritma penilaian berhasil membuat dokumen .json sebagai hasil penilaian otomatis model bahasa Inggris	Algoritma penilaian telah berhasil membuat dokumen .json sebagai hasil penilaian otomatis model bahasa Inggris
	F2B-5	Aplikasi dapat menampilkan hasil penilaian bahasa Inggris berdasarkan data .json yang dihasilkan oleh algoritma penilaian bahasa Inggris pada halaman web	Aplikasi web berhasil menggunakan dokumen .json yang telah dihasilkan penilaian otomatis bahasa Inggris sebelumnya untuk ditampilkan	Aplikasi web telah berhasil menggunakan dokumen .json yang telah dihasilkan penilaian otomatis bahasa Inggris sebelumnya untuk ditampilkan

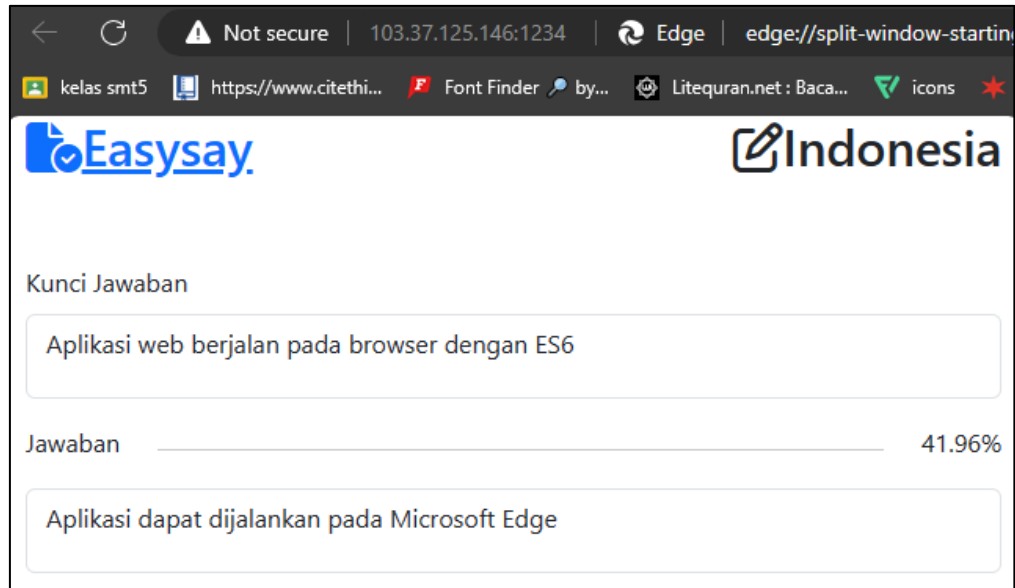
PARENT ID	ID	Pengujian	Ekspektasi	Hasil
			pada halaman web	pada halaman web
	F2B-2	Aplikasi web dapat membuat dokumen .json berdasarkan pengujian F1A-1, dan F1A-2 atau F2A-1 agar cocok dengan syarat penilaian pada algoritma penilaian otomatis bahasa Indonesia	Aplikasi web berhasil membuat dokumen .json setelah menentukan pasangan nama dan jawaban pada berkas CSV	Aplikasi web telah berhasil membuat dokumen .json setelah menentukan pasangan nama dan jawaban pada berkas CSV
	F2B-4	Algoritma penilaian otomatis bahasa Indonesia dapat membuat dokumen .json setelah melakukan penilaian otomatis	Algoritma penilaian otomatis bahasa Indonesia berhasil membuat dokumen .json untuk ditampilkan pada halaman web	Algoritma penilaian otomatis bahasa Indonesia telah berhasil membuat dokumen .json untuk ditampilkan pada halaman web
F2	F2	Aplikasi penilaian otomatis dapat menampilkan hasil penilaian ke halaman web berdasarkan dokumen .json yang dihasilkan oleh algoritma penilaian otomatis.	Dokumen .json yang merupakan hasil penilaian otomatis dari algoritma bahasa Indonesia maupun bahasa Inggris berhasil ditampilkan pada halaman web	Dokumen .json yang merupakan hasil penilaian otomatis dari algoritma bahasa Indonesia maupun bahasa Inggris telah berhasil ditampilkan pada halaman web

Pada Tabel 4.7 diatas dapat diperhatikan hasil pengujian fungsional *black box* terhadap aplikasi yang dikembangkan pada penelitian ini telah berhasil dilakukan untuk setiap pengujiaannya. Tentunya ini merupakan hasil akhir pengujian dari peluncuran terakhir, karena sebelum ini didapatkan beberapa *error* yang telah diperbaiki oleh peneliti dan diluncurkan kembali pada server untuk kembali digunakan.

