RANCANG BANGUN APLIKASI PENILAIAN ESAI SINGKAT BERBAHASA INDONESIA DAN INGGRIS MENGGUNAKAN METODE TEST-DRIVEN DEVELOPMENT

TUGAS AKHIR

Diajukan sebagai syarat menyelesaikan jenjang strata Satu (S-1) di Program Studi Teknik Informatika, Jurusan Teknologi, Produksi dan Industri, Institut Teknologi Sumatera

Oleh:

GEIZKA ROZILIA RUICOSTA

119140114

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI, PRODUKSI DAN INDUSTRI

INSTITUT TEKNOLOGI SUMATERA

LAMPUNG SELATAN

2023

LEMBAR PENGESAHAN

Tugas Akhir dengan judul "Tulis Judul Disini" adalah benar dibuat oleh saya sendiri dan belum pernah dibuat dan diserahkan sebelumnya, baik sebagian ataupun seluruhnya, baik oleh saya ataupun orang lain, baik di Institut Teknologi Sumatera maupun di institusi pendidikan lainnya.

Lampung Selatan, DD-MM-YYYY
Penulis,

PHOTO
BERWARNA

Geizka Rozilia Ruicosta
NIM. 119140114

Diperiksa dan disetujui oleh,

Pembimbing	Tanda Tangan
1. Nama Pembimbing 1 + Gelar NIP. XXXXXX	……
2. Nama Pembimbing 2 + Gelar NIP. XXXXXX	……

Penguji	Tanda Tangan
1. Nama Penguji 1 + Gelar NIP. XXXXXXXXXXXX	……
2. Nama Penguji 2+ Gelar NIP. XXXXXXXXXXXX	……

Disahkan oleh,
Koordinator Program Studi Teknik Informatika
Jurusan Teknologi, Produksi dan Industri
Institut Teknologi Sumatera

Nama Kaprodi + Gelar
NIP. XXXXXXXXXXXXXX

HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir dengan judul "TULIS JUDUL DISINI" adalah karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan benar.

Nama	**: Geizka Rozilia Ruicosta**
NIM	**: 11940114**
Tanda Tangan	**: …….**
Tanggal	**: …….**

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai civitas akademik Institut Teknologi Sumatera, saya yang bertanda tangan di bawah ini:

Nama	: Geizka Rozilia Ruicosta
NIM	: 119140114
Program Studi	: Teknik Informatika
Jurusan	: Jurusan Teknologi, Produksi dan Industri
Jenis Karya	: Tugas Akhir

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut Teknologi Sumatera **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty Free Right*)** atas karya ilmiah saya yang berjudul:

TULIS JUDUL DISINI

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini, Institut Teknologi Sumatera berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Lampung Selatan

Pada tanggal DD Bulan YYYY

Yang menyatakan,

Geizka Rozilia Ruicosta

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas limpahan rahmat, karunia, serta petunjuk-Nya sehingga penyusunan tugas akhir ini telah terselesaikan dengan baik. Dalam penyusunan tugas akhir ini penulis telah banyak mendapatkan arahan, bantuan, serta dukungan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapan terima kasih kepada:

1. <isi dengan nama Rektor ITERA>
2. <isi dengan nama Kajur JTPI>
3. <isi dengan nama Kaprodi IF>
4. <isi dengan nama Sesprodi IF>
5. <isi dengan nama Koordinator TA>
6. <isi dengan nama Dosen Pembimbing>
7. Kedua Orang Tua, kakak dan adik yang selalu memberikan arahan selama belajar dan menyelesaikan tugas akhir ini.
8. <isi dengan nama orang lainnya>

Akhir kata penulis berharap semoga tugas akhir ini dapat memberikan manfaat bagi kita semua, amin. [Contoh]

RINGKASAN

Judul TA

Nama Mahasiswa

Halaman Ringkasan berisi uraian singkat tentang latar belakang masalah, rumusan masalah, tujuan, metodologi penelitian, hasil dan analisis data, serta kesimpulan dan saran. Isi ringkasan tidak lebih dari 1500 kata (sekitar 3 halaman).

ABSTRAK

Judul TA

Nama Mahasiswa

Halaman ABSTRAK berisi uraian tentang latar belakang, tujuan, metodologi penelitian, hasil / kesimpulan. Ditulis dalam BAHASA INDONESIA tidak lebih dari 250 kata, dengan jarak antar baris satu spasi.

Pada akhir abstrak ditulis kata “Kata Kunci” yang dicetak tebal, diikuti tanda titik dua dan kata kunci yang tidak lebih dari 5 kata. Kata kunci terdiri dari kata-kata yang khusus menunjukkan dan berkaitan dengan bahan yang diteliti, metode/instrumen yang digunakan, topik penelitian. Kata kunci diketik pada jarak dua spasi dari baris akhir isi abstrak.

Kata Kunci : Penambangan Data, Kecerdasan Buatan, Lampung Selatan

ABSTRACT

Judul TA (Bahasa Inggris)

Nama Mahasiswa

Halaman ABSTRACT berisi uraian tentang latar belakang, tujuan, metodologi penelitian, hasil / kesimpulan. Ditulis dalam BAHASA INGGRIS tidak lebih dari 250 kata, dengan jarak antar baris satu spasi. Secara khusus, kata dan kalimat pada halaman ini tidak perlu ditulis dengan huruf miring meskipun menggunakan Bahasa Inggris, kecuali terdapat huruf asing lain yang ditulis dengan huruf miring (misalnya huruf Latin atau Greek, dll).

Pada akhir abstract ditulis kata “Keywords” yang dicetak tebal, diikuti tanda titik dua dan kata kunci yang tidak lebih dari 5 kata. Keywords terdiri dari kata-kata yang khusus menunjukkan dan berkaitan dengan bahan yang diteliti, metode/instrumen yang digunakan, topik penelitian. Keywords diketik pada jarak dua spasi dari baris akhir isi abstrak.

Keywords : Data Mining, Artificial Intelligence, Lampung Selatan

DAFTAR ISI

LEMBAR PENGESAHAN i

HALAMAN PERNYATAAN ORISINALITAS ii

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS iii

BAB I PENDAHULUAN 1

1.1 Latar Belakang Masalah 1

1.2 Rumusan Masalah 3

1.3 Tujuan Penelitian 4

1.4 Batasan Masalah 4

1.5 Manfaat Penelitian 4

1.6 Sistematika Penulisan 5

1.6.1 BAB I PENDAHULUAN 5

1.6.2 BAB II TINJAUAN PUSTAKA 5

1.6.3 BAB III METODE PENELITIAN 5

BAB II TINJAUAN PUSTAKA 6

2.1 Tinjauan Pustaka 6

2.2 Dasar Teori 10

2.2.1 Test-Driven Development 10

2.2.2 Use Case Diagram 12

2.2.3 Activity Diagram 14

2.2.4 Website 15

2.2.5 JavaScript 15

2.2.6 Node.js 15

2.2.7 Express.js 16
2.2.8 JavaScript Object Notation 16
2.2.9 Esai Singkat 17
2.2.10 Python 17
2.2.11 System Usability Scale 18
2.2.12 Black Box Testing 19
2.2.13 Integration Testing 20
2.2.14 Peramban Web 20
2.2.15 Peluncuran Perangkat Lunak 20
BAB III METODE PENELITIAN 21
3.1 Alur Penelitian 21
3.2 Penjabaran Langkah Penelitian 21
3.2.1 Identifikasi Masalah 22
3.2.2 Studi Literatur 22
3.2.3 Penerapan *Test-Driven Development* 22
3.2.4 Integrasi Aplikasi dan Model 23
3.2.5 Pengujian 24
3.3 Alat dan Bahan Tugas Akhir 24
3.3.1 Alat 24
3.3.2 Bahan 25
3.4 Metode Pengembangan 25
3.4.1 Kebutuhan Fitur / Fungsi 26
3.4.2 Pembuatan Uji Coba (Scenario Testing) 36
3.4.3 Pembuatan Kode (Coding) 37
3.4.4 Refaktor (Refactor) 37
3.5 Pre-Deployment Testing (Integration Testing) 37

3.6 Peluncuran 38
3.7 Post-Deployment Testing 38
3.7.1 System Usability Scale 38
3.7.2 Black Box 39
BAB IV HASIL DAN PEMBAHASAN 43
4.1 Hasil Pengembangan Metode TDD 43
4.1.1 Pengembangan Frontend 44
4.1.2 Pengembangan Backend 48
4.1.3 Child-Process Model Machine Learning 53
4.1.4 Hasil Scenario Testing 54
4.2 Hasil Pre-Deployment Testing (Integration Testing) 57
4.3 Hasil Peluncuran 61
4.4 Hasil Post-Deployment Testing 61
4.4.1 Hasil Pengujian System Usability Scale 61
4.4.2 Hasil Pengujian Black Box 66
4.5 Pembahasan 73
BAB V KESIMPULAN DAN SARAN 78
5.1 Kesimpulan 78
5.2 Saran 79
LAMPIRAN 84
Lampiran I Kode Program 84
Lampiran II Dokumentasi dengan Narasumber 256
Lampiran III Software Requirement Specification 260

DAFTAR TABEL

Tabel 2.1 Perbandingan Refrensi 8
Tabel 2.2 Simbol *Use Case Diagram* 13
Tabel 2.3 Simbol *Activity Diagram* 14
Tabel 2.4 Standar Hasil SUS [14] 19

Tabel 3.1 Data Narasumber 27
Tabel 3.2 Hasil Wawancara 27
Tabel 3.3 Kumpulan *User Story* 28
Tabel 3.4 Kebutuhan Fungsional 29
Tabel 3.5 Kebutuhan Non-Fungsional 30
Tabel 3.6 Skenario Uji Coba 36
Tabel 3.7 *Integration Testing* 37
Tabel 3.8 Kuesioner *System Usability Scale* 39
Tabel 3.9 Daftar Pengujian Fungsional Black Box 39
Tabel 3.10 Daftar Pengujian Non Fungsional Black Box 42

Tabel 4.1 Kode Program Rute / 48
Tabel 4.2 Kode Program Rute /help 49
Tabel 4.3 Kode Program Rute /option 49
Tabel 4.4 Kode Program Rute /ens 50
Tabel 4.5 Kode Program Rute /enb 50
Tabel 4.6 Kode Program Rute /ids 51
Tabel 4.7 Kode Program Rute /idbulk 52
Tabel 4.8 Kode Konversi Hasil Penilaian Menjadi PDF 52
Tabel 4.9 Kode Konversi Berkas CSV ke JSON 53
Tabel 4.10 Model Bahasa Inggris Menerima Nama Berkas CSV 53
Tabel 4.11 Model Bahasa Inggris Mengirim Hasil Penilaian 54
Tabel 4.12 Model Bahasa Indonesia Menerima Teks Jawaban 54
Tabel 4.13 Model Bahasa Indonesia Selesai Melakukan Penilaian 54
Tabel 4.14 Hasil Scenario Testing 55

Tabel 4.15 Hasil Integration Testing 58
Tabel 4.16 Responden SUS 62
Tabel 4.17 Rata-Rata Skor Tiap Pertanyaan SUS 63
Tabel 4.18 Skor SUS Ganjil dan Genap 64
Tabel 4.19 Total Skor SUS Setiap Responden Dikali 2.5 65
Tabel 4.20 Hasil Pengujian Fungsional *Black Box* 66
Tabel 4.21 Hasil Pengujian Non Fungsional Black Box 69

DAFTAR GAMBAR

Gambar 2.1 SDLC Test-Driven Development (TDD) [20]**Error! Bookmark not defined.**

Gambar 2.2 SUS *Score Percentile Rank* [13].....**Error! Bookmark not defined.**

Gambar 3.1 *Flowchart* Penelitian.....**Error! Bookmark not defined.**

Gambar 3.2 Alur Penelitian *Test-Driven Development* 26

Gambar 3.3 *Use Case Diagram* Aplikasi 30

Gambar 3.4 *Activity Diagram* Aplikasi 31

Gambar 3.5 *Mockup* Halaman Home 32

Gambar 3.6 Mockup Metode Penilaian Tunggal..... 33

Gambar 3.7 *Mockup* Hasil Metode Penilaian Tunggal..... 34

Gambar 3.8 *Mockup* Metode Penilaian Jamak 34

Gambar 3.9 *Mockup* Hasil Metode Penilaian Jamak..... 35

Gambar 3.10 *Mockup* Halaman Bantuan..... 35

DAFTAR RUMUS

Rumus perhitungan SUS (1) 18

DAFTAR LAMPIRAN

Lampiran I Kode Program 84
Lampiran II Dokumentasi dengan Narasumber 256
Lampiran III Software Requirement Specification 260

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Berdasarkan data statistik dari Perserikatan Bangsa-Bangsa (*United Nations*), sejak bulan November 2022 jumlah populasi manusia telah menyentuh angka delapan milyar dengan negara Indonesia berada di peringkat 4 dengan angka 280 juta jiwa [1][2]. Kemudian pada tahun 2021, Badan Pusat Statistik menyatakan bahwa penduduk di Indonesia terdiri dari 22 juta jiwa anak-anak dengan rentang umur 0 s/d 4 tahun dan 22 juta jiwa anak-anak dengan rentang umur 5 s/d 9 tahun dengan total anak-anak dengan umur dibawah 10 tahun berjumlah 44 juta jiwa yang merupakan 15.71% dari total populasi penduduk di Indonesia [3].

Dengan kemungkinan banyaknya pelajar yang akan bersekolah, maka teknologi di bidang pendidikan juga diperlukan untuk meningkatkan efisiensi kegiatan belajar mengajar di Indonesia, khususnya pada teknik penilaian esai karena esai memperbolehkan pelajar untuk memberikan jawaban rinci terkait pertanyaan yang diberikan oleh pengajar [4]. Menurut studi sosial yang dilakukan pada tahun 2018, semakin banyak esai yang harus dikoreksi oleh pengajar, maka kualitas penilaian juga akan menurun dan tidak lagi objektif (bias) yang menyebabkan evaluasi penilaian pengajar tidak akurat [5].

Permasalahan efisiensi waktu pada penilaian esai dapat dilakukan apabila evaluasi esai pelajar dilakukan oleh beberapa tim pengajar sehingga waktu yang diperlukan untuk menilai seluruh esai dapat terbagi dengan banyaknya jumlah tim pengajar. Solusi pertama walau dapat mempersingkat waktu yang dibutuhkan untuk melakukan evaluasi terhadap jawaban esai pelajar, tetap memiliki permasalahan terkait subjektivitas yang dapat mempengaruhi akurasi penilaian yang dilakukan oleh tim pengajar. Selain itu, setiap pengajar yang melakukan evaluasi juga mungkin meyakini jawaban yang berbeda terkait sebuah pertanyaan dalam suatu esai, hal ini dapat menyebabkan penilaian tim pengajar semakin tidak akurat. Solusi lainnya adalah menggunakan teknologi untuk mengevaluasi esai pelajar berdasarkan jawaban

pengajar untuk menilai seberapa sesuai jawaban tersebut. Teknologi dapat berupa aplikasi yang terpasang pada sebuah perangkat tertentu (aplikasi *mobile* atau *desktop*) atau berupa aplikasi *online* yang dapat diakses melalui internet (*website*). Solusi ini tentunya akan memberikan penilaian lebih objektif terhadap jawaban esai yang dievaluasi.

Aplikasi ini kemudian juga harus dapat diakses dimanapun dan kapanpun agar penggunaan aplikasi tidak terbatas di satu perangkat saja. Namun, pengembangan aplikasi *mobile* perlu disesuaikan untuk setiap *platform* yang akan semakin membatasi dimana aplikasi dapat digunakan [6]. Maka dari itu, aplikasi berbasis *website* adalah solusi terbaik untuk permasalahan ini karena aplikasi akan dapat digunakan kapan saja dan dimana saja, tanpa membatasi perangkat apa yang dapat mengakses aplikasi tersebut.

Aplikasi akan dikembangkan mengikuti siklus hidup perangkat lunak *Test-Driven Development* (TDD) yang merupakan salah satu *Software Development Life-Cycle* (SDLC) *Agile* [7]. Praktik TDD dimulai dengan memikirkan cara menguji sebuah fungsi. Programmer kemudian menulis kode yang dapat memenuhi pengujian [8]. Dibanding dengan *eXtreme Programming* (XP), dan *Personal eXtreme Programming* (PXP), yang juga termasuk dalam SDLC *Agile*, TDD terbukti dapat menghasilkan aplikasi yang sangat kecil kemungkinannya untuk memiliki *bug* dan *error* karena pada setiap fungsi yang akan dikembangkan perlu dibuat syarat keberhasilan pengujian terlebih dahulu, kemudian jika kode yang dituliskan berhasil memenuhi syarat keberhasilan pengujian tersebut, pengembang dapat melanjutkan pengembangan fungsi selanjutnya [9].

Aplikasi yang dikembangkan akan dilakukan pada aplikasi yang dikembangkan untuk memastikan kelayakannya menggunakan dua metode pengujian, yaitu *Integration Testing*, *System Usability Scale* (SUS) dan *Black Box*. *Integration Testing* dilakukan untuk memastikan bahwa setiap fungsi yang dibangun pada aplikasi telah memenuhi syarat kebutuhannya [10]. SUS memiliki tingkat keabsahan dan keandalan tinggi untuk mengukur tingkat kemudahan perangkat lunak saat

digunakan [11][12]. SUS digunakan karena terfokus pada kegunaan yang dirasakan pengguna melalui 10 kuesioner yang dapat diisi dengan cepat, dibanding *Post-Study System Usability Questionaire* (PSSUQ) yang lebih terfokus untuk menilai kenyamanan yang dirasakan pengguna saat menggunuakan sebuah sistem [13]. SUS juga tidak membutuhkan biaya lebih besar untuk melakukan pengujiannya dibanding dengan *Heuristic Evaluation* (HE) yang merupakan salah satu metode pengujian perangkat lunak lainnya [14]. Sedangkan pengujian *Black Box* dilakukan untuk menguji tampilan, fungsionalitas, masukkan, dan luaran perangkat lunak tanpa harus mengetahui bagaimana baris-per-baris program kode perangkat lunak bekerja dengan harapan bahwa perangkat lunak yang dikembangkan bekerja semestinya [15]. Pada penelitian ini, SUS digunakan karena pengembang ingin memfokuskan aplikasi pada sisi kegunaannya.

Dengan ini, akan dikembangkan aplikasi berbasis *website* yang dapat membantu pengajar dalam menilai jawaban esai pelajar berdasarkan jawaban yang sudah diberikan oleh pengajar untuk dijadikan landasan untuk program menilai jawaban pelajar. Berdasarkan wawancara yang dilakukan pada beberapa guru yang merupakan calon pengguna aplikasi yang akan dibangun, aplikasi akan memiliki dua fitur utama, yaitu untuk menilai satu jawaban esai singkat dan menilai banyak jawaban sekaligus, hal ini akan mengurangi lebih banyak beban kerja pengajar karena tidak perlu memasukkan jawaban pelajar satu-persatu. Fitur lainnya yaitu pengajar dapat memilih bahasa esai yang ingin diperiksa, Indonesia atau Inggris.

1.2 Rumusan Masalah

Untuk menyelesaikan permasalahan yang sudah dipaparkan pada latar belakang, maka didapat rumusan masalah sebagai berikut:

1. Bagaimana implementasi aplikasi penilaian esai singkat menggunakan metode *test-driven development* dapat dilakukan?
2. Bagaimana menerapkan *pre-trained machine learning model* untuk menilai esai singkat pada sebuah aplikasi?
3. Bagaimana memastikan hasil penilaian pada aplikasi sama dengan penilaian pada model *machine learning* berdasarkan skor yang dihasilkan?

4. Bagaimana aplikasi penilaian esai singkat dapat digunakan?

1.3 Tujuan Penelitian

Tujuan dilakukannya penelitian tugas akhir ini berdasarkan latar belakang dan rumusan masalah adalah:

1. Mengetahui implementasi metode *test-driven development* pada pembangunan aplikasi implementasi penilaian esai singkat.
2. Menerapkan *pre-trained machine learning model* python pada aplikasi berbasis JavaScript.
3. Memastikan hasil penilaian dari aplikasi sama dengan hasil penilaian pada model *machine learning* berdasarkan skor.
4. Mengetahui penggunan aplikasi penilaian esai singkat.

1.4 Batasan Masalah

Batasan masalah yang ada pada penelitian tugas akhir ini yaitu:

1. Aplikasi berbasis web hanya dapat digunakan melalui *browser*.
2. Aplikasi hanya dapat memproses penilaian esai yang diketik, bukan ditulis tangan.
3. Aplikasi hanya dapat menerima masukkan berupa teks untuk jawaban guru.
4. Aplikasi hanya dapat menerima masukkan berupa teks atau berkas *comma-separated values* (CSV) berisi kumpulan jawaban untuk jawaban murid.
5. Aplikasi hanya dapat melakukan penilaian terhadap 1 pertanyaan pada sekali pemakaian.
6. Penelitian ini terfokus pada implementasi aplikasi menggunakan model *machine learning* penilaian esai bahasa Indonesia dan bahasa Inggris yang sudah ada.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat meningkatkan efisiensi kegiatan belajar-mengajar (KBM) pada seluruh jenjang pendidikan dengan mempermudah proses penilaian esai singkat pada soal dengan jawaban berbahasa Indonesia atau Inggris.

1.6 Sistematika Penulisan

1.6.1 BAB I PENDAHULUAN

Pada bab ini membahas mengenai latar belakang dalam pembuatan proyek tugas kerja praktik, rumusan masalah, tujuan, manfaat, ruang lingkup, metodologi dan sistematika penulisan dari laporan kerja praktik.

1.6.2 BAB II TINJAUAN PUSTAKA

Pada bab ini menjelaskan mengenai landasan teori atau pustaka yang digunakan dalam membuat laporan ataupun dalam pembuatan sistem yang dibuat.

1.6.3 BAB III METODE PENELITIAN

Pada bab ini menjelaskan langkah-langkah dilakukannya penelitian untuk mencapai hasil penelitian yang diharapkan.

BAB II
TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Pada penelitian ini, ada beberapa referensi yang dijadikan landasan pendukung untuk pelaksanaan setiap langkah penelitian. Referensi yang digunakan merupakan penelitian terdahulu yang akan menjadi acuan dalam membangun perangkat lunak berupa aplikasi berbasis web. Pada Tabel 2.1 berikut dapat diperhatikan referensi yang digunakan pada penelitian ini.

Aplikasi yang dikembangkan pada penelitian ini akan dibuat berbasis *website* menggunakan metode siklus hidup pengembangan perangkat lunak (SDLC) *Test-Driven Development* (TDD) yang merupakan salah satu bentuk SDLC *Agile* seperti yang sudah dilakukan pada penelitian sebelumnya dengan judul Comparative Study of Test-Driven Development (TDD), Behavior-Driven Development (BDD) and Acceptance Test–Driven Development (ATDD) yang dilakukan oleh Myint Myint Moe [7]. Pada penelitian yang dilakukannnya, didapatkan bahwa TDD digunakan saat pengembang melakukan pengujian, hal ini berlaku untuk memastikan seluruh fitur dan fungsi yang dikembangkan pada aplikasi ini terbebas dari error dan bug.

Kemudian pada penelitian yang dilakukan oleh Xinyu Chang dan Jing Li dengan judul Improvement of excel data processing function based on Spring MVC Framework dijelaskan bahwa penggunaan JavaScript Object Notation (JSON) untuk memproses data dari excel agar dapat diproses lebih cepat [16]. Hal ini akan digunakan pada penelitian tugas akhir kali ini pada aplikasi saat melakukan penilaian otomatis untuk jawaban secara massal dengan format berkas CSV.

Penelitian ini juga akan menggunakan teknologi Express.js karena memiliki performa yang cepat dibandingkan pesaingnya yaitu Ktor sebagaimana yang telah diteliti pada penelitian sebelumnya dengan judul Express.js and Ktor web server performance A comparative study yang dilakukan oleh Isac Glantz dan Hampus Hurtig pada tahun 2022 [17].

Untuk memeriksa jawaban secara otomatis, akan digunakan model pemrosesan bahasa alami yang sudah terlatih berupa model *Python* dalam bahasa Inggris dan Indonesia. Diperlukan penelitian pendukung yang dapat menjembatani antara bahasa JavaScript yang akan digunakan pada website dan Python yang akan menjalankan pemeriksaan jawaban, penelitian tersebut adalah TENSORFLOW.JS: MACHINE LEARNING FOR THE WEB AND BEYOND [18]. Dengan ini, aplikasi yang dikembangkan akan menggunakan *Node.js* sebagai teknologi penghubung *JavaScript* dan *Python*.

Model yang digunakan dapat membandingkan jawaban siswa terhadap guru dalam bahasa Inggris dan Indonesia menggunakan teknik sebagaimana dilakukan pada penelitian oleh Svanhvít Ingólfsdóttir tentang *lemmatization* untuk bahasa Islandia dengan judul Nefnir: A high accuracy lemmatizer for Icelandic [19].

Aplikasi yang dikembangkan juga akan menggunakan basis data untuk menyimpan jawaban khususnya jawaban yang ingin diperiksa otomatis secara masal menggunakan basis data non-SQL MongoDB. MongoDB digunakan karena dapat melakukan ekspor data berupa JSON yang selanjutnya akan digunakan untuk memeriksa jawaban satu-persatu pada model *Python*. Kemudian, menggunakan basis data non-SQL memiliki performa lebih cepat daripada basis data SQL khususnya pada eksekusi *Query*. Hal ini didukung oleh penelitian berjudul Performance analysis of NoSQL and relational databases with MongoDB and MySQL yang dilakukan oleh Benymol Jose dan Sajimon Abraham pada tahun 2020 [20].

Tabel 2.1 Perbandingan Refrensi

No	Judul	Permasalahan	Metode	Hasil	Perbandingan
1	TENSORFLOW.JS: MACHINE LEARNING FOR THE WEB AND BEYOND (2019)	*Library* pembelajaran mesin biasanya ditulis dalam bahasa *Python* atau C++. Namun, jumlah pengguna *JavaScript* untuk frontend dan backend semakin bertambah. Jurnal ini berisi panduan untuk menjembatani *Python* dengan *JavaScript*	-API (*Application Programming Interface*)	Menggunakan *Layers API*, didapatkan performa waktu sebesar - 3426 ms dengan *JavaScript* biasa dengan 1x percepatan, - 10 ms dengan *WebGL* dengan 342x percepatan, - 3 ms dengan *Node.js* dengan 1105x percepatan.	Aplikasi yang akan dikembangkan akan menggunakan API dibangun dengan *Node.js* tanpa menggunakan *TensorFlow*.
2	Improvement of excel data processing function based on Spring MVC Framework (2022)	*Excel* yang sangat umum digunakan, masih dinilai kurang pada pemrosesan sekelompok data dan verifikasi data. Dengan berkembangnya bidang sains dan teknologi, diperlukan pemrosesan data berskala besar dan berkelanjutan.	-JSON(*JavaScript Object Notation*)	Berdasarkan *Spring MVC + EasyUI* dengan pengembangan *Java J2EE IDE*. Diciptakan *kit* pengembangan yang bisa menkonversi *excel* dan melakukan ekspor dan impor sekumpulan data dengan menkonversinya menjadi JSON.	CSV akan dikonversi menjadi bentuk JSON menggunakan teknik *spawn JavaScript*.
3	Nefnir: A high accuracy lemmatizer for Icelandic (2019)	Mencari bentuk morfologi dasar pada sebuah kumpulan tulisan dengan bahasa yang kaya akan morfologi.	-*Nefnir* -*part-of-speech*	*Nefnir* meraih akurasi sebesari 99.55% untuk teks yang ditandai dengan benar. *Part-of-speech* (PoS) mendapat akurasi sebesar 96.88% dengan teks yang ditandai dengan *PoS tagger*.	Model yang digunakan pada penelitian ini dicapai menggunakan teknik *stemming* dan *lemmatization*.

No	Judul	Permasalahan	Metode	Hasil	Perbandingan
4	Express.js and Ktor web server performance A comparative study (2022)	Membandingkan dua *framework* web berdasarkan waktu merespon untuk membantu *developer* memilih antara *Express.js* dan *Ktor*	Melakukan test terhadap waktu respon menggunakan database melalui *Object Relational Mapper* (ORM) *Sequelize* untuk *Express.js*, dan *Exposed* untuk *Ktor*	*Express.js* memiliki waktu respon yang lebih baik (3 ms) secara keseluruhan daripada *Ktor* (106 ms). Namun penggunaan *Object Relational Mapper* pada *Ktor* lebih berpengaruh pada hasil daripada *Express.js*.	Aplikasi yang akan dikembangkan pada penelitian ini akan dibangun menggunakan *Express.js* sebagai framework modul HTTP dari *Node.js*.
5	Performance analysis of NoSQL and relational databases with MongoDB and MySQL (2020)	Basis data relasional dinilai tidak efektif untuk bekerja dengan bervariasi informasi yang besar, karena basis data relasional seperti *MySQL* menyimpan data secara terorganisir.	-Basis data non-relasional (*NoSQL*) seperti MongoDB.	Berdasarkan waktu eksekusi *query* untuk *insert*, menggunakan MySQL membutuhkan waktu 302156 ms, sedangkan saat menggunakan MongoDB hanya memerlukan waktu 56985 ms.	Aplikasi akan menggunakan basis data non-SQL MongoDB.
6	Comparative Study of Test-Driven Development (TDD), Behavior-Driven Development (BDD) and Acceptance Test–Driven Development (ATDD) (2019)	Membandingkan perbedaan antara *Test-Driven Development* (TDD), *Behavior-Driven Development* (BDD), dan *Acceptance Test-Driven Development* (ATDD) dalam pengembangan perangkat lunak dengan lingkungan pengembangan yang berbeda.	Studi banding metode pengembangan perangkat lunak.	TDD digunakan saat pengembang menulis dan menjalankan pengujian. BDD merincikan perilaku fitur menggunakan bahasa yang dapat dimengerti semua orang yang terkait dalam pembangunan. ATDD membuat implementasi lebih efektif.	SDLC TDD akan digunakan pada penelitian ini untuk mengembangkan aplikasi penilaian esai otomatis guna meminimalisir kemungkinan adanya *bug* pada tahap penulisan kode.

Pada Tabel 2.1 sebelumnya, dapat diperhatikan beberapa tinjauan pustaka yang menjadi landasan pada penelitian ini lengkap dengan permasalahan, metode, hasil, dan perbedaan penelitian penulis dengan landasan teori penelitian sebelumnya.

2.2 Dasar Teori

Ada beberapa teori yang akan digunakan untuk mendukung penelitian tugas akhir ini. Teori yang digunakan akan menjadi landasan, dan memperkuat pemahaman selama penelitian ini dilakukan. Berikut adalah teori yang digunakan pada penelitian tugas akhir ini:

2.2.1 Test-Driven Development

Test-Driven Development (TDD) merupakan metode pengembangan perangkat lunak *agile* yang dikenalkan oleh *Extreme Programming* (XP). TDD menggunakan pendekatan pengujian unit yang terfokus pada pengembang [7]. TDD bekerja dengan cara menuliskan kode pengujian dan fungsionalitas hingga pengujian berhasil, kemudian menuliskan kode pengujian dan fungsionalitas selanjutnya [9]. Kode pengujian yang digunakan dapat berupa *integration testing* atau *scenario testing*. *Scenario testing* digunakan untuk memastikan bahwa fungsi pada aplikasi bekerja pada kondisi spesifik yang biasa digunakan dengan metode pengujian lain [21], sedangkan *integration testing* digunakan untuk memastikan bahwa sebuah fungsi pada aplikasi dapat bekerja sesuai ekspektasi secara keseluruhan. Metode ini meningkatkan kualitas produk perangkat lunak yang dikembangkan dan produktifitas pengembang.

Studi kasus pada tahun 2019 menyebutkan kelebihan saat menggunakan metode pengembangan perangkat lunak TDD sebagai berikut:

1. Membantu mencegah kecacatan produk
2. Membantu dokumentasi kode dengan contoh eksekusi
3. Membantu programmer untuk sangat memahami kode mereka
4. Mendukung refaktor sebagai kebutuhan dan perubahan desain
5. Mendorong desain yang lebih baik
6. Menyediakan peringatan awal terkait masalah desain

7. Membuat rangkaian uji regresi otomatis
8. Programmer mempelajari cara menulis jenis pengujian lain
9. Mendorong langkah kecil dan prinsip bahwa lebih baik untuk menjaga sistem tetap bekerja

Kemudian, pada studi yang sama juga menyebutkan kekurangan saat menggunakan metode pengembangan perangkat lunak TDD, yaitu sebagai berikut:

1. Sulit untuk dipelajari
2. Sulit untuk diaplikasikan kepada kode yang telah dibuat sebelumnya dari orang lain (*legacy code*)
3. Banyak kesalahpahaman yang mempersulit programmer untuk mempelajarinya.

Test-Driven Development (TDD) memiliki siklus pengembangan perangkat lunak seperti Gambar 2.1 berikut ini:

Gambar 2.1 SDLC Test-Driven Development (TDD) [22]

1. Langkah pertama yaitu membaca, memahami, dan memproses fitur atau bug yang diminta.

2. Berdasarkan kebutuhan yang diminta, unit pengujian akan dibuat dan dijalankan dengan kode pada langkah 3.
3. Membuat dan mengimplementasikan kode yang memenuhi kebutuhan pengujian. Saat dijalankan, semua pengujian harus lolos, jika tidak, langkah ini akan diulang hingga berhasil.
4. Apabila sudah berhasil, maka kode akan dibersihkan/dirapikan (*refactoring*).
5. Ulang dari langkah 1 dengan permintaan fitur atau bug yang baru.

Alur kerja *Test-Driven Development* (TDD) sebagaimana dilihat pada Gambar 2.1 disebut juga sebagai *Red-Green-Refactoring* (Merah-Hijau-Refaktor), yang merupakan status dari pengujian dalam setiap siklusnya [22].

Penelitian tugas akhir ini akan menggunakan siklus hidup pembangunan perangkat lunak (SDLC) *Test-Driven Development* (TDD) karena kebutuhan yang dapat bervariasi dari aplikasi yang akan dikembangkan baik dari fitur maupun kebutuhan utama. Dengan menggunakan TDD, aplikasi yang dikembangkan dapat diminimalisir adanya *error* atau *bug* dengan melakukan unit pengujian untuk setiap langkah pembangunannya[8]. Pengembangan aplikasi penilaian esai singkat otomatis ini juga tidak menggunakan *legacy code* pada sisi yang dikerjakan oleh penulis, sehingga tidak perlu khawatir dengan kekurangan pada penerapan TDD yang telah disebutkan sebelumnya tentang kesulitan penerapan TDD pada *legacy code* [7].

2.2.2 Use Case Diagram

Use Case Diagram adalah salah satu diagram dalam *Unified Modeling* Language. *Use Case* Diagram merupakan diagram perilaku yang mendeskripsikan kebutuhan fungsional pada sebuah perangkat lunak. Diagram ini juga digunakan untuk memahami bagaimana sebuah sistem seharusnya bekerja [23]. Berikut ini merupakan manfaat dari digunakannya *Use Case Diagram*, yaitu:

1. Mendapatkan tampilan luar dari suatu sistem.
2. Mendapatkan seluruh kebutuhan sistem.
3. Mengenali faktor-faktor yang dapat mempengaruhi sistem secara eksternal maupun internal.