Tabel 4.8 Hasil Pengujian Non Fungsional Black Box

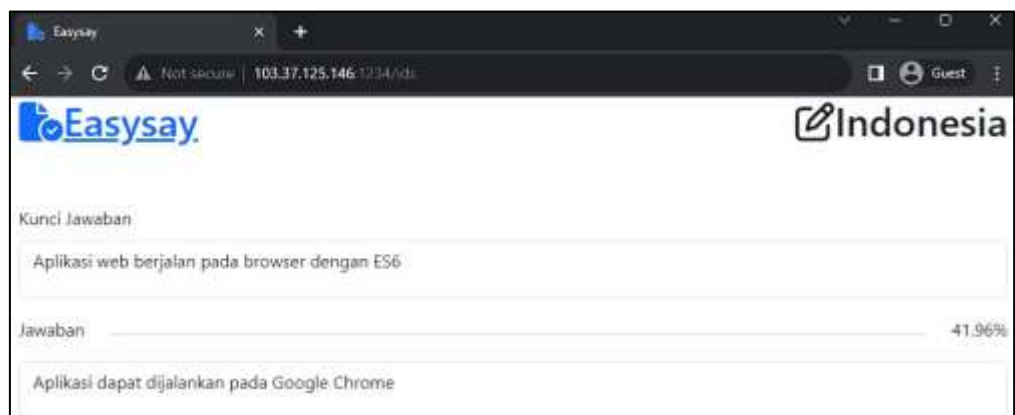
PARENT ID	ID	Pengujian	Ekspektasi	Hasil
NF1	NF1-1	Aplikasi dapat dijalankan pada ponsel	Aplikasi web berhasil diakses dan digunakan pada ponsel	Aplikasi web telah berhasil diakses dan digunakan pada ponsel
	NF1-2	Aplikasi dapat dijalankan pada komputer	Aplikasi web berhasil diakses dan digunakan pada komputer	Aplikasi web telah berhasil diakses dan digunakan pada komputer
NF2	NF2	Aplikasi dapat dijalankan pada peramban web yang mendukung ES6	Aplikasi web berhasil diakses dan dijalankan pada peramban web yang telah mendukung ES6	Aplikasi web telah berhasil diakses dan dijalankan pada peramban web yang telah mendukung ES6
NF3	NF3	Aplikasi tidak memiliki tampilan tumpang-tindih pada resolusi yang berbeda	Aplikasi web berhasil beradaptasi pada setiap resolusi dimana perlu ditampilkan tanpa adanya komponen yang tumpang-tindih	Aplikasi web telah berhasil beradaptasi pada setiap resolusi dimana perlu ditampilkan tanpa adanya komponen yang tumpang-tindih

Pada Tabel 4.8, merupakan hasil pengujian *black box* non fungsional. Pada pengujian ini dapat disimpulkan bahwa aplikasi yang dikembangkan penelitian ini memiliki tampilan responsif, tidak memiliki tampilan tumpang-tindih (sebagaimana pada sub bab 4.1.1), dan dapat dijalankan pada setiap peramban web yang mendukung ES6. Gambar dibawah ini merupakan bukti bahwa aplikasi ini dapat digunakan untuk peramban web yang mendukung ES6.



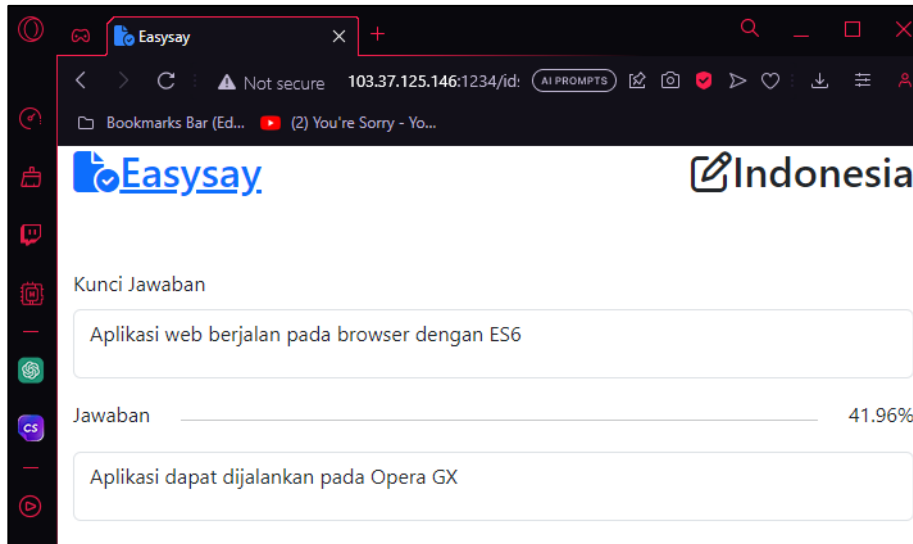
Gambar 4.11 Aplikasi pada *Microsoft Edge*

Aplikasi dapat dijalankan pada peramban web *Microsoft Edge* sebagaimana mestinya tanpa adanya kekurangan pada tampilan maupun fungsi yang dikembangkan.

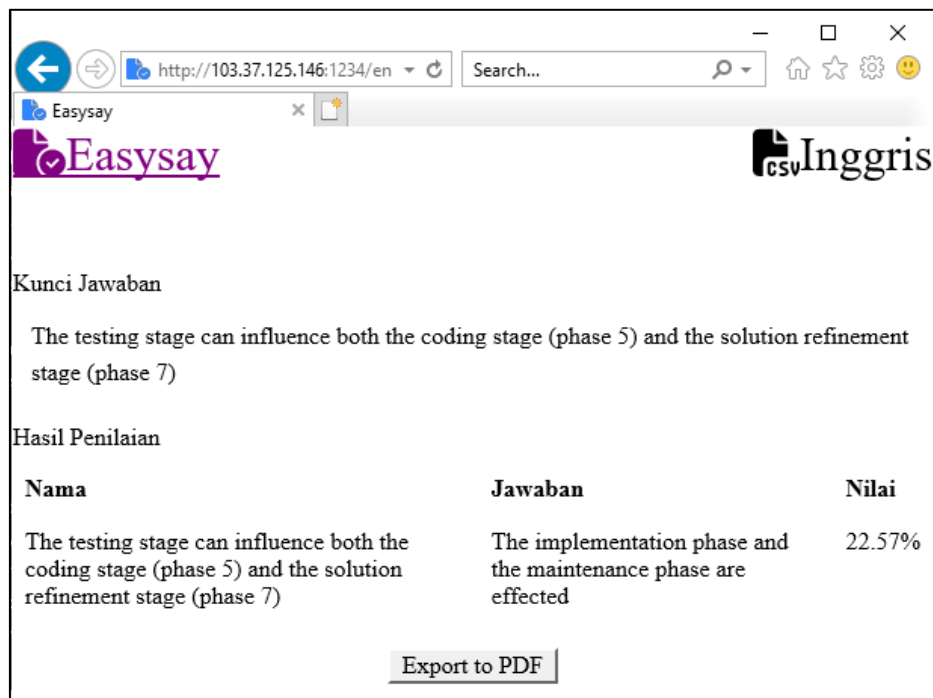


Gambar 4.12 Aplikasi pada *Google Chrome*

Aplikasi dapat dijalankan pada peramban web *Google Chrome* sebagaimana mestinya tanpa adanya kekurangan pada tampilan maupun fungsi yang dikembangkan.

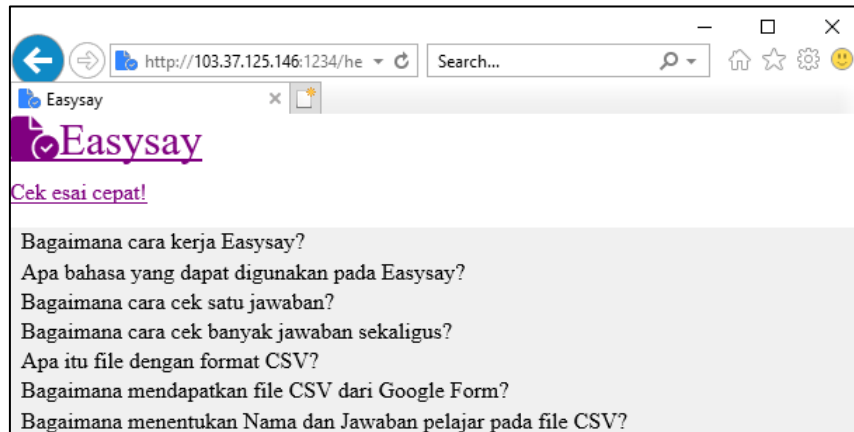
Gambar 4.13 Aplikasi pada *Opera GX*

Aplikasi dapat dijalankan pada peramban web *Opera GX* sebagaimana mestinya tanpa adanya kekurangan pada tampilan maupun fungsi yang dikembangkan.

Gambar 4.14 Aplikasi pada *Internet Explorer* (ekspor ke PDF)

Saat menggunakan *Internet Explorer* yang tidak mendukung ES6, aplikasi tetap dapat digunakan namun terdapat perbedaan pada tampilan yang ditunjukkan. Selain

itu, karena fitur “Export to PDF” dijalankan menggunakan *JavaScript*, *Internet Explorer* tidak dapat mengekspor hasil penilaian menjadi PDF.