4. Mendemonstrasikan interaksi antara sistem dan aktor.

Pada Tabel 2.2 dibawah ini, dapat diperhatikan nama dan deskripsi dari simbol yang akan digunakan dalam sebuah *Use Case Diagram*.

Tabel 2.2 Simbol *Use Case Diagram*

Simbol	**Nama**	**Deskripsi**
	Aktor	Peranan eksternal yang berinteraksi dengan sistem.
	Use Case	Fungsionalitas atau kebutuhan yang akan diimplementasikan pada sebuah sistem.
	Generalisasi	Merepresentasiskan hubungan antara aktor dengan aktor, maupun *use case* dengan *use case* lainnya.
	Asosiasi	Merepresentasikan komunikasi dua arah antara aktor dengan *use case*, setiap kasus dimulai dengan aktor utama yang harus mendapatkan respon dari *use case*.
	Include	Merepresentasikan hubungan antara dua *use case*. Saat *use case* A membutuhkan *use case* B untuk menyelesaikan tugasnya, walau *use case* B dapat menyelesaikan tugasnya sendiri.
	Extend	Merepresentasikan hubungan antara dua *use case*. Saat *use case* A mungkin memerlukan *use case* B untuk menyelesaikan tugasnya, namun *use case* B tidak bisa ada sendiri.

Penelitian ini akan menggunakan *Use Case Diagram* untuk memperjelas skenario dan kebutuhan dari perilaku sistem yang akan dikembangkan. Hal ini dilakukan

untuk memastikan bahwa sistem yang akan dikembangkan memenuhi skenario dan perilaku sistem yang telah diajukan.

2.2.3 Activity Diagram

Sama seperti *Use Case Diagram*, *Activity Diagram* merupakan salah satu diagram dalam *Unified Modeling Language*. *Activity Diagram* dapat menampilkan alur kegiatan dari sebuah sistem perangkat lunak [24]. *Activity Diagram* biasanya berisi aktifitas, transaksi, keputusan, swimlane, dan aktifitas paralel [25]. Berdasarkan itu, simbol, nama, dan deskripsi dari *Activity Diagram* dapat dilihat pada Tabel 2.3 berikut ini.

Tabel 2.3 Simbol *Activity Diagram*

Simbol	Nama	Deskripsi
	Start	Menunjukkan kondisi awal / dimulainya sebuah aktifitas
	End	Menunjukkan berakhirnya semua kondisi dari sebuah aktifitas
	Aktifitas	Menunjukkan aktifitas dari sebuah proses
	Keputusan	Menunjukkan kondisi percabangan dari 1 masukkan dengan banyak keluaran
	Swimlane	Mengorganisir aktifitas berdasarkan peran dari setiap proses
	Aktifitas Paralel	Menunjukkan proses yang berjalan secara paralel, biasanya terdiri dari dua aktifitas yang dijalankan bersamaan

Penelitian ini akan menggunakan *Activity Diagram* untuk menggambarkan alur aktifitas dari sistem perangkat lunak yang akan dikembangkan.

2.2.4 Website

Website adalah keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi [26]. Layanan yang terdapat pada *website* dapat diakses melalui jaringan internet, dengan permintaan layanan dieksekusi melalui sistem jarak jauh[27].

2.2.5 JavaScript

JavaScript adalah bahasa pemrograman yang populer, digunakan bukan hanya untuk web sisi klien, tapi juga sisi *server* [28]. *JavaScript* juga memiliki tipe dinamis, yang artinya pembangun tidak perlu menspesifikasikan tipe dalam kodenya. Hal ini yang membuat IDE *JavaScript* sering gagal mensugestikan tipe yang akurat karena tipe dari elemen kode tidak diketahui hingga memiliki isi. Namun kekurangan ini bisa diatasi dengan membuat kode yang bagus dan tidak ambigu.

Pada penelitian tugas akhir ini, *JavaScript* akan digunakan untuk membangun aplikasi berbasis website penilaian esai singkat pada sisi klien dan juga sisi *server* dengan *Node.js*.

2.2.6 Node.js

Node.js adalah *run timer* sisi server *JavaScript* yang digunakan pada *browser* untuk memproses HTTP. *Node.js* memungkinkan penulis untuk membuat *backend* dari sebuah *website* yang dapat menerima *request GET* dan *POST* menggunakan bahasa pemrograman *JavaScript* [29].

Node.js memiliki *package manager* yaitu NPM, gudang perangkat lunak terbesar di dunia dengan jutaan paket yang dapat digunakan pada perangkat lunak siapapun menjadi bantuan (*dependencies*).

Penelitian ini menggunakan *Node.js* sebagai *run timer* sisi *server* untuk aplikasi berbasis *website* yang akan dibangun karena dapat menangani lebih banyak permintaan dalam satuan waktu dibanding PHP [30]. Hal ini membuat *Node.js* lebih ideal untuk membangun *website* dengan masukkan dan keluaran yang intens. Selain

itu, *Node.js* juga *developer friendly*, yang mana mudah dipahami karena cara penggunaannya yang konsisten.

2.2.7 Express.js

Pada teori sebelumnya, dapat diketahui bahwa *Node.js* merupakan *run timer JavaScript* dan dapat menerima permintaan HTTP *GET* dan *POST*. *Express.js* merupakan paket NPM berupa *framework* untuk mempermudah penggunaan modul HTTP pada *Node.js* [31] sehingga penulisan kode untuk melakukan permintaan HTTP dapat dilakukan lebih cepat.

Sebagai *framework* untuk menangani permintaan HTTP, performa *Express.js* terbilang cepat yaitu hanya memerlukan waktu 3 ms untuk memproses permintaan *Object Relational Model* [17], dan memerlukan waktu rata-rata 17 *ms* untuk memproses halaman dinamis dari permintaan sebanyak 200 pengguna.

Atas alasan tersebut, perangkat lunak yang akan dikembangkan dalam penelitian ini akan menggunakan *Express.js* untuk menangani permintaan HTTP sebagai *framework* dari *Node.js* karena kemampuan paket NPM ini diperlukan untuk memproses nilai dan jawaban pada aplikasi website penilaian esai singkat.

2.2.8 JavaScript Object Notation

JavaScript Object Notation (JSON) adalah format data dengan penulisan seperti tipe data objek pada bahasa pemrograman *JavaScript*. JSON merupakan format data berisi objek atau kumpulan objek-objek dalam suatu objek yang ringan, mudah dibaca dan ditulis oleh manusia, serta dibuat dan diterjemahkan dengan komputer. Format data ini merupakan bagian dari bahasa pemrograman *JavaScript* [32].

JSON umumnya berisikan sebuah objek yang dapat berisi banyak objek lain. Dalam format dokumen ini, objek harus dimulai dan diakhiri oleh kurung keriting ({}), kemudian setiap objek dalam JSON dipisah dengan simbol koma (,) setelah kurung keriting tutup (}) [32].

Untuk menggunakan dokumen JSON pada aplikasi website khususnya yang menggunakan JavaScript, objek dapat diakses dengan menggunakan dokumen JSON dan menggunakan simbol titik (.) untuk mengakses setiap sub-objeknya.

Pada aplikasi web yang akan dikembangkan pada penelitian ini, akan ada fitur untuk mengunggah seluruh jawaban pelajar berupa berkas CSV ke aplikasi website penilaian esai. Berkas CSV kemudian akan diubah menjadi bentuk JSON untuk selanjutnya diproses kedalam penilaian.

2.2.9 Esai Singkat

Esai merupakan karangan yang membahas suatu masalaah berdasarkan sudut pandang penulis, pada kasus ini yaitu jawaban dari pertanyaan yang diberikan oleh pengajar menurut pelajar. Esai merupakan salah satu metode penilaian untuk mengukur tingkat pemahaman pelajar pada suatu topik yang kerap diberikan pada saat ujian [33].

Esai singkat menurut Steven Burrows pada tahun 2014 merupakan sebuah teks deskriptif tentang suatu topik khusus yang terdiri dari sebuah frasa atau paling banyak 100 kata [34].

2.2.10 Python

Python, seperti *JavaScript*, merupakan salah satu bahasa pemrograman yang banyak digunakan oleh pembangun perangkat lunak di bidang teknologi. Bedanya, *Python* lebih banyak digunakan untuk sains data dengan kapabilitasnya untuk pembelajaran mesin [35].

Python merupakan bahasa pemrograman yang banyak dipilih bagi praktisi pemrosesan bahasa alami, yang merupakan cabang dari kecerdasan buatan. *Python* memiliki *library* yang dapat mendukung pemrosesan bahasa alami yang mencakup banyak bahasa manusia seperti *Stanza* [36].

Aplikasi penilaian esai singkat yang akan dikembangkan penulis akan menggunakan model terlatih berbasis *Python* untuk melakukan proses penilaian dan perbandingan jawaban.

2.2.11 System Usability Scale

Dalam membangun sebuah sistem perangkat lunak, diperlukan adanya skala standar yang dapat digunakan untuk memastikan bahwa sistem yang dikembangkan dapat digunakan oleh pengguna sebaik-baiknya. *System Usability Scale* (SUS) adalah kuesioner yang digunakan untuk mengukur tingkat kemudahan perangkat lunak saat digunakan [11]. SUS adalah alat psikometrik gratis yang digunakan diseluruh dunia dengan keabsahan dan keandalan tinggi [12].

Pada penelitian ini, penulis menggunakan SUS untuk mengukur tingkat kemudahan pengguna dalam menggunakan aplikasi penilaian esai singkat otomatis berbasis web melalui antarmuka yang akan disediakan untuk mengoperasikan aplikasi yang akan dibangun.

Kuesioner SUS akan diajukan kepada sejumlah pengguna relawan yang tertarik untuk menggunakan aplikasi penilaian esai otomatis berbasis web yang akan dikembangkan pada penelitian tugas akhir ini. Kemudian hasil dari kuesioner SUS ini akan berupa rata-rata skor dari semua responden dengan rentang nilai dari 0 s/d 100. Nilai prinsip SUS adalah untuk menyediakan skor referensi dari pandangan peserta terhadap kegunaan sebuah produk / sistem [37].

Nilai akan diperoleh dengan cara memisahkan skor pertanyaan dengan nomor ganjil dari nomor genap dengan kalkulasi pada rumus (1) berikut ini :

$$\boxed{\bar{x} = \frac{\sum(2.5 \times (\frac{a-1}{5-b}))}{n}}$$

Rumus perhitungan SUS (1)

Keterangan:

α : skor pertanyaan bernomor ganjil

b : skor pertanyaan bernomor genap

$\bar{x}$: skor rata-rata

n : jumlah responden

Kemudian pada Tabel 2.4 berikut ini, dapat dilihat interpretasi dari setiap nilai yang diperoleh kalkulasi pengujian SUS.

Tabel 2.4 Standar Hasil SUS [14]

Skor SUS	Nilai	Sifat Nilai
>80.3	A	Sangat Baik
74 – 80.3	B	Baik
68 - 74	C	Oke
51 – 68	D	Buruk
<51	F	Sangat Buruk

Melalui standar hasil SUS pada Tabel 2.4, aplikasi penilaian esai otomatis berbasis web yang dikembangkan sebaiknya meraih minimal nilai C untuk dapat dinyatakan berhasil dan dapat digunakan oleh banyak pengguna sebagaimana terdapat pada Gambar 2.2 berikut ini [14].

Gambar 2.2 SUS *Score Percentile Rank* [14]

2.2.12 Black Box Testing

Teknik pengujian selanjutnya yaitu teknik pengujian *Black Box*. Teknik pengujian aplikasi ini membahas aplikasi perangkat lunak dari sisi luar seperti tampilan, fungsionalitas, masukkan, dan luaran [15].

Aplikasi yang akan dikembangkan dari penelitian ini akan diuji menggunakan pengujian *Black Box*. Aplikasi akan diuji oleh peneliti untuk menguji masukkan dan keluaran dari aplikasi ini dengan maksud untuk memastikan bahwa aplikasi yang dibangun bekerja semestinya.

2.2.13 Integration Testing

Untuk meningkatkan kepastian bahwa suatu program berfungsi sebagaimana diinginkan, perlu proses validasi untuk melakukannya. Proses validasi dapat dilakukan menggunakan Integration Testing karena dapat membuktikan bahwa sebuah fungsi pada program telah memenuhi spesifikasinya [10]. Integration Testing dilakukan dengan memastikan bahwa komponen-komponen pada program saling cocok dan memberikan hasil sesuai harapan [38].

2.2.14 Peramban Web

Dengan semakin bertambahnya kemampuan teknis internet, bertambah juga kebutuhan untuk sebuah aplikasi web yang mempengaruhi berubahnya cara sebuah halaman web untuk menampilkan sebuah data dan informasi [39].

JavaScript telah mencapai platform web secara luas dan menjadi bahasa pemrograman multifungsi dengan adanya ES6 yang diimplementasikan pada mesin peramban web sejak tahun 2017 [40].

2.2.15 Peluncuran Perangkat Lunak

Peluncuran (*deployment*) perangkat lunak merupakan proses eksekusi dari sebuah perangkat lunak. Peluncuran adalah aktifitas dari sebuah perangkat lunak setelah proses pembangunan perangkat lunak [41]. Peluncuran dilakukan untuk mendapatkan timbal balik berupa perbaruan, konfigurasi ulang, adaptasi, penonaktifan, dan peluncuran ulang pada sebuah perangkat lunak [42].

BAB III
METODE PENELITIAN

3.1 Alur Penelitian

Penelitian tugas akhir ini akan dilakukan membangun aplikasi penilaian esai singkat otomatis berbasis web dengan metode pengembangan *Test-Driven Development* (TDD) dan diuji dengan metode dua metode pengujian, yakni *System Usability Scale* (SUS) dan *Black Box*. Alur penelitian dapat dilihat pada *flowchart* pada Gambar 3.1 dibawah ini.

Gambar 3.1 *Flowchart* Penelitian

3.2 Penjabaran Langkah Penelitian

Dapat diperhatikan melalui Gambar 3.1 sebelumnya bahwa pada penelitian ini memiliki 4 fase sebagaimana pada gambar tersebut. Penjelasan setiap fase penelitian ini adalah sebagai berikut:

3.2.1 Identifikasi Masalah

Pada tahap ini, peneliti akan menyediakan alasan dilakukannya penelitian dengan mengumpulkan permasalahan nyata yang memerlukan tindakan dan dapat diselesaikan dengan disiplin teknik informatika.

3.2.2 Studi Literatur

Dalam sebuah penelitian, diperlukan pengetahuan landasan secara teoritis terhadap permasalahan yang dihadapi dan ingin ditindak. Pengetahuan teoritis ini dapat diperoleh melalui jurnal penelitian terdahulu maupun buku yang berisi materi yang dibutuhkan. Jurnal penelitian terdahulu dan buku yang digunakan akan memiliki kesimpulan terkait satu atau lebih teori yang akan diperlukan pada penelitian ini. Kemudian teori yang sudah dikumpulkan dan dijadikan landasan akan menjadi arahan kemana kemungkinan penelitian yang dilakukan akan berakhir.

3.2.3 Penerapan *Test-Driven Development*

Penelitian ini akan dilakukan menggunakan metode siklus hidup perangkat lunak (SDLC) *Test-Driven Development* (TDD) dengan harapan menjadi solusi atas permasalahan yang ingin diselesaikan. Dengan ini, penelitian akan menekankan pengembangan melalui beberapa kebutuhan yang dijadikan pengujian atas dibuatnya fitur maupun fungsi aplikasi yang dikembangkan. Dengan metode ini, aplikasi yang dikembangkan dapat dipastikan kelayakannya dalam memenuhi semua fitur dan fungsi berdasarkan pengujian sebelumnya.

Telah diketahui sebelumnya bahwa TDD memiliki 3 fase utama yaitu membuat pengujian, membuat kode yang berhasil memenuhi pengujian tersebut, lalu *refactor* kode yang telah berhasil tersebut. Tahap lainnya seperti pemahaman terhadap fitur atau *bug* yang ingin diuji sebelum masuk ke tiga fase utama, dan tahap iterasi atau tindakan terhadap fitur atau *bug* lain yang ingin diuji dan diselesaikan apabila pengujian sebelumnya sudah berhasil tanpa error. Apabila seluruh fitur dan *bug* yang ingin ditindak sudah selesai dibangun, maka aplikasi siap memasuki tahap pengujian

3.2.4 Integrasi Aplikasi dan Model

Pada penelitian ini, aplikasi yang akan dikembangkan akan menggunakan dua model pembelajaran mesin yang telah dilatih sebelumnya. Dua model tersebut yakni dalam bahasa Indonesia dan bahasa Inggris. Alur integrasi aplikasi dengan model yang digunakan dapat dilihat pada Gambar 3.2 berikut ini.

Gambar 3.2 Alur Integrasi Aplikasi dan Model

Integrasi model pembelajaran mesin pada aplikasi digunakan dengan pengguna memilih bahasa yang akan digunakan pada penilaian yang dilakukannya, dengan ini aplikasi akan memilih model dengan bahasa yang tepat.

Selanjutnya aplikasi akan menjalankan model terkait dengan masukkan dari pengguna untuk diproses. Setelah model selesai melakukan penilaian otomatis, model akan menghasilkan objek dan mengembalikannya pada aplikasi. Kemudian aplikasi akan menggunakan objek untuk menampilkan hasil penilaian kepada pengguna untuk dilihat atau diunduh berupa PDF.

3.2.5 Pengujian

Aplikasi yang dikembangkan pada penelitian tugas akhir ini akan diuji menggunakan *System Usability Scale* (SUS) dan *Black Box* untuk memastikan apakah aplikasi penilaian esai otomatis berbasis web yang dikembangkan dapat meningkatkan proses penilaian esai singkat menjadi lebih efisien. Pengujian dilakukan dengan melakukan perbandingan subjektif antara nilai yang diusulkan dari aplikasi dengan nilai yang diperiksa oleh pengajar suatu subjek terhadap jawaban esai dari pertanyaan esai yang diberikan pengajar tersebut. Apabila hasil nilai akhir dari aplikasi dan pribadi pengajar kurang lebih sama, dapat dipastikan bahwa aplikasi dapat bekerja dengan baik.

3.3 Alat dan Bahan Tugas Akhir

Alat dan bahan yang dibutuhkan untuk mendukung jalannya penelitian tugas akhir ini. Alat dan bahan yang dimaksud dapat dilihat sebagai berikut ini:

3.3.1 Alat

Alat yang mendukung dan dibutuhkan dalam pelaksanaan penelitian tugas akhir ini adalah berikut ini:

1. *Notebook* dengan spesifikasi minimum, sistem operasi Windows 8, *processor* Intel Core i3 2330M CPU @ 2,2 GHz, memori 2GB DDR3, grafis NVIDIA GeForce GT 610 (4GB), *hardisk* 500GB. Pada tugas akhir ini digunakan komputer dengan spesifikasi minimum sistem operasi Windows 10, processor AMD Ryzen 5 2500U @ 2 GHz, memori 16GB DDR4, grafis AMD Radeon Vega 8, dan *Solid-State Drive* 512GB.
2. *Smartphone* dengan spesifikasi tipe minimum, OS Android OS v4.1.2 (Jelly Bean), CPU Dual-core 800 MHz, GPU Mali-400, Internal 4 GB, 768 MB RAM. Pada tugas akhir ini digunakan OS Android 9.0 (Pie), CPU Octa-core 2.3GHz, GPU PowerVR GE8320, Internal 128 GB, 4GB RAM.
3. *Node.js* dengan versi minimal 16.16.0. Pada tugas akhir ini menggunakan *Node.js* dengan versi 18.12.1.
4. *Python* dengan versi minimal 3.11.5.

5. *Visual Studio Code* (VSCode) untuk menulis kode program menggunakan versi 1.74.2. Versi dari VSCode dapat berubah seiring adanya pembaruan dari pembangun aplikasi VSCode tersebut sewaktu-waktu.

3.3.2 Bahan

Bahan yang digunakan / diperlukan untuk melakukan penelitian tugas akhir ini adalah sebagai berikut:

1. Model Terlatih untuk memproses penilaian esai singkat berbahasa Indonesia.
2. Dataset dari model terlatih penilaian esai singkat untuk bahasa Indonesia
3. Model Terlatih untuk memproses penilaian esai singkat berbahasa Inggris.
4. Dataset dari model terlatih penilaian esai singkat untuk bahasa Inggris

3.4 Metode Pengembangan

Dalam pengembangan aplikasi penilaian esai singkat otomatis berbasis web pada penelitian ini, akan digunakan salah satu metode pengembangan siklus hidup perangkat lunak *Agile*, khususnya *Test-Driven Development* (TDD). Alur pengembangan perangkat lunak menggunakan TDD pada penelitian ini dapat diperhatikan pada Gambar 3.3 dibawah ini.

Gambar 3.3 Alur Penelitian *Test-Driven Development*

Berdasarkan diagram alir yang dapat dilihat pada Gambar 3.3, TDD memiliki tahapan berupa kebutuhan, pembuatan uji coba, pembuatan kode, refaktor, dan peluncuran. Kemudian dibawah ini merupakan penjelasan dari masing-masing tahapan pada metode pengembangan TDD:

3.4.1 Kebutuhan Fitur / Fungsi

Tahap ini merupakan tahap pertama dari metode pengembangan TDD guna mendapatkan kebutuhan atau fungsi pada aplikasi yang akan dibangun. Pada tahap ini, pengembang diharapkan dapat mengerti kebutuhan yang diminta untuk dapat dilakukan oleh perangkat lunak yang sedang dikembangkan.

3.4.1.1 Wawancara

Dalam penelusuran kebutuhan fungsi untuk aplikasi yang akan dibangun pada tugas akhir ini, dibutuhkan informasi dari calon pengguna yaitu pengajar yang sudah biasa melakukan penilaian esai pada pelajar. Penulis melakukan wawancara kepada

sepuluh guru/dosen. Pada Tabel 3.1 dapat dilihat data narasumber, dan pada Tabel 3.2 merupakan pertanyaan dan jawaban dari wawancara yang dilakukan.

Tabel 3.1 Data Narasumber

No	Nama Responden	Peran	Lokasi
1	Zamroni Riza S.E	Guru	SMA Sapta Kharisma Jakarta
2	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M.	Dosen	Universitas Suryadarma
3	Sukma Erawan, S.Kom	Guru	SMA Negeri 68 Jakarta
4	Alan Syahwandi, S.Pd.I	Guru	SMP Negeri 3 Bogor
5	Ajeng Yeni Ratnasari, S.Pd	Guru	Pondok Pesantren Alfassalam
6	Dita Putri Bestari S.Pd	Guru	SMA Negeri 65 Jakarta
7	Ayu Nulantika S.Pd	Guru	SMA Negeri 65 Jakarta
8	Virgawaty Dewy, S.Pd	Guru	SMA Negeri 65 Jakarta
9	Ahmad Arfandi, S.Kom	Guru	SMA Negeri 65 Jakarta
10	Muhammad Lutfi, S.Pd	Guru	SMA Negeri 113 Jakarta

Tabel 3.2 Hasil Wawancara

No	Pertanyaan	Jawaban
1	Apabila ada aplikasi penilai esai otomatis, apa fungsi/fitur yang ibu/bapak harapkan dari aplikasi tersebut?	Aplikasi harus bisa menentukan kata kunci dari jawaban yang mana bisa digunakan untuk penilaian. Aplikasi juga harus bisa menilai sekaligus banyak karena apabila tidak, akan tetap memakan energi seperti menilai esai biasa
2	Apakah ada permasalahan terkait jumlah pelajar dan pengajar?	Sebagai pembuat soal, biasanya perlu waktu lama untuk mengoreksi 300 jawaban hanya untuk satu soal, maka perlu alat bantu untuk mempercepat ini.
3	Apa ada fungsi/fitur lain yang kira-kira diperlukan?	Aplikasi harus bisa menilai jawaban dari singkatan, karena tidak seluruh jawaban akan menggunakan kata formal untuk setiap kalimatnya, khususnya untuk pelajar yang masih SD atau SMP.
4	Terkait efisiensi, apa yang sekiranya dibutuhkan untuk sebuah aplikasi penilaian esai otomatis?	Aplikasi harus bisa menilai banyak jawaban sekaligus. Kalau bisa terintegrasi kepada media yang sering digunakan untuk jawaban esai seperti Google Form
5	Apakah ada fitur terkait efektifitas aplikasi yang dibutuhkan?	Aplikasi harus bisa menampilkan perbandingan jawaban pengajar dan pelajar serta bobot nilai dari

No	Pertanyaan	Jawaban
		aplikasinya. Karena nilai untuk sebuah soal esai itu tidak satu dan nol, bisa saja nilainya setengah (50% benar).
6	Aplikasi sebaiknya dapat diakses hanya pada komputer atau bebas?	Untuk utamanya pada komputer, karena pengajar biasanya menilai dari komputer sekolah, namun sebaiknya dibuat juga untuk versi ponsel untuk mengantisipasi adanya kebutuhan khusus saat menggunakan aplikasi tersebut, seperti saat ada pelajar yang ketinggalan jawabannya untuk dinilai bersama yang lain.

3.4.1.2 User Story

Tahap ini dilakukan untuk mendapatkan fungsi dan fitur utama pada aplikasi yang dikembangkan pada kebutuhan fungsional dan non-fungsional. *User story* didapatkan dengan melakukan wawancara kepada calon pengguna sistem dengan mengetahui apa yang diinginkan dan mengapa mereka menginginkan hal tersebut. *User story* dari hasil wawancara yang dilakukan dapat dilihat pada Tabel 3.3 dibawah ini:

Tabel 3.3 Kumpulan *User Story*

No	Nama	Objective	Alasan
1	Zamroni Riza S.E	Dapat melakukan penilaian secara masal	Aplikasi harus bisa dilakukan sekaligus agar proses penilaian bisa cepat
2	Alan Syahwandi, S.Pd.I		
3	Ajeng Yeni Ratnasari, S.Pd		
4	Dita Putri Bestari S.Pd	Dapat menilai esai dengan pembobotan nilai dan melihat bobot nilai setiap jawaban	Nilai kebenaran esai tidak selalu 1 atau 0, karena ada esai yang mungkin hanya benar 50 persen.
5	Ayu Nulantika S.Pd		
6	Muhammad Lutfi, S.Pd		
7	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M.	Dapat dipakai langsung dari aplikasi ujian	Aplikasi harus dapat dipakai langsung dari aplikasi yang sering digunakan ujian esai
8	Ahmad Arfandi, S.Kom	Dapat melakukan penilaian untuk satu jawaban	Aplikasi bisa dipakai untuk menilai satu esai apabila ada kebutuhan khusus
9	Virgawaty Dewy, S.Pd		
10	Sukma Erawan, S.Kom		

Berdasarkan wawancara yang dilakukan, calon pengguna membutuhkan aplikasi yang bisa melakukan penilaian esai secara masal untuk sehingga pengajar tidak perlu melakukan aktifitas yang berulang (menyalin dan menempel jawaban) untuk setiap jawaban pelajar dan penilaian esai secara tunggal dengan memasukkan satu pasangan jawaban pengajar dan pelajar untuk kasus khusus (seperti esai yang telat dikumpul oleh pelajar). Aplikasi juga diharapkan dapat menampilkan bobot nilai jawaban pelajar karena menurut narasumber nilai pada esai tidak benar atau salah seluruhnya, melainkan ada bobot pada seberapa benar jawaban esai tersebut. Kemudian untuk penilaian esai secara masal harus menggunakan tipe dokumen yang sama dari aplikasi ujian pada umumnya. Kebutuhan ini diuraikan menjadi kebutuhan teknis yang dijelaskan pada sub-bab 3.4.1.2 Kebutuhan Fungsional.

3.4.1.3 Kebutuhan Fungsional

Pada pengimplementasian aplikasi penilaian esai otomatis berbasis *website* ini, perlu beberapa kebutuhan fungsional yang harus dipenuhi untuk aplikasi dapat bekerja semestinya. Kebutuhan fungsional aplikasi berbasis *website* dapat dilihat pada Tabel 3.4 berikut ini.

Tabel 3.4 Kebutuhan Fungsional

ID	Deskripsi
F0	Aplikasi *website* dapat menentukan algoritma penilaian yang digunakan berdasarkan permintaan pengguna melalui tombol
F1A	Aplikasi *website* dapat menerima masukkan dua buah jawaban berupa teks melalui *input form*
F1B	Aplikasi *website* dapat melakukan penilaian otomatis dari kedua masukkan pada fungsi dengan ID F1A
F2A	Aplikasi *website* dapat menerima masukkan berkas dengan ekstensi CSV, sebuah jawaban berupa teks, nama kolom nama pelajar berupa teks, dan nama kolom jawaban pelajar berupa teks.
F2B	Aplikasi *website* dapat melakukan penilaian berdasarkan masukkan pada fungsi dengan ID F2A

3.4.1.4 Kebutuhan Non-Fungsional

Untuk meningkatkan kenyamanan pengguna saat menggunakan aplikasi penilaian esai otomatis berbasis website ini, diperlukan beberapa kebutuhan non-fungsional

yang ingin dikembangkan. Kebutuhan ini dapat meningkatkan retensi pengguna agar lebih nyaman saat menggunakan aplikasi ini. Pada Tabel 3.5 dibawah ini dapat dilihat kebutuhan non-fungsional yang akan dipenuhi.

Tabel 3.5 Kebutuhan Non-Fungsional

ID	Deskripsi
NF1	Aplikasi *website* dapat digunakan pada berbagai perangkat, komputer dan mobile
NF2	Aplikasi *website* dapat dijalankan pada peramban web yang mendukung ES6
NF3	Aplikasi *website* memiliki tampilan yang menyesuaikan dengan ukuran layar perangkat yang digunakan (responsif)

3.4.1.5 Use Case Diagram

Use Case Diagram adalah diagram yang digunakan untuk menggambarkan sebuah interaksi yang dapat terjadi antara pengguna dan sistem. Pada Gambar 3.4 dibawah ini dapat diperhatikan interaksi pengguna pada sistem yang terjadi pada sistem yang hendak dibangun.

Gambar 3.4 *Use Case Diagram* Aplikasi

Pada Gambar 3.4 tersebut, aktor merupakan pengajar yang ingin melakukan pengecekkan jawaban esai milik pelajar. Pengguna dapat melihat hasil jawaban yang diperiksa dengan mengirimkan esai yang ingin diperiksa, kemudian memilih bahasa dan teknik pengecekkan yang diinginkan.

3.4.1.6 Activity Diagram

Activity diagram digunakan untuk menggambarkan alur pekerjaan dari aktivitas pada suatu sistem. Pada Gambar 3.5 berikut ini dapat dilihat *activity diagram* dari aplikasi yang akan dibangun pada penelitian ini.

Gambar 3.5 *Activity Diagram* Aplikasi

Pada Gambar 3.5 tersebut dapat diperhatikan bahwa aplikasi dimulai dengan pengguna memilih bahasa dan metode penilaian yang akan digunakan. Kemudian berdasarkan pilihan tersebut, sistem akan menentukan masukkan yang dapat dilakukan oleh pengguna. Apabila semua masukkan telah diisi, kemudian sistem akan meminta *server* melakukan penilaian yang diminta oleh pengguna berdasarkan bahasa dan metode penilaian. Setelah *server* selesai melakukan penilaian, *server* akan mengirimkan hasil penilaian untuk ditampilkan pada aplikasi agar dapat dilihat oleh pengguna.

3.4.1.7 Mockup

Untuk pengguna dapat menggunakan aplikasi web yang akan dibangun, perlu dibuat sebuah desain antarmuka agar dapat mengoperasikan aplikasinya. Perancangan antarmuka ini dilakukan pada tahap kebutuhan sebagai perencanaan terhadap gambaran aplikasi yang akan dikembangkan. Rancangan antarmuka aplikasi dapat dilihat sebagai berikut.

Gambar 3.6 *Mockup* Halaman Home

Saat pengguna mengakses halaman *website*, pengguna akan ditunjukkan halaman beranda seperti pada Gambar 3.6 dimana pengguna dapat memilih bahasa yang akan

digunakan, dan metode penilaian yang akan digunakan dengan sebelah kiri merupakan metode tunggal dan sebelah kanan merupakan metode jamak. Pengguna juga dapat mengakses halaman bantuan dengan menekan logo tanda tanya yang ada pada pojok kanan atas halaman beranda web.

Gambar 3.7 *Mockup* Metode Penilaian Tunggal

Pada Gambar 3.7 merupakan tampilan yang akan dilihat pengguna apabila memilih metode penilaian tunggal. Pada halaman ini, pengguna sebagai pengajar dapat mengisi jawaban pengajar di sebelah kiri dan jawaban pelajar untuk dinilai di sebelah kanan.

Gambar 3.8 *Mockup* Hasil Metode Penilaian Tunggal

Setelah memproses penilaian otomatis, aplikasi akan mengarahkan pengguna ke halaman hasil penilaian dengan metode penilaian 1 sebagaimana ditunjukkan pada Gambar 3.8 dimana pengguna dapat melihat jawban pengajar dan jawaban pelajar dengan persentase hasil jawabannya.

Gambar 3.9 *Mockup* Metode Penilaian Jamak

Pada Gambar 3.9 diatas merupakan halaman yang akan ditampilkan apabila pengguna memilih metode penilaian secara jamak. Pada halaman ini, pengguna dapat mengisi jawaban pengajar dan memasukkan berkas CSV pada aplikasi dan menentukan judul kolom berisi nama dan jawaban pelajar untuk diperiksa terhadap jawaban pengajar.

Gambar 3.10 merupakan halaman yang akan ditampilkan kepada pengguna untuk melihat hasil penilaian otomatis oleh aplikasi. Aplikasi akan menampilkan halaman yang akan menunjukkan jawaban pengajar pada bagian atas dan tabel berisi nama, jawaban, dan persentase nilai pelajar.

Gambar 3.10 *Mockup* Hasil Metode Penilaian Jamak

Gambar 3.11 *Mockup* Halaman Bantuan

Apabila pada halaman beranda seperti pada Gambar 3.6 pengguna memutuskan untuk menekan logo tanda tanya, maka pengguna akan diarahkan ke halaman bantuan seperti pada Gambar 3.11. Pada halaman ini, pengguna dapat melihat cara menggunakan aplikasi penilaian esai otomatis.

3.4.2 Pembuatan Uji Coba (Scenario Testing)

Pada tahap ini, pengembang membuat uji coba lengkap dengan ekspektasi berupa hasil dari pengujian tersebut. Pengembang membuat uji coba berdasarkan satu kebutuhan yang telah didapat dari tahap sebelumnya.

Aplikasi yang dikembangkan akan memiliki 3 skenario uji coba yang harus berhasil setiapnya dengan ekspektasi pada Tabel 3.6 berikut ini:

Tabel 3.6 Skenario Uji Coba

Skenario	**Masukkan Jawaban Pengajar**	**Masukkan Jawaban Pelajar**	**Ekspektasi Keluaran**
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
SC2	Teks untuk uji coba masal “To sumulate the behaviour of portions of the desired software product.”	Dokumen CSV berisi jawaban pelajar (“dataset.csv”)	Persentase Hasil Penilaian
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 asam dan basa dalam tubuh “	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 sebagai pemasok energi. Makanan berkarbohidrat dan serat tinggi dapat membantu memperlancar proses pada pencernaan. ”	55 dari 100

3.4.3 Pembuatan Kode (Coding)

Pada tahap ini, pengembang akan membuat fungsi pada perangkat lunak yang dikembangkan yang dapat memenuhi uji coba pada tahapan sebelumnya. Apabila ekspektasi hasil uji coba tidak dapat terpenuhi, maka kode dianggap gagal dan harus dibuat ulang hingga berhasil meraih ekspektasi yang diharapkan dari uji coba tersebut.