Gambar 4.15 Aplikasi pada *Internet Explorer* (bantuan)

Selain itu, pada halaman bantuan dimana pengguna dapat mengetahui cara menggunakan aplikasi ini juga tidak dapat digunakan. Pengguna tidak dapat melihat isi dari setiap bantuan yang disediakan oleh pengembang. Dengan ini dapat disimpulkan bahwa aplikasi yang dikembangkan hanya layak untuk digunakan pada peramban web yang mendukung ES6.

Setelah pengujian fungsional dan non-fungsional, berikut ini merupakan hasil dari pengujian teks yang dilakukan pada aplikasi yang dikembangkan. Dari Tabel 4.9 dibawah ini dapat diperhatikan apa yang terjadi apabila pengguna melakukan aksi dengan kondisi ekstrem seperti mencampur dua bahasa pada satu teks yang ingin dilakukan penilaian otomatis, ataupun salah memilih bahasa saat menggunakan aplikasi yang dibangun.

Tabel 4.9 Hasil Pengujian Teks Black Box

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi	Hasil
ID-ID-ID	Pengguna memasukkan teks berbahasa Indonesia sebagai jawaban pelajar	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	Menunjukkan skor penilaian normal.	100.0 dari 100.0

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi	Hasil
	dan jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia.				
ID- IDEN- ID	Pengguna memasukkan teks berbahasa Indonesia sebagai jawaban pelajar dan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-ID-ID.	41.96 dari 100.0
IDEN- IDEN- ID	Pengguna memasukkan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban pengajar dan jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia.	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-ID-ID, namun kemungkinan mirip dengan skor pengujian ID-ID-ID.	100.0 dari 100.0
EN-EN- EN	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan jawaban pelajar saat menggunakan model penilaian berbahasa Inggris.	My children work at South Jakarta	South Jakarta is where my children work	Menunjukkan skor penilaian normal.	5 dari 5
EN- ENID-	Pengguna memasukkan teks	My children work at	Jakarta Selatan is	Menunjukkan skor	3.014 dari 5

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi	Hasil
EN	berbahasa Inggris pada jawaban pengajar dan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pelajar saat menggunakan model penilaian berbahasa Inggris	South Jakarta	where my children work	penilaian yang berbeda dari pengujian EN-EN-EN.	
ENID-ENID-EN	Pengguna memasukkan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Inggris.	My children work at Jakarta Selatan	Jakarta Selatan is where my children work	Menunjukkan skor penilaian yang berbeda dari pengujian EN-ENID-EN, namun kemungkinan mirip dengan skor pengujian EN-EN-EN.	5 dari 5
EN-EN-ID	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Indonesia.	My children work at South Jakarta	South Jakarta is where my children work	Menunjukkan skor penilaian.	71.01 dari 100.0
EN-ENID-ID	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pelajar	My children work at South Jakarta	Jakarta Selatan is where my children work	Menunjukkan skor penilaian yang berbeda dari pengujian EN-EN-ID.	53.87 dari 100.0

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi	Hasil
	saat menggunakan model penilaian berbahasa Indonesia.				
ENID-ENID-ID	Pengguna memasukkan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Indonesia.	My children work at Jakarta Selatan	My children work at Jakarta Selatan	Menunjukkan skor penilaian yang berbeda dari pengujian EN-ENID-ID, namun mirip dengan EN-EN-ID.	71.01 dari 100.0
ID-ID-EN	Pengguna memasukkan teks berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Inggris.	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	Menunjukkan skor penilaian.	5 dari 5
ID-IDEN-EN	Pengguna memasukkan teks berbahasa Indonesia pada jawaban pengajar dan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris saat menggunakan model penilaian berbahasa Inggris.	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-ID-EN.	3.347 dari 5
IDEN-IDEN-EN	Pengguna memasukkan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-IDEN-EN,	5 dari 5

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi	Hasil
	pengajar dan jawaban pelajar saat menggunakan metode penilaian berbahasa Inggris.			namun mirip dengan ID-ID-EN.	

Pada Tabel 4.9 diatas dapat dilihat hasil pengujian terhadap total 12 kemungkinan aksi yang dilakukan oleh pengguna. Pada pengujian teks tersebut, didapatkan hasil yang sesuai dengan ekspektasi penulis sebelumnya. Kemudian pada Tabel 4.10 dibawah ini merupakan hasil kata kunci yang dinilai oleh aplikasi pada setiap ID pengujiannya.

Tabel 4.10 Penilaian Kata Kunci Teks

ID	Masukkan Pengajar	Masukkan Pelajar	Kata Kunci Pengajar	Kata Kunci Pelajar	Hasil
ID-ID-ID	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	'pemilik', 'rekreasi'	'pemilik', 'rekreasi'	100.0 dari 100.0
ID-IDEN-ID	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	'pemilik', 'rekreasi'	'owner', 'rekreasi'	41.96 dari 100.0
IDEN-IDEN-ID	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	'owner', 'rekreasi'	'owner', 'rekreasi'	100.0 dari 100.0
EN-EN-EN	My children work at South Jakarta	South Jakarta is where my children work	"children", "work", "south", "makarka"	"south", "makarka", "children", "work"	5 dari 5
EN-ENID-EN	My children work at South Jakarta	Jakarta Selatan is where my children work	"children", "work", "south", "makarka"	"makarka", "gelatin", "children", "work"	3.014 dari 5
ENID-ENID-EN	My children work at Jakarta Selatan	Jakarta Selatan is where my children work	"children", "work", "makarka", "gelatin"	"makarka", "gelatin", "children", "work"	5 dari 5
EN-EN-ID	My children work at South Jakarta	South Jakarta is where my children work	'my', 'children', 'work', 'at', 'south',	'south', 'jakarta', 'is', 'where', 'my', 'children',	71.01 dari 100.0

ID	Masukkan Pengajar	Masukkan Pelajar	Kata Kunci Pengajar	Kata Kunci Pelajar	Hasil
			'jakarta'	'work'	
EN-ENID-ID	My children work at South Jakarta	Jakarta Selatan is where my children work	'my', 'children', 'work', 'at', 'south', 'jakarta'	'jakarta', 'selatan', 'is', 'where', 'my', 'children', 'work'	53.87 dari 100.0
ENID-ENID-ID	My children work at Jakarta Selatan	My children work at Jakarta Selatan	'my', 'children', 'work', 'at', 'jakarta', 'selatan'	'jakarta', 'selatan', 'is', 'where', 'my', 'children', 'work'	71.01 dari 100.0
ID-ID-EN	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	"say", "allah", "emilie", "tempt", "rekreasi"	"emilie", "tempt", "rekreasi", "allah", "say"	5 dari 5
ID-IDEN-EN	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	"say", "allah", "emilie", "tempt", "rekreasi"	"owner", "tempt", "rekreasi", "allah", "say"	3.347 dari 5
IDEN-IDEN-EN	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	"say", "allah", "owner", "tempt", "rekreasi"	"owner", "tempt", "rekreasi", "allah", "say"	5 dari 5

Pada Tabel 4.10 diatas, didapati bahwa pada kasus normal (ID-ID-ID dan EN-EN-EN), yaitu menggunakan model penilaian berbahasa Indonesia atau berbahasa Inggris untuk teks dengan bahasa terkait tanpa campuran bahasa lainnya menghasilkan skor sempurna. Hal ini dapat terjadi karena model *machine learning* berhasil mengabaikan kata yang tidak memiliki makna pada masing-masing teks dan menjadikan kata yang tepat sebagai kata kunci untuk pembobotan nilai teks terkait.

Selanjutnya pada kasus yang sedikit berbeda menggunakan model penilaian berbahasa Indonesia (ID-IDEN-ID) dimana terdapat kata campuran berbahasa Inggris pada masukkan teks pelajar menghasilkan skor lebih rendah walau memiliki maksud jawaban yang sama dengan kasus pengujian ID-ID-ID. Hal ini dikarenakan model penilaian berbahasa Indonesia tidak menemukan korelasi antara kata

“pemilik” pada teks pengajar dan “owner” pada teks pelajar, sehingga pembobotan nilai hanya diberikan untuk teks “rekreasi” yang dimiliki kedua teks tersebut. Kasus ini juga terjadi pada model penilaian berbahasa Inggris pada pengujian (EN-ENID-EN) dimana model penilaian berbahasa Inggris tidak menemukan hubungan antara kata “south” dan “selatan” walau memiliki arti sama. Pada kasus ini, kata “selatan” dianggap sebagai kesalahan ketik (*typo*) sehingga dibenarkan menurut model penilaian berbahasa Inggris ini menjadi kata “gelatin”. Kemudian untuk kata “Jakarta” juga dikoreksi menjadi “makarka”, namun tetap memberi penilaian positif pada teks dikarenakan kedua teks milik pengajar dan pelajar memiliki kata “Jakarta”.

Pada kasus lainnya, pengguna juga dapat memasukkan teks campuran pada model penilaian berbahasa Indonesia seperti pada pengujian IDEN-IDEN-ID, namun tetap menghasilkan skor sempurna dikarenakan tidak ada perbedaan kata kunci pada teks pengajar dan pelajar. Pada kasus ini, model penilaian tidak mengenal kata “owner” sehingga model penilaian berbahasa Indonesia menganggap kata “owner” sebagai kata unik yang dijadikan kata kunci. Hal yang sama juga terjadi pada kasus pengujian ENID-ENID-EN yang mana pengguna memiliki kata campuran berbahasa Indonesia pada teks berbahasa Inggris yang dilakukan penilaian pada model berbahasa Inggris. Penjelasan lebih lengkap terkait pemilahan kata kunci dari teks yang dimasukkan akan dijelaskan lebih lanjut pada paragraf berikutnya terkait kasus pengujian lainnya.