3.4.4 Refaktor (Refactor)

Saat kode untuk suatu fungsi berdasarkan suatu uji coba sudah berhasil dibuat, maka akan dilakukan pembersihan kode. Pembersihan disini berarti kode akan tulis sedemikian rupa agar tidak bersifat ambigu bagi pengembang lain yang mungkin akan melanjutkan pengembangan perangkat lunak yang sedang dikembangkan saat ini di masa depan. Setelah melakukan refaktorisasi kode, pengembang dapat kembali ke tahap kebutuhan dan mengulang seluruh tahapan ini dengan kebutuhan yang berbeda hingga semua kebutuhan berhasil dibuat.

3.5 Pre-Deployment Testing (Integration Testing)

Pengujian ini dibuat untuk memastikan bahwa komponen-komponen berbeda pada program berjalan semestinya berdasarkan skenario yang telah dibuat sebelumnya. Akan dibuat sebuah fungsi pada *server* untuk dapat memproses skenario pengujian dengan rancangan pada Tabel 3.7 berikut:

Tabel 3.7 *Integration Testing*

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

ID	Metode	Input 1	Input 2	Output
		Indonesia.	berbahasa Indonesia.	berbahasa Indonesia

Pada Tabel 3.7 tersebut, akan dibuat 3 fungsi yang masing-masing dapat menerima teks jawaban pengajar dalam bahasa Indonesia atau Inggris, kemudian menerima jawaban pelajar untuk diperiksa melalui teks atau berkas CSV. Fungsi tersebut selanjutnya akan memproses kedua masukkan, dan menampilkan persentase hasil penilaian jawaban pada *console*.

3.6 Peluncuran

Apabila semua kebutuhan terhadap perangkat lunak yang dibangun sudah dibuat dan telah teruji bekerja dengan baik berdasasrkan pengujian sebelumya, maka akan dilakukan tahap peluncuran, yaitu kondisi dimana perangkat lunak yang dikembangkan sudah siap digunakan secara umum. Tahap ini dilakukan untuk mendapatkan timbal balik dari pengguna terkait perangkat lunak yang dikembangkan.

3.7 Post-Deployment Testing

Setelah aplikasi diluncurkan, maka akan didapatkan timbal balik dari pengguna untuk pengujian SUS dan *Blackbox*. Dengan ini dapat dilakukan perbaikan apabila terdapat bug yang tidak terdeteksi selama pembangunan aplikasi dan dapat diluncurkan kembali. Apabila tahap ini telah membuahkan hasil yang memuaskan, maka penelitian ini dapat dianggap selesai.

3.7.1 System Usability Scale

Seperti yang telah dilansir pada Bab 2, aplikasi yang dihasilkan penelitian ini akan diuji menggunakan SUS pada sekelompok orang untuk mendapatkan hasil kegunaan aplikasi dari penelitian yang dilakukan. Dengan itu, pada Tabel 3.8 berikut adalah daftar kuesioner yang akan diajukan kepada pengguna setelah menggunakan aplikasi yang bersangkutan.

Tabel 3.8 Kuesioner *System Usability Scale*

No	Pertanyaan
1	Saya berpikir akan menggunakan aplikasi penilaian esai otomatis berbasis web ini lagi
2	Saya merasa aplikasi penilaian esai otomatis berbasis web ini sulit digunakan
3	Saya merasa aplikasi penilaian esai otomatis berbasis web ini mudah digunakan
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini
5	Saya merasa fitur-fitur aplikasi penilaian esai otomatis berbasis web ini berjalan dengan semestinya
6	Saya merasa banyak hal yang tidak konsisten / tidak serasi pada aplikasi penilaian esai otomatis berbasis web ini
7	Saya merasa orang lain akan memahami cara menggunakan aplikasi penilaian esai otomatis berbasis web ini dengan cepat
8	Saya merasa aplikasi penilaian esai otomatis berbasis web ini membingungkan
9	Saya merasa tidak ada hambatan dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan aplikasi penilaian esai otomatis berbasis web ini

3.7.2 Black Box

Pada waktu yang bersamaan, akan dilakukan pengujian Black Box untuk memastikan bahwa aplikasi yang dikembangkan bekerja sesuai dengan yang diharapkan. Pada tahap ini, penulis mungkin akan mendapatkan kecacatan program (*bug*), namun dengan ini penulis dapat memperbaiki aplikasi terkait dan menyempurnakannya. Daftar pengujian Black Box dapat dilihat secara rinci pada Tabel 3.9 dibawah ini.

Tabel 3.9 Daftar Pengujian Fungsional Black Box

PARENT ID	ID	Pengujian	Ekspektasi
F1A	F1A-1	Aplikasi web dapat menerima masukkan teks sebagai jawaban pengajar	Aplikasi web 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
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

PARENT ID	ID	Pengujian	Ekspektasi
			Inggris
	F1B-2	Aplikasi web dapat menerima masukka dari fungsi F1A untuk penilaian otomatis tunggal bahasa Indonesia	Aplikasi 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.
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
	F2A-2	Aplikasi web dapat menerima masukkan teks sebagai kolom nama pada kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan nama kolom yang berisi nama pelajar pada 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
	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
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
	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
	F2B-5	Aplikasi dapat menampilkan hasil penilaian bahasa Inggris berdasarkan data .json yang dihasilkan oleh algoritma penilaian bahasa Inggris pada	Aplikasi web berhasil menggunakan dokumen .json yang telah dihasilkan penilaian otomatis bahasa Inggris sebelumnya untuk

PARENT ID	ID	Pengujian	Ekspektasi
		halaman web	ditampilkan 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
	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
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

Pada Tabel 3.9 tersebut dapat diperhatikan ID dan PARENT ID untuk setiap pengujian *black box*. PARENT ID melambangkan bahwa pengujian memiliki fokus utama untuk suatu pengujian yang dipecah menjadi beberapa pengujian lainnya yang dilambangkan sebagai ID.

PARENT ID F1A merupakan pengujian untuk memastikan bahwa aplikasi web dapat menerima masukkan teks dari pengguna sebagai variabel pada *backend* untuk dilakukan penilaian jawaban.

PARENT ID F1B merupakan pengujian untuk memastikan bahwa masukkan teks dari F1A dapat digunakan sebagai variabel penilaian pada model *machine learning* yang akan digunakan yang penentuannya merupakan pengujian dari ID F0.

Selanjutnya, PARENT ID F2A adalah sekumpulan pengujian untuk menguji fungsi penilaian jamak melalui aplikasi web yang akan dikembangkan. Pengujian F2A memastikan bahwa *backend* akan menerima berkas CSV dan rinciannya yang akan

digunakan untuk memproses penilaian jamak untuk setiap model *machine learning* terpilih yang mana pengujiannya merupakan bagian dari PARENT ID F2B.

Setelah pengujian fungsional, dilakukan juga pengujian non-fungsional pada aplikasi yang dikembangkan penelitian ini dengan pengujian sebagai berikut.

Tabel 3.10 Daftar Pengujian Non Fungsional Black Box

PARENT ID	**ID**	**Pengujian**	**Ekspektasi**
NF1	NF1-1	Aplikasi dapat dijalankan pada ponsel	Aplikasi web berhasil diakses dan digunakan pada ponsel
	NF1-2	Aplikasi dapat dijalankan pada komputer	Aplikasi web 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
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

Pada Tabel 3.10, PARENT ID NF1 merupakan pengujian untuk tampilan yang dapat berubah (*responsive*) pada ukuran layar apapun yang sifatnya non-fungsional.

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

4.1.1.2 Frontend Halaman Metode Penilaian Tunggal

Pengguna dapat memilih “Cek satu jawaban” pada halaman home pada Gambar 4.1, kemudian pengguna akan diarahkan ke halaman metode penilaian tunggal untuk memasukkan 1 jawaban pengajar dan 1 jawaban pelajar untuk diperiksa seperti pada Gambar 4.2. Pengguna juga dapat melihat bahasa yang digunakan di pojok kanan atas halaman untuk memastikan pengguna tidak salah pilih bahasa yang digunakan untuk menilai esai.

Gambar 4.2 *Frontend* Halaman Metode Penilaian Tunggal

4.1.1.3 Frontend Halaman Hasil Metode Penilaian Tunggal

Setelah menekan tombol “periksa” pada halaman Gambar 4.2, pengguna akan diarahkan ke halaman hasil penilaian seperti pada Gambar 4.3 untuk melihat hasil penilaian dari metode penilaian tunggal.

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 Tabel 4.1.

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

Tabel 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 Tabel 4.2. Rute ini hanya akan menampilkan halaman bantuan.

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

Tabel 4.2 Kode Program Rute /help

4.1.2.3 Rute Untuk Menentukan Pilihan

Pada halaman *home*, pengguna dapat menetukan bahasa esai yang akan digunakan untuk melakukan penilaian dan metode penilaian esainya melalui permintaan *post* ke rute "/option". Pada Tabel 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("/");
  }
});
```

Tabel 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 Tabel 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) => {
    });
});
```

Tabel 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 Tabel 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) => {
    });
});
```

Tabel 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 Tabel 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) => {
  });
});
```

Tabel 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 Tabel 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 {
  }
});
```

Tabel 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;
});
```

Tabel 4.8 Kode Konversi Hasil Penilaian Menjadi PDF

Pada kode 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);
```

Tabel 4.9 Kode Konversi Berkas CSV ke JSON

Kode program 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 jawban 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 Tabel 4.10.

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

Tabel 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 Tabel 4.11.

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

Tabel 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 Tabel 4.12.

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

Tabel 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 Tabel 4.13 berikut ini.

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

Tabel 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 4.14 berikut dapat diperhatikan hasil *scenario testing* yang dilakukan pada pengembangan aplikasi terkait.

Tabel 4.14 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 4.14, 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 penilaia 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 4.14. Model *machine learing* 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 masukkan jawaban pengajar dan jawaban pelajar untuk diperiksa, kemudian membandingkan keluaran hasil jawabannya untuk dibandingkan. Hasil percobaan ini berhasil karena keluaran dari pengecekkan 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 masukkan dan keluaran spesifik. Pada Tabel 4.15 dibawah ini dapat dilihat hasil pengujian unit yang telah dilakukan beserta hasilnya.

Tabel 4.15 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 4.15 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 4.15.

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 UT1 *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 4.7 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.

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 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. Dengan ini, aplikasi Easysay dapat diakses melalui peramban web pada tautan https://shercosta.github.io/easysay/.

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 dilucurkan, 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.

Gambar 4.10 Sampel Google Form SUS

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

Tabel 4.16 Responden SUS

Nama	**Peran**	**Skor Untuk Pertannyaan**									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Muhammad Lutfi, S.Pd	Guru	5	1	4	1	5	1	4	2	3	2
Ajeng Yeni Ratnasari, S.Pd	Guru	5	3	4	2	5	3	5	2	4	1
Dita Putri Bestari S.Pd	Guru	5	1	3	1	4	2	5	1	5	2
Ayu Nulantika S.Pd	Guru	4	2	4	1	5	3	4	1	4	2
Ahmad Arfandi, S.Kom	Guru	4	1	5	1	4	2	5	2	5	3
Virgawaty Dewy, S.Pd	Guru	5	2	5	1	4	3	3	1	4	1
Isma Nurfitriah,	Guru	4	3	5	2	5	2	4	1	3	2

Nama	Peran	Skor Untuk Pertannyaan									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
S.Pd											
Renny Noviani Riandy, S.S	Guru	5	1	4	1	5	1	5	2	5	1
Sukma Erawan, S.Kom	Guru	3	2	3	1	5	2	5	1	4	2
Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	4	1	5	2	4	3	4	3	4	1
Zamroni Riza S.E	Guru	5	2	5	1	5	2	5	2	3	3
Alan Syahwandi, S.Pd.I	Guru	5	2	5	1	3	1	4	1	5	2
Sahlul Efendi, S.pd	Guru	4	2	5	1	4	1	3	2	4	2
Yayah Shulhiyyah, S.Pd	Guru	5	3	4	3	3	3	4	1	5	3
Iyan Maulana, M.pd.i	Guru	4	3	3	1	4	1	5	2	5	3
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 sebagai berikut

Tabel 4.17 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

No	Pertanyaan	Skor Rata-Rata
7	Saya merasa orang lain akan memahami cara menggunakan aplikasi penilaian esai otomatis berbasis web ini dengan cepat	4.3125
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 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 sebagai berikut.

Tabel 4.18 Skor SUS Ganjil dan Genap

Nama	Peran	Skor Untuk Pertannyaan									
		1	2	3	4	5	6	7	8	9	10
Muhammad Lutfi, S.Pd	Guru	4	4	3	4	4	4	3	3	2	3
Ajeng Yeni Ratnasari, S.Pd	Guru	4	2	3	3	4	2	4	3	3	4
Dita Putri Bestari S.Pd	Guru	4	4	2	4	3	3	4	4	4	3
Ayu Nulantika	Guru	3	3	3	4	4	2	3	4	3	3

Nama	Peran	Skor Untuk Pertannyaan									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
S.Pd											
Ahmad Arfandi, S.Kom	Guru	3	4	4	4	3	3	4	3	4	2
Virgawaty Dewy, S.Pd	Guru	4	3	4	4	3	2	2	4	3	4
Isma Nurfitriah, S.Pd	Guru	3	2	4	3	4	3	3	4	2	3
Renny Noviani Riandy, S.S	Guru	4	4	3	4	4	4	4	3	4	4
Sukma Erawan, S.Kom	Guru	2	3	2	4	4	3	4	4	3	3
Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	3	4	4	3	3	2	3	2	3	4
Zamroni Riza S.E	Guru	4	3	4	4	4	3	4	3	2	2
Alan Syahwandi, S.Pd.I	Guru	4	3	4	4	2	4	3	4	4	3
Sahlul Efendi, S.pd	Guru	3	3	4	4	3	4	2	3	3	3
Yayah Shulhiyyah, S.Pd	Guru	4	2	3	2	2	2	3	4	4	2
Iyan Maulana, M.pd.i	Guru	3	2	2	4	3	4	4	3	4	2
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 4.19 Total Skor SUS Setiap Responden Dikali 2.5

Nama	Peran	Total Skor x2.5
Muhammad Lutfi, S.Pd	Guru	85
Ajeng Yeni Ratnasari, S.Pd	Guru	80
Dita Putri Bestari S.Pd	Guru	87.5
Ayu Nulantika S.Pd	Guru	80

Nama	Peran	Total Skor x2.5
Ahmad Arfandi, S.Kom	Guru	85
Virgawaty Dewy, S.Pd	Guru	82.5
Isma Nurfitriah, S.Pd	Guru	77.5
Renny Noviani Riandy, S.S	Guru	95
Sukma Erawan, S.Kom	Guru	80
Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M	Dosen	77.5
Zamroni Riza S.E	Guru	82.5
Alan Syahwandi, S.Pd.I	Guru	87.5
Sahlul Efendi, S.pd	Guru	80
Yayah Shulhiyyah, S.Pd	Guru	70
Iyan Maulana, M.pd.i	Guru	77.5
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 dibawah ini dapat diperhatikan hasil pengujian *black box* yang dilakukan.

Tabel 4.20 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

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
	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 kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan nama kolom yang berisi nama pelajar pada berkas CSV	Aplikasi web telah berhasil menyimpan nama kolom yang berisi nama pelajar pada berkas CSV
	F2A-3	Aplikasi web dapat menerima masukkan teks	Aplikasi web berhasil menyimpan nama	Aplikasi web telah berhasil menyimpan nama

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
		sebagai kolom jawaban pada kumpulan jawaban pelajar	kolom yang berisi jawaban pelajar pada berkas CSV	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 pada halaman web	Aplikasi web telah berhasil menggunakan dokumen .json yang telah dihasilkan penilaian otomatis bahasa Inggris sebelumnya untuk ditampilkan pada halaman web
	F2B-2	Aplikasi web dapat membuat dokumen .json berdasarkan pengujian F1A-1,	Aplikasi web berhasil membuat dokumen .json setelah menentukan	Aplikasi web telah berhasil membuat dokumen .json setelah menentukan

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
		dan F1A-2 atau F2A-1 agar cocok dengan syarat penilaian pada algoritma penilaian otomatis bahasa Indonesia	pasangan nama dan jawaban pada berkas CSV	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 4.15 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 4.21 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	Aplikasi web berhasil diakses	Aplikasi web telah berhasil diakses

PARENT ID	ID	Pengujian	Ekspektasi	Hasil
		komputer	dan digunakan pada komputer	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 4.16, 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

Gambar 4.12 Aplikasi pada Google Chrome

Gambar 4.13 Aplikasi pada Opera GX

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.

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. Setelahya, 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 masukkan 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.

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 *runtimer* 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 pennilaian 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.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil akhir penelitian yang dilakukan ini, maka dapat disimpulkan bahwa:

1. Implementasi aplikasi penilaian esai singkat dapat dilakukan menggunakan SDLC *Test-Driven Development* (TDD) dengan membangun sebuah fungsi / fitur hingga berhasil terlebih dahulu, kemudian melakukan refaktor agar kode tidak berantakan. Setelahya, pengembang dapat membangun fungsi / fitur selanjutnya, dan proses ini akan berulang hingga tidak ada lagi fungsi / fitur yang hendak ditambahkan.
2. Model *pre-trained machine learning* dapat diterapkan pada aplikasi dengan menerapkan konsep *child-process* dimana sebuah proses menjalankan sebuah proses turunan lainnya. Pada aplikasi yang dibangun, *backend* dari aplikasi akan menjalankan model untuk melakukan penilaian pada esai yang ingin dinilai dan mendapatkan hasilnya apabila proses penilaian telah selesai dilakukan.
3. Hasil penilaian pada aplikasi dan pada model *machine learning* dapat dipastikan sama dengan membandingkan hasil penilaian yang didapat dari aplikasi dan model *machine learning* terkait secara langsung. Skor yang dihasilkan kedua sumber ini kemudian dikonversi menjadi rentang 0 hingga 100 dan dibandingkan hasilnya. Hasil menunjukkan bahwa hasil penilaian pada aplikasi yang dikembangkan dan model *machine learning* sama, yang berarti tidak ada perubahan selama proses implementasi model pada saat mengembangkan aplikasi. Hasil pengujian yang dilakukan menunjukkan skor 1.128299 dari 5.0 pada model dan 22% pada aplikasi yang menerapkan model. Hal ini menunjukkan bahwa hasil yang didapat adalah sama, karena 1.128299 merupakan 22% dari 5.0
4. Untuk pengguna, aplikasi penilaian esai singkat dapat dilakukan dengan menjalankannya pada peramban web yang mendukung ES6 dan pengguna dapat melihat tahapan cara menggunakan aplikasi pada halaman bantuan. Sedangkan untuk sisi teknikal, aplikasi dapat digunakan pada mesin yang menjalankan

runtimer Node.js untuk menjalankan *server* aplikasi, dan Python sebagai interpreter untuk menjalankan model *machine learning* yang digunakan pada penelitian ini.

5.2 Saran

Pada akhir penelitian ini, peneliti menemukan beberapa saran yang dapat diterapkan terkait penelitian ini ataupun kelanjutan penelitian ini kedepannya.

1. **Saran untuk Lembaga Edukasi**: Seperti yang telah dijelaskan pada latar belakang terkait tidak efisiennya penilaian esai apabila dilakukan secara manual. Dengan landasan itu, sebaiknya setiap lembaga edukasi (sekolah, universitas, atau institusi) dapat menyediakan aplikasi ini pada server untuk dapat digunakan oleh pihak pengajar yang hendak melakukan penilaian esai agar proses penilaian esai dapat dilakukan lebih cepat.
2. **Saran untuk Peneliti Selanjutnya**: Pada aplikasi yang dikembangkan penelitian ini, hanya memiliki dua bahasa yang dapat digunakan untuk penilaian esai yaitu bahasa Indonesia dan Inggris. Hal ini dikarenakan penelitian ini menggunakan 2 model *machine learning* yang dapat melakukan pemeriksaan esai otomatis menggunakan bahasa Indonesia dan Inggris. Apabila di masa depan ada model *machine learning* yang dapat melakukan hal serupa dengan bahasa lain, maka peneliti selanjutnya dapat mengimplementasi model dengan bahasa baru tersebut kedalam aplikasi ini agar semakin dapat digunakan untuk banyak bahasa.

DAFTAR PUSTAKA (minimal 20)

[1] U. Nations, "Population." Accessed: Dec. 25, 2022. [Online]. Available: https://www.un.org/en/global-issues/population

[2] B. P. Statistik, "Jumlah Penduduk Pertengahan Tahun (Ribu Jiwa), 2020-2022." Accessed: Dec. 26, 2022. [Online]. Available: https://www.bps.go.id/indicator/12/1975/1/jumlah-penduduk-pertengahan-tahun.html

[3] B. P. Statistik, "Jumlah Penduduk Menurut Kelompok Umur dan Jenis Kelamin, 2021." Accessed: Dec. 26, 2022. [Online]. Available: https://www.bps.go.id/indikator/indikator/view_data_pub/0000/api_pub/YW40a21pdTU1cnJxOGt6dm43ZEdoZz09/da_03/1

[4] F. Oktariano and H. Hastuti, "Buku Panduan penulisan Esai Berdasarkan Analisis Historical Thinking," vol. 2, no. 4, 2020.

[5] E. Amorim, M. Cançado, and A. Veloso, "Automated Essay Scoring in the Presence of Biased Ratings," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana: Association for Computational Linguistics, 2018, pp. 229–237. doi: 10.18653/v1/N18-1021.

[6] D. Fortunato and J. Bernardino, "Progressive web apps: An alternative to the native mobile Apps," in *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, Caceres: IEEE, Jun. 2018, pp. 1–6. doi: 10.23919/CISTI.2018.8399228.

[7] M. M. Moe and University of Computer Studies, Hpa-An, Kayin State, Myanmar, "Comparative Study of Test-Driven Development TDD, Behavior-Driven Development BDD and Acceptance Test–Driven Development ATDD," *Int. J. Trend Sci. Res. Dev.*, vol. Volume-3, no. Issue-4, pp. 231–234, Jun. 2019, doi: 10.31142/ijtsrd23698.

[8] B. George and L. Williams, "A structured experiment of test-driven development," *Inf. Softw. Technol.*, vol. 46, no. 5, pp. 337–342, Apr. 2004, doi: 10.1016/j.infsof.2003.09.011.

[9] J. B. Ibarra *et al.*, "Development of the Low Cost Classroom Response System Using Test-Driven Development Approach and Analysis of the Adaptive Capability of Students Using Sequential Minimal Optimization Algorithm," in *2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA)*, Tokyo, Japan: IEEE, Apr. 2019, pp. 689–693. doi: 10.1109/IEA.2019.8714889.

[10] H. K. N. Leung and L. White, "A study of integration testing and software regression at the integration level," in *Proceedings. Conference on Software Maintenance 1990*, San Diego, CA, USA: IEEE Comput. Soc. Press, 1990, pp. 290–301. doi: 10.1109/ICSM.1990.131377.

[11] D. Pal and V. Vanijja, "Perceived usability evaluation of Microsoft Teams as an online learning platform during COVID-19 using system usability scale and technology acceptance model in India," *Child. Youth Serv. Rev.*, vol. 119, p. 105535, Dec. 2020, doi: 10.1016/j.childyouth.2020.105535.

[12] P. Vlachogianni and N. Tselios, “Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review,” *J. Res. Technol. Educ.*, vol. 54, no. 3, pp. 392–409, May 2022, doi: 10.1080/15391523.2020.1867938.

[13] J. R. Lewis, “Psychometric Evaluation of the PSSUQ Using Data from Five Years of Usability Studies,” *Int. J. Hum.-Comput. Interact.*, vol. 14, no. 3–4, pp. 463–488, Sep. 2002, doi: 10.1080/10447318.2002.9669130.

[14] U. Ependi, T. B. Kurniawan, and F. Panjaitan, “SYSTEM USABILITY SCALE VS HEURISTIC EVALUATION: A REVIEW,” *Simetris J. Tek. Mesin Elektro Dan Ilmu Komput.*, vol. 10, no. 1, pp. 65–74, Apr. 2019, doi: 10.24176/simet.v10i1.2725.

[15] T. Hidayat and M. Muttaqin, “Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis,” vol. 6, 2018.

[16] xinyu chang and jing Li, “Improvement of Excel data processing function based on Spring MVC framework,” in *Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)*, Y. Lu and C. Cheng, Eds., Beijing, China: SPIE, Dec. 2022, p. 25. doi: 10.1117/12.2661778.

[17] I. Glantz and H. Hurtig, “Express.js and Ktor web server performance A comparative study”.

[18] D. Smilkov *et al.*, “TensorFlow.js: Machine Learning for the Web and Beyond”.

[19] S. L. Ingólfsdóttir, H. Loftsson, J. F. Daðason, and K. Bjarnadóttir, “Nefnir: A high accuracy lemmatizer for Icelandic.” arXiv, Jul. 27, 2019. Accessed: Jan. 05, 2023. [Online]. Available: http://arxiv.org/abs/1907.11907

[20] B. Jose and S. Abraham, “Performance analysis of NoSQL and relational databases with MongoDB and MySQL,” *Mater. Today Proc.*, vol. 24, pp. 2036–2043, 2020, doi: 10.1016/j.matpr.2020.03.634.

[21] C. Kaner, “An Introduction to Scenario Testing”.

[22] G. Steinfeld, “5 steps of test-driven development,” *IBM Developer*, Feb. 06, 2020. Accessed: Jan. 30, 2023. [Online]. Available: https://developer.ibm.com/articles/5-steps-of-test-driven-development/

[23] Institut Teknologi Sepuluh Nopember *et al.*, “A Different Approach on Automated Use Case Diagram Semantic Assessment,” *Int. J. Intell. Eng. Syst.*, vol. 14, no. 1, pp. 496–505, Feb. 2021, doi: 10.22266/ijies2021.0228.46.

[24] Dept. of Computer Science & Engineering, BITM, VTU, Ballari, India., Dr. R. N. Kulkarni, C. K. Srinivasa, and Dept. of Computer Science & Engineering, BITM, VTU, Ballari, India., “Novel approach to transform UML Sequence diagram to Activity diagram,” *J. Univ. Shanghai Sci. Technol.*, vol. 23, no. 07, pp. 1247–1255, Jul. 2021, doi: 10.51201/JUSST/21/07300.

[25] M. Shirole and R. Kumar, “Constrained permutation-based test scenario generation from concurrent activity diagrams,” *Innov. Syst. Softw. Eng.*, vol. 17, no. 4, pp. 343–353, Dec. 2021, doi: 10.1007/s11334-021-00389-4.

[26] A. Firdaus, S. Widodo, A. Sutrisman, S. G. F. Nasution, and R. Mardiana, “RANCANG BANGUN SISTEM INFORMASI PERPUSTAKAAN MENGGUNAKAN WEB SERVICE PADA JURUSAN TEKNIK KOMPUTER POLSRI,” vol. 5, 2019.

[27] A. Ismail and K. S. Kuppusamy, “Web accessibility investigation and identification of major issues of higher education websites with statistical measures: A case study of college websites,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 3, pp. 901–911, Mar. 2022, doi: 10.1016/j.jksuci.2019.03.011.

[28] R. S. Malik, J. Patra, and M. Pradel, “NL2Type: Inferring JavaScript Function Types from Natural Language Information,” in *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, Montreal, QC, Canada: IEEE, May 2019, pp. 304–315. doi: 10.1109/ICSE.2019.00045.

[29] M. Shcherbakov, M. Balliu, and C.-A. Staicu, “Silent Spring: Prototype Pollution Leads to Remote Code Execution in Node.js”.

[30] H. Brar, T. Kaur, and Y. Rajoria, “The Better Comparison between PHP, Python-web & Node.js,” *web* ...

[31] A. Romanelli, S. Serbout, and C. Pautasso, “ExpressO: From Express.js implementation code to OpenAPI interface descriptions”.

[32] G. Langdale and D. Lemire, “Parsing gigabytes of JSON per second,” *VLDB J.*, vol. 28, no. 6, pp. 941–960, Dec. 2019, doi: 10.1007/s00778-019-00578-5.

[33] S. Hamza, M. Sarosa, and P. B. Santoso, “Sistem Koreksi Soal Essay Otomatis Dengan Menggunakan Metode Rabin Karp,” vol. 7, no. 2, 2013.

[34] S. Burrows, I. Gurevych, and B. Stein, “The Eras and Trends of Automatic Short Answer Grading,” *Int. J. Artif. Intell. Educ.*, vol. 25, no. 1, pp. 60–117, Mar. 2015, doi: 10.1007/s40593-014-0026-8.

[35] J. Hao and T. K. Ho, “Machine Learning Made Easy: A Review of *Scikit-learn* Package in Python Programming Language,” *J. Educ. Behav. Stat.*, vol. 44, no. 3, pp. 348–361, Jun. 2019, doi: 10.3102/1076998619832248.

[36] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, “Stanza: A Python Natural Language Processing Toolkit for Many Human Languages.” arXiv, Apr. 23, 2020. Accessed: Jan. 09, 2023. [Online]. Available: http://arxiv.org/abs/2003.07082

[37] A. Bangor, P. T. Kortum, and J. T. Miller, “An Empirical Evaluation of the System Usability Scale,” *Int. J. Hum.-Comput. Interact.*, vol. 24, no. 6, pp. 574–594, Jul. 2008, doi: 10.1080/10447310802205776.

[38] M. E. Delamaro, J. C. Maidonado, and A. P. Mathur, “Interface Mutation: an approach for integration testing,” *IEEE Trans. Softw. Eng.*, vol. 27, no. 3, pp. 228–247, Mar. 2001, doi: 10.1109/32.910859.

[39] A. Popov, J. Bilokin, T. Solianyk, and K. Vasylchenko, “Development of the system to provide cross-browser compatibility of web application,” in *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Kyiv, Ukraine: IEEE, May 2018, pp. 117–122. doi: 10.1109/DESSERT.2018.8409111.

[40] K. Paltoglou, V. E. Zafeiris, N. A. Diamantidis, and E. A. Giakoumakis, “Automated refactoring of legacy JavaScript code to ES6 modules,” *J. Syst. Softw.*, vol. 181, p. 111049, Nov. 2021, doi: 10.1016/j.jss.2021.111049.

[41] A. Dearle, “Software Deployment, Past, Present and Future,” in *Future of Software Engineering (FOSE ’07)*, Minneapolis, MN: IEEE, May 2007, pp. 269–284. doi: 10.1109/FOSE.2007.20.

[42] R. S. Hall, D. Heimbigner, and A. L. Wolf, “A Cooperative Approach to Support Software Deployment Using the Software Dock”.

LAMPIRAN

Lampiran I Kode Program

1. views/header.ejs

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Easysay</title>

    <link
      rel="icon"
      href="/images/easysay-icon.png"
      type="image"
    />

    <!-- Bootstrap CSS -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
      crossorigin="anonymous"
    />

    <!-- our CSS -->
    <link rel="stylesheet" href="/css/master.css" />

    <!-- FontAwesome JS -->
    <script
      src="https://kit.fontawesome.com/6b522f63d7.js"
      crossorigin="anonymous"
    ></script>

    <!-- XLS to JSON -->
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/xlsx/0.16.2/xlsx.full.mi
n.js"></script>
  </head>

  <body>
    <!-- <h2>Header</h2> -->
```

2. views/footer.ejs

```
<!-- <h3>Footer</h3> -->
<!-- Bootstrap JS -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
```

```
ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfAdVQSZe"
      crossorigin="anonymous"
    ></script>
</body>

</html>
```

3. views/title.ejs

```
<div>
  <a href="/">
    <h1><i class="fa-solid fa-file-circle-check"></i>Easysay</h1>
  </a>
</div>
```

4. views/titleSubtitle.ejs

```
<div>
  <a href="/">
    <h1><i class="fa-solid fa-file-circle-check"></i>Easysay</h1>
    <p>Cek esai cepat!</p>
  </a>
</div>
```