Kemudian untuk kasus ekstrem lainnya, pengguna juga bisa salah memilih model bahasa yang ingin digunakan. Kasus ini dapat dilihat pada pengujian ID-ID-EN maupun EN-EN-ID. Pada kasus ini, pengguna memasukkan teks berbahasa Indonesia pada kedua teks namun memilih metode penilaian dengan model berbahasa Inggris atau sebaliknya. Dapat dilihat pada kata kunci yang dihasilkan dari pengujian ID-ID-EN bahwa model memasukkan 5 dari 6 kata menjadi kata kunci dan mengubah kata tersebut. Hal ini dikarenakan model menganggap kata yang dimasukkan merupakan sebuah kesalahan dan mengubahnya menjadi kata yang dianggap benar dalam bahasa Inggris. Disisi lain, pada pengujian EN-EN-ID model penilaian berbahasa Indonesia menjadikan seluruh kata yang dimasukkan pada teks jawaban pengajar maupun

pelajar sebagai kata kunci. Hal ini membuat model penilaian gagal dalam membuang kata yang tidak memiliki makna dan menganggap semua kata yang dimasukkan adalah unik. Pada kasus pengujian normal (EN-EN-EN) kedua teks ini seharusnya menghasilkan skor sempurna (5 dari 5), sedangkan pada kasus pengujian ekstrem (EN-EN-ID) kedua teks ini menghasilkan skor tidak sempurna (71.01 dari 100.0).

Untuk kasus pengujian ID-ID-EN dan IDEN-IDEN-EN, merupakan kasus ekstrem yang seharusnya tidak boleh terjadi. Namun pada hasil pengujian didapatkan hasil sempurna. Hal ini dikarenakan kedua teks kebetulan memiliki kata unik yang sama menurut model *machine learning* berbahasa Inggris. Walau begitu, menggunakan model penilaian berbahasa Inggris untuk menilai teks berbahasa Indonesia bukanlah hal yang dianjurkan karena pada teks lain bisa menghasilkan skor tidak sempurna yang mana seharusnya menghasilkan skor sempurna apabila dilakukan penilaian pada model penilaian bahasa yang benar.

Hasil analisis lainnya terhadap kata campuran berbahasa lain yang digunakan pada bahasa utama seperti kasus pengujian ENID-ENID-EN atau IDEN-IDEN-ID dapat dilakukan apabila kedua teks memiliki kata campuran yang sama. Hal ini dikarenakan sebuah soal esai bisa saja memiliki jawaban dengan kata dari bahasa lain, seperti pertanyaan esai untuk soal terjemahan “apa yang dimaksud dari kata *obvious* pada teks tersebut?” dan soal lain semacamnya.

Analisis akhir dari penulis terkait pengujian teks ini adalah untuk tidak mencampur bahasa apabila tidak diperlukan, dan tidak menggunakan model bahasa lain saat melakukan penilaian menggunakan aplikasi yang dikembangkan penelitian ini. Hal ini tidak dianjurkan karena dapat menyebabkan perbedaan skor yang berpotensi menyebabkan kerugian sepihak pada penilaian esai yang dilakukan secara otomatis.


4.5 Pembahasan

Penelitian ini menghasilkan sebuah produk untuk hasil akhirnya yaitu berupa aplikasi penilaian esai singkat yang dapat melakukan penilaian esai otomatis menggunakan bahasa Indonesia dan bahasa Inggris. Pada tahap pengembangan aplikasi, peneliti

mengimplementasikan metode pengembangan perangkat lunak *Test-Driven Development* (TDD) yang merupakan SDLC Agile.

Pengembangan dilakukan dengan mengumpulkan kebutuhan dari 10 narasumber yang merupakan pengajar di Indonesia, tahap ini termasuk dalam tahap *red* pada metode pengembangan TDD. Setelah terkumpulnya kebutuhan ini, peneliti menuliskan pengujian berupa *scenario testing* yang perlu diloloskan mencapai kesuksesan aplikasi yang dibangun. Setelahnya, dibuatlah *mockup* aplikasi yang dapat memenuhi kebutuhan yang terkumpul, seperti masukkan teks, tombol memilih metode penilaian, dan lainnya. Setelahnya, peneliti akan masuk ke tahap antara *red* dan *green*, yaitu menuliskan kode untuk setiap kebutuhan fungsional dan non-fungsional yang terdata pada Tabel 3.4 dan 3.5. Apabila suatu fungsi / fitur berhasil memenuhi uji coba skenario yang dituliskan, berarti peneliti akan memasuki tahap *green* dari metode pengembangan TDD, dengan ini pengembang dapat membersihkan / refaktor kode pada aplikasi yang dibangunnya yang merupakan tahap *refactor* pada metode pengembangan TDD. Setelah refaktor kode dilakukan, pengembang dapat langsung mengembangkan fungsi / fitur selanjutnya. Pada siklus hidup ini lah aplikasi penilaian esai singkat otomatis Easysay dibuat.

Ada 2 model *machine learning* yang digunakan pada aplikasi yang dikembangkan, model yang bekerja bahasa Indonesia, dan bahasa Inggris. Masing-masing model memiliki cara kerja yang berbeda.



```

Importing Dataset

full_df = pd.read_csv("dataset.csv")
full_df

```

Gambar 4.16 Cara Penilaian Model *Machine Learning* Bahasa Inggris

Pada model *machine learning* bahasa Inggris yang dapat dilihat pada Gambar 4.16 diatas, model bekerja dengan cara menilai berkas CSV yang sudah memiliki

pasangan jawaban pengajar dan jawaban pengajar pada berkas bernama “dataset.csv”, hal ini menunjukkan bahwa model bahasa Inggris secara mendasar melakukan penilaian secara jamak. Pada implementasi, untuk melakukan penilaian tunggal, *backend* aplikasi yang dikembangkan akan membuat berkas CSV dengan satu pasang jawaban pengajar dan pelajar. Selanjutnya untuk melakukan penilaian jamak, *backend* aplikasi membuat berkas yang dapat mendeteksi nama dari *header* kolom berisi jawaban pelajar, dan membuat kolom baru berisi jawaban pengajar yang didapat dari masukkan pengguna. Dengan ini, dapat dibuat berkas CSV dengan pasangan jawaban pengajar dan jawaban pelajar yang berbeda. Kemudian dengan teknik *child-process*, *backend* menjalankan model *machine learning* bahasa Inggris ini dan mengirimkan juga berkas CSV yang perlu diperiksa. Setelah selesai, model akan mengirimkan ke *backend* sebuah JSON yang berisi hasil penilaian yang dapat ditampilkan pada *frontend*. Setelah selesai, berkas CSV yang telah selesai digunakan akan dihapus untuk tidak memenuhi penyimpanan server.

```
print('Masukkan Kunci Jawaban:')
kj = input()
print('Masukkan Jawaban Siswa:')
jawaban = input()
```

Gambar 4.17 Cara Penilaian Model *Machine Learning* Bahasa Indonesia

Kemudian pada model *machine learning* bahasa Indonesia yang dapat dilihat pada Gambar 4.17, model bekerja dengan memeriksa satu masukkan jawaban pengajar dan pelajar melalui *input*. Dengan ini, model diketahui untuk bekerja secara tunggal dalam melakukan penilaian esainya. Untuk penilaian esai tunggal, *backend* menjalankan model dengan metode *child-process* dengan argumen bawaan yaitu jawaban pengajar dan jawaban pelajar untuk diperiksa. Selanjutnya model akan membuat berkas teks yang berisi skor penilaian dan mengirimkan argumen berisikan “done” pada *backend* untuk menandakan bahwa proses penilaian pada model telah selesai. Dengan ini, *backend* akan membuka berkas teks berisi skor tersebut sebagai JSON untuk ditampilkan pada *frontend* dan menghapus berkas teks tersebut agar tidak memenuhi penyimpanan pada server. Sedangkan untuk penilaian jamak, *backend* akan mendapatkan pasangan jawaban pengajar dan pelajar dari setiap baris

pada berkas CSV yang diunggah oleh pengguna dan menjalankan *child-process* pada model untuk setiap pasangan jawaban tersebut, dalam kata lain, proses penilaian jamak yang dilakukan pada model bahasa Indonesia adalah proses penilaian tunggal yang dilakukan secara berulang. Apabila penilaian selesai, maka *backend* akan menghapus berkas CSV dan teks berisi penilaian setelah menampilkannya pada *frontend*.

Untuk memastikan bahwa selama proses implementasi model *machine learning* pada aplikasi web ini tidak terjadi pengaruh terhadap penilaian yang dilakukan, dilakukanlah pengujian skenario. Pengujian dilakukan dengan melakukan penilaian dengan memasukkan yang sama pada aplikasi web dan pada model *machine learning* langsung. Skor yang dihasilkan oleh model bahasa Inggris adalah nilai numerik dari 0.0 hingga 5.0, dan nilai numerik dari 0 hingga 100 untuk model bahasa Indonesia. Pada pengujian yang dilakukan pada *scenario testing* SC1, didapatkan skor dari model bahasa Inggris senilai 1.128299 yang merupakan 22% dari 5.0, yang mana aplikasi web mendapatkan hasil 22%. Kemudian pada model bahasa Indonesia, khususnya pada *scenario testing* SC3, didapatkan skor senilai 55 dari 100, yang mana aplikasi web mendapatkan skor senilai 55%. Dengan ini, dapat dipastikan bahwa tidak ada perbedaan antara hasil penilaian pada model yang telah diimplementasikan pada aplikasi web dan model aslinya.