5. views/help/help.ejs

```
<%-include('header')%>

<div class="container-fluid">
  <%-include('titleSubtitle')%>

  <div class="accordion accordion-flush" id="accordionFlushExample">
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingOne">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseOne"
          aria-expanded="false"
          aria-controls="flush-collapseOne"
        >
          Bagaimana cara kerja Easysay?
        </button>
      </h2>
      <div
        id="flush-collapseOne"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingOne"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Easysay bekerja dengan cara membandingkan jawaban
          pelajar(siswa/mahasiswa) yang ingin diperiksa dengan
jawaban yang
```

```
          menurut pengajar(guru/dosen) benar. Maka dari itu aplikasi
ini hanya
          dapat digunakan untuk memeriksa esai yang memiliki jawaban
benar /
          teoritis.
          <br />
          <img
            src="https://i.ibb.co/xFbWHXn/Help-1-1.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 1.1"
          />
          <br />
          Bagian <strong>Kunci Jawaban</strong> diisi oleh jawaban
pengajar,
          sedangkan bagian <strong>Jawaban</strong> diisi oleh
jawaban pelajar
          yang ingin diperiksa. Kedua kolom diisi oleh pemeriksa
jawaban.
          <br />
          Easysay <em>bukan</em> aplikasi E-Learning dimana siswa
dapat menjawab
          soal dan langsung diperiksa.
          <br />
          Pada gambar diatas, 58.64 merupakan nilai hasil
pemeriksaan esai.
        </div>
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingTwo">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseTwo"
          aria-expanded="false"
          aria-controls="flush-collapseTwo"
        >
          Apa bahasa yang dapat digunakan pada Easysay?
        </button>
      </h2>
      <div
        id="flush-collapseTwo"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingTwo"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Easysay dapat memeriksa esai dengan jawaban berbahasa
          <strong>Indonesia</strong> atau <strong>Inggris</strong>.
          <em>"Atau"</em> berarti esai yang diperiksa sebaiknya
tidak berisi
          campuran dari kedua bahasa tersebut.
          <br />
          <img
            src="https://i.ibb.co/rvpY3m0/Help-2-1-JPG.png"
            class="rounded img-fluid shadow-lg"
            alt="Help 2.1"
          />
```

```
          <br />
          Bahasa yang ingin digunakan untuk memeriksa esai dapat
dipilih pada
          halaman utama aplikasi.
        </div>
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingThree">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseThree"
          aria-expanded="false"
          aria-controls="flush-collapseThree"
        >
          Bagaimana cara cek satu jawaban?
        </button>
      </h2>
      <div
        id="flush-collapseThree"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingThree"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Untuk memeriksa satu jawaban, anda harus memilih terlebih
dahulu
          bahasa yang akan digunakan. Kemudian anda dapat menekan
tombol
          <strong>Cek satu jawaban</strong> seperti pada gambar.
          <br />
          <img
            src="https://i.ibb.co/h1kNXhj/Help-3-1.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 3.1"
          />
          <br />
          Kemudian anda dapat memasukkan jawaban pengajar dan
jawaban pelajar
          pada kolomnya masing-masing, setelah itu menekan tombol
          <strong>Periksa</strong>.
          <br />
          <img
            src="https://i.ibb.co/cQgtTZ4/Help-3-2.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 3.2"
          />
          <br />
          Setelah itu, anda akan dialihkan ke halaman hasil
penilaian.
          <br />
          <img
            src="https://i.ibb.co/j6ymQJg/Help-3-3.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 3.3"
          />
          <br />
```

```
        </div>
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingFour">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseFour"
          aria-expanded="false"
          aria-controls="flush-collapseFour"
        >
          Bagaimana cara cek banyak jawaban sekaligus?
        </button>
      </h2>
      <div
        id="flush-collapseFour"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingFour"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Anda dapat melakukan pengecekan banyak jawaban pelajar
dengan satu
          jawaban pengajar untuk dijadikan landasan perbandingan
esai.
          <br />
          Anda harus memilih bahasa yang ingin digunakan terlebih
dahulu,
          kemudian meenekan tombol <strong>Cek banyak
jawaban</strong> seperti
          pada gambar.
          <br />
          <img
            src="https://i.ibb.co/g9KQNq1/Help-4-1.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 4.1"
          />
          <br />
          Kemudian anda dapat memasukkan jawaban pengajar pada kolom
yang sudah
          disediakan. <br />
          Setelah itu, anda perlu mengunggah <strong>file</strong>
yang berisi
          nama dan jawaban pelajar untuk diperiksa masing-masing
terhadap
          jawaban pengajar.
          <br />
          Jangan lupa untuk menspesifikasikan kolom yang berisi
          <strong>nama pelajar</strong> pada <em>Kolom Nama
Pelajar</em>, dan
          kolom yang berisi <strong>jawaban pelajar</strong> pada
          <em>Kolom Jawaban Pelajar</em>.
          <br />
          <img
            src="https://i.ibb.co/n03MSss/Help-4-2.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 4.2"
```

```
          />
          <br />
          Setelah itu, anda dapat menekan tombol
<strong>Periksa</strong> dan
          anda akan dialihkan ke halaman hasil penilaian esai.
          <br />
          <img
            src="https://i.ibb.co/WVW5WJQ/Help-4-3.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 4.3"
          />
          <br />
          Kemudian apabila anda dapat mengunduh hasil penilaian
dengan menekan
          tombol <strong>Export to PDF</strong>
        </div>
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingFive">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseFive"
          aria-expanded="false"
          aria-controls="flush-collapseFive"
        >
          Apa itu file dengan format CSV?
        </button>
      </h2>
      <div
        id="flush-collapseFive"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingFive"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          File CSV adalah salah satu file Excel yang berisi tabel
dan kolom.
          File ini dapat dibaca oleh Easysay untuk diperiksa setiap
tabel dan
          kolomnya untuk pasangan nama dan jawaban yang telah
dispesifikasikan
          saat anda hendak melakukan pemeriksaan banyak esai
sekaligus.
          <br />
          <img
            src="https://i.ibb.co/z4fw16g/Help-5-1.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 5.1"
          />
          <br />
          Pengembang menggunakan file CSV karena Google Form
merupakan salah
          satu teknologi yang sering digunakan untuk Ujian Online di
semua
          kalangan.
        </div>
```

```
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingSix">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseSix"
          aria-expanded="false"
          aria-controls="flush-collapseSix"
        >
          Bagaimana mendapatkan file CSV dari Google Form?
        </button>
      </h2>
      <div
        id="flush-collapseSix"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingSix"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Anda dapat membuka Google Form anda dan mengunduh jawaban
dengan
          bentuk CSV seperti tertera pada gambar berikut ini.
          <br />
          <img
            src="https://i.ibb.co/LdTd1bt/Help-5-2.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 5.2"
          />
          <br />
        </div>
      </div>
    </div>
    <div class="accordion-item">
      <h2 class="accordion-header" id="flush-headingSeven">
        <button
          class="accordion-button collapsed"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#flush-collapseSeven"
          aria-expanded="false"
          aria-controls="flush-collapseSeven"
        >
          Bagaimana menentukan Nama dan Jawaban pelajar pada file
CSV?
        </button>
      </h2>
      <div
        id="flush-collapseSeven"
        class="accordion-collapse collapse"
        aria-labelledby="flush-headingSeven"
        data-bs-parent="#accordionFlushExample"
      >
        <div class="accordion-body">
          Anda dapat memeriksa file CSV yang ingin digunakan.
          <br />
          <img
```

```
            src="https://i.ibb.co/BC4dchL/Help-6-1.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 6.1"
          />
          <br />
          <br />
          Pada contoh, kolom nama adalah "Name" karena pada kolom
dibawahnya
          terdapat nama pelajar yang ingin diperiksa.
          <br />
          Kemudian kolom jawaban pelajar adalah "I would like (come)
to the
          party with you." karena pada kolom dibawahnya terdapat
jawaban pelajar
          yang ingin diperiksa.
          <br />
          <img
            src="https://i.ibb.co/vHjKPYh/Help-6-2.jpg"
            class="rounded img-fluid shadow-lg"
            alt="Help 6.2"
          />
          <br />
          Nama kolom nama dan nama kolom jawaban pelajar dapat
disalin langsung
          dari excel saat membuka file CSV dan ditempel langsung
pada aplikasi
          Easysay.
          <br />
        </div>
      </div>
    </div>
  </div>
</div>

<%-include('footer')%>
```

6. views/home/home.ejs

```
<%-include('header')%>
<div class="container-fluid">
  <div class="help_icon">
    <a href="/help" class="fs-3">
      <abbr title="Help"><i class="fa-solid fa-circle-
question"></i></abbr
    ></a>
  </div>
  <div class="text-center">
    <%-include('titleSubtitle')%>
    <form action="/option" method="post">
      <select class="form_language select_local" name="language"
id="language">
        <option selected disabled>Pilih Bahasa Esai</option>
        <option value="Indonesia">Indonesia</option>
        <option value="Inggris">Inggris</option>
      </select>
      <br />
      <br />
```

```
      <div class="container text-center">
        <div class="row">
          <div class="col col-12 col-lg-5">
            <!-- <h2>Cek satu jawaban</h2> -->
            <button
              id="single"
              type="submit"
              name="select"
              value="single"
              class="btn btn-primary btn-lg"
            >
              Cek satu jawaban
            </button>
            <br />
            <p class="service_explanation">
              Cek satu jawaban esai pelajar berdasarkan jawaban yang
anda anggap
              benar dengan mengetikkan jawabannya.
            </p>
          </div>
          <div class="col col-md-2 d-none d-lg-block">
            <div class="vr h-100"></div>
          </div>
          <div class="col d-lg-none">
            <hr />
          </div>
          <div class="col col-12 col-lg-5">
            <!-- <h2>Cek banyak jawaban</h2> -->
            <button
              id="bulk"
              type="submit"
              name="select"
              value="bulk"
              class="btn btn-outline-primary btn-lg"
            >
              Cek banyak jawaban
            </button>
            <br />
            <p class="service_explanation">
              Cek banyak jawaban esai pelajar berdasarkan satu
jawaban yang anda
              anggap benar dengan mengunggah dokumen berisi jawaban
esai para
              pelajar.
            </p>
          </div>
        </div>
      </div>
    </form>
  </div>
</div>

<%-include('footer')%>
```

7. views/indonesia/bulkJQuery.ejs

```
<%-include('header')%>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%-include('title')%></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-solid fa-file-csv"></i>Indonesia</h1>
    </div>
  </div>
  <form
    action="/idbulk"
    id="idbulk"
    enctype="multipart/form-data"
    method="post"
    class=""
  >
    <input type="hidden" name="language" value="Indonesia" />
    <div class="row">
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="teacherAnswer">Jawaban
Pengajar</label>
        <textarea
          class="form-control"
          name="teacherAnswer"
          id="teacherAnswer"
        ></textarea>
      </div>
      <div class="col col-12 col-lg-6 mb-5">
        <div>
          <label class="form-label" for="studentAnswer"
            >Excel Jawaban Pelajar</label
          >
          <input
            class="form-control mb-3"
            type="file"
            accept=".csv"
            multiple="multiple"
            name="studentAnswers"
            id="studentAnswer"
          />
          <label class="form-label" for="nameColumns">Kolom Nama
Pelajar</label>
          <input
            class="form-control mb-3"
            type="text"
            name="nameColumns"
            id="nameColumns"
          />
          <label class="form-label" for="answerColumns"
            >Kolom Jawaban Pelajar</label
          >
          <input
            class="form-control mb-3"
            type="text"
```

```
            name="answerColumns"
            id="answerColumns"
          />
        </div>
      </div>
    </div>
    <div class="text-center">
      <button class="btn btn-lg btn-danger mb-5"
type="submit">Periksa</button>
    </div>
  </form>
</div>

<%-include('footer')%>
```

8. views/indonesia/single.ejs

```
<%-include('header')%>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%-include('title')%></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-regular fa-pen-to-square"></i>Indonesia</h1>
    </div>
  </div>
  <form action="/ids" method="post" class="">
    <input type="hidden" name="language" value="Indonesia" />
    <input type="hidden" name="docid" id="docid" />
    <div class="row">
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="teacherAnswer">Jawaban
Pengajar</label>
        <textarea
          class="form-control"
          name="teacherAnswer"
          id="teacherAnswer"
        ></textarea>
      </div>
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="studentAnswer">Jawaban
Pelajar</label>
        <textarea
          class="form-control"
          name="studentAnswer"
          id="studentAnswer"
        ></textarea>
      </div>
    </div>
    <div class="text-center">
      <button class="btn btn-lg btn-warning mb-5"
type="submit">Periksa</button>
    </div>
  </form>
</div>

<script src="./js/milisecondDate.js"></script>
```

```
<%-include('footer')%>
```

9. views/inggris/bulk.ejs

```
<%-include('header')%>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%-include('title')%></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-solid fa-file-csv"></i>Inggris</h1>
    </div>
  </div>
  <form
    action="/enb"
    id="enbulk"
    enctype="multipart/form-data"
    method="post"
    class=""
  >
    <input type="hidden" name="language" value="Inggris" />
    <div class="row">
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="teacherAnswer">Jawaban
Pengajar</label>
        <textarea
          class="form-control"
          name="teacherAnswer"
          id="teacherAnswer"
        ></textarea>
      </div>
      <div class="col col-12 col-lg-6 mb-5">
        <div>
          <label class="form-label" for="studentAnswer"
            >Excel Jawaban Pelajar</label
          >

          <input
            class="form-control mb-3"
            type="file"
            accept=".csv"
            multiple="multiple"
            name="studentAnswers"
            id="studentAnswer"
          />
          <label class="form-label" for="nameColumns">Kolom Nama
Pelajar</label>
          <input
            class="form-control mb-3"
            type="text"
            name="nameColumns"
            id="nameColumns"
          />
          <label class="form-label" for="answerColumns"
            >Kolom Jawaban Pelajar</label
          >
```

```
          <input
            class="form-control mb-3"
            type="text"
            name="answerColumns"
            id="answerColumns"
          />
        </div>
      </div>
    </div>
    <div class="text-center">
      <button
        class="btn btn-lg btn-outline-danger mb-5"
        id="button"
        type="submit"
      >
        Periksa
      </button>
    </div>
  </form>
</div>

<%-include('footer')%>
```

10. views/inggris/single.ejs

```
<%-include('header')%>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%-include('title')%></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-regular fa-pen-to-square"></i>Inggris</h1>
    </div>
  </div>
  <form action="/ens" method="post" class="">
    <input type="hidden" name="language" value="Inggris" />
    <input type="hidden" name="docid" id="docid" />
    <div class="row">
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="teacherAnswer">Jawaban
Pengajar</label>
        <textarea
          class="form-control"
          name="teacherAnswer"
          id="teacherAnswer"
        ></textarea>
      </div>
      <div class="col col-12 col-lg-6 mb-5">
        <label class="form-label" for="studentAnswer">Jawaban
Pelajar</label>
        <textarea
          class="form-control"
          name="studentAnswer"
          id="studentAnswer"
        ></textarea>
      </div>
    </div>
```

```
    <div class="text-center">
      <button class="btn btn-lg btn-outline-warning mb-5"
type="submit">
        Periksa
      </button>
    </div>
  </form>
</div>

<script src="./js/milisecondDate.js"></script>

<%-include('footer')%>
```

11. views/result/bulkResult.ejs

```
<%- include("header") %>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%- include("title") %></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-solid fa-file-csv"></i><%= language %></h1>
    </div>
  </div>
  <div>
    <p class="form-label">Kunci Jawaban</p>
    <div class="form-control">
      <p><%= keyAnswer %></p>
    </div>
  </div>
  <!-- <div class="text-center container-fluid">
    <p>Mohon mengisi survey kepuasan anda terhadap Easysay</p>
    <button type="button" class="btn btn-info">
      <a
        href="https://forms.gle/FApH5uzhVbRB4NtY9"
        target="_blank"
        class="link-body-emphasis"
        >Survey 3 menit</a
      >
    </button>
  </div> -->
  <div>
    <p class="form-label">Hasil Penilaian</p>
    <div class="row">
      <table class="table">
        <thead>
          <tr>
            <th scope="col">Nama</th>
            <th scope="col">Jawaban</th>
            <th scope="col">Nilai</th>
          </tr>
        </thead>
        <tbody>
          <% for( let i = 0; i < givenObjectKeyLength; i++ ) { %>
          <tr>
            <td><%= objectName[i] %></td>
            <td><%= objectAnswer[i] %></td>
```

```
            <td><%= ((objectScore[i]/5)*100).toFixed(2) %>%</td>
          </tr>
          <br />
          <% } %>
        </tbody>
      </table>
    </div>
  </div>
</div>
<div class="container-fluid text-center mb-2">
  <button type="button" id="print" class="btn btn-secondary">
    Export to PDF
  </button>
</div>
<script src="js/printResult.js"></script>

<%- include("footer") %>
```

12. views/result/singleResult.ejs

```
<%- include("header") %>

<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><%- include("title") %></div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-regular fa-pen-to-square"></i><%= language
%></h1>
    </div>
  </div>

  <div>
    <p class="form-label">Kunci Jawaban</p>
    <div class="form-control">
      <p><%= keyAnswer %></p>
    </div>
  </div>
  <!-- <div class="text-center container-fluid">
    <p>Mohon mengisi survey kepuasan anda terhadap Easysay</p>
    <button type="button" class="btn btn-info">
      <a
        href="https://forms.gle/FApH5uzhVbRB4NtY9"
        target="_blank"
        class="link-body-emphasis"
        >Survey 3 menit</a
      >
    </button>
  </div> -->
  <div>
    <div class="form-label">
      <div class="row mt-3">
        <div class="w-auto">Jawaban</div>
        <div class="col"><hr /></div>
        <div class="w-auto"><%= score %></div>
      </div>
    </div>
    <div class="form-control">
```

```
      <p><%= answer %></p>
    </div>
  </div>
</div>

<%- include("footer") %>
```

13. models/english/aprilModel.py

```
# %%

import nltk
from nltk.tokenize import word_tokenize

# nltk.download("punkt")
# nltk.download("averaged_perceptron_tagger")
from nltk.stem import WordNetLemmatizer

# nltk.download("wordnet")
# nltk.download("omw-1.4")
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
from textblob import TextBlob
import pandas as pd
import string
import re
import sys
from nltk.corpus import stopwords

# nltk.download("stopwords")
from nltk.stem import SnowballStemmer
from textblob import TextBlob
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.metrics.pairwise import cosine_similarity

# %% [markdown]
# **Importing Dataset**

# %%

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


# %% [markdown]
# **Text Preprocessing**

# %% [markdown]
# Spelling Correction

# %%
# Spelling Correction

full_df["keyAnswer"] = full_df["keyAnswer"].astype(str)
```

```
full_df["keySpell"] = full_df["keyAnswer"].apply(
    lambda txt: "".join(TextBlob(txt).correct())
)

# %%
full_df["studentAnswer"] = full_df["studentAnswer"].astype(str)
full_df["studentSpell"] = full_df["studentAnswer"].apply(
    lambda txt: "".join(TextBlob(txt).correct())
)

# %%
full_df[["keyAnswer", "keySpell"]]

# %% [markdown]
# Converting Abbreviation

# %%
contractions_dict = {
    "'s": " is",
    "ain't": "is not",
    "aren't": "are not",
    "can't": "cannot",
    "can't've": "cannot have",
    "'cause": "because",
    "could've": "could have",
    "couldn't": "could not",
    "couldn't've": "could not have",
    "didn't": "did not",
    "doesn't": "does not",
    "don't": "do not",
    "hadn't": "had not",
    "hadn't've": "had not have",
    "hasn't": "has not",
    "haven't": "have not",
    "he'd": "he would",
    "he'd've": "he would have",
    "he'll": "he will",
    "he'll've": "he he will have",
    "he's": "he is",
    "how'd": "how did",
    "how'd'y": "how do you",
    "how'll": "how will",
    "how's": "how is",
    "I'd": "I would",
    "I'd've": "I would have",
    "I'll": "I will",
    "I'll've": "I will have",
    "I'm": "I am",
    "I've": "I have",
    "i'd": "i would",
    "i'd've": "i would have",
    "i'll": "i will",
    "i'll've": "i will have",
    "i'm": "i am",
    "i've": "i have",
    "isn't": "is not",
    "it'd": "it would",
    "it'd've": "it would have",
    "it'll": "it will",
```

```
    "it'll've": "it will have",
    "it's": "it is",
    "let's": "let us",
    "ma'am": "madam",
    "mayn't": "may not",
    "might've": "might have",
    "mightn't": "might not",
    "mightn't've": "might not have",
    "must've": "must have",
    "mustn't": "must not",
    "mustn't've": "must not have",
    "needn't": "need not",
    "needn't've": "need not have",
    "o'clock": "of the clock",
    "oughtn't": "ought not",
    "oughtn't've": "ought not have",
    "shan't": "shall not",
    "sha'n't": "shall not",
    "shan't've": "shall not have",
    "she'd": "she would",
    "she'd've": "she would have",
    "she'll": "she will",
    "she'll've": "she will have",
    "she's": "she is",
    "should've": "should have",
    "shouldn't": "should not",
    "shouldn't've": "should not have",
    "so've": "so have",
    "so's": "so as",
    "that'd": "that would",
    "that'd've": "that would have",
    "that's": "that is",
    "there'd": "there would",
    "there'd've": "there would have",
    "there's": "there is",
    "they'd": "they would",
    "they'd've": "they would have",
    "they'll": "they will",
    "they'll've": "they will have",
    "they're": "they are",
    "they've": "they have",
    "to've": "to have",
    "wasn't": "was not",
    "we'd": "we would",
    "we'd've": "we would have",
    "we'll": "we will",
    "we'll've": "we will have",
    "we're": "we are",
    "we've": "we have",
    "weren't": "were not",
    "what'll": "what will",
    "what'll've": "what will have",
    "what're": "what are",
    "what's": "what is",
    "what've": "what have",
    "when's": "when is",
    "when've": "when have",
    "where'd": "where did",
    "where's": "where is",
```

```
    "where've": "where have",
    "who'll": "who will",
    "who'll've": "who will have",
    "who's": "who is",
    "who've": "who have",
    "why's": "why is",
    "why've": "why have",
    "will've": "will have",
    "won't": "will not",
    "won't've": "will not have",
    "would've": "would have",
    "wouldn't": "would not",
    "wouldn't've": "would not have",
    "y'all": "you all",
    "y'all'd": "you all would",
    "y'all'd've": "you all would have",
    "y'all're": "you all are",
    "y'all've": "you all have",
    "you'd": "you would",
    "you'd've": "you would have",
    "you'll": "you will",
    "you'll've": "you will have",
    "you're": "you are",
    "you've": "you have",
}

# %%
# Converting Abbreviation

contractions_re = re.compile("(%s)" %
"|".join(contractions_dict.keys()))


def expand_contractions(text, contractions_dict=contractions_dict):
    def replace(match):
        return contractions_dict[match.group(0)]

    return contractions_re.sub(replace, text)


full_df["studentSpell"] = full_df["studentSpell"].apply(
    lambda x: expand_contractions(x)
)
full_df["keySpell"] = full_df["keySpell"].apply(lambda x:
expand_contractions(x))

# %%
full_df[["studentAnswer", "studentSpell"]]

# %% [markdown]
# Case Folding

# %%
# Case Folding

full_df["keyLower"] = full_df["keySpell"].str.lower()
full_df["studentLower"] = full_df["studentSpell"].str.lower()

full_df[["studentSpell", "studentLower"]]
```

```
# %% [markdown]
# Removing Punctuation

# %%
# Removing Punctuation

PUNCT_TO_REMOVE = string.punctuation


def remove_punctuation(text):
    """custom function to remove the punctuation"""
    return text.translate(str.maketrans("", "", PUNCT_TO_REMOVE))


full_df["keyPunct"] = full_df["keyLower"].apply(lambda text:
remove_punctuation(text))
full_df["studentPunct"] = full_df["studentLower"].apply(
    lambda text: remove_punctuation(text)
)

full_df[["studentLower", "studentPunct"]]

# %% [markdown]
# Stopword Removal

# %%
# Stopword Removal

STOPWORDS = set(stopwords.words("english"))


def remove_stopwords(text):
    """custom function to remove the stopwords"""
    return " ".join([word for word in str(text).split() if word not
in STOPWORDS])


full_df["keyStopword"] = full_df["keyPunct"].apply(lambda text:
remove_stopwords(text))
full_df["studentStopword"] = full_df["studentPunct"].apply(
    lambda text: remove_stopwords(text)
)

full_df[["studentPunct", "studentStopword"]]

# %% [markdown]
# Stemming

# %%
# Stemming

stemmer = SnowballStemmer("english")


def stem_words(text):
    return " ".join([stemmer.stem(word) for word in text.split()])
```

```
full_df["studentStem"] = full_df["studentStopword"].apply(lambda
text: stem_words(text))
full_df["keyStem"] = full_df["keyStopword"].apply(lambda text:
stem_words(text))

full_df[["studentStopword", "studentStem"]]

# %% [markdown]
# POS Tagging

# %%
studentSplit = full_df["studentStopword"]
keySplit = full_df["keyStopword"]

studentSplitWords = [nltk.word_tokenize(str(sentence)) for sentence
in studentSplit]
keySplitWords = [nltk.word_tokenize(str(sentence)) for sentence in
keySplit]
full_df["studentSplit"] = studentSplitWords
full_df["keySplit"] = keySplitWords

# %%
# POS Tagging

full_df["keyPostag"] = full_df["keySplit"].apply(nltk.pos_tag)
full_df["studentPostag"] =
full_df["studentSplit"].apply(nltk.pos_tag)

full_df[["keyPostag", "studentPostag"]]

# %% [markdown]
# Lemmatizing

# %%
# Lemmatizing

lemmatizer = WordNetLemmatizer()
wordnet_map = {"N": wordnet.NOUN, "V": wordnet.VERB, "J":
wordnet.ADJ, "R": wordnet.ADV}


def lemmatize_words(text):
    pos_tagged_text = nltk.pos_tag(text.split())
    return " ".join(
        [
            lemmatizer.lemmatize(word, wordnet_map.get(pos[0],
wordnet.NOUN))
            for word, pos in pos_tagged_text
        ]
    )


full_df["studentLemmatized"] = full_df["studentStopword"].apply(
    lambda text: lemmatize_words(text)
)
full_df["keyLemmatized"] = full_df["keyStopword"].apply(
    lambda text: lemmatize_words(text)
)
full_df[["studentStopword", "studentLemmatized"]]
```

```
# %% [markdown]
# **TF-IDF**

# %% [markdown]
# TF

# %%
# TF for Lemmatized

full_df["keyLemmatized"] = full_df["keyLemmatized"].astype(str)
full_df["studentLemmatized"] =
full_df["studentLemmatized"].astype(str)

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)

    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)

    columns = countvectorizer.get_feature_names_out()
    tf = pd.DataFrame(
        terms.toarray(), index=["keyLemmatized",
"studentLemmatized"], columns=columns
    )

    # print(tf)

# %%
# TF for Stem

full_df["keyStem"] = full_df["keyStem"].astype(str)
full_df["studentStem"] = full_df["studentStem"].astype(str)

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer2 = CountVectorizer()
    terms2 = countvectorizer2.fit_transform(docs2)

    columns2 = countvectorizer2.get_feature_names_out()
    tf2 = pd.DataFrame(
        terms2.toarray(), index=["keyStem", "studentStem"],
columns=columns2
    )

    # print(tf2)

# %% [markdown]
# IDF

# %%
# IDF for Lemmatized

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
```

```
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)

    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)

    tfidf_transformer = TfidfTransformer(smooth_idf=True,
use_idf=True)
    tfidf_transformer.fit(terms)

    df_idf = pd.DataFrame(
        tfidf_transformer.idf_,
        index=countvectorizer.get_feature_names_out(),
        columns=["idf_weights"],
    )

    # print(df_idf)


# %%
# IDF for Stem

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer = CountVectorizer()
    terms2 = countvectorizer.fit_transform(docs2)

    tfidf_transformer = TfidfTransformer(smooth_idf=True,
use_idf=True)
    tfidf_transformer.fit(terms2)

    df_idf2 = pd.DataFrame(
        tfidf_transformer.idf_,
        index=countvectorizer.get_feature_names_out(),
        columns=["idf_weights"],
    )

    # print(df_idf2)

# %% [markdown]
# TF-IDF

# %%
# TF-IDF for Lemmatized

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)
    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)
    X = tfidf_transformer.fit_transform(terms)
    tf_idf = pd.DataFrame(X.toarray(),
columns=countvectorizer.get_feature_names_out())
    # print(tf_idf)

# %%
```

```
# TF-IDF for Stem

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer = CountVectorizer()
    terms2 = countvectorizer.fit_transform(docs2)

    X2 = tfidf_transformer.fit_transform(terms2)
    tf_idf2 = pd.DataFrame(
        X2.toarray(),
columns=countvectorizer.get_feature_names_out()
    )
    # print(tf_idf2)

# %% [markdown]
# **Cosine Similarity**

# %%
# Cosine Similarity for Lemmatized

tfidf_vectorizer = TfidfVectorizer()
similarity = []

full_df["keyLemmatized"] = full_df["keyLemmatized"].astype(str)
full_df["studentLemmatized"] =
full_df["studentLemmatized"].astype(str)

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)
    tfidf_matrix = tfidf_vectorizer.fit_transform(docs)
    tfidf_tokens = tfidf_vectorizer.get_feature_names_out()
    df_tfidfvect = pd.DataFrame(
        data=tfidf_matrix.toarray(),
        index=["keyLemmatized", "studentLemmatized"],
        columns=tfidf_tokens,
    )
    # print(df_tfidfvect)

    cosine_sim = cosine_similarity(tfidf_matrix[0], tfidf_matrix[1])
    # print("Cosine Similarity =", cosine_sim[0][0])
    # print("\n")
    similarity.append(cosine_sim[0][0])

full_df["cosineSimilarityLemma"] = similarity

# cosinelemma = full_df['cosineSimilarityLemma']
# cosinelemma.to_csv('cosinelemma.csv')

# %%
# Cosine Similarity for Stem

tfidf_vectorizer2 = TfidfVectorizer()
similarity2 = []

full_df["keyStem"] = full_df["keyStem"].astype(str)
```

```
full_df["studentStem"] = full_df["studentStem"].astype(str)

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    tfidf_matrix2 = tfidf_vectorizer2.fit_transform(docs2)
    tfidf_tokens2 = tfidf_vectorizer2.get_feature_names_out()
    df_tfidfvect2 = pd.DataFrame(
        data=tfidf_matrix2.toarray(),
        index=["keyStem", "studentStem"],
        columns=tfidf_tokens2,
    )
    # print(df_tfidfvect2)

    cosine_sim2 = cosine_similarity(tfidf_matrix2[0],
tfidf_matrix2[1])
    # print("Cosine Similarity =", cosine_sim2[0][0])
    # print("\n")
    similarity2.append(cosine_sim2[0][0])

full_df["cosineSimilarityStem"] = similarity2

# %% [markdown]
# **Scoring**

# %%
# Scoring for Lemmatized
scoring = full_df["cosineSimilarityLemma"]

scoring = scoring.mul(5)

# full_df = full_df.drop('scoreModel', axis=1)
full_df["scoreModelLemma"] = scoring

full_df[["scoreModelLemma"]]

# %%
# Scoring for Stemming
scoring2 = full_df["cosineSimilarityStem"]

scoring2 = scoring2.mul(5)

full_df["scoreModelStem"] = scoring2

full_df[["scoreModelStem"]]

# %% [markdown]
# Uji Analisis (Abaikan)

# %%
full_df.to_json()

# %%
# newfile = open("HasilPenilaianENG.json", "w")
# newfile.write(full_df.to_json())
# newfile.close()

print(full_df.to_json())
```

14. models/english/aprilModelTab.py

```
# %%

import nltk
from nltk.tokenize import word_tokenize

# nltk.download("punkt")
# nltk.download("averaged_perceptron_tagger")
from nltk.stem import WordNetLemmatizer

# nltk.download("wordnet")
# nltk.download("omw-1.4")
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
from textblob import TextBlob
import pandas as pd
import string
import re
import sys
from nltk.corpus import stopwords

# nltk.download("stopwords")
from nltk.stem import SnowballStemmer
from textblob import TextBlob
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.metrics.pairwise import cosine_similarity

# %% [markdown]
# **Importing Dataset**

# %%

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


# %% [markdown]
# **Text Preprocessing**

# %% [markdown]
# Spelling Correction

# %%
# Spelling Correction

full_df["keyAnswer"] = full_df["keyAnswer"].astype(str)
full_df["keySpell"] = full_df["keyAnswer"].apply(
    lambda txt: "".join(TextBlob(txt).correct())
)

# %%
full_df["studentAnswer"] = full_df["studentAnswer"].astype(str)
full_df["studentSpell"] = full_df["studentAnswer"].apply(
    lambda txt: "".join(TextBlob(txt).correct())
)
```

```
# %%
full_df[["keyAnswer", "keySpell"]]

# %% [markdown]
# Converting Abbreviation

# %%
contractions_dict = {
    "'s": " is",
    "ain't": "is not",
    "aren't": "are not",
    "can't": "cannot",
    "can't've": "cannot have",
    "'cause": "because",
    "could've": "could have",
    "couldn't": "could not",
    "couldn't've": "could not have",
    "didn't": "did not",
    "doesn't": "does not",
    "don't": "do not",
    "hadn't": "had not",
    "hadn't've": "had not have",
    "hasn't": "has not",
    "haven't": "have not",
    "he'd": "he would",
    "he'd've": "he would have",
    "he'll": "he will",
    "he'll've": "he he will have",
    "he's": "he is",
    "how'd": "how did",
    "how'd'y": "how do you",
    "how'll": "how will",
    "how's": "how is",
    "I'd": "I would",
    "I'd've": "I would have",
    "I'll": "I will",
    "I'll've": "I will have",
    "I'm": "I am",
    "I've": "I have",
    "i'd": "i would",
    "i'd've": "i would have",
    "i'll": "i will",
    "i'll've": "i will have",
    "i'm": "i am",
    "i've": "i have",
    "isn't": "is not",
    "it'd": "it would",
    "it'd've": "it would have",
    "it'll": "it will",
    "it'll've": "it will have",
    "it's": "it is",
    "let's": "let us",
    "ma'am": "madam",
    "mayn't": "may not",
    "might've": "might have",
    "mightn't": "might not",
    "mightn't've": "might not have",
    "must've": "must have",
```

```
    "mustn't": "must not",
    "mustn't've": "must not have",
    "needn't": "need not",
    "needn't've": "need not have",
    "o'clock": "of the clock",
    "oughtn't": "ought not",
    "oughtn't've": "ought not have",
    "shan't": "shall not",
    "sha'n't": "shall not",
    "shan't've": "shall not have",
    "she'd": "she would",
    "she'd've": "she would have",
    "she'll": "she will",
    "she'll've": "she will have",
    "she's": "she is",
    "should've": "should have",
    "shouldn't": "should not",
    "shouldn't've": "should not have",
    "so've": "so have",
    "so's": "so as",
    "that'd": "that would",
    "that'd've": "that would have",
    "that's": "that is",
    "there'd": "there would",
    "there'd've": "there would have",
    "there's": "there is",
    "they'd": "they would",
    "they'd've": "they would have",
    "they'll": "they will",
    "they'll've": "they will have",
    "they're": "they are",
    "they've": "they have",
    "to've": "to have",
    "wasn't": "was not",
    "we'd": "we would",
    "we'd've": "we would have",
    "we'll": "we will",
    "we'll've": "we will have",
    "we're": "we are",
    "we've": "we have",
    "weren't": "were not",
    "what'll": "what will",
    "what'll've": "what will have",
    "what're": "what are",
    "what's": "what is",
    "what've": "what have",
    "when's": "when is",
    "when've": "when have",
    "where'd": "where did",
    "where's": "where is",
    "where've": "where have",
    "who'll": "who will",
    "who'll've": "who will have",
    "who's": "who is",
    "who've": "who have",
    "why's": "why is",
    "why've": "why have",
    "will've": "will have",
    "won't": "will not",
```

```
    "won't've": "will not have",
    "would've": "would have",
    "wouldn't": "would not",
    "wouldn't've": "would not have",
    "y'all": "you all",
    "y'all'd": "you all would",
    "y'all'd've": "you all would have",
    "y'all're": "you all are",
    "y'all've": "you all have",
    "you'd": "you would",
    "you'd've": "you would have",
    "you'll": "you will",
    "you'll've": "you will have",
    "you're": "you are",
    "you've": "you have",
}

# %%
# Converting Abbreviation

contractions_re = re.compile("(%s)" %
"|".join(contractions_dict.keys()))


def expand_contractions(text, contractions_dict=contractions_dict):
    def replace(match):
        return contractions_dict[match.group(0)]

    return contractions_re.sub(replace, text)


full_df["studentSpell"] = full_df["studentSpell"].apply(
    lambda x: expand_contractions(x)
)
full_df["keySpell"] = full_df["keySpell"].apply(lambda x:
expand_contractions(x))

# %%
full_df[["studentAnswer", "studentSpell"]]

# %% [markdown]
# Case Folding

# %%
# Case Folding

full_df["keyLower"] = full_df["keySpell"].str.lower()
full_df["studentLower"] = full_df["studentSpell"].str.lower()

full_df[["studentSpell", "studentLower"]]

# %% [markdown]
# Removing Punctuation

# %%
# Removing Punctuation

PUNCT_TO_REMOVE = string.punctuation
```

```
def remove_punctuation(text):
    """custom function to remove the punctuation"""
    return text.translate(str.maketrans("", "", PUNCT_TO_REMOVE))


full_df["keyPunct"] = full_df["keyLower"].apply(lambda text:
remove_punctuation(text))
full_df["studentPunct"] = full_df["studentLower"].apply(
    lambda text: remove_punctuation(text)
)

full_df[["studentLower", "studentPunct"]]

# %% [markdown]
# Stopword Removal

# %%
# Stopword Removal

STOPWORDS = set(stopwords.words("english"))


def remove_stopwords(text):
    """custom function to remove the stopwords"""
    return " ".join([word for word in str(text).split() if word not
in STOPWORDS])


full_df["keyStopword"] = full_df["keyPunct"].apply(lambda text:
remove_stopwords(text))
full_df["studentStopword"] = full_df["studentPunct"].apply(
    lambda text: remove_stopwords(text)
)

full_df[["studentPunct", "studentStopword"]]

# %% [markdown]
# Stemming

# %%
# Stemming

stemmer = SnowballStemmer("english")


def stem_words(text):
    return " ".join([stemmer.stem(word) for word in text.split()])


full_df["studentStem"] = full_df["studentStopword"].apply(lambda
text: stem_words(text))
full_df["keyStem"] = full_df["keyStopword"].apply(lambda text:
stem_words(text))

full_df[["studentStopword", "studentStem"]]

# %% [markdown]
# POS Tagging
```

```
# %%
studentSplit = full_df["studentStopword"]
keySplit = full_df["keyStopword"]

studentSplitWords = [nltk.word_tokenize(str(sentence)) for sentence
in studentSplit]
keySplitWords = [nltk.word_tokenize(str(sentence)) for sentence in
keySplit]
full_df["studentSplit"] = studentSplitWords
full_df["keySplit"] = keySplitWords

# %%
# POS Tagging

full_df["keyPostag"] = full_df["keySplit"].apply(nltk.pos_tag)
full_df["studentPostag"] =
full_df["studentSplit"].apply(nltk.pos_tag)

full_df[["keyPostag", "studentPostag"]]

# %% [markdown]
# Lemmatizing

# %%
# Lemmatizing

lemmatizer = WordNetLemmatizer()
wordnet_map = {"N": wordnet.NOUN, "V": wordnet.VERB, "J":
wordnet.ADJ, "R": wordnet.ADV}


def lemmatize_words(text):
    pos_tagged_text = nltk.pos_tag(text.split())
    return " ".join(
        [
            lemmatizer.lemmatize(word, wordnet_map.get(pos[0],
wordnet.NOUN))
            for word, pos in pos_tagged_text
        ]
    )


full_df["studentLemmatized"] = full_df["studentStopword"].apply(
    lambda text: lemmatize_words(text)
)
full_df["keyLemmatized"] = full_df["keyStopword"].apply(
    lambda text: lemmatize_words(text)
)
full_df[["studentStopword", "studentLemmatized"]]

# %% [markdown]
# **TF-IDF**

# %% [markdown]
# TF

# %%
# TF for Lemmatized
```

```
full_df["keyLemmatized"] = full_df["keyLemmatized"].astype(str)
full_df["studentLemmatized"] =
full_df["studentLemmatized"].astype(str)

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)

    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)

    columns = countvectorizer.get_feature_names_out()
    tf = pd.DataFrame(
        terms.toarray(), index=["keyLemmatized",
"studentLemmatized"], columns=columns
    )

    # print(tf)

# %%
# TF for Stem

full_df["keyStem"] = full_df["keyStem"].astype(str)
full_df["studentStem"] = full_df["studentStem"].astype(str)

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer2 = CountVectorizer()
    terms2 = countvectorizer2.fit_transform(docs2)

    columns2 = countvectorizer2.get_feature_names_out()
    tf2 = pd.DataFrame(
        terms2.toarray(), index=["keyStem", "studentStem"],
columns=columns2
    )

    # print(tf2)

# %% [markdown]
# IDF

# %%
# IDF for Lemmatized

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)

    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)

    tfidf_transformer = TfidfTransformer(smooth_idf=True,
use_idf=True)
    tfidf_transformer.fit(terms)
```

```
    df_idf = pd.DataFrame(
        tfidf_transformer.idf_,
        index=countvectorizer.get_feature_names_out(),
        columns=["idf_weights"],
    )

    # print(df_idf)


# %%
# IDF for Stem

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer = CountVectorizer()
    terms2 = countvectorizer.fit_transform(docs2)

    tfidf_transformer = TfidfTransformer(smooth_idf=True,
use_idf=True)
    tfidf_transformer.fit(terms2)

    df_idf2 = pd.DataFrame(
        tfidf_transformer.idf_,
        index=countvectorizer.get_feature_names_out(),
        columns=["idf_weights"],
    )

    # print(df_idf2)

# %% [markdown]
# TF-IDF

# %%
# TF-IDF for Lemmatized

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)
    countvectorizer = CountVectorizer()
    terms = countvectorizer.fit_transform(docs)
    X = tfidf_transformer.fit_transform(terms)
    tf_idf = pd.DataFrame(X.toarray(),
columns=countvectorizer.get_feature_names_out())
    # print(tf_idf)

# %%
# TF-IDF for Stem

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    countvectorizer = CountVectorizer()
    terms2 = countvectorizer.fit_transform(docs2)
```

```
    X2 = tfidf_transformer.fit_transform(terms2)
    tf_idf2 = pd.DataFrame(
        X2.toarray(),
columns=countvectorizer.get_feature_names_out()
    )
    # print(tf_idf2)

# %% [markdown]
# **Cosine Similarity**

# %%
# Cosine Similarity for Lemmatized

tfidf_vectorizer = TfidfVectorizer()
similarity = []

full_df["keyLemmatized"] = full_df["keyLemmatized"].astype(str)
full_df["studentLemmatized"] =
full_df["studentLemmatized"].astype(str)

for i in range(len(full_df)):
    doc1 = full_df["keyLemmatized"][i]
    doc2 = full_df["studentLemmatized"][i]
    docs = (doc1, doc2)
    tfidf_matrix = tfidf_vectorizer.fit_transform(docs)
    tfidf_tokens = tfidf_vectorizer.get_feature_names_out()
    df_tfidfvect = pd.DataFrame(
        data=tfidf_matrix.toarray(),
        index=["keyLemmatized", "studentLemmatized"],
        columns=tfidf_tokens,
    )
    # print(df_tfidfvect)

    cosine_sim = cosine_similarity(tfidf_matrix[0], tfidf_matrix[1])
    # print("Cosine Similarity =", cosine_sim[0][0])
    # print("\n")
    similarity.append(cosine_sim[0][0])

full_df["cosineSimilarityLemma"] = similarity

# cosinelemma = full_df['cosineSimilarityLemma']
# cosinelemma.to_csv('cosinelemma.csv')

# %%
# Cosine Similarity for Stem

tfidf_vectorizer2 = TfidfVectorizer()
similarity2 = []

full_df["keyStem"] = full_df["keyStem"].astype(str)
full_df["studentStem"] = full_df["studentStem"].astype(str)

for i in range(len(full_df)):
    doc12 = full_df["keyStem"][i]
    doc22 = full_df["studentStem"][i]
    docs2 = (doc12, doc22)
    tfidf_matrix2 = tfidf_vectorizer2.fit_transform(docs2)
    tfidf_tokens2 = tfidf_vectorizer2.get_feature_names_out()
    df_tfidfvect2 = pd.DataFrame(
```

```
        data=tfidf_matrix2.toarray(),
        index=["keyStem", "studentStem"],
        columns=tfidf_tokens2,
    )
    # print(df_tfidfvect2)

    cosine_sim2 = cosine_similarity(tfidf_matrix2[0],
tfidf_matrix2[1])
    # print("Cosine Similarity =", cosine_sim2[0][0])
    # print("\n")
    similarity2.append(cosine_sim2[0][0])

full_df["cosineSimilarityStem"] = similarity2

# %% [markdown]
# **Scoring**

# %%
# Scoring for Lemmatized
scoring = full_df["cosineSimilarityLemma"]

scoring = scoring.mul(5)

# full_df = full_df.drop('scoreModel', axis=1)
full_df["scoreModelLemma"] = scoring

full_df[["scoreModelLemma"]]

# %%
# Scoring for Stemming
scoring2 = full_df["cosineSimilarityStem"]

scoring2 = scoring2.mul(5)

scoring3 = (scoring2 / 5) * 100

full_df["scoreModelStem"] = scoring3

full_df[["scoreModelStem"]]

# %% [markdown]
# Uji Analisis (Abaikan)

# %%
full_df.to_json()

# %%
# newfile = open("HasilPenilaianENG.json", "w")
# newfile.write(full_df.to_json())
# newfile.close()

print(full_df.to_json())
```

15. models/indonesia/ASAG.py

```
import json
import math
```

```
from numpy.linalg import norm
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import nltk
import re
import sys
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import string
from string import digits
from sklearn.pipeline import Pipeline
from spell import spell_check

# from Lemma import lemmatization

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


def case_folding(text):
    pattern = r"[" + string.punctuation + "]"
    punct = re.sub(pattern, " ", str(text))
    case_fold = punct.lower()
    return case_fold


def spellcheck(text):
    text = spell_check(text)
    return text


def tokenization(text):
    tokens = re.split(" ", text)
    return tokens


def remove_digits(text):
    text = [item for item in text if item.isalpha()]
    return text


sw = nltk.corpus.stopwords.words("indonesian")


def remove_SW(text):
    text = [item for item in text if not item in sw]
    return text


factory = StemmerFactory()
stemmer = factory.create_stemmer()


def stemming(text):
    text = [stemmer.stem(item) for item in text]
    return text
```

```
# def lemma(text):
#     text = lemmatization(text)
#     return text


def term(q, ans):
    for i in q:
        if i == "":
            q.remove("")
    for i in ans:
        if i == "":
            ans.remove("")

    BoWQ = set(q)
    BoWA = set(ans)

    uniqueWords = BoWQ.union(BoWA)
    # print(uniqueWords)

    numOfWordsQ = dict.fromkeys(uniqueWords, 0)
    for word in q:
        numOfWordsQ[word] += 1

    numOfWordsA = dict.fromkeys(uniqueWords, 0)
    for word in ans:
        numOfWordsA[word] += 1

    # print('Unique words', numOfWordsA)

    term = pd.DataFrame([numOfWordsQ, numOfWordsA])
    term = term.transpose()
    term.columns = ["TF_Q", "TF_Ans"]

    # display(term)

    dfQ = dict.fromkeys(uniqueWords, 0)
    for word in BoWQ:
        dfQ[word] += 1

    dfA = dict.fromkeys(uniqueWords, 0)
    for word in BoWA:
        dfA[word] += 1

    term["DF_Q"] = dfQ.values()
    term["DF_A"] = dfA.values()

    DF = []
    for i in range(len(uniqueWords)):
        DF.append(term["DF_Q"][i] + term["DF_A"][i])
    term["DF"] = DF
    # display(term)

    idfDict = []

    for i in range(len(term["DF"])):
        idfDict.append(math.log10((2 + 1) / (term["DF"][i] + 1)) +
1)
        # print(idfDict)
```

```
    term["IDF"] = idfDict

    # display(term)

    tfidfQ = []
    tfidfA = []
    for i in range(len(uniqueWords)):
        tfidfQ.append(term["TF_Q"][i] * term["IDF"][i])
        tfidfA.append(term["TF_Ans"][i] * term["IDF"][i])

    term["TF-IDF_Q"] = np.array(tfidfQ)
    term["TF-IDF_A"] = np.array(tfidfA)

    cosine = np.dot(tfidfQ, tfidfA) / (np.linalg.norm(tfidfQ) *
np.linalg.norm(tfidfA))

    if math.isnan(cosine):
        cosine = 0
    skor = round((cosine * 100), 2)
    return skor


case_fold_q = case_folding(kj)
case_fold_ans = case_folding(jawaban)
spell_q = spell_check(case_fold_q)
spell_ans = spell_check(case_fold_ans)
token_q = tokenization(spell_q)
token_ans = tokenization(spell_ans)
token_q = remove_digits(token_q)
token_ans = remove_digits(token_ans)
filter_q = remove_SW(token_q)
filter_ans = remove_SW(token_ans)

stem_q = stemming(filter_q)
stem_ans = stemming(filter_ans)

# lemma_q = lemma(filter_q)
# lemma_ans = lemma(filter_ans)

# print("Skor Tanpa pre-processing: ",
term(tokenization(kj),tokenization(jawaban)))
# print("Skor Menggunakan Stemming: ", term(stem_q, stem_ans))
# print("Skor Menggunakan Lemmatization: ", term(lemma_q,
lemma_ans))

sctScore = str(term(stem_q, stem_ans))
newfile = open(f"{sys.argv[3]}.txt", "a")
# newfile.write(
#     '{"kj":"'
#     + case_fold_q
#     + '",\n"ans":"'
#     + case_fold_ans
#     + '",\n"score":'
#     + sctScore
#     + "}"
# )
newfile.write(f"{sctScore}\n")
newfile.close()
print("done")
```

16. models/indonesia/ASAGbulk.py

```
import json
import math
from numpy.linalg import norm
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import nltk
import re
import sys
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import string
from string import digits
from sklearn.pipeline import Pipeline
from spell import spell_check

# from Lemma import lemmatization

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


def case_folding(text):
    pattern = r"[" + string.punctuation + "]"
    punct = re.sub(pattern, " ", str(text))
    case_fold = punct.lower()
    return case_fold


def spellcheck(text):
    text = spell_check(text)
    return text


def tokenization(text):
    tokens = re.split(" ", text)
    return tokens


def remove_digits(text):
    text = [item for item in text if item.isalpha()]
    return text


sw = nltk.corpus.stopwords.words("indonesian")


def remove_SW(text):
    text = [item for item in text if not item in sw]
    return text


factory = StemmerFactory()
stemmer = factory.create_stemmer()
```

```
def stemming(text):
    text = [stemmer.stem(item) for item in text]
    return text


# def lemma(text):
#     text = lemmatization(text)
#     return text


def term(q, ans):
    for i in q:
        if i == "":
            q.remove("")
    for i in ans:
        if i == "":
            ans.remove("")

    BoWQ = set(q)
    BoWA = set(ans)

    uniqueWords = BoWQ.union(BoWA)
    # print(uniqueWords)

    numOfWordsQ = dict.fromkeys(uniqueWords, 0)
    for word in q:
        numOfWordsQ[word] += 1

    numOfWordsA = dict.fromkeys(uniqueWords, 0)
    for word in ans:
        numOfWordsA[word] += 1

    # print('Unique words', numOfWordsA)

    term = pd.DataFrame([numOfWordsQ, numOfWordsA])
    term = term.transpose()
    term.columns = ["TF_Q", "TF_Ans"]

    # display(term)

    dfQ = dict.fromkeys(uniqueWords, 0)
    for word in BoWQ:
        dfQ[word] += 1

    dfA = dict.fromkeys(uniqueWords, 0)
    for word in BoWA:
        dfA[word] += 1

    term["DF_Q"] = dfQ.values()
    term["DF_A"] = dfA.values()

    DF = []
    for i in range(len(uniqueWords)):
        DF.append(term["DF_Q"][i] + term["DF_A"][i])
    term["DF"] = DF
    # display(term)

    idfDict = []
```

```
    for i in range(len(term["DF"])):
        idfDict.append(math.log10((2 + 1) / (term["DF"][i] + 1)) +
1)
        # print(idfDict)
    term["IDF"] = idfDict

    # display(term)

    tfidfQ = []
    tfidfA = []
    for i in range(len(uniqueWords)):
        tfidfQ.append(term["TF_Q"][i] * term["IDF"][i])
        tfidfA.append(term["TF_Ans"][i] * term["IDF"][i])

    term["TF-IDF_Q"] = np.array(tfidfQ)
    term["TF-IDF_A"] = np.array(tfidfA)

    cosine = np.dot(tfidfQ, tfidfA) / (np.linalg.norm(tfidfQ) *
np.linalg.norm(tfidfA))

    if math.isnan(cosine):
        cosine = 0
    skor = round((cosine * 100), 2)
    return skor


case_fold_q = case_folding(kj)
case_fold_ans = case_folding(jawaban)
spell_q = spell_check(case_fold_q)
spell_ans = spell_check(case_fold_ans)
token_q = tokenization(spell_q)
token_ans = tokenization(spell_ans)
token_q = remove_digits(token_q)
token_ans = remove_digits(token_ans)
filter_q = remove_SW(token_q)
filter_ans = remove_SW(token_ans)

stem_q = stemming(filter_q)
stem_ans = stemming(filter_ans)

# lemma_q = lemma(filter_q)
# lemma_ans = lemma(filter_ans)

# print("Skor Tanpa pre-processing: ",
term(tokenization(kj),tokenization(jawaban)))
# print("Skor Menggunakan Stemming: ", term(stem_q, stem_ans))
# print("Skor Menggunakan Lemmatization: ", term(lemma_q,
lemma_ans))

sctScore = str(term(stem_q, stem_ans))
# newfile = open(f"{sys.argv[3]}.txt", "a")
# # newfile.write(
# #     '{"kj":"'
# #     + case_fold_q
# #     + '",\n"ans":"'
# #     + case_fold_ans
# #     + '",\n"score":'
# #     + sctScore
# #     + "}"
```

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

17. models/indonesia/slang.dic

```
{
    "1pun": "satupun",
    "7an": "tujuan",
    "Dr.": "doktor",
    "dr.": "dokter",
    "drg.": "dokter gigi",
    "Jkt": "Jakarta",
    "Jkw": "jokowi",
    "Napza": "narkoba psikotropika dan zat adiktif",
    "Nasihat": "nasehat",
    "ababil": "abg labil",
    "abis": "habis",
    "acc": "accord",
    "accord": "sesuai",
    "ad": "ada",
    "adl": "adalah",
    "adlah": "adalah",
    "adlh": "adalah",
    "administ": "administrasi",
    "adoh": "aduh",
    "afaik": "as far as i know",
    "agma": "agama",
    "aha": "tertawa",
    "ahaha": "haha",
    "ahiok": "ahok",
    "mehong": "mahal",
    "ahoax": "ahok",
    "ahokncc": "ahok",
    "aing": "saya",
    "aj": "saja",
    "aja": "saja",
    "ajak2": "ajak-ajak",
    "ajep-ajep": "dunia gemerlap",
    "ajj": "saja",
    "ak": "aku",
    "aka": "dikenal juga sebagai",
    "akherat": "akhirat",
    "akhirx": "akhirnya",
    "akika": "aku",
    "akko": "aku",
    "akkoh": "aku",
    "akku": "aku",
    "akn": "akan",
    "aktifis2": "aktivis-aktivis",
    "aktivi": "aktivis",
    "aktivis2": "aktivis-aktivis",
    "akun2": "akun-akun",
    "akuwh": "aku",
    "akyu": "aku",
    "alay": "norak",
```

```
    "aleg": "caleg",
    "alih2": "alih-alih",
    "aljasa": "asal jadi saja",
    "almaidahyg": "almaidah",
    "alow": "halo",
    "ama": "sama",
    "ambilin": "ambilkan",
    "ambl": "ambil",
    "amerik": "amarika",
    "anak2": "anak-anak",
    "analisise": "analisis",
    "ancur": "hancur",
    "angis": "nangis",
    "anjay": "anjing",
    "anjir": "anjing",
    "anjrit": "anjing",
    "ank": "anak",
    "anter": "antar",
    "ap": "apa",
    "ap2": "apa-apa",
    "apaan": "apa",
    "apasih": "apa sih",
    "ape": "apa",
    "apes": "sial",
    "aplot": "unggah",
    "aps": "apa",
    "apva": "apa",
    "aq": "saya",
    "aqu": "aku",
    "aquwh": "aku",
    "as": "amerika serikat",
    "asap": "sesegera mungkin",
    "asbun": "asal bunyi",
    "aseek": "asyik",
    "aseekk": "asyik",
    "asek": "asyik",
    "asekk": "asyik",
    "aseknya": "asyiknya",
    "asem": "asam",
    "asoy": "asyik",
    "aspal": "asli tetapi palsu",
    "astrojim": "astagfirullahaladzim",
    "astul": "asal tulis",
    "ath": "kalau begitu",
    "ato": "atau",
    "ats": "atas",
    "atuh": "kalau begitu",
    "atvsi": "asosiasi televisi swasta indonesia",
    "au": "angkatan udara",
    "au ah": "tidak mau tahu",
    "ava": "avatar",
    "awak": "saya",
    "aws": "awas",
    "ay": "sayang",
    "ayang": "sayang",
    "ayank": "sayang",
    "ayat2": "ayat-ayat",
    "ayok": "ayo",
    "b2": "berdua",
```

```
    "b4": "sebelum",
    "bacot": "banyak bicara",
    "bajir": "banjir",
    "bakalan": "akan",
    "bales": "balas",
    "bandes": "bantuan desa",
    "bangdes": "pembangunan desa",
    "bangedh": "banget",
    "bangkotan": "tua",
    "banpol": "bantuan polisi",
    "banpres": "bantuan presiden",
    "banpur": "bantuan tempur",
    "bansarkas": "bantuan sarana kesehatan",
    "baper": "bawa perasaan",
    "basbang": "basi",
    "bazis": "badan amal, zakat, infak, dan sedekah",
    "bbrp": "beberapa",
    "bcanda": "bercanda",
    "bcoz": "karena",
    "bdaya": "budaya",
    "bdg": "bandung",
    "bdk": "budak",
    "beb": "sayang",
    "bebal": "bodoh",
    "bedakin": "bedak",
    "begajulan": "nakal",
    "bejibun": "banyak",
    "beliin": "belikan",
    "belom": "belum",
    "bem": "badan eksekutif mahasiswa",
    "benarjujur": "jujur",
    "bencong": "banci",
    "bener": "benar",
    "bener2": "bener bener",
    "bentar": "sebentar",
    "ber2": "berdua",
    "ber3": "bertiga",
    "berdikari": "berdiri di atas kaki sendiri",
    "beresin": "membereskan",
    "berhenti2": "berhenti berhenti",
    "bet": "banget",
    "bete": "bosan",
    "beti": "beda tipis",
    "beud": "banget",
    "beut": "banget",
    "bg": "bagi",
    "bgd": "banget",
    "bgm": "bagaimana",
    "bgmn": "bagaimana",
    "bgs": "bagus",
    "bgst": "bangsat",
    "bgt": "banget",
    "bgtu": "begitu",
    "bhai": "bye",
    "bhkan": "bahkan",
    "bhubu": "tidur",
    "bhw": "bahwa",
    "bhy": "bahaya",
    "bijimane": "bagaimana",
```

```
    "bilanh": "bilang",
    "bimbuluh": "bimbingan dan penyuluhan",
    "bintal": "bimbingan mental",
    "bisi": "kalau-kalau",
    "bkannya": "bukannya",
    "bkl": "akan",
    "bkmt": "badan kontak majelis taklim",
    "bkn": "bukan",
    "bknnya": "bukannya",
    "bl": "beli",
    "blas": "balas",
    "blegug": "bodoh",
    "blg": "bilang",
    "blh": "boleh",
    "blm": "belum",
    "bln": "bulan",
    "bls": "balas",
    "blum": "belum",
    "bnchi": "benci",
    "bnci": "benci",
    "bnda": "benda",
    "bngung": "bingung",
    "bnran": "yang benar",
    "bntr": "sebentar",
    "bnyk": "banyak",
    "bodor": "lucu",
    "bohay": "badan aduhai",
    "bokap": "ayah",
    "bokep": "video porno",
    "boker": "buang air besar",
    "bokin": "pacar",
    "bokis": "bohong",
    "bole": "boleh",
    "boljug": "boleh juga",
    "bolot": "bodoh",
    "bonek": "bocah nekat",
    "bonyok": "ayah ibu",
    "bopi": "badan olahraga profesional indonesia",
    "boyeh": "boleh",
    "bpk": "bapak",
    "br": "baru",
    "brb": "segera kembali",
    "brg": "bareng",
    "brhadapan": "berhadapan",
    "brngkt": "berangkat",
    "bro": "saudara laki-laki",
    "brp": "berapa",
    "brsdia": "bersedia",
    "brstu": "bersatu",
    "bru": "baru",
    "brur": "saudara laki-laki",
    "bs": "bisa",
    "bsa": "bisa",
    "bsen": "bosan",
    "bsk": "besok",
    "bsr": "besar",
    "bt": "buat",
    "btol": "botol",
    "btw": "ngomong-ngomong",
```

```
    "bu_bu": "tidur",
    "buangbuang": "buang",
    "buaya": "tidak setia",
    "bubarin": "bubarkan",
    "bubbu": "tidur",
    "buber": "buka bersama",
    "bubu": "tidur",
    "bujubune": "luar biasa",
    "bumil": "ibu hamil",
    "bunuh2an": "bunuh-bunuhan",
    "buser": "buru sergap",
    "buzzerbuzzer": "buzzer",
    "bw": "bawa",
    "bwhn": "bawahan",
    "bwt": "buat",
    "byar": "bayar",
    "byk": "banyak",
    "byr": "bayar",
    "byrin": "bayarkan",
    "c8": "chat",
    "cabal": "sabar",
    "cabut": "pergi",
    "cadas": "keren",
    "caem": "cakep",
    "cakap": "berbicara",
    "calo": "makelar",
    "cama-cama": "sama-sama",
    "can": "belum",
    "cangcut": "celana dalam",
    "capcus": "pergi",
    "cape": "capek",
    "capek": "lelah",
    "caper": "cari perhatian",
    "capresss": "calon presiden",
    "caur": "jelek",
    "cayalah": "percaya",
    "cb": "coba",
    "cc": "carbon copy",
    "ce": "cewek",
    "cekak": "tidak ada uang",
    "cekal": "cegah tangkal",
    "cekidot": "coba lihat",
    "cemen": "penakut",
    "cemplungin": "cemplungkan",
    "cengengesan": "tertawa",
    "cepat2": "cepat cepat",
    "ceper": "pendek",
    "cepet": "cepat",
    "ceu": "kakak perempuan",
    "cew": "cewek",
    "cewe": "cewek",
    "chuyunk": "sayang",
    "cibuk": "sibuk",
    "cimeng": "ganja",
    "cin": "cinta",
    "cipika cipiki": "cium pipi kanan cium pipi kiri",
    "ciye": "cie",
    "ciyh": "sih",
    "cjr": "coboy junior",
```

```
    "ckck": "ck",
    "ckepp": "cakep",
    "ckp": "cukup",
    "clbk": "cinta lama bersemi kembali",
    "cm": "cuma",
    "cmiiw": "correct me if i'm wrong",
    "cmpr": "campur",
    "cmpur": "campur",
    "cnenk": "senang",
    "cong": "banci",
    "congor": "mulut",
    "conlok": "cinta lokasi",
    "cos": "cause",
    "cow": "cowok",
    "cowwyy": "maaf",
    "coz": "karena",
    "cp": "siapa",
    "cpa": "siapa",
    "cpe": "capek",
    "cppa": "siapa",
    "cppe": "capek",
    "cr": "cara",
    "crny": "caranya",
    "cth": "contoh",
    "cucok": "cocok",
    "cuek": "tidak acuh",
    "cuex": "cuek",
    "cuman": "Cuma",
    "cumi": "Cuma miscall",
    "cungkring": "kurus tinggi",
    "cups": "culun",
    "cupu": "culun",
    "curanmor": "pencurian kendaraan bermotor",
    "curas": "pencurian dengan kekerasan",
    "curcol": "curahan hati colongan",
    "curhat": "curahan hati",
    "cust": "customer",
    "cwek": "cewek",
    "cwo": "cowo",
    "cwok": "cowok",
    "cyin": "cinta",
    "cz": "karena",
    "d": "di",
    "dagh": "sudah",
    "dah": "sudah",
    "damang": "sehat",
    "dapet": "dapat",
    "dateng": "datang",
    "de": "adik",
    "dech": "deh",
    "dek": "adik",
    "deket": "dekat",
    "demen": "suka",
    "denger": "dengar",
    "dengerin": "dengarkan",
    "deyh": "deh",
    "dg": "dengan",
    "dgn": "dengan",
    "dgr": "dengar",
```

```
    "dh": "sudah",
    "diancurin": "dihancurkan",
    "die": "dia",
    "diem2": "diam diam",
    "digosipin": "digosipkan",
    "dikti": "direktoral jendral pendidikan tinggi",
    "dimaafin": "dimaafkan",
    "dimana2": "dimana-mana",
    "dimintak": "diminta",
    "dimna": "dimana",
    "direhap": "direhab",
    "disono": "di sana",
    "diuber": "dikejar",
    "dket": "dekat",
    "dki": "daerah khusus ibukota",
    "dkit": "sedikit",
    "dkk": "dan kawan-kawan",
    "dl": "dulu",
    "dll": "dan lain-lain",
    "dlm": "dalam",
    "dlu": "dulu",
    "dmaksud": "dimaksud",
    "dmn": "dimana",
    "dn": "dan",
    "dngn": "dengan",
    "doang": "saja",
    "dodol": "bodoh",
    "doi": "dia",
    "doku": "uang",
    "donat": "donut",
    "doughnat": "donut",
    "don juan": "playboy",
    "dongok": "bodoh",
    "dongs": "dong",
    "donk": "dong",
    "donlot": "unduh",
    "dpr": "dewan perwakilan rakyat",
    "dprd": "dewan perwakilan rakyat daerah",
    "dpt": "dapat",
    "dr": "dari",
    "dri": "dari",
    "drmana": "darimana",
    "drmn": "darimana",
    "drpd": "daripada",
    "drtd": "dari tadi",
    "dsb": "dan sebagainya",
    "dst": "dan seterusnya",
    "dstu": "disitu",
    "dtg": "datang",
    "dugem": "dunia gemerlap",
    "duh": "aduh",
    "duit": "uang",
    "dulu": "dahulu",
    "dungu": "bodoh",
    "duren": "durian",
    "dy": "dia",
    "dzolim": "zalim",
    "ed": "edisi",
    "edan": "gila",
```

```
    "ego": "emosi",
    "egp": "emang gue pikirin",
    "eike": "aku",
    "eke": "aku",
    "elo": "kamu",
    "elu": "kamu",
    "emang": "memang",
    "emangnya": "memangnya",
    "emank": "memang",
    "emg": "memang",
    "emng": "memang",
    "emte": "makan teman",
    "endak": "tidak",
    "engga": "tidak",
    "enggak": "tidak",
    "entar": "nanti",
    "envy": "iri",
    "eps": "episode",
    "etc": "et cetera",
    "ex": "mantan",
    "ext": "extension",
    "fahami": "pahami",
    "fak": "fakultas",
    "fakta2": "fakta-fakta",
    "faktakta": "fakta",
    "fax": "facsimile",
    "fb": "facebook",
    "fdsi": "forum diskusi suporter indonesia",
    "fifo": "first in first out",
    "finek": "finansial ekonomi",
    "fisip": "fakultas ilmu sosial dan ilmu politik",
    "fitnag": "fitnah",
    "folbek": "follow back",
    "fpi": "front pembela islam",
    "fwd": "diterukan",
    "fyi": "sebagai informasi",
    "g": "tidak",
    "ga": "tidak",
    "gaada": "tidak ada uang",
    "gabisa": "tidak bisa",
    "gag": "tidak",
    "gahar": "sangar",
    "gaje": "tidak jelas",
    "gak": "tidak",
    "gak papa": "tidak apa-apa",
    "gamau": "tidak mau",
    "gan": "juragan",
    "gantengbs": "ganteng",
    "gantengsaya": "ganteng",
    "gapapa": "tidak apa-apa",
    "gaptek": "gagap teknologi",
    "gatau": "tidak tahu",
    "gatek": "gagap teknologi",
    "gatot": "gagal total",
    "gawe": "kerja",
    "gax": "tidak",
    "gbs": "tidak bisa",
    "gd": "gusdur",
    "gda": "tidak ada uang",
```

```
    "gebetan": "orang yang disuka",
    "geer": "gede rasa",
    "gegara": "gara-gara",
    "geje": "tidak jelas",
    "gelo": "gila",
    "gene": "begini",
    "gepeng": "gelandangan dan pengemis",
    "ghiey": "lagi",
    "ghiy": "lagi",
    "gi": "lagi",
    "gib": "gerakan indonesia beradab",
    "gile": "gila",
    "giman": "gigi mancung",
    "gimana": "bagaimana",
    "gini": "begini",
    "gingerbread": "ginger bread",
    "gino": "gigi nongol",
    "gitchu": "gitu",
    "githu": "gitu",
    "gitu": "begitu",
    "gj": "tidak jelas",
    "gk": "tidak",
    "gmana": "bagaimana",
    "gmn": "bagamana",
    "gn": "begini",
    "goblog": "goblok",
    "goblok": "bodoh",
    "gokil": "gila",
    "golongan2": "golongan-golongan",
    "golput": "golongan putih",
    "gombal": "suka merayu",
    "good": "bagus",
    "governor": "gubernur",
    "gowes": "mengayuh sepeda",
    "gpl": "tidak pakai lama",
    "gpny": "tidak punya",
    "gpp": "tidak apa-apa",
    "gr": "gede rasa",
    "gretong": "gratis",
    "gretongan": "gratisan",
    "gt": "gitu",
    "gtau": "tidak tahu",
    "gtw": "tidak tahu",
    "gua": "saya",
    "gub": "gubernur",
    "gue": "saya",
    "guna": "menggunakan",
    "guoblok": "goblok",
    "guys": "teman-teman",
    "gw": "saya",
    "gws": "cepat sembuh",
    "ha": "tertawa",
    "habis2an": "habis-habisan",
    "haghaghag": "tertawa",
    "haha": "tertawa",
    "hak2": "hak hak",
    "hakhak": "tertawa",
    "hallow": "halo",
    "ham": "hak asasi manusia",
```

```
    "handak": "bahan peledak",
    "hankam": "pertahanan dan keamanan",
    "hansip": "pertahanan sipil",
    "happy": "bahagia",
    "hardcore": "keras",
    "hehe": "he",
    "hellow": "halo",
    "helo": "halo",
    "helow": "halo",
    "heter": "pembenci",
    "heters": "pembenci",
    "hey": "hai",
    "hi": "hai",
    "hlm": "halaman",
    "hlng": "hilang",
    "hny": "hanya",
    "hnya": "hanya",
    "hoax": "isu bohong",
    "hot2nya": "hot-hotnya",
    "houm": "rumah",
    "hr": "hari",
    "hrs": "harus",
    "hrus": "harus",
    "hubad": "hubungan angkatan darat",
    "hubdar": "perhubungan darat",
    "hubla": "perhubungan laut",
    "huff": "mengeluh",
    "huft": "mengeluh",
    "hum": "rumah",
    "humas": "hubungan masyarakat",
    "humoris": "lucu",
    "humz": "rumah",
    "hurts": "pedih",
    "icecream": "ice cream",
    "icecreamsandwich": "ice cream sandwich",
    "ide2": "ide-ide",
    "idk": "saya tidak tahu",
    "idup": "hidup",
    "ikon": "lambang",
    "ilang": "hilang",
    "ilfeel": "tidak suka",
    "ilfil": "tidak suka",
    "imba": "jago sekali",
    "imho": "in my humble opinion",
    "imoet": "imut",
    "imoetz": "imut",
    "indinesia": "indonesia",
    "indo": "indonesia",
    "indon": "Indonesia",
    "info": "informasi",
    "insom": "insomnia",
    "ipb": "institut pertanian bogor",
    "irit": "hemat",
    "isengin": "bercanda",
    "ista": "nista",
    "istaa": "nista",
    "isu2": "isu-isu",
    "item": "hitam",
    "itj": "Indonesia tanpa jil",
```

```
    "ito": "itu",
    "itu2": "itu-itu",
    "itung": "hitung",
    "itungan": "hitungan",
    "iy": "iya",
    "iyala": "iya lah",
    "iye": "iya",
    "iyo": "iya",
    "ja": "saja",
    "jabar": "jawa barat",
    "jablay": "jarang dibelai",
    "jadiin": "jadikan",
    "jadul": "jaman dulu",
    "jaim": "jaga image",
    "jan": "jangan",
    "jancuk": "anjing",
    "janjinyaa": "janji",
    "jargon2": "jargon jargon",
    "jatim": "jawa timur",
    "jatohin": "menjatuhkan",
    "jawapan": "jawaban",
    "jayus": "tidak lucu",
    "jd": "jadi",
    "jdi": "jadi",
    "jdikan": "jadikan",
    "je": "saja",
    "jelas2": "jelas-jelas",
    "jem": "jam",
    "jentelmen": "berani",
    "jg": "juga",
    "jga": "juga",
    "jgk": "juga",
    "jgn": "jangan",
    "jgnkan": "jangankan",
    "ji2k": "jijik",
    "jijay": "jijik",
    "jil": "jaringan islam liberal",
    "jir": "anjing",
    "jjk": "jijik",
    "jk": "jika",
    "jkt": "jakarta",
    "jln": "jalan",
    "jng": "jangan",
    "jnj": "janji",
    "joget2": "joget-joget",
    "jomblo": "tidak punya pacar",
    "jth": "jatuh",
    "jual2": "jual-jual",
    "jubir": "juru bicara",
    "jurdil": "jujur adil",
    "jutek": "galak",
    "jwb": "jawab",
    "k": "ke",
    "ka": "kakak",
    "kab": "kabupaten",
    "kabag": "kepala bagian",
    "kabor": "kabur",
    "kacian": "kasihan",
    "kacrut": "kacau",
```

```
    "kadang2": "kadang kadang",
    "kadit": "kepala direktorat",
    "kadiv": "kepala divisi",
    "kaga": "tidak",
    "kagak": "tidak",
    "kaka": "kakak",
    "kalo": "kalau",
    "kaltim": "kalimantan timur",
    "kampret": "sialan",
    "kamtib": "keamanan dan ketertiban",
    "kamtibmas": "keamanan dan ketertiban masyarakat",
    "kamuh": "kamu",
    "kamuwh": "kamu",
    "kamyu": "kamu",
    "kan": "akan",
    "kanwil": "kantor wilayah",
    "kapolri": "Kepala Polisi Republik Indonesia",
    "kapt": "kapten",
    "karna": "karena",
    "kasubbag": "kepala subbagian",
    "kasubbid": "kepala subbidang",
    "kata2": "kata-kata",
    "katrok": "kampungan",
    "kau": "kamu",
    "kawan2": "kawan-kawan",
    "kayanya": "kayaknya",
    "kbar": "kabar",
    "kbr": "kabar",
    "kcian": "kasihan",
    "kdng": "kadang",
    "kdu": "harus",
    "ke": "kah",
    "kebranian": "keberanian",
    "keburu": "terlanjur",
    "kec": "kecamatan",
    "kedubes": "kedutaan besar",
    "kejar2": "kejar-kejar",
    "kejurnas": "kejuaraan nasional",
    "kek": "seperti",
    "kekeuh": "keras kepala",
    "keknya": "kayanya",
    "kel": "kelurahan",
    "keliatan": "kelihatan",
    "kemaren": "kemarin",
    "kemenag": "kementrian agama",
    "kemenkominfo": "kementrian komunikasi dan informatika",
    "kemrin": "kemarin",
    "keneh": "masih",
    "kentot": "senggama",
    "kepengen": "mau",
    "kepikiran": "terpikirkan",
    "kepingin": "mau",
    "kepo": "mau tahu urusan orang",
    "kepsek": "kepala sekolah",
    "kere": "tidak punya uang",
    "kesbang": "kesatuan bangsa",
    "kesian": "kasihan",
    "kesihatan": "kesehatan",
    "kesra": "kesejahteraan rakyat",
```

```
    "ketauan": "ketahuan",
    "ketrima": "diterima",
    "ketum": "ketua umum",
    "keukeuh": "keras kepala",
    "kgiatan": "kegiatan",
    "khan": "kan",
    "kibul": "bohong",
    "kibus": "kaki busuk",
    "kimpoi": "kawin",
    "kk": "kakak",
    "kl": "kalau",
    "klian": "kalian",
    "klianz": "kalian",
    "klmpk": "kelompok",
    "klo": "kalau",
    "kloter": "kelompok terbang",
    "kls": "kelas",
    "kluarga": "keluarga",
    "klw": "kalau",
    "klwrga": "keluarga",
    "km": "kamu",
    "kmaksiatan": "kemaksiatan",
    "kmari": "kemari",
    "kmna": "kemana",
    "kmps": "kampus",
    "kmpus": "kampus",
    "kmrm": "kemeramen",
    "kmrn": "kemarin",
    "kn": "kan",
    "knal": "kenal",
    "kne": "terlibat",
    "knl": "kenal",
    "knp": "kenapa",
    "knpa": "kenapa",
    "koar": "teriak",
    "kodya": "kota madya",
    "kog": "kok",
    "komdis": "komisi disiplin",
    "kompi": "komputer",
    "komsov": "komunis sovyet",
    "komtiong": "komunis Tiongkok",
    "komun": "komunitas",
    "kongkow": "kumpul bareng teman-teman",
    "konjen": "konsulat jenderal",
    "konpers": "konferensi pers",
    "kopdar": "kopi darat",
    "koq": "kok",
    "korup": "korupsi",
    "kowe": "kamu",
    "kpai": "komisi perlindungan anak indonesia",
    "kpd": "kepada",
    "kpid": "komisi penyiaran indonesia daerah",
    "kpk": "komisi pemberantas koruptor",
    "kpn": "kapan",
    "kptsan": "keputusan",
    "krenz": "keren",
    "krik": "garing",
    "krim": "kirim",
    "krm": "kirim",
```

```
    "krn": "karena",
    "ksadaran": "kesadaran",
    "kt": "kita",
    "ktauan": "ketahuan",
    "ktipu": "tipu",
    "ktk": "ketika",
    "ktmu": "ketemu",
    "ktny": "katanya",
    "ktp": "kartu tanda penduduk",
    "ktr": "kantor",
    "kudu": "harus",
    "kuper": "kurang pergaulan",
    "kuq": "kok",
    "kurng": "kurang",
    "kw": "imitasi",
    "kwn": "kawan",
    "ky": "seperti",
    "kyk": "seperti",
    "kykny": "kayanya",
    "la": "lah",
    "laka": "kecelakaan",
    "laki2": "laki-laki",
    "lam": "salam",
    "lambreta": "lambat",
    "lamp": "lampiran",
    "langi": "langit",
    "lansia": "lanjut usia",
    "lanud": "landasan udara",
    "lapas": "lembaga pemasyarakatan",
    "larang2": "larang-larang",
    "latgab": "latihan gabungan",
    "lawyer2": "lawyer lawyer",
    "lbh": "lebih",
    "lbur": "libur",
    "lebay": "berlebihan",
    "leh": "boleh",
    "lekong": "laki-laki",
    "lelet": "lambat",
    "lemot": "lemah otak",
    "lesbia": "lesbian",
    "less": "kurang",
    "lg": "lagi",
    "lgi": "lagi",
    "lgkp": "lengkap",
    "lgsg": "langsung",
    "lha": "lah",
    "lht": "lihat",
    "liat": "lihat",
    "light": "terang",
    "linmas": "perlindungan masyarakat",
    "litbang": "penelitian dan pengembangan",
    "liwat": "lewat",
    "lknt": "laknat",
    "lmyan": "lumayan",
    "lmyn": "lumayan",
    "lngkp": "lengkap",
    "lo": "kamu",
    "loch": "loh",
    "loe": "kamu",
```

```
    "lol": "tertawa",
    "lola": "lambat berfikir",
    "lom": "belum",
    "louph": "cinta",
    "loupz": "cinta",
    "low": "kalau",
    "lowh": "kamu",
    "lp": "lupa",
    "lsm": "lembaga swadaya masyarakat",
    "lu": "kamu",
    "luber": "langsung, umum, bebas, dan rahasia",
    "luchu": "lucu",
    "luchuw": "lucu",
    "luff": "cinta",
    "lum": "belum",
    "luph": "cinta",
    "luthu": "lucu",
    "luv": "love",
    "lw": "kamu",
    "lwn": "lawan",
    "lwt": "lewat",
    "lyf": "live",
    "ma": "sama",
    "maacih": "terima kasih",
    "maaciw": "terima kasih",
    "mabal": "bolos",
    "mabes": "markas besar",
    "macem": "macam",
    "macem-macem": "macam-macam",
    "macet2an": "macet-macetan",
    "macih": "masih",
    "madesu": "masa depan suram",
    "maem": "makan",
    "maen": "main",
    "magabut": "makan gaji buta",
    "mahatma": "maju sehat bersama",
    "maho": "homo",
    "majority": "mayoritas",
    "mak": "ibu",
    "mak jang": "kaget",
    "makasih": "terima kasih",
    "maksain": "memaksa",
    "malah": "bahkan",
    "malem": "malam",
    "malu2in": "memalukan",
    "mam": "makan",
    "mamz": "makan",
    "mana2": "mana-mana",
    "maneh": "kamu",
    "manen": "panen",
    "manggil": "panggil",
    "manies": "manis",
    "maniez": "manis",
    "mantabss bagus": "",
    "mantep": "mantap",
    "mao": "mau",
    "maren2": "kemarin-kemarin",
    "markus": "makelar kasus",
    "masama": "sama - sama",
```

```
    "masing2": "masing-masing",
    "masukin": "masukkan",
    "masy": "masyarakat",
    "mati2": "mati-mati",
    "maybe": "mungkin",
    "mb": "mbak",
    "mba": "mbak",
    "mbabibuta": "ceroboh",
    "mbahas": "membahas",
    "mbaur": "baur",
    "mbelain": "belain",
    "mbual": "membual",
    "mcm": "macam",
    "mdh": "mudah",
    "mdh2an": "mudah-mudahan",
    "medsos": "media sosial",
    "melayakni": "melayani",
    "melu": "ikut",
    "mencarut2": "mencarut",
    "mending": "lebih baik",
    "mendpt": "mendapat",
    "menjdi": "menjadi",
    "menklarifikasi": "mengklarifikasi",
    "menolak": "tolak",
    "menpora": "menteri pemuda dan olahraga",
    "menristek": "mentri riset, teknologi dan pendidikan",
    "mensos": "menteri sosial",
    "mepet": "dekat sekali",
    "mesjid": "masjid",
    "met": "selamat",
    "metro": "metropolitan",
    "mga": "moga",
    "mgkin": "mungkin",
    "mgkn": "mungkin",
    "mgu": "minggu",
    "mhn": "mohon",
    "migas": "minyak dan gas bumi",
    "miker": "minuman keras",
    "mikol": "minuman beralkohol",
    "milis": "mailing list",
    "miras": "minuman keras",
    "miumi": "majelis intelektual dan ulama muda indonesia",
    "mjadikan": "menjadikan",
    "mjd": "menjadi",
    "mksd": "maksud",
    "mlah": "malah",
    "mlalui": "melalui",
    "mlm": "malam",
    "mls": "malas",
    "mmbuat": "membuat",
    "mmg": "memang",
    "mmng": "memang",
    "mndukung": "mendukung",
    "mngajarkan": "mengajarkan",
    "mngkn": "mungkin",
    "mninggal": "meninggal",
    "mnipu": "tipu",
    "mnjd": "menjadi",
    "mnolak": "tolak",
```

```
    "mnrtmu": "menurutmu",
    "mnt": "minta",
    "mnyumbang": "menyumbang",
    "mo": "mau",
    "modar": "mati",
    "moga": "semoga",
    "moge": "motor gede",
    "mokad": "mati",
    "mokat": "mati",
    "moso": "masa",
    "mosok": "masa",
    "most": "paling",
    "mpe": "sampai",
    "mpr": "majelis permusyawaratan rakyat",
    "mrk": "mereka",
    "msh": "masih",
    "msk": "masuk",
    "mskpn": "meskipun",
    "mslh": "masalah",
    "msng2": "masing-masing",
    "msuh": "musuh",
    "mt": "makan teman",
    "muahal": "mahal",
    "mubes": "musyawarah besar",
    "mui": "majelis ulama indonesia",
    "muker": "musyawarah kerja",
    "mulu": "melulu",
    "mumet": "pusing",
    "mumpung": "selagi",
    "muna": "munafik",
    "munas": "musyawarah nasional",
    "munaslub": "musyawarah nasional luar biasa",
    "munfik": "munafik",
    "muntaber": "muntah dan berak",
    "musda": "musyawarah daerah",
    "musti": "mesti",
    "muter2": "muter-muter",
    "muup": "maaf",
    "muupz": "maaf",
    "muuv": "maaf",
    "mw": "now watching",
    "mz": "mas",
    "n": "dan",
    "na": "nya",
    "naas": "nahas",
    "nak": "hendak",
    "nal": "kenal",
    "nan": "dan",
    "nanam": "menanam",
    "nangis": "menangis",
    "nanya": "bertanya",
    "naon": "apa",
    "napa": "kenapa",
    "napi": "narapidana",
    "napol": "narapidana politik",
    "napza": "narkotika, alkohol, psikotropika, dan zat adiktif",
    "naq": "anak",
    "narkoba": "narkotika, psikotropika, dan obat terlarang",
    "narsis": "bangga pada diri sendiri",
```

```
    "nasgor": "nasi goreng",
    "nax": "anak",
    "nda": "tidak",
    "ndak": "tidak",
    "ndiri": "sendiri",
    "ndut": "gendut",
    "ne": "ini",
    "negri": "negeri",
    "nekolim": "neokolonialisme",
    "nekolin": "neokolonialisme",
    "nelfon": "menelepon",
    "nembak": "menyatakan cinta",
    "next": "lanjut",
    "ngabis2in": "menghabiskan",
    "ngabuburit": "menunggu berbuka puasa",
    "ngakak": "tertawa",
    "ngaku": "mengaku",
    "ngambek": "marah",
    "ngambil": "mengambil",
    "ngampus": "pergi ke kampus",
    "ngamuk2": "ngamuk-ngamuk",
    "nganggur": "tidak punya pekerjaan",
    "ngantri": "mengantri",
    "ngapah": "kenapa",
    "ngapain": "sedang apa",
    "ngaret": "terlambat",
    "ngaruh": "berpengaruh",
    "ngasih": "memberikan",
    "ngawur": "berbicara sembarangan",
    "ngebandel": "berbuat bandel",
    "ngebully": "intimidasi",
    "ngeceng": "kumpul bareng-bareng",
    "ngegosip": "bergosip",
    "ngeh": "sadar",
    "ngehe": "kesal",
    "ngeklaim": "mengklaim",
    "ngekos": "tinggal di kos",
    "ngeksis": "menjadi eksis",
    "ngelamar": "melamar",
    "ngeles": "berkilah",
    "ngeliat": "melihat",
    "ngelidur": "menggigau",
    "ngemeng": "bicara terus-terusan",
    "ngentot": "senggama",
    "ngentotin": "senggama",
    "ngerampok": "merampok",
    "ngeri": "seram",
    "ngerti": "mengerti",
    "ngga": "tidak",
    "nggak": "tidak",
    "ngibul": "berbohong",
    "ngikut": "ikut",
    "ngiler": "mau",
    "nginep": "menginap",
    "ngiri": "iri",
    "ngisi": "mengisi",
    "ngisiin": "mengisikan",
    "ngmg": "bicara",
    "ngmng": "bicara",
```

```
    "ngn": "dengan",
    "ngo": "non governmental organization",
    "ngocol": "lucu",
    "ngomong": "bicara",
    "ngomongin": "membicarakan",
    "ngubek2": "mencari-cari",
    "ngumpet": "sembunyi",
    "ngumpul": "berkumpul",
    "ngurus": "mengurus",
    "ngusir2": "usir-usir",
    "nh": "nih",
    "ni": "ini",
    "nice": "bagus",
    "nie": "ini",
    "niech": "nih",
    "nih": "ini",
    "nikahkah": "nikahkan",
    "nimbrung": "ikut serta",
    "nipu": "tipu",
    "nitizen": "netizen",
    "niyh": "nih",
    "nk": "ingin",
    "nkah": "nikah",
    "nmenarik": "menarik",
    "nmr": "nomor",
    "nnt": "nanti",
    "nnti": "nanti",
    "nntn": "nonton",
    "no": "nomor",
    "nobar": "nonton bareng",
    "nongkrong": "kumpul-kumpul",
    "np": "now playing",
    "nrp": "narapidana",
    "ntap": "mantap",
    "ntar": "sebentar",
    "ntn": "nonton",
    "nu": "nahdatul ulama",
    "nuhun": "terima kasih",
    "numpuk": "bertumpuk",
    "nunggu": "menunggu",
    "nutupin": "menutupi",
    "ny": "nya",
    "nyaingin": "saing",
    "nyalahin": "salah",
    "nyampah": "membuat sampah",
    "nyampyr": "nyampur",
    "nyang": "yang",
    "nyari": "mencari",
    "nyariin": "mencari",
    "nyasar": "tersesat",
    "nyekar": "menyekar",
    "nyela": "cela",
    "nyembah": "sembah",
    "nyemot": "ambil",
    "nyerang": "serang",
    "nyiapin": "mempersiapkan",
    "nyicil": "mencicil",
    "nyiram": "menyiram",
    "nympr": "nyampur",
```

```
    "nyoblos": "mencoblos",
    "nyok": "ayo",
    "nyokap": "ibu",
    "o": "oleh",
    "o/": "oleh",
    "odmk": "orang dengan masalah kejiwaan",
    "ogah": "tidak mau",
    "oia": "oh iya",
    "ok": "ok",
    "okt": "oktober",
    "ol": "online",
    "omdo": "omong doang",
    "omg": "oh my god",
    "omong2": "ngomong-ngomong",
    "ongkir": "ongkos kirim",
    "oon": "bodoh",
    "oot": "out of topic",
    "orange": "oranye",
    "org": "orang",
    "org2": "orang-orang",
    "orglaen": "orang lain",
    "orkes": "olahraga dan kesehatan",
    "ortu": "orang tua",
    "ospek": "orientasi studi dan pengenalan kampus",
    "otda": "otonomi daerah",
    "otre": "ok",
    "otw": "on the way, sedang di jalan",
    "our": "kami",
    "p": "per",
    "p.s": "pesan",
    "p0rn": "porno",
    "pa": "apa",
    "pacal": "pacar",
    "pahe": "paket hemat",
    "pake": "pakai",
    "pakek": "pakai",
    "pala": "kepala",
    "panja": "panitia kerja",
    "pansus": "panitia khusus",
    "pantura": "pantai utara",
    "paranoid": "takut",
    "parno": "takut",
    "parpol": "partai politik",
    "parte": "partai",
    "pas2an": "pas-pasan",
    "pasutri": "pasangan suami istri",
    "patas": "cepat dan terbatas",
    "pd": "pada",
    "pdhl": "padahal",
    "pdip": "partai demokrasi Indonesia perjuangan",
    "pdskji": "perhimpunan dokter spesialis kedokteran jiwa
indonesia",
    "pdt": "padat",
    "pecahan2": "pecahan-pecahan",
    "pede": "percaya diri",
    "pelanggarab": "pelanggaran",
    "pelatda": "pemusatan latihan daerah",
    "pelatnas": "pemusatan latihan nasional",
    "pelcehn": "pelecehan",
```

```
    "pembokat": "pembantu",
    "pemda": "pemerintah daerah",
    "pemilu": "pemilihan umum",
    "pemko": "pemerintah kota",
    "pemkot": "pemerintah kota",
    "pemprov": "pemerintah provinsi",
    "pemred": "pemimpin redaksi",
    "pendukung2": "pendukung-pendukung",
    "penertiban": "tertib",
    "pengen": "ingin",
    "penjas": "pendidikan jasmani",
    "penmas": "pendidikan masyarakat",
    "perda": "peraturan daerah",
    "perek": "wanita tuna susila",
    "peremp": "perempuan",
    "pergub": "peraturan gubernur",
    "perhatiin": "perhatikan",
    "pertamax": "pertama",
    "pesenan": "pesanan",
    "pesenin": "pesankan",
    "pgang": "pegang",
    "pgi": "pergi",
    "pi": "tapi",
    "pihak2": "pihak pihak",
    "pijetin": "pijitkan",
    "pilgub": "pemilihan gubernur",
    "pilkada": "pemilihan kepala daerah",
    "pimpro": "pemimpin proyek",
    "pinter2": "pintar-pintar",
    "pisan": "sangat",
    "pisgor": "pisang goreng",
    "pk": "penjahat kelamin",
    "pke": "pakai",
    "pket": "paket",
    "pkr": "pakar",
    "pks": "partai keadilan sejahtera",
    "plg": "paling",
    "plh": "pilih",
    "pls": "tolong",
    "pm": "private message",
    "pmhaman": "pemahaman",
    "pmrnth": "pemerintah",
    "pndkung": "pendukung",
    "pndukung": "pendukung",
    "pnjilat": "penjilat",
    "pns": "pegawai negeri sipil",
    "pny": "punya",
    "poin2": "poin-poin",
    "polantas": "polisi lalu lintas",
    "polda": "kepolisian daerah",
    "polres": "kepolisian resor",
    "polwan": "polisi wanita",
    "pon": "juga",
    "ponpes": "pondok pesantren",
    "ponsel": "telepon seluler",
    "poster2": "poster-poster",
    "pp": "pulang pergi",
    "ppa": "apa",
    "presh": "segar",
```

```
    "prg": "pergi",
    "prikitiw": "cie",
    "priksa": "periksa",
    "prilaku": "perilaku",
    "prnh": "pernah",
    "pro": "profesional",
    "problem": "masalah",
    "prototype": "contoh",
    "prov": "provinsi",
    "psen": "pesan",
    "psn": "pesan",
    "psr": "pasar",
    "pst": "pasti",
    "psti": "pasti",
    "pswt": "pesawat",
    "ptemuan": "pertemuan",
    "puanas": "panas",
    "pulgar": "kasar",
    "puskesmas2": "puskesmas-puskesmas",
    "puyeng": "pusing",
    "pw": "posisi nyaman",
    "qmo": "kamu",
    "qmu": "kamu",
    "qt": "kita",
    "rakor": "rapat koordinasi",
    "rame": "ramai",
    "rang": "orang",
    "ranmor": "kendaraan bermotor",
    "raskin": "rakyat miskin",
    "rata2": "rata-rata",
    "re": "reply",
    "red": "redaksi",
    "ref": "referensi",
    "reg": "register",
    "rehab": "rehabilitasi",
    "rejeki": "rezeki",
    "rekan2": "rekan-rekan",
    "reklamuk": "reklamasi",
    "rempah2": "rempah-rempah",
    "rempong": "sulit",
    "renstra": "rencana strategis",
    "repp": "balas",
    "reskrim": "reserse kriminal",
    "restik": "reserse narkotika",
    "rhs": "rahasia",
    "rights": "benar",
    "rja": "raja",
    "rkyt": "rakyat",
    "rmbt": "rambut",
    "rmh": "rumah",
    "rofl": "rolling on the floor laughing",
    "ru": "baru",
    "rubahnn": "rubah",
    "rudal": "peluru kendali",
    "ruko": "rumah toko",
    "rusun": "rumah susun",
    "rusunawa": "rumah susun sewa",
    "rutan": "rumah tahanan",
    "ruu": "rancangan undang-undang",
```

```
    "ruz": "terus",
    "rwt": "rawat",
    "s4": "sempat",
    "sadaar": "sadar",
    "sah2": "sah-sah",
    "saia": "saya",
    "saiank": "sayang",
    "salting": "salah tingkah",
    "saltum": "salah kostum",
    "sampe": "sampai",
    "sampeyan": "anda",
    "samsek": "sama sekali",
    "sapa": "siapa",
    "sapose": "siapa",
    "satkorlak": "satuan koordinator pelaksana",
    "satpam": "satuan pengamanan",
    "say": "sayang",
    "sayan": "sayang",
    "sbaliknya": "sebaliknya",
    "sbb": "sebagai berikut",
    "sbg": "sebagai",
    "sbgn": "sebagian",
    "sbh": "sebuah",
    "sblm": "sebelum",
    "sbnr": "sebenar",
    "sbnrny": "sebenarnya",
    "sbntr": "sebentar",
    "scara": "secara",
    "scr": "secara",
    "sdg": "sedang",
    "sdgkn": "sedangkan",
    "sdh": "sudah",
    "sdkt": "sedikit",
    "sdngkn": "sedangkan",
    "se7": "setuju",
    "seabrek": "banyak",
    "sebelas dua belas": "mirip",
    "sebelom": "sebelum",
    "sebgmn": "sebagaimana",
    "sebuk": "sibuk",
    "sekjen": "sekretaris jenderal",
    "selaen": "selain",
    "selmt": "selamat",
    "semau": "asal",
    "sembako": "sembilan bahan pokok",
    "sempak": "celana dalam",
    "sempet": "sempat",
    "sempet2nya": "sempat-sempatnya",
    "sendratari": "seni drama tari",
    "seorg": "seorang",
    "sept": "september",
    "serasiii": "serasi",
    "sertijab": "serah terima jabatan",
    "sesok": "besok",
    "seting": "akting",
    "setting": "atur",
    "sgl": "segala",
    "sgrc": "support group and resource center",
    "sgt": "sangat",
```

```
    "sgtunya": "segitunya",
    "shg": "sehingga",
    "shrsnya": "seharusnya",
    "siap2": "siap-siap",
    "sidak": "inspeksi mendadak",
    "siech": "sih",
    "sieyh": "sih",
    "sigh": "sih",
    "sik": "sih",
    "sikon": "situasi dan kondisi",
    "simiskin": "miskin",
    "simpedes": "simpanan pedesaan",
    "sinetron": "sinema elektronik",
    "sip": "ok",
    "siramin": "siramkan",
    "sis": "kakak perempuan",
    "sj": "saja",
    "sjk": "sejak",
    "skali": "sekali",
    "skalian": "sekalian",
    "skl": "sekali",
    "sklh": "sekolah",
    "skrg": "sekarang",
    "skt": "sakit",
    "skul": "sekolah",
    "slalu": "selalu",
    "slama": "selama",
    "slesai": "selesai",
    "slh": "salah",
    "sll": "selalu",
    "sllu": "selalu",
    "slm": "salam",
    "slma": "selama",
    "slmt": "selamat",
    "slsai": "selesai",
    "slt": "sulit",
    "sm": "sama",
    "sma": "sama",
    "smg": "semoga",
    "smkn": "semakin",
    "smpt": "sempat",
    "smua": "semua",
    "smw": "semua",
    "smwny": "semuanya",
    "sndiri": "sendiri",
    "sng": "senang",
    "sni": "sini",
    "soft": "halus",
    "sohib": "teman",
    "sokong": "dukung",
    "soljum": "sholat jumat",
    "solve": "solusi",
    "somse": "sombong sekali",
    "songong": "sombong",
    "sorry": "maaf",
    "sory": "maaf",
    "sosbud": "sosial-budaya",
    "sosek": "sosial-ekonomi",
    "sospol": "sosial-politik",
```

```
    "sotoy": "sok tahu",
    "sowry": "maaf",
    "spa": "siapa",
    "spd": "sepeda",
    "sppa": "siapa",
    "sprti": "seperti",
    "spt": "seperti",
    "spy": "supaya",
    "srt": "surat",
    "srtfkt": "sertifikat",
    "stelah": "setelah",
    "stiap": "setiap",
    "still": "masih",
    "stlh": "setelah",
    "stop": "henti",
    "subbag": "subbagian",
    "suk": "masuk",
    "sumbangin": "sumbangkan",
    "sumpek": "sempit",
    "sy": "saya",
    "sya": "saya",
    "syg": "sayang",
    "syp": "siapa",
    "t4": "tempat",
    "tabanas": "tabungan pembangunan nasional",
    "tahi": "kotoran",
    "tai": "kotoran",
    "taik": "tahi",
    "tajir": "kaya",
    "tak": "tidak",
    "tar": "nanti",
    "tarik2": "tarik-tarik",
    "tau": "tahu",
    "tauk": "tahu",
    "taun": "tahun",
    "taw": "tahu",
    "tawh": "tahu",
    "tayangan2": "tayangan-tayangan",
    "td": "tadi",
    "tdi": "tadi",
    "tdk": "tidak",
    "te2p": "tetap",
    "teh": "kakak perempuan",
    "teken": "tekan",
    "tekor": "rugi",
    "telat": "terlambat",
    "telkom": "telekomunikasi",
    "telmi": "telat berpikir",
    "telp": "telepon",
    "teman2": "teman-teman",
    "temen": "teman",
    "temen2": "teman-teman",
    "tenang2": "tenang tenang",
    "tengil": "menyebalkan",
    "tengok": "menjenguk",
    "tepar": "terkapar",
    "terbitin": "terbitkan",
    "tetp": "tetap",
    "teve": "televisi",
```

```
    "tggu": "tunggu",
    "tgl": "tanggal",
    "tgok": "tengok",
    "tgu": "tunggu",
    "thanks": "terima kasih",
    "thankz": "terima kasih",
    "thd": "terhadap",
    "thdp": "terhadap",
    "thn": "tahun",
    "thx": "thanks",
    "tilang": "bukti pelanggaran",
    "tilep": "tilap",
    "tioghoa": "tionghoa",
    "tipi": "TV",
    "tipiwan": "TvOne",
    "tirani": "tiran",
    "tk": "tidak",
    "tkg": "tukang",
    "tks": "terima kasih",
    "tl": "timeline",
    "tll": "terlalu",
    "tlp": "telepon",
    "tlpn": "telepon",
    "tls": "tulis",
    "tman": "teman",
    "tmbah": "tambah",
    "tmbh": "tambah",
    "tmen2": "teman-teman",
    "tmn": "teman",
    "tmn2": "teman-teman",
    "tmpah": "tumpah",
    "tmph": "tumpah",
    "tmpt": "tempat",
    "tnda": "tanda",
    "tngu": "tunggu",
    "tnh": "tanah",
    "tni": "tentara nasional indonesia",
    "tnp": "tanpa",
    "tntg": "tentang",
    "tnya": "tanya",
    "tnyta": "ternyata",
    "togel": "toto gelap",
    "tokai": "tai",
    "tokoh2": "tokoh-tokoh",
    "toserba": "toko serba ada",
    "toys": "main",
    "tp": "tapi",
    "tpi": "tetapi",
    "tq": "terima kasih",
    "trdhulu": "terdahulu",
    "tren": "trend",
    "trending": "trend",
    "trfkir": "terfikir",
    "trgntg": "tergantung",
    "trh": "terhadap",
    "trhdp": "terhadap",
    "trima": "terima kasih",
    "trims": "terima kasih",
    "trimz": "terima kasih",
```

```
    "trkhr": "terakhir",
    "trlalu": "terlalu",
    "trllu": "terlalu",
    "trm": "terima",
    "trmsk": "termasuk",
    "trnyt": "ternyata",
    "trs": "terus",
    "trus": "terus",
    "trutama": "terutama",
    "trz": "terus",
    "ts": "penulis",
    "tsangka": "tersangka",
    "tsb": "tersebut",
    "tsk": "tersangka",
    "tst": "tahu sama tahu",
    "ttd": "tertanda",
    "ttg": "tentang",
    "ttng": "tentang",
    "ttp": "tetap",
    "tu": "itu",
    "tuch": "tuh",
    "tuh": "itu",
    "tuir": "tua",
    "tuwh": "tuh",
    "tv": "televisi",
    "tw": "tahu",
    "twit2": "tweet-tweet",
    "twt": "tweet",
    "tyap": "tiap",
    "u": "untuk",
    "u/": "untuk",
    "uasai": "kuasa",
    "ud": "sudah",
    "uda": "sudah",
    "udah": "sudah",
    "udh": "sudah",
    "ugm": "universitas gajah mada",
    "ui": "universitas indonesia",
    "ujg": "ujung",
    "ujung2nya": "ujung-ujungnya",
    "ul": "ulangan",
    "ultah": "ulang tahun",
    "unair": "universitas airlangga",
    "undang2": "undang undang",
    "undip": "universitas diponegoro",
    "univ": "universitas",
    "unyu": "menggemaskan",
    "uplot": "unggah",
    "uppu": "apa",
    "urang": "saya",
    "urg": "saya",
    "usah": "perlu",
    "ust": "ustadz",
    "ut": "untuk",
    "utang": "hutang",
    "utip": "kutip",
    "utk": "untuk",
    "utul": "ujian tertulis",
    "uu": "undang-undang",
```

```
    "valas": "valuta asing",
    "valid": "sah",
    "victory": "menang",
    "vol": "volume",
    "vote": "pilih",
    "w/": "dengan",
    "wad": "buat",
    "wadir": "wakil direktur",
    "wagub": "wakil gubernur",
    "wamil": "wajib militer",
    "war": "perang",
    "warbiyasak": "luar biasa",
    "waria": "wanita pria",
    "warkop": "warung kopi",
    "warnet": "warung internet",
    "warteg": "warung tegal",
    "wartel": "warung telekomunikasi",
    "waspadalah": "waspada",
    "wat": "buat",
    "watuk": "batuk",
    "who": "world health organization",
    "wk": "tertawa",
    "wkt": "waktu",
    "wkwk": "tertawa",
    "wkwkkw": "tertawa",
    "wtf": "what the fuck",
    "wth": "what the hell",
    "x": "tidak",
    "xan": "tak akan",
    "xew": "apa",
    "xixixi": "tertawa",
    "y": "ya",
    "ya": "iya",
    "yad": "yang akan datang",
    "yap": "iya",
    "yapz": "iya",
    "yaudah": "ya sudah",
    "yauwda": "ya sudah",
    "yawdah": "ya sudah",
    "ybs": "yang bersangkutan",
    "yg": "yang",
    "yi": "yaitu",
    "yl": "yang lain",
    "yng": "yang",
    "yo": "iya",
    "yoha": "iya",
    "yowes": "ya sudah",
    "yuk": "ayo",
    "yup": "iya",
    "ywdh": "ya sudah",
    "zonk": "bodoh"
}
```

18. models/indonesia/spell.py

```
import json
import pandas as pd
```

```
with open('slang.dic') as f:
    slang = json.load(f)

kamus_slang = pd.read_json('slang.dic',typ = 'series')
pd.DataFrame(kamus_slang.items(), columns = ['contraction',
'original']).head()
contractions_list = list(kamus_slang.keys())

def spell_check(text):
    words = []
    for word in text.split():
        if word in contractions_list:
            words = words + kamus_slang[word].split()
        else:
            words = words + word.split()

    text_converted = " ".join(words)
    return text_converted
```

19. public/css/master.css

```
/* ELEMENTS */

textarea.form-control {
  height: 40vmin;
}

/* CLASSES */

.form_language {
  /* display: block; */
  /* width: 100%; */
  padding: 0.375rem 2.25rem 0.375rem 0.75rem;
  -moz-padding-start: calc(0.75rem - 3px);
  font-size: 1rem;
  font-weight: 400;
  line-height: 1.5;
  color: #212529;
  background-color: #fff;
  background-image: url("data:image/svg+xml,%3csvg
xmlns='http://www.w3.org/2000/svg' viewBox='0 0 16 16'%3e%3cpath
fill='none' stroke='%23343a40' stroke-linecap='round' stroke-
linejoin='round' stroke-width='2' d='m2 5 6 6 6-6'/%3e%3c/svg%3e");
  background-repeat: no-repeat;
  background-position: right 0.75rem center;
  background-size: 16px 12px;
  border: 1px solid #ced4da;
  border-radius: 0.375rem;
  transition: border-color 0.15s ease-in-out, box-shadow 0.15s ease-
in-out;
  -webkit-appearance: none;
  -moz-appearance: none;
  appearance: none;
}
@media (prefers-reduced-motion: reduce) {
  .form_language {
```

```
    transition: none;
  }
}

.help_icon {
  margin-left: auto;
  width: fit-content;
}

.score_line {
  width: 100%;
}

.select_local {
  width: fit-content;
}

.service_explanation {
  text-align: justify;
  margin: 1rem;
}
```

20. public/js/milisecondDate.js

```
let documentID = new Date().getMilliseconds();

document.getElementById("docid").value = documentID;
```

21. public/js/printResult.js

```
document.getElementById("print").addEventListener("click", () => {
  document.getElementById("print").style.opacity = 0;
  window.print();
  document.getElementById("print").style.opacity = 1;
  // setTimeout(() => {
  // }, 1000);
});
```

22. public/pages/bulkID1.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Document</title>

    <!-- Bootstrap CSS -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
```

```
      crossorigin="anonymous"
    />

    <!-- our CSS -->
    <link rel="stylesheet" href="/css/master.css" />

    <!-- FontAwesome JS -->
    <script
      src="https://kit.fontawesome.com/6b522f63d7.js"
      crossorigin="anonymous"
    ></script>

    <!-- XLS to JSON -->
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/xlsx/0.16.2/xlsx.full.mi
n.js"></script>
  </head>

  <body>
    <!-- <h2>Header</h2> -->



<div class="container-fluid">
  <div class="row">
    <div class="col col-6"><div>
  <a href="/">
    <h1><i class="fa-solid fa-file-circle-check"></i> A Title</h1>
  </a>
</div>
</div>
    <div class="col col-6 d-flex justify-content-end mb-5">
      <h1><i class="fa-regular fa-file-excel"></i>Indonesia</h1>
    </div>
  </div>
  <div>
    <p class="form-label">Kunci Jawaban</p>
    <div class="form-control">
```

23. public/pages/bulkID2.html

```
</div>
  </div>
  <!-- <div class="text-center container-fluid">
    <p>Mohon mengisi survey kepuasan anda terhadap Easysay</p>
    <button type="button" class="btn btn-info">
      <a
        href="https://forms.gle/FApH5uzhVbRB4NtY9"
        target="_blank"
        class="link-body-emphasis"
        >Survey 3 menit</a
      >
    </button>
  </div> -->
  <div>
    <p class="form-label">Hasil Penilaian</p>
    <div class="row">
```

```
      <table class="table">
        <thead>
          <tr>
            <th scope="col">Nama</th>
            <th scope="col">Jawaban</th>
            <th scope="col">Nilai</th>
          </tr>
        </thead>
        <tbody>
```

24. public/pages/bulkID3.html

```
</tbody>
      </table>
    </div>
  </div>
</div>
<div class="container-fluid text-center mb-2">
  <button type="button" id="print" class="btn btn-secondary">
    Export to PDF
  </button>
</div>
<script src="js/printResult.js"></script>

<!-- <h3>Footer</h3> -->
<!-- Bootstrap JS -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfAdVQSZe"
      crossorigin="anonymous"
    ></script>
</body>

</html>
```

25. app.js

```
const express = require("express");
const bodyParser = require("body-parser");
const path = require("path");
const CSVToJSON = require("csvtojson");
const JSONToCSV = require("json2csv").parse;
var fs = require("fs");
const { PythonShell } = require("python-shell");
const multer = require("multer");

let bulkDocumentID;
const storage = multer.diskStorage({
  destination: (req, file, cb) => {
    cb(null, "./");
  },
  filename: (req, file, cb) => {
    bulkDocumentID = new Date().getMilliseconds();
    cb(null, bulkDocumentID + ".csv");
```

```
  },
});
const upload = multer({ storage: storage });

const app = express();

app.set("views", [path.join(__dirname, "views")]);
app.set("view engine", "ejs");

app.use(bodyParser.urlencoded({ extended: true, limit: "50mb" }));
app.use(express.static("public"));
app.use(express.json({ limit: "50mb" }));

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

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

app.route("/option").post((req, res) => {
  const obj = {
    language: req.body.language,
    select: req.body.select,
  };
  if (obj.language === "Indonesia") {
    if (obj.select === "single") {
      res.render("indonesia/single", { language: obj.language });
    } else {
      res.render("indonesia/bulkJQuery");
    }
  } else if (obj.language === "Inggris") {
    if (obj.select === "single") {
      res.render("inggris/single");
    } else {
      res.render("inggris/bulk");
    }
  } else {
    res.redirect("/");
  }
});

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 = {
    mode: "json",
    pythonOptions: ["-u"],
```

```
    args: [docid],
  };

  PythonShell.run("models/english/aprilModel.py", options)
    .then((messages) => {
      scoreLoad = messages;
    })
    .then(() => {
      fs.unlink(docid + ".csv", (err) => {
        //delete the csv file from crowding
        if (err) throw err;
      });
    })
    .then(() => {
      res.render("result/singleResult", {
        language: language,
        keyAnswer: scoreLoad[0].keyAnswer["0"],
        answer: scoreLoad[0].studentAnswer["0"],
        score: ((scoreLoad[0].scoreModelStem["0"] / 5) *
100).toFixed(2) + "%",
      });
    })
    .catch((err) => {
      res.render("result/singleResult", {
        language: language,
        keyAnswer: keyAnswer,
        answer: answer,
        score:
          "Submitted paragraphs may not contain meaningful words for
the app to check, so we cannot continue the calculation",
      });
    });
});

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) => {
      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);

      let options = {
        mode: "json",
        pythonOptions: ["-u"],
        args: [docid],
      };
```

```
      PythonShell.run("models/english/aprilModel.py", options)
        .then((messages) => {
          scoreLoad = messages;
        })
        .then(() => {
          fs.unlink(docid + ".csv", (err) => {
            if (err) throw err;
          });
        })
        .then(() => {
          res.render("result/bulkResult", {
            language: language,
            keyAnswer: keyAnswer,
            givenObjectKeyLength:
Object.keys(scoreLoad[0].studentName).length,
            objectName: scoreLoad[0].studentName,
            objectAnswer: scoreLoad[0].studentAnswer,
            objectScore: scoreLoad[0].scoreModelStem,
          });
        });
    });
});

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

  let option = {
    mode: "text",
    pythonOptions: ["-u"],
    args: [teacherAnswer, studentAnswer, docid],
  };

  PythonShell.run("./models/indonesia/ASAG.py",
option).then((messages) => {
    let txtfile = fs
      .readFileSync(docid + ".txt", "utf-8")
      .replace(/\r/g, "")
      .split("\n");

    fs.unlink(docid + ".txt", (err) => {
      if (err) throw err;
    });
    res.render("result/singleResult", {
      language: language,
      keyAnswer: teacherAnswer,
      score: txtfile[0] + "%",
      answer: studentAnswer,
    });
  });
});

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
  ) {
    fs.unlink(docid + ".csv", (err) => {
      if (err) throw err;
    });
    res.write(
      '<div class="alert alert-warning" role="alert">Kolom Nama dan
/ atau kolom jawaban pelajar sepertinya tidak cocok dengan apa yang
ada di file CSV, mohon di cek lagi karena harus sama persis
namanya</div>'
    );
    res.end();
  } else {
    for (i = 0; i < sources.length; i++) {
      sources[i].studentName = sources[i][nameColumns];
      sources[i].studentAnswer = sources[i][answerColumns];
    }
    for (let i = 0; i < sources.length; i++) {
      const options = {
        mode: "text",
        pythonOptions: ["-u"],
        args: [teacherAnswer, sources[i][answerColumns], docid],
      };
      await PythonShell.run("./models/indonesia/ASAGbulk.py",
options).then(
        (messages, err) => {
          console.log("error LINE 267");
          res.write("<tr>");
          res.write("<td>");
          res.write(sources[i].studentName);
          res.write("</td>");
          res.write("<td>");
          res.write(sources[i].studentAnswer);
          res.write("</td>");
          res.write("<td>");
          res.write(messages[0]);
          res.write("</td>");
          res.write("</tr>");
          res.write("<br />");
        }
      );
    }
    res.write(idbulk3);
```

```
    fs.unlink(docid + ".csv", (err) => {
      if (err) throw err;
    });
    res.end();
  }
});

app.route("/ScenarioEn").post((req, res) => {
  const { key, ans } = req.body; //json recieved from postman has
key and ans, put in their own variable

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

  fs.writeFileSync("ScenarioEn.csv", Dataset, "utf8");

  let options = {
    mode: "json",
    pythonOptions: ["-u"],
    args: ["ScenarioEn"],
  };

  PythonShell.run("models/english/aprilModel.py",
options).then((messages) => {
    // res.send(((messages[0].scoreModelStem["0"] / 5) *
100).toFixed(2) + "%");
    // code above is to send just the score in a 0 to 100% percent
range
    res.send(messages);
    // code above is to send the whole scoring processes as the
model does
  });
});

app.route("/ScenarioEnTab").post((req, res) => {
  const { key } = req.body;

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

      let options = {
        mode: "json",
        pythonOptions: ["-u"],
        args: ["dataset"],
      };

      PythonShell.run("models/english/aprilModelTab.py",
options).then(
        (messages) => {
          res.send(messages[0].scoreModelStem);
          // res.send(messages);
        }
      );
    });
```

```
});

app.route("/ScenarioId").post((req, res) => {
  const { key, ans } = req.body; //json recieved from postman has
key and ans, put in their own variable

  let option = {
    mode: "text",
    pythonOptions: ["-u"],
    args: [key, ans, "ScenarioId"],
  };

  PythonShell.run("./models/indonesia/ASAG.py",
option).then((messages) => {
    let txtfile = fs
      .readFileSync("ScenarioId" + ".txt", "utf-8")
      .replace(/\r/g, "")
      .split("\n");

    fs.unlink("ScenarioId" + ".txt", (err) => {
      if (err) throw err;
    });
    res.send(txtfile[0] + "%");
  });
});

app.listen(process.env.PORT || 1234, () => {
  console.log("Listening to http://localhost:1234");
});
```

26. index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0" />
    <title>Easysay</title>
  </head>
  <body>
    <script>
      window.location.href = "http://103.37.125.146:1234/";
    </script>
  </body>
</html>
```

27. package-lock.json

```
{
  "name": "product",
  "version": "1.0.0",
  "lockfileVersion": 2,
  "requires": true,
  "packages": {
    "": {
```

```
      "name": "product",
      "version": "1.0.0",
      "license": "ISC",
      "dependencies": {
        "body-parser": "^1.20.1",
        "csv": "^6.3.0",
        "csv-generate": "^4.2.6",
        "csvtojson": "^2.0.10",
        "ejs": "^3.1.8",
        "express": "^4.18.2",
        "formidable": "^2.1.1",
        "json-to-csv-export": "^2.1.1",
        "json2csv": "^6.0.0-alpha.2",
        "localtunnel": "^2.0.2",
        "multer": "^1.4.5-lts.1",
        "nodemon": "^2.0.22",
        "papaparse": "^5.4.1",
        "python-shell": "^5.0.0"
      }
    },
    "node_modules/@streamparser/json": {
      "version": "0.0.6",
      "resolved": "https://registry.npmjs.org/@streamparser/json/-
/json-0.0.6.tgz",
      "integrity": "sha512-
vL9EVn/v+OhZ+Wcs6O4iKE9EUpwHUqHmCtNUMWjqp+6dr85+XPOSGTEsqYNq1Vn04uk9
SWlOVmx9J48ggJVT2Q=="
    },
    "node_modules/abbrev": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/abbrev/-/abbrev-
1.1.1.tgz",
      "integrity": "sha512-
nne9/IiQ/hzIhY6pdDnbBtz7DjPTKrY00P/zvPSm5pOFkl6xuGrGnXn/VtTNNfNtAfZ9
/1RtehkszU9qcTii0Q=="
    },
    "node_modules/accepts": {
      "version": "1.3.8",
      "resolved": "https://registry.npmjs.org/accepts/-/accepts-
1.3.8.tgz",
      "integrity": "sha512-
PYAthTa2m2VKxuvSD3DPC/Gy+U+sOA1LAuT8mkmRuvw+NACSaeXEQ+NHcVF7rONl6qca
xV3Uuemwawk+7+SJLw==",
      "dependencies": {
        "mime-types": "~2.1.34",
        "negotiator": "0.6.3"
      },
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/ansi-regex": {
      "version": "5.0.1",
      "resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-
regex-5.0.1.tgz",
      "integrity": "sha512-
quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMW
v37qOOb9pdJlMUEKFQ==",
      "engines": {
```

```
        "node": ">=8"
      }
    },
    "node_modules/ansi-styles": {
      "version": "4.3.0",
      "resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-
styles-4.3.0.tgz",
      "integrity": "sha512-
zbB9rCJAT1rbjiVDb2hqKFHNYLxgtk8NURxZ3IZwD3F6NtxbXZQCnnSi1Lkx+IDohdPl
Fp222wVALIheZJQSEg==",
      "dependencies": {
        "color-convert": "^2.0.1"
      },
      "engines": {
        "node": ">=8"
      },
      "funding": {
        "url": "https://github.com/chalk/ansi-styles?sponsor=1"
      }
    },
    "node_modules/anymatch": {
      "version": "3.1.3",
      "resolved": "https://registry.npmjs.org/anymatch/-/anymatch-
3.1.3.tgz",
      "integrity": "sha512-
KMReFUr0B4t+D+OBkjR3KYqvocp2XaSzO55UcB6mgQMd3KbcE+mWTyvVV7D/zsdEbNnV
6acZUutkiHQXvTr1Rw==",
      "dependencies": {
        "normalize-path": "^3.0.0",
        "picomatch": "^2.0.4"
      },
      "engines": {
        "node": ">= 8"
      }
    },
    "node_modules/append-field": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/append-field/-/append-
field-1.0.0.tgz",
      "integrity": "sha512-
klpgFSWLW1ZEs8svjfb7g4qWY0YS5imI82dTg+QahUvJ8YqAY0P10Uk8tTyh9ZGuYEZE
MaeJYCF5BFuX552hsw=="
    },
    "node_modules/array-flatten": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/array-flatten/-/array-
flatten-1.1.1.tgz",
      "integrity": "sha512-
PCVAQswWemu6UdxsDFFX/+gVeYqKAod3D3UVm91jHwynguOwAvYPhx8nNlM++NqRcK6C
xxpUafjmhIdKiHibqg=="
    },
    "node_modules/asap": {
      "version": "2.0.6",
      "resolved": "https://registry.npmjs.org/asap/-/asap-
2.0.6.tgz",
      "integrity": "sha512-
BSHWgDSAiKs50o2Re8ppvp3seVHXSRM44cdSsT9FfNEUUZLOGWVCsiWaRPWM1Znn+mqZ
1OfVZ3z3DWEzSp7hRA==",
    },
```

```
    "node_modules/async": {
      "version": "3.2.4",
      "resolved": "https://registry.npmjs.org/async/-/async-
3.2.4.tgz",
      "integrity": "sha512-
iAB+JbDEGXhyIUavoDl9WP/Jj106Kz9DEn1DPgYw5ruDn0e3Wgi3sKFm55sASdGBNOQB
8F59d9qQ7deqrHA8wQ=="
    },
    "node_modules/axios": {
      "version": "0.21.4",
      "resolved": "https://registry.npmjs.org/axios/-/axios-
0.21.4.tgz",
      "integrity": "sha512-
ut5vewkiu8jjGBdqpM44XxjuCjq9LAKeHVmoVfHVzy8eHgxxq8SbAVQNovDA8mVi05kP
0Ea/n/UzcSHcTJQfNg==",
      "dependencies": {
        "follow-redirects": "^1.14.0"
      }
    },
    "node_modules/balanced-match": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/balanced-match/-
/balanced-match-1.0.2.tgz",
      "integrity": "sha512-
3oSeUO0TMV67hN1AmbXsK4yaqU7tjiHlbxRDZOpH0KW9+CeX4bRAaX0Anxt0tx2MrpRp
WwQaPwIlISEJhYU5Pw=="
    },
    "node_modules/binary-extensions": {
      "version": "2.2.0",
      "resolved": "https://registry.npmjs.org/binary-extensions/-
/binary-extensions-2.2.0.tgz",
      "integrity": "sha512-
jDctJ/IVQbZoJykoeHbhXpOlNBqGNcwXJKJog42E5HDPUwQTSdjCHdihjj0DlnheQ7bl
bT6dHOafNAiS8ooQKA==",
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/bluebird": {
      "version": "3.7.2",
      "resolved": "https://registry.npmjs.org/bluebird/-/bluebird-
3.7.2.tgz",
      "integrity": "sha512-
XpNj6GDQzdfW+r2Wnn7xiSAd7TM3jzkxGXBGTtWKuSXv1xUV+azxAm8jdWZN06QTQk+2
N2XB9jRDkvbmQmcRtg=="
    },
    "node_modules/body-parser": {
      "version": "1.20.1",
      "resolved": "https://registry.npmjs.org/body-parser/-/body-
parser-1.20.1.tgz",
      "integrity": "sha512-
jWi7abTbYwajOytWCQc37VulmWiRae5RyTpaCyDcS5/lMdtwSz5lOpDE67srw/HYe35f
1z3fDQw+3txg7gNtWw==",
      "dependencies": {
        "bytes": "3.1.2",
        "content-type": "~1.0.4",
        "debug": "2.6.9",
        "depd": "2.0.0",
        "destroy": "1.2.0",
```

```
        "http-errors": "2.0.0",
        "iconv-lite": "0.4.24",
        "on-finished": "2.4.1",
        "qs": "6.11.0",
        "raw-body": "2.5.1",
        "type-is": "~1.6.18",
        "unpipe": "1.0.0"
      },
      "engines": {
        "node": ">= 0.8",
        "npm": "1.2.8000 || >= 1.4.16"
      }
    },
    "node_modules/brace-expansion": {
      "version": "1.1.11",
      "resolved": "https://registry.npmjs.org/brace-expansion/-
/brace-expansion-1.1.11.tgz",
      "integrity": "sha512-
iCuPHDFgrHX7H2vEI/5xpz07zSHB00TpugqhmYtVmMO6518mCuRMoOYFldEBl0g187uf
ozdaHgWKcYFb61qGiA==",
      "dependencies": {
        "balanced-match": "^1.0.0",
        "concat-map": "0.0.1"
      }
    },
    "node_modules/braces": {
      "version": "3.0.2",
      "resolved": "https://registry.npmjs.org/braces/-/braces-
3.0.2.tgz",
      "integrity": "sha512-
b8um+L1RzM3WDSzvhm6gIz1yfTbBt6YTlcEKAvsmqCZZFw46z626lVj9j1yEPW33H5H+
lBQpZMP1k8l+78Ha0A==",
      "dependencies": {
        "fill-range": "^7.0.1"
      },
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/buffer-from": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/buffer-from/-/buffer-
from-1.1.2.tgz",
      "integrity": "sha512-
E+XQCRwSbaaiChtv6k6Dwgc+bx+Bs6vuKJHHl5kox/BaKbhiXzqQOwK4cO22yElGp2OC
mjwVhT3HmxgyPGnJfQ=="
    },
    "node_modules/busboy": {
      "version": "1.6.0",
      "resolved": "https://registry.npmjs.org/busboy/-/busboy-
1.6.0.tgz",
      "integrity": "sha512-
8SFQbg/0hQ9xy3UNTB0YEnsNBbWfhf7RtnzpL7TkBiTBRfrQ9Fxcnz7VJsleJpyp6rVL
vXiuORqjlHi5q+PYuA==",
      "dependencies": {
        "streamsearch": "^1.1.0"
      },
      "engines": {
        "node": ">=10.16.0"
```

```
      }
    },
    "node_modules/bytes": {
      "version": "3.1.2",
      "resolved": "https://registry.npmjs.org/bytes/-/bytes-
3.1.2.tgz",
      "integrity": "sha512-
/Nf7TyzTx6S3yRJObOAV7956r8cr2+Oj8AC5dt8wSP3BQAoeX58NoHyCU8P8zGkNXStj
TSi6fzO6F0pBdcYbEg==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/call-bind": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/call-bind/-/call-bind-
1.0.2.tgz",
      "integrity": "sha512-
7O+FbCihrB5WGbFYesctwmTKae6rOiIzmz1icreWJ+0aA7LJfuqhEso2T9ncpcFtzMQt
zXf2QGGueWJGTYsqrA==",
      "dependencies": {
        "function-bind": "^1.1.1",
        "get-intrinsic": "^1.0.2"
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/chalk": {
      "version": "4.1.2",
      "resolved": "https://registry.npmjs.org/chalk/-/chalk-
4.1.2.tgz",
      "integrity": "sha512-
oKnbhFyRIXpUuez8iBMmyEa4nbj4IOQyuhc/wy9kY7/WVPcwIO9VA668Pu8RkO7+0G76
SLROeyw9CpQ061i4mA==",
      "dependencies": {
        "ansi-styles": "^4.1.0",
        "supports-color": "^7.1.0"
      },
      "engines": {
        "node": ">=10"
      },
      "funding": {
        "url": "https://github.com/chalk/chalk?sponsor=1"
      }
    },
    "node_modules/chokidar": {
      "version": "3.5.3",
      "resolved": "https://registry.npmjs.org/chokidar/-/chokidar-
3.5.3.tgz",
      "integrity": "sha512-
Dr3sfKRP6oTcjf2JmUmFJfeVMvXBdegxB0iVQ5eb2V10uFJUCAS8OByZdVAyVb8xXNz3
GjjTgj9kLWsZTqE6kw==",
      "funding": [
        {
          "type": "individual",
          "url": "https://paulmillr.com/funding/"
        }
      ],
```

```
      "dependencies": {
        "anymatch": "~3.1.2",
        "braces": "~3.0.2",
        "glob-parent": "~5.1.2",
        "is-binary-path": "~2.1.0",
        "is-glob": "~4.0.1",
        "normalize-path": "~3.0.0",
        "readdirp": "~3.6.0"
      },
      "engines": {
        "node": ">= 8.10.0"
      },
      "optionalDependencies": {
        "fsevents": "~2.3.2"
      }
    },
    "node_modules/cliui": {
      "version": "7.0.4",
      "resolved": "https://registry.npmjs.org/cliui/-/cliui-
7.0.4.tgz",
      "integrity": "sha512-
OcRE68cOsVMXp1Yvonl/fzkQOyjLSu/8bhPDfQt0e0/Eb283TKP20Fs2MqoPsr9SwA59
5rRCA+QMzYc9nBP+JQ==",
      "dependencies": {
        "string-width": "^4.2.0",
        "strip-ansi": "^6.0.0",
        "wrap-ansi": "^7.0.0"
      }
    },
    "node_modules/color-convert": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/color-convert/-/color-
convert-2.0.1.tgz",
      "integrity": "sha512-
RRECPsj7iu/xb5oKYcsFHSppFNnsj/52OVTRKb4zP5onXwVF3zVmmToNcOfGC+CRDpfK
/U584fMg38ZHCaElKQ==",
      "dependencies": {
        "color-name": "~1.1.4"
      },
      "engines": {
        "node": ">=7.0.0"
      }
    },
    "node_modules/color-name": {
      "version": "1.1.4",
      "resolved": "https://registry.npmjs.org/color-name/-/color-
name-1.1.4.tgz",
      "integrity": "sha512-
dOy+3AuW3a2wNbZHIuMZpTcgjGuLU/uBL/ubcZF9OXbDo8ff4O8yVp5Bf0efS8uEoYo5
q4Fx7dY9OgQGXgAsQA=="
    },
    "node_modules/commander": {
      "version": "6.2.1",
      "resolved": "https://registry.npmjs.org/commander/-/commander-
6.2.1.tgz",
      "integrity": "sha512-
U7VdrJFnJgo4xjrHpTzu0yrHPGImdsmD95ZlgYSEajAn2JKzDhDTPG9kBTefmObL2w/n
geZnilk+OV9CG3d7UA==",
      "engines": {
```

```
        "node": ">= 6"
      }
    },
    "node_modules/concat-map": {
      "version": "0.0.1",
      "resolved": "https://registry.npmjs.org/concat-map/-/concat-
map-0.0.1.tgz",
      "integrity": "sha512-
/Srv4dswyQNBfohGpz9o6Yb3Gz3SrUDqBH5rTuhGR7ahtlbYKnVxw2bCFMRljaA7EXHa
XZ8wsHdodFvbkhKmqg=="
    },
    "node_modules/concat-stream": {
      "version": "1.6.2",
      "resolved": "https://registry.npmjs.org/concat-stream/-
/concat-stream-1.6.2.tgz",
      "integrity": "sha512-
27HBghJxjiZtIk3Ycvn/4kbJk/1uZuJFfuPEns6LaEvpvG1f0hTea8lilrouyo9mVc2G
WdcEZ8OLoGmSADlrCw==",
      "engines": [
        "node >= 0.8"
      ],
      "dependencies": {
        "buffer-from": "^1.0.0",
        "inherits": "^2.0.3",
        "readable-stream": "^2.2.2",
        "typedarray": "^0.0.6"
      }
    },
    "node_modules/content-disposition": {
      "version": "0.5.4",
      "resolved": "https://registry.npmjs.org/content-disposition/-
/content-disposition-0.5.4.tgz",
      "integrity": "sha512-
FveZTNuGw04cxlAiWbzi6zTAL/lhehaWbTtgluJh4/E95DqMwTmha3KZN1aAWA8cFIhH
zMZUvLevkw5Rqk+tSQ==",
      "dependencies": {
        "safe-buffer": "5.2.1"
      },
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/content-type": {
      "version": "1.0.5",
      "resolved": "https://registry.npmjs.org/content-type/-
/content-type-1.0.5.tgz",
      "integrity": "sha512-
nTjqfcBFEipKdXCv4YDQWCfmcLZKm81ldF0pAopTvyrFGVbcR6P/VAAd5G7N+0tTr8Qq
iU0tFadD6FK4NtJwOA==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/cookie": {
      "version": "0.5.0",
      "resolved": "https://registry.npmjs.org/cookie/-/cookie-
0.5.0.tgz",
      "integrity": "sha512-
YZ3GUyn/o8gfKJlnlX7g7xq4gyO6OSuhGPKaaGssGB2qgDUS0gPgtTvoyZLTt9Ab6dC4
```

```
hfc9dV5arkvc/OCmrw==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/cookie-signature": {
      "version": "1.0.6",
      "resolved": "https://registry.npmjs.org/cookie-signature/-
/cookie-signature-1.0.6.tgz",
      "integrity": "sha512-
QADzlaHc8icV8I7vbaJXJwod9HWYp8uCqf1xa4OfNu1T7JVxQIrUgOWtHdNDtPiywmFb
iS12VjotIXLrKM3orQ==""
    },
    "node_modules/core-util-is": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/core-util-is/-/core-
util-is-1.0.3.tgz",
      "integrity": "sha512-
ZQBvi1DcpJ4GDqanjucZ2Hj3wEO5pZDS89BWbkcrvdxksJorwUDDZamX9ldFkp9aw2lm
BDLgkObEA4DWNJ9FYQ==""
    },
    "node_modules/csv": {
      "version": "6.3.0",
      "resolved": "https://registry.npmjs.org/csv/-/csv-6.3.0.tgz",
      "integrity": "sha512-
wXakaMNIz6qe/a15AXQ6JA9urN2lEx08J/3xpxv7Ke3GOrio300Ikbqzrlg1ML0fuEOU
LHhswvSxtN5h/72sHg==",
      "dependencies": {
        "csv-generate": "^4.2.6",
        "csv-parse": "^5.3.10",
        "csv-stringify": "^6.4.0",
        "stream-transform": "^3.2.6"
      },
      "engines": {
        "node": ">= 0.1.90"
      }
    },
    "node_modules/csv-generate": {
      "version": "4.2.6",
      "resolved": "https://registry.npmjs.org/csv-generate/-/csv-
generate-4.2.6.tgz",
      "integrity": "sha512-
VtnYqhWLcsUocA346ewFOk+rrqcoT663j11vXzD2uelXq9WguQ3QzDeVD8ISso7hhVtk
DSHcWl9psdemeiEHDA==""
    },
    "node_modules/csv-parse": {
      "version": "5.3.10",
      "resolved": "https://registry.npmjs.org/csv-parse/-/csv-parse-
5.3.10.tgz",
      "integrity": "sha512-
cTXY6iy0gN5Ha/cGILeDgQE+nKiKDU2m0DjSRdJhr86BN3cM7oefBsTk2aH0LQeaYtL7
Z7HvW+jYoadmdhzeDA==""
    },
    "node_modules/csv-stringify": {
      "version": "6.4.0",
      "resolved": "https://registry.npmjs.org/csv-stringify/-/csv-
stringify-6.4.0.tgz",
      "integrity": "sha512-
HQsw0QXiN5fdlO+R8/JzCZnR3Fqp8E87YVnhHlaPtNGJjt6ffbV0LpOkieIb1x6V1+xt
```

```
878IYq77SpXHWAqKkA=="
    },
    "node_modules/csvtojson": {
      "version": "2.0.10",
      "resolved": "https://registry.npmjs.org/csvtojson/-/csvtojson-
2.0.10.tgz",
      "integrity": "sha512-
lUWFxGKyhraKCW8Qghz6Z0f2l/PqB1W3AO0HKJzGIQ5JRSlR651ekJDiGJbBT4sRNNv5
ddnSGVEnsxP9XRCVpQ==",
      "dependencies": {
        "bluebird": "^3.5.1",
        "lodash": "^4.17.3",
        "strip-bom": "^2.0.0"
      },
      "bin": {
        "csvtojson": "bin/csvtojson"
      },
      "engines": {
        "node": ">=4.0.0"
      }
    },
    "node_modules/debug": {
      "version": "2.6.9",
      "resolved": "https://registry.npmjs.org/debug/-/debug-
2.6.9.tgz",
      "integrity": "sha512-
bC7ElrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7A
dgFBVeAPVMNcKGsHMA==",
      "dependencies": {
        "ms": "2.0.0"
      }
    },
    "node_modules/depd": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/depd/-/depd-
2.0.0.tgz",
      "integrity": "sha512-
g7nH6P6dyDioJogAAGprGpCtVImJhpPk/roCzdb3fIh61/s/nPsfR6onyMwkCAR/OlC3
yBC0lESvUoQEAssIrw==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/destroy": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/destroy/-/destroy-
1.2.0.tgz",
      "integrity": "sha512-
2sJGJTaXIIaR1w4iJSNoN0hnMY7Gpc/n8D4qSCJw8QqFWXf7cuAgnEHxBpweaVcPevC2
l3KpjYCx3NypQQgaJg==",
      "engines": {
        "node": ">= 0.8",
        "npm": "1.2.8000 || >= 1.4.16"
      }
    },
    "node_modules/dezalgo": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/dezalgo/-/dezalgo-
1.0.4.tgz",
```

```
      "integrity": "sha512-
rXSP0bf+5n0Qonsb+SVVfNfIsimO4HEtmnIpPHY8Q1UCzKlQrDMfdobr8nJOOsRgWCyM
RqeSBQzmWUMq7zvVig==",
      "dependencies": {
        "asap": "^2.0.0",
        "wrappy": "1"
      }
    },
    "node_modules/ee-first": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-
1.1.1.tgz",
      "integrity": "sha512-
WMwm9LhRUo+WUaRN+vRuETqG89IgZphVSNkdFgeb6sS/E4OrDIN7t48CAewSHXc6C8le
fD8KKfr5vY61brQlow=="
    },
    "node_modules/ejs": {
      "version": "3.1.8",
      "resolved": "https://registry.npmjs.org/ejs/-/ejs-3.1.8.tgz",
      "integrity": "sha512-
/sXZeMlhS0ArkfX2Aw780gJzXSMPnKjtspYZv+f3NiKLlubezAHDU5+9xz6gd3/NhG3t
xQCo6xlglmTS+oTGEQ==",
      "dependencies": {
        "jake": "^10.8.5"
      },
      "bin": {
        "ejs": "bin/cli.js"
      },
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/emoji-regex": {
      "version": "8.0.0",
      "resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-
regex-8.0.0.tgz",
      "integrity": "sha512-
MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8E
u3TdM14SawnVUmGE6A=="
    },
    "node_modules/encodeurl": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-
1.0.2.tgz",
      "integrity": "sha512-
TPJXq8JqFaVYm2CWmPvnP2Iyo4ZSM7/QKcSmuMLDObfpH5fi7RUGmd/rTDf+rut/saiD
iQEeVTNgAmJEdAOx0w==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/escalade": {
      "version": "3.1.1",
      "resolved": "https://registry.npmjs.org/escalade/-/escalade-
3.1.1.tgz",
      "integrity": "sha512-
k0er2gUkLf8O0zKJiAhmkTnJlTvINGv7ygDNPbeIsX/TJjGJZHuh9B2UxbsaEkmlEo9M
fhrSzmhIlhRlI2GXnw==",
      "engines": {
```

```
        "node": ">=6"
      }
    },
    "node_modules/escape-html": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/escape-html/-/escape-
html-1.0.3.tgz",
      "integrity": "sha512-
NiSupZ4OeuGwr68lGIeym/ksIZMJodUGOSCZ/FSnTxcrekbvqrgdUxlJOMpijaKZVjAJ
rWrGs/6Jy8OMuyj9ow==",
    },
    "node_modules/etag": {
      "version": "1.8.1",
      "resolved": "https://registry.npmjs.org/etag/-/etag-
1.8.1.tgz",
      "integrity": "sha512-
aIL5Fx7mawVa300al2BnEE4iNvo1qETxLrPI/o05L7z6go7fCw1J6EQmbK4FmJ2AS7kg
VF/KEZWufBfdClMcPg==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/express": {
      "version": "4.18.2",
      "resolved": "https://registry.npmjs.org/express/-/express-
4.18.2.tgz",
      "integrity": "sha512-
5/PsL6iGPdfQ/lKM1UuielYgv3BUoJfz1aUwU9vHZ+J7gyvwdQXFEBIEIaxeGf0GIcre
ATNyBExtalisDbuMqQ==",
      "dependencies": {
        "accepts": "~1.3.8",
        "array-flatten": "1.1.1",
        "body-parser": "1.20.1",
        "content-disposition": "0.5.4",
        "content-type": "~1.0.4",
        "cookie": "0.5.0",
        "cookie-signature": "1.0.6",
        "debug": "2.6.9",
        "depd": "2.0.0",
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "etag": "~1.8.1",
        "finalhandler": "1.2.0",
        "fresh": "0.5.2",
        "http-errors": "2.0.0",
        "merge-descriptors": "1.0.1",
        "methods": "~1.1.2",
        "on-finished": "2.4.1",
        "parseurl": "~1.3.3",
        "path-to-regexp": "0.1.7",
        "proxy-addr": "~2.0.7",
        "qs": "6.11.0",
        "range-parser": "~1.2.1",
        "safe-buffer": "5.2.1",
        "send": "0.18.0",
        "serve-static": "1.15.0",
        "setprototypeof": "1.2.0",
        "statuses": "2.0.1",
        "type-is": "~1.6.18",
```

```
        "utils-merge": "1.0.1",
        "vary": "~1.1.2"
      },
      "engines": {
        "node": ">= 0.10.0"
      }
    },
    "node_modules/filelist": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/filelist/-/filelist-
1.0.4.tgz",
      "integrity": "sha512-
w1cEuf3S+DrLCQL7ET6kz+gmlJdbq9J7yXCSjK/OZCPA+qEN1WyF4ZAf0YYJa4/shHJr
a2t/d/r8SV4Ji+x+8Q==",
      "dependencies": {
        "minimatch": "^5.0.1"
      }
    },
    "node_modules/filelist/node_modules/brace-expansion": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/brace-expansion/-
/brace-expansion-2.0.1.tgz",
      "integrity": "sha512-
XnAIvQ8eM+kC6aULx6wuQiwVsnzsi9d3WxzV3FpWTGA19F621kwdbsAcFKXgKUHZWsy+
mY6iL1sHTxWEFCytDA==",
      "dependencies": {
        "balanced-match": "^1.0.0"
      }
    },
    "node_modules/filelist/node_modules/minimatch": {
      "version": "5.1.6",
      "resolved": "https://registry.npmjs.org/minimatch/-/minimatch-
5.1.6.tgz",
      "integrity": "sha512-
lKwV/1brpG6mBUFHtb7NUmtABCb2WZZmm2wNiOA5hAb8VdCS4B3dtMWyvcoViccwAW/C
OERjXLt0zP1zXUN26g==",
      "dependencies": {
        "brace-expansion": "^2.0.1"
      },
      "engines": {
        "node": ">=10"
      }
    },
    "node_modules/fill-range": {
      "version": "7.0.1",
      "resolved": "https://registry.npmjs.org/fill-range/-/fill-
range-7.0.1.tgz",
      "integrity": "sha512-
qOo9F+dMUmC2Lcb4BbVvnKJxTPjCm+RRpe4gDuGrzkL7mEVl/djYSu2OdQ2Pa302N4oq
kSg9ir6jaLWJ2USVpQ==",
      "dependencies": {
        "to-regex-range": "^5.0.1"
      },
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/finalhandler": {
      "version": "1.2.0",
```

```
      "resolved": "https://registry.npmjs.org/finalhandler/-
/finalhandler-1.2.0.tgz",
      "integrity": "sha512-
5uXcUVftlQMFnWC9qu/svkWv3GTd2PfUhK/3PLkYNAe7FbqJMt3515HaxE6eRL74Gdsr
iiwujiawdaB1BpEISg==",
      "dependencies": {
        "debug": "2.6.9",
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "on-finished": "2.4.1",
        "parseurl": "~1.3.3",
        "statuses": "2.0.1",
        "unpipe": "~1.0.0"
      },
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/follow-redirects": {
      "version": "1.15.2",
      "resolved": "https://registry.npmjs.org/follow-redirects/-
/follow-redirects-1.15.2.tgz",
      "integrity": "sha512-
VQLG33o04KaQ8uYi2tVNbdrWp1QWxNNea+nmIB4EVM28v0hmP17z7aG1+wAkNzVq4KeX
Tq3221ye5qTJP91JwA==",
      "funding": [
        {
          "type": "individual",
          "url": "https://github.com/sponsors/RubenVerborgh"
        }
      ],
      "engines": {
        "node": ">=4.0"
      },
      "peerDependenciesMeta": {
        "debug": {
          "optional": true
        }
      }
    },
    "node_modules/formidable": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/formidable/-
/formidable-2.1.1.tgz",
      "integrity": "sha512-
0EcS9wCFEzLvfiks7omJ+SiYJAiD+TzK4Pcw1UlUoGnhUxDcMKjt0P7x8wEb0u6OHu8N
b98WG3nxtlF5C7bvUQ==",
      "dependencies": {
        "dezalgo": "^1.0.4",
        "hexoid": "^1.0.0",
        "once": "^1.4.0",
        "qs": "^6.11.0"
      },
      "funding": {
        "url": "https://ko-fi.com/tunnckoCore/commissions"
      }
    },
    "node_modules/forwarded": {
      "version": "0.2.0",
```

```
      "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-
0.2.0.tgz",
      "integrity": "sha512-
buRG0fpBtRHSTCOASe6hD258tEubFoRLb4ZNA6NxMVHNw2gOcwHo9wyablzMzOA5z9xA
9L1KNjk/Nt6MT9aYow==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/fresh": {
      "version": "0.5.2",
      "resolved": "https://registry.npmjs.org/fresh/-/fresh-
0.5.2.tgz",
      "integrity": "sha512-
zJ2mQYM18rEFOudeV4GShTGIQ7RbzA7ozbU9I/XBpm7kqgMywgmylMwXHxZJmkVoYkna
9d2pVXVXPdYTP9ej8Q==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/fsevents": {
      "version": "2.3.2",
      "resolved": "https://registry.npmjs.org/fsevents/-/fsevents-
2.3.2.tgz",
      "integrity": "sha512-
xiqMQR4xAeHTuB9uWm+fFRcIOgKBMiOBP+eXiyT7jsgVCq1bkVygt00oASowB7EdtpOH
aaPgKt812P9ab+DDKA==",
      "hasInstallScript": true,
      "optional": true,
      "os": [
        "darwin"
      ],
      "engines": {
        "node": "^8.16.0 || ^10.6.0 || >=11.0.0"
      }
    },
    "node_modules/function-bind": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/function-bind/-
/function-bind-1.1.1.tgz",
      "integrity": "sha512-
yIovAzMX49sF8Yl58fSCWJ5svSLuaibPxXQJFLmBObTuCr0Mf1KiPopGM9NiFjiYBCbf
aa2Fh6breQ6ANVTI0A=="
    },
    "node_modules/get-caller-file": {
      "version": "2.0.5",
      "resolved": "https://registry.npmjs.org/get-caller-file/-/get-
caller-file-2.0.5.tgz",
      "integrity": "sha512-
DyFP3BM/3YHTQOCUL/w0OZHR0lpKeGrxotcHWcqNEdnltqFwXVfhEBQ94eIo34AfQpo0
rGki4cyIiftY06h2Fg==",
      "engines": {
        "node": "6.* || 8.* || >= 10.*"
      }
    },
    "node_modules/get-intrinsic": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/get-intrinsic/-/get-
intrinsic-1.2.0.tgz",
```

```
      "integrity": "sha512-
L049y6nFOuom5wGyRc3/gdTLO94dySVKRACj1RmJZBQXlbTMhtNIgkWkUHq+jYmZvKf1
4EW1EoJnnjbmoHij0Q==",
      "dependencies": {
        "function-bind": "^1.1.1",
        "has": "^1.0.3",
        "has-symbols": "^1.0.3"
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/glob-parent": {
      "version": "5.1.2",
      "resolved": "https://registry.npmjs.org/glob-parent/-/glob-
parent-5.1.2.tgz",
      "integrity": "sha512-
AOIgSQCepiJYwP3ARnGx+5VnTu2HBYdzbGP45eLw1vr3zB3vZLeyed1sC9hnbcOc9/Sr
MyM5RPQrkGz4aS9Zow==",
      "dependencies": {
        "is-glob": "^4.0.1"
      },
      "engines": {
        "node": ">= 6"
      }
    },
    "node_modules/has": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/has/-/has-1.0.3.tgz",
      "integrity": "sha512-
f2dvO0VU6Oej7RkWJGrehjbzMAjFp5/VKPp5tTpWIV4JHHZK1/BxbFRtf/siA2SWTe09
caDmVtYYzWEIbBS4zw==",
      "dependencies": {
        "function-bind": "^1.1.1"
      },
      "engines": {
        "node": ">= 0.4.0"
      }
    },
    "node_modules/has-flag": {
      "version": "4.0.0",
      "resolved": "https://registry.npmjs.org/has-flag/-/has-flag-
4.0.0.tgz",
      "integrity": "sha512-
EykJT/Q1KjTWctppgIAgfSO0tKVuZUjhgMr17kqTumMl6Afv3EISleU7qZUzoXDFTAHT
DC4NOoG/ZxU3EvlMPQ==",
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/has-symbols": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/has-symbols/-/has-
symbols-1.0.3.tgz",
      "integrity": "sha512-
l3LCuF6MgDNwTDKkdYGEihYjt5pRPbEg46rtlmnSPlUbgmB8LOIrKJbYYFBSbnPaJexM
KtiPO8hmeRjRz2Td+A==",
      "engines": {
        "node": ">= 0.4"
```

```
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/hexoid": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/hexoid/-/hexoid-
1.0.0.tgz",
      "integrity": "sha512-
QFLV0taWQOZtvIRIAdBChesmogZrtuXvVWsFHZTk2SU+anspqZ2vMnoLg7IE1+Uk16N1
9APic1BuF8bC8c2m5g==",
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/http-errors": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/http-errors/-/http-
errors-2.0.0.tgz",
      "integrity": "sha512-
FtwrG/euBzaEjYeRqOgly7G0qviiXoJWnvEH2Z1plBdXgbyjv34pHTSb9zoeHMyDy33+
DWy5Wt9Wo+TURtOYSQ==",
      "dependencies": {
        "depd": "2.0.0",
        "inherits": "2.0.4",
        "setprototypeof": "1.2.0",
        "statuses": "2.0.1",
        "toidentifier": "1.0.1"
      },
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/iconv-lite": {
      "version": "0.4.24",
      "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-
lite-0.4.24.tgz",
      "integrity": "sha512-
v3MXnZAcvnywkTUEZomIActle7RXXeedOR31wwl7VlyoXO4Qi9arvSenNQWne1TcRwhC
L1HwLI21bEqdpj8/rA==",
      "dependencies": {
        "safer-buffer": ">= 2.1.2 < 3"
      },
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/ignore-by-default": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/ignore-by-default/-
/ignore-by-default-1.0.1.tgz",
      "integrity": "sha512-
Ius2VYcGNk7T90CppJqcIkS5ooHUZyIQK+ClZfMfMNFEF9VSE73Fq+906u/CWu92x4gz
ZMWOwfFYckPObzdEbA==",
    },
    "node_modules/inherits": {
      "version": "2.0.4",
      "resolved": "https://registry.npmjs.org/inherits/-/inherits-
```

```
2.0.4.tgz",
      "integrity": "sha512-
k/vGaX4/Yla3WzyMCvTQOXYeIHvqOKtnqBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIF
eMpXKxa4dYeZIQqewQ=="
    },
    "node_modules/ipaddr.js": {
      "version": "1.9.1",
      "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-
1.9.1.tgz",
      "integrity": "sha512-
0KI/607xoxSToH7GjN1FfSbLoU0+btTicjsQSWQlh/hZykN8KpmMf7uYwPW3R+akZ6R/
w18ZlXSHBYXiYUPO3g==",
      "engines": {
        "node": ">= 0.10"
      }
    },
    "node_modules/is-binary-path": {
      "version": "2.1.0",
      "resolved": "https://registry.npmjs.org/is-binary-path/-/is-
binary-path-2.1.0.tgz",
      "integrity": "sha512-
ZMERYes6pDydyuGidse7OsHxtbI7WVeUEozgR/g7rd0xUimYNlvZRE/K2MgZTjWy725I
felLeVcEM97mmtRGXw==",
      "dependencies": {
        "binary-extensions": "^2.0.0"
      },
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/is-extglob": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/is-extglob/-/is-
extglob-2.1.1.tgz",
      "integrity": "sha512-
SbKbANkN603Vi4jEZv49LeVJMn4yGwsbzZworEoyEiutsN3nJYdbO36zfhGJ6QEDpOZI
FkDtnq5JRxmvl3jsoQ==",
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/is-fullwidth-code-point": {
      "version": "3.0.0",
      "resolved": "https://registry.npmjs.org/is-fullwidth-code-
point/-/is-fullwidth-code-point-3.0.0.tgz",
      "integrity": "sha512-
zymm5+u+sCsSWyD9qNaejV3DFvhCKclKdizYaJUuHA83RLjb7nSuGnddCHGv0hk+KY7B
MAlsWeK4Ueg6EV6XQg==",
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/is-glob": {
      "version": "4.0.3",
      "resolved": "https://registry.npmjs.org/is-glob/-/is-glob-
4.0.3.tgz",
      "integrity": "sha512-
xelSayHH36ZgE7ZWhli7pW34hNbNl8Ojv5KVmkJD4hBdD3th8Tfk9vYasLM+mXWOZhFk
gZfxhLSnrwRr4elSSg==",
```

```
      "dependencies": {
        "is-extglob": "^2.1.1"
      },
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/is-number": {
      "version": "7.0.0",
      "resolved": "https://registry.npmjs.org/is-number/-/is-number-
7.0.0.tgz",
      "integrity": "sha512-
41Cifkg6e8TylSpdtTpeLVMqvSBEVzTttHvERD741+pnZ8ANv0004MRL43QKPDlK9cGv
Np6NZWZUBlbGXYxxng==",
      "engines": {
        "node": ">=0.12.0"
      }
    },
    "node_modules/is-utf8": {
      "version": "0.2.1",
      "resolved": "https://registry.npmjs.org/is-utf8/-/is-utf8-
0.2.1.tgz",
      "integrity": "sha512-
rMYPYvCzsXywIsldgLaSoPlw5PfoB/ssr7hY4pLfcodrA5M/eArza1a9VmTiNIBNMjOG
r1Ow9mTyU2o69U6U9Q=="
    },
    "node_modules/isarray": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/isarray/-/isarray-
1.0.0.tgz",
      "integrity": "sha512-
VLghIWNM6ELQzo7zwmcg0NmTVyWKYjvIeM83yjp0wRDTmUnrM678fQbcKBo6n2CJEF0s
zoG//ytg+TKla89ALQ=="
    },
    "node_modules/jake": {
      "version": "10.8.5",
      "resolved": "https://registry.npmjs.org/jake/-/jake-
10.8.5.tgz",
      "integrity": "sha512-
sVpxYeuAhWt0OTWITwT98oyV0GsXyMlXCF+3L1SuafBVUIr/uILGRB+NqwkzhgXKvoJp
DIpQvqkUALgdmQsQxw==",
      "dependencies": {
        "async": "^3.2.3",
        "chalk": "^4.0.2",
        "filelist": "^1.0.1",
        "minimatch": "^3.0.4"
      },
      "bin": {
        "jake": "bin/cli.js"
      },
      "engines": {
        "node": ">=10"
      }
    },
    "node_modules/json-to-csv-export": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/json-to-csv-export/-
/json-to-csv-export-2.1.1.tgz",
      "integrity": "sha512-
```

```
JQLYqGfaJP4dVV0zIxH6RmHOdnJ32Qch3yDxDtWQy8KsOv73SthVRo38ANXckJxl/WUf
GYvvtOlVwPURMTWqlA=="
    },
    "node_modules/json2csv": {
      "version": "6.0.0-alpha.2",
      "resolved": "https://registry.npmjs.org/json2csv/-/json2csv-
6.0.0-alpha.2.tgz",
      "integrity": "sha512-
nJ3oP6QxN8z69IT1HmrJdfVxhU1kLTBVgMfRnNZc37YEY+jZ4nU27rBGxT4vaqM/KUCa
vLRhntmTuBFqZLBUcA==",
      "dependencies": {
        "@streamparser/json": "^0.0.6",
        "commander": "^6.2.0",
        "lodash.get": "^4.4.2"
      },
      "bin": {
        "json2csv": "bin/json2csv.js"
      },
      "engines": {
        "node": ">= 12",
        "npm": ">= 6.13.0"
      }
    },
    "node_modules/localtunnel": {
      "version": "2.0.2",
      "resolved": "https://registry.npmjs.org/localtunnel/-
/localtunnel-2.0.2.tgz",
      "integrity": "sha512-
n418Cn5ynvJd7m/N1d9WVJISLJF/ellZnfsLnx8WBWGzxv/ntNcFkJ1o6se5quUhCplf
LGBNL5tYHiq5WF3Nug==",
      "dependencies": {
        "axios": "0.21.4",
        "debug": "4.3.2",
        "openurl": "1.1.1",
        "yargs": "17.1.1"
      },
      "bin": {
        "lt": "bin/lt.js"
      },
      "engines": {
        "node": ">=8.3.0"
      }
    },
    "node_modules/localtunnel/node_modules/debug": {
      "version": "4.3.2",
      "resolved": "https://registry.npmjs.org/debug/-/debug-
4.3.2.tgz",
      "integrity": "sha512-
mOp8wKcvj7XxC78zLgw/ZA+6TSgkoE2C/ienthhRD298T7UNwAg9diBpLRxC0mOezLl4
B0xV7M0cCO6P/O0Xhw==",
      "dependencies": {
        "ms": "2.1.2"
      },
      "engines": {
        "node": ">=6.0"
      },
      "peerDependenciesMeta": {
        "supports-color": {
          "optional": true
```

```
        }
      }
    },
    "node_modules/localtunnel/node_modules/ms": {
      "version": "2.1.2",
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.2.tgz",
      "integrity": "sha512-
sGkPx+VjMtmA6MX27oA4FBFELFCZZ4S4XqeGOXCv68tT+jb3vk/RyaKWP0PTKyWtmLSM
0b+adUTEvbs1PEaH2w=="
    },
    "node_modules/lodash": {
      "version": "4.17.21",
      "resolved": "https://registry.npmjs.org/lodash/-/lodash-
4.17.21.tgz",
      "integrity": "sha512-
v2kDEe57lecTulaDIuNTPy3Ry4gLGJ6Z1O3vE1krgXZNrsQ+LFTGHVxVjcXPs17LhbZV
GedAJv8XZ1tvj5FvSg=="
    },
    "node_modules/lodash.get": {
      "version": "4.4.2",
      "resolved": "https://registry.npmjs.org/lodash.get/-
/lodash.get-4.4.2.tgz",
      "integrity": "sha512-
z+Uw/vLuy6gQe8cfaFWD7p0wVv8fJl3mbzXh33RS+0oW2wvUqiRXiQ69gLWSLpgB5/6s
U+r6BlQR0MBILadqTQ=="
    },
    "node_modules/media-typer": {
      "version": "0.3.0",
      "resolved": "https://registry.npmjs.org/media-typer/-/media-
typer-0.3.0.tgz",
      "integrity": "sha512-
dq+qelQ9akHpcOl/gUVRTxVIOkAJ1wR3QAvb4RsVjS8oVoFjDGTc679wJYmUmknUF5Hw
MLOgb5O+a3KxfWapPQ==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/merge-descriptors": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/merge-descriptors/-
/merge-descriptors-1.0.1.tgz",
      "integrity": "sha512-
cCi6g3/Zr1iqQi6ySbseM1Xvooa98N0w31jzUYrXPX2xqObmFGHJ0tQ5u74H3mVh7wLo
uTseZyYIq39g8cNp1w=="
    },
    "node_modules/methods": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/methods/-/methods-
1.1.2.tgz",
      "integrity": "sha512-
iclAHeNqNm68zFtnZ0e+1L2yUIdvzNoauKU4WBA3VvH/vPFieF7qfRlwUZU+DA9P9bPX
IS90ulxoUoCH23sV2w==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/mime": {
      "version": "1.6.0",
      "resolved": "https://registry.npmjs.org/mime/-/mime-
```

```
1.6.0.tgz",
      "integrity": "sha512-
x0Vn8spI+wuJ1O6S7gnbaQg8Pxh4NNHb7KSINmEWKiPE4RKOplvijn+NkmYmmRgP68mc
70j2EbeTFRsrswaQeg==",
      "bin": {
        "mime": "cli.js"
      },
      "engines": {
        "node": ">=4"
      }
    },
    "node_modules/mime-db": {
      "version": "1.52.0",
      "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-
1.52.0.tgz",
      "integrity": "sha512-
sPU4uV7dYlvtWJxwwxHD0PuihVNiE7TyAbQ5SWxDCB9mUYvOgroQOwYQQOKPJ8CIbE+1
ETVlOoK1UC2nU3gYvg==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/mime-types": {
      "version": "2.1.35",
      "resolved": "https://registry.npmjs.org/mime-types/-/mime-
types-2.1.35.tgz",
      "integrity": "sha512-
ZDY+bPm5zTTF+YpCrAU9nK0UgICYPT0QtT1NZWFv4s++TNkcgVaT0g6+4R2uI4MjQjzy
sHB1zxuWL50hzaeXiw==",
      "dependencies": {
        "mime-db": "1.52.0"
      },
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/minimatch": {
      "version": "3.1.2",
      "resolved": "https://registry.npmjs.org/minimatch/-/minimatch-
3.1.2.tgz",
      "integrity": "sha512-
J7p63hRiAjw1NDEww1W7i37+ByIrOWO5XQQAzZ3VOcL0PNybwpfmV/N05zFAzwQ9USyE
cX6t3UO+K5aqBQOIHw==",
      "dependencies": {
        "brace-expansion": "^1.1.7"
      },
      "engines": {
        "node": "*"
      }
    },
    "node_modules/minimist": {
      "version": "1.2.8",
      "resolved": "https://registry.npmjs.org/minimist/-/minimist-
1.2.8.tgz",
      "integrity": "sha512-
2yyAR8qBkN3YuheJanUpWC5U3bb5osDywNB8RzDVlDwDHbocAJveqqj1u8+SVD7jkWT4
yvsHCpWqqWqAxb0zCA==",
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
```

```
      }
    },
    "node_modules/mkdirp": {
      "version": "0.5.6",
      "resolved": "https://registry.npmjs.org/mkdirp/-/mkdirp-
0.5.6.tgz",
      "integrity": "sha512-
FP+p8RB8OWpF3YZBCrP5gtADmtXApB5AMLn+vdyA+PyxCjrCs00mjyUozssO33cwDeT3
wNGdLxJ5M//YqtHAJw==",
      "dependencies": {
        "minimist": "^1.2.6"
      },
      "bin": {
        "mkdirp": "bin/cmd.js"
      }
    },
    "node_modules/ms": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",
      "integrity": "sha512-
Tpp60P6IUJDTuOq/5Z8cdskzJujfwqfOTkrwIwj7IRISpnkJnT6SyJ4PCPnGMoFjC9dd
hal5KVIYtAt97ix05A=="
    },
    "node_modules/multer": {
      "version": "1.4.5-lts.1",
      "resolved": "https://registry.npmjs.org/multer/-/multer-1.4.5-
lts.1.tgz",
      "integrity": "sha512-
ywPWvcDMeH+z9gQq5qYHCCy+ethsk4goepZ45GLD63fOu0YcNecQxi64nDs3qluZB+mu
rG3/D4dJ7+dGctcCQQ==",
      "dependencies": {
        "append-field": "^1.0.0",
        "busboy": "^1.0.0",
        "concat-stream": "^1.5.2",
        "mkdirp": "^0.5.4",
        "object-assign": "^4.1.1",
        "type-is": "^1.6.4",
        "xtend": "^4.0.0"
      },
      "engines": {
        "node": ">= 6.0.0"
      }
    },
    "node_modules/negotiator": {
      "version": "0.6.3",
      "resolved": "https://registry.npmjs.org/negotiator/-
/negotiator-0.6.3.tgz",
      "integrity": "sha512-
+EUsqGPLsM+j/zdChZjsnX51g4XrHFOIXwfnCVPGlQk/k5giakcKsuxCObBRu6DSm9op
w/O6slWbJdghQM4bBg==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/nodemon": {
      "version": "2.0.22",
      "resolved": "https://registry.npmjs.org/nodemon/-/nodemon-
2.0.22.tgz",
      "integrity": "sha512-
```

```
B8YqaKMmyuCO7BowF1Z1/mkPqLk6cs/163Ojtd6otKjMx47Dq1utxfRxcavH1I7VSaL8
n5BUaoutadnsX3AAVQ==",
      "dependencies": {
        "chokidar": "^3.5.2",
        "debug": "^3.2.7",
        "ignore-by-default": "^1.0.1",
        "minimatch": "^3.1.2",
        "pstree.remy": "^1.1.8",
        "semver": "^5.7.1",
        "simple-update-notifier": "^1.0.7",
        "supports-color": "^5.5.0",
        "touch": "^3.1.0",
        "undefsafe": "^2.0.5"
      },
      "bin": {
        "nodemon": "bin/nodemon.js"
      },
      "engines": {
        "node": ">=8.10.0"
      },
      "funding": {
        "type": "opencollective",
        "url": "https://opencollective.com/nodemon"
      }
    },
    "node_modules/nodemon/node_modules/debug": {
      "version": "3.2.7",
      "resolved": "https://registry.npmjs.org/debug/-/debug-
3.2.7.tgz",
      "integrity": "sha512-
CFjzYYAi4ThfiQvizrFQevTTXHtnCqWfe7x1AhgEscTz6ZbLbfoLRLPugTQyBth6f8ZE
RVUSyWHFD/7Wu4t1XQ==",
      "dependencies": {
        "ms": "^2.1.1"
      }
    },
    "node_modules/nodemon/node_modules/has-flag": {
      "version": "3.0.0",
      "resolved": "https://registry.npmjs.org/has-flag/-/has-flag-
3.0.0.tgz",
      "integrity": "sha512-
sKJf1+ceQBr4SMkvQnBDNDtf4TXpVhVGateu0t918bl30FnbE2m4vNLX+VWe/dpjlb+H
ugGYzW7uQXH98HPEYw==",
      "engines": {
        "node": ">=4"
      }
    },
    "node_modules/nodemon/node_modules/ms": {
      "version": "2.1.3",
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",
      "integrity": "sha512-
6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9x
AOtHnSO/tXtF3WRTlA=="
    },
    "node_modules/nodemon/node_modules/supports-color": {
      "version": "5.5.0",
      "resolved": "https://registry.npmjs.org/supports-color/-
/supports-color-5.5.0.tgz",
      "integrity": "sha512-
```

```
QjVjwdXIt408MIiAqCX4oUKsgU2EqAGzs2Ppkm4aQYbjm+ZEWEcW4SfFNTr4uMNZma0e
y4f5lgLrkB0aX0QMow==",
      "dependencies": {
        "has-flag": "^3.0.0"
      },
      "engines": {
        "node": ">=4"
      }
    },
    "node_modules/nopt": {
      "version": "1.0.10",
      "resolved": "https://registry.npmjs.org/nopt/-/nopt-
1.0.10.tgz",
      "integrity": "sha512-
NWmpvLSqUrgrAC9HCuxEvb+PSloHpqVu+FqcO4eeF2h5qYRhA7ev6KvelyQAKtegUbC6
RypJnlEOhd8vloNKYg==",
      "dependencies": {
        "abbrev": "1"
      },
      "bin": {
        "nopt": "bin/nopt.js"
      },
      "engines": {
        "node": "*"
      }
    },
    "node_modules/normalize-path": {
      "version": "3.0.0",
      "resolved": "https://registry.npmjs.org/normalize-path/-
/normalize-path-3.0.0.tgz",
      "integrity": "sha512-
6eZs5Ls3WtCisHWp9S2GUy8dqkpGi4BVSz3GaqiE6ezub0512ESztXUwUB6C6IKbQkY2
Pnb/mD4WYojCRwcwLA==",
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/object-assign": {
      "version": "4.1.1",
      "resolved": "https://registry.npmjs.org/object-assign/-
/object-assign-4.1.1.tgz",
      "integrity": "sha512-
rJgTQnkUnH1sFw8yT6VSU3zD3sWmu6sZhIseY8VX+GRu3P6F7Fu+JNDoXfklElbLJSnc
3FUQHVe4cU5hj+BcUg==",
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/object-inspect": {
      "version": "1.12.3",
      "resolved": "https://registry.npmjs.org/object-inspect/-
/object-inspect-1.12.3.tgz",
      "integrity": "sha512-
geUvdk7c+eizMNUDkRpW1wJwgfOiOeHbxBR/hLXK1aT6zmVSO0jsQcs7fj6MGw89jC/c
jGfLcNOrtMYtGqm81g==",
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
```

```
    "node_modules/on-finished": {
      "version": "2.4.1",
      "resolved": "https://registry.npmjs.org/on-finished/-/on-
finished-2.4.1.tgz",
      "integrity": "sha512-
oVlzkg3ENAhCk2zdv7IJwd/QUD4z2RxRwpkcGY8psCVcCYZNq4wYnVWALHM+brtuJjeP
WiYF/ClmuDr8Ch5+kg==",
      "dependencies": {
        "ee-first": "1.1.1"
      },
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/once": {
      "version": "1.4.0",
      "resolved": "https://registry.npmjs.org/once/-/once-
1.4.0.tgz",
      "integrity": "sha512-
lNaJgI+2Q5URQBkccEKHTQOPaXdUxnZZElQTZY0MFUAuaEqe1E+Nyvgdz/aIyNi6Z9Mz
O5dv1H8n58/GELp3+w==",
      "dependencies": {
        "wrappy": "1"
      }
    },
    "node_modules/openurl": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/openurl/-/openurl-
1.1.1.tgz",
      "integrity": "sha512-
d/gTkTb1i1GKz5k3XE3XFV/PxQ1k45zDqGP2OA7YhgsaLoqm6qRvARAZOFer1fcXritW
lGBRCu/UgeS4HAnXAA=="
    },
    "node_modules/papaparse": {
      "version": "5.4.1",
      "resolved": "https://registry.npmjs.org/papaparse/-/papaparse-
5.4.1.tgz",
      "integrity": "sha512-
HipMsgJkZu8br23pW15uvo6sib6wne/4woLZPlFf3rpDyMe9ywEXUsuD7+6K9PRkJlVT
51j/sCOYDKGGS3ZJrw=="
    },
    "node_modules/parseurl": {
      "version": "1.3.3",
      "resolved": "https://registry.npmjs.org/parseurl/-/parseurl-
1.3.3.tgz",
      "integrity": "sha512-
CiyeOxFT/JZyN5m0z9PfXw4SCBJ6Sygz1Dpl0wqjlhDEGGBP1GnsUVEL0p63hoG1fcj3
fHynXi9NYO4nWOL+qQ==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/path-to-regexp": {
      "version": "0.1.7",
      "resolved": "https://registry.npmjs.org/path-to-regexp/-/path-
to-regexp-0.1.7.tgz",
      "integrity": "sha512-
5DFkuoqlv1uYQKxy8omFBeJPQcdoE07Kv2sferDCrAq1ohOU+MSDswDIbnx3YAM60qIO
nYa53wBhXW0EbMonrQ==",
```

```
    },
    "node_modules/picomatch": {
      "version": "2.3.1",
      "resolved": "https://registry.npmjs.org/picomatch/-/picomatch-
2.3.1.tgz",
      "integrity": "sha512-
JU3teHTNjmE2VCGFzuY8EXzCDVwEqB2a8fsIvwaStHhAWJEeVd1o1QD80CU6+ZdEXXSL
bSsuLwJjkCBWqRQUVA==",
      "engines": {
        "node": ">=8.6"
      },
      "funding": {
        "url": "https://github.com/sponsors/jonschlinkert"
      }
    },
    "node_modules/process-nextick-args": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/process-nextick-args/-
/process-nextick-args-2.0.1.tgz",
      "integrity": "sha512-
3ouUOpQhtgrbOa17J7+uxOTpITYWaGP7/AhoR3+A+/1e9skrzelGi/dXzEYyvbxubEF6
Wn2ypscTKiKJFFn1ag=="
    },
    "node_modules/proxy-addr": {
      "version": "2.0.7",
      "resolved": "https://registry.npmjs.org/proxy-addr/-/proxy-
addr-2.0.7.tgz",
      "integrity": "sha512-
llQsMLSUDUPT44jdrU/O37qlnifitDP+ZwrmmZcoSKyLKvtZxpyV0n2/bD/N4tBAAZ/g
JEdZU7KMraoK1+XYAg==",
      "dependencies": {
        "forwarded": "0.2.0",
        "ipaddr.js": "1.9.1"
      },
      "engines": {
        "node": ">= 0.10"
      }
    },
    "node_modules/pstree.remy": {
      "version": "1.1.8",
      "resolved": "https://registry.npmjs.org/pstree.remy/-
/pstree.remy-1.1.8.tgz",
      "integrity": "sha512-
77DZwxQmxKnu3aR542U+X8FypNzbfJ+C5XQDk3uWjWxn6151aIMGthWYRXTqT1E5oJvg
+ljaa2OJi+VfvCOQ8w=="
    },
    "node_modules/python-shell": {
      "version": "5.0.0",
      "resolved": "https://registry.npmjs.org/python-shell/-/python-
shell-5.0.0.tgz",
      "integrity": "sha512-
RUOOOjHLhgR1MIQrCtnEqz/HJ1RMZBIN+REnpSUrfft2bXqXy69fwJASVziWExfFXsR1
bCY0TznnHooNsCo0/w==",
      "engines": {
        "node": ">=0.10"
      }
    },
    "node_modules/qs": {
      "version": "6.11.0",
```

```
      "resolved": "https://registry.npmjs.org/qs/-/qs-6.11.0.tgz",
      "integrity": "sha512-
MvjoMCJwEarSbUYk5O+nmoSzSutSsTwF85zcHPQ9OrlFoZOYIjaqBAJIqIXjptyD5vTh
xGq52Xu/MaJzRkIk4Q==",
      "dependencies": {
        "side-channel": "^1.0.4"
      },
      "engines": {
        "node": ">=0.6"
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/range-parser": {
      "version": "1.2.1",
      "resolved": "https://registry.npmjs.org/range-parser/-/range-
parser-1.2.1.tgz",
      "integrity": "sha512-
Hrgsx+orqoygnmhFbKaHE6c296J+HTAQXoxEF6gNupROmmGJRoyzfG3ccAveqCBrwr/2
yxQ5BVd/GTl5agOwSg==",
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/raw-body": {
      "version": "2.5.1",
      "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-
2.5.1.tgz",
      "integrity": "sha512-
qqJBtEyVgS0ZmPGdCFPWJ3FreoqvG4MVQln/kCgF7Olq95IbOp0/BWyMwbdtn4VTvkM8
Y7khCQ2Xgk/tcrCXig==",
      "dependencies": {
        "bytes": "3.1.2",
        "http-errors": "2.0.0",
        "iconv-lite": "0.4.24",
        "unpipe": "1.0.0"
      },
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/readable-stream": {
      "version": "2.3.8",
      "resolved": "https://registry.npmjs.org/readable-stream/-
/readable-stream-2.3.8.tgz",
      "integrity": "sha512-
8p0AUk4XODgIewSi0l8Epjs+EVnWiK7NoDIEGU0HhE7+ZyY8D1IMY7odu5lRrFXGg71L
15KG8QrPmum45RTtdA==",
      "dependencies": {
        "core-util-is": "~1.0.0",
        "inherits": "~2.0.3",
        "isarray": "~1.0.0",
        "process-nextick-args": "~2.0.0",
        "safe-buffer": "~5.1.1",
        "string_decoder": "~1.1.1",
        "util-deprecate": "~1.0.1"
      }
    },
```

```
    "node_modules/readable-stream/node_modules/safe-buffer": {
      "version": "5.1.2",
      "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-
buffer-5.1.2.tgz",
      "integrity": "sha512-
Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOI
rt8ud/wPtojys4G6+g=="
    },
    "node_modules/readdirp": {
      "version": "3.6.0",
      "resolved": "https://registry.npmjs.org/readdirp/-/readdirp-
3.6.0.tgz",
      "integrity": "sha512-
hOS089on8RduqdbhvQ5Z37A0ESjsqz6qnRcffsMU3495FuTdqSm+7bhJ29JvIOsBDEEn
an5DPu9t3To9VRlMzA==",
      "dependencies": {
        "picomatch": "^2.2.1"
      },
      "engines": {
        "node": ">=8.10.0"
      }
    },
    "node_modules/require-directory": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/require-directory/-
/require-directory-2.1.1.tgz",
      "integrity": "sha512-
fGxEI7+wsG9xrvdjsrlmL22OMTTiHRwAMroiEeMgq8gzoLC/PQr7RsRDSTLUg/bZAZtF
+TVIkHc6/4RIKrui+Q==",
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/safe-buffer": {
      "version": "5.2.1",
      "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-
buffer-5.2.1.tgz",
      "integrity": "sha512-
rp3So07KcdmmKbGvgaNxQSJr7bGVSVk5S9Eq1F+ppbRo70+YeaDxkw5Dd8NPN+GD6bjn
Ym2VuPuCXmpuYvmCXQ==",
      "funding": [
        {
          "type": "github",
          "url": "https://github.com/sponsors/feross"
        },
        {
          "type": "patreon",
          "url": "https://www.patreon.com/feross"
        },
        {
          "type": "consulting",
          "url": "https://feross.org/support"
        }
      ]
    },
    "node_modules/safer-buffer": {
      "version": "2.1.2",
      "resolved": "https://registry.npmjs.org/safer-buffer/-/safer-
buffer-2.1.2.tgz",
```

```
      "integrity": "sha512-
YZo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXXevb5dJI7tpyN2ADxGcQbHG7
vcyRHk0cbwqcQriUtg