Kemudian untuk pengujian hasil pengujian setelah peluncuran, didapatkan skor SUS senilai 81.875 atau rentang nilai A. Hasil SUS ini didapat setelah peneliti mendapat responden sebanyak 16 orang pengguna yang sudah melakukan penilaian otomatis pada aplikasi yang dibuat. Selain itu, aplikasi yang dikembangkan pada penelitian ini juga mendapatkan hasil pengujian sukses pada kebutuhan fungsional dan non-fungsionalnya yang diujicobakan menggunakan metode pengujian Black Box dengan total pengujian fungsional sebanyak 5 pengujian umum dengan total 15 pengujian spesifik, dan total pengujian non-fungsional sebanyak 3 pengujian umum dengan total 4 pengujian spesifik, dan pengujian teks sebanyak 12 pengujian. Khususnya untuk pengujian teks ini didapatkan kesimpulan bahwa sebaiknya tidak mencampurkan bahasa pada sebuah teks / esai yang ingin diperiksa apabila tidak

diperlukan saat melakukan penilaian otomatis dikarenakan dapat menghasilkan skor yang tidak sempurna dibanding dengan kondisi normal. Kasus ini tentunya dapat diabaikan apabila memang pencampuran bahasa diperlukan pada sebuah soal esai, yang mana jawaban pengajar dan jawaban pelajar akan masing-masing memiliki kata campuran tersebut. Selanjutnya kesimpulan dari pengujian teks ini juga didapati untuk tidak salah memilih bahasa untuk penilaian, apabila bahasa esai merupakan bahasa Indonesia atau bahasa Indonesia dengan beberapa kata bahasa Inggris, maka gunakanlah model penilaian bahasa Indonesia. Begitu juga untuk kasus penilaian bahasa Inggris.

Aplikasi penilai esai singkat ini dapat digunakan pada peramban web apabila pengguna berhak mengakses laman web yang sudah diluncurkan, atau menjalankannya pada komputer lokal.

Untuk menjalankannya pada komputer lokal, pengguna perlu *runtime* Node.js dengan minimal versi 16.16.0 dan *Python* dengan minimal versi 3.11.5. Untuk spesifikasi *processor*, *ram*, dan penyimpanan tidak memiliki minimal, namun apabila spesifikasi yang digunakan lebih besar maka akan lebih baik untuk performa. Selanjutnya juga pengguna perlu menginstal seluruh *dependencies Node.js* dan *Python* yang digunakan pada aplikasi ini.

Kemudian apabila pengguna ingin melakukan penilaian, pengguna dapat membuka aplikasi pada laman web ataupun lokal, pada tahap ini pengguna akan diarahkan ke halaman utama. Pada halaman ini, pengguna dapat memilih bahasa yang ingin digunakan pada penilaian esai. Selanjutnya pengguna dapat memilih metode penilaian apa yang ingin digunakan untuk melakukan penilaian, dengan ini pengguna akan diarahkan ke halaman yang bersangkutan untuk bahasa dan metode penilaian terkait.

Apabila pengguna memilih metode penilaian tunggal, pengguna akan berada di halaman dimana pengguna dapat memasukkan satu teks pada kolom “Jawaban Pelajar” dan satu teks pada kolom “Jawaban Pengajar”. Setelahnya, pengguna dapat

menekan “periksa” dan aplikasi web akan melakukan penilaian otomatis pada esai yang disediakan. Setelah selesai, pengguna akan diarahkan ke halaman hasil penilaian tunggal dimana pengguna dapat melihat hasil persentase penilaian esai.

Sedangkan apabila pengguna memilih metode penilaian jamak, pengguna akan diarahkan ke halaman dimana pengguna dapat memasukkan jawaban pengajar, mengunggah berkas CSV berisi jawaban pelajar yang ingin diperiksa, dan menyediakan nama kolom “nama” dan “jawaban pelajar” pada kolom yang disediakan. Hal ini dilakukan agar aplikasi mengetahui mana kolom yang berisi nama dan jawaban pelajar pada berkas CSV untuk dibandingkan dengan jawaban pengajar. Setelah pengguna menekan tombol “periksa”, pengguna diarahkan ke halaman hasil penilaian jamak dimana pengguna dapat melihat pasangan nama dan jawaban pelajar serta nilai hasil penilaian. Pengguna juga dapat mengekspor hasil penilaian ke PDF apabila diperlukan untuk kepentingan pribadi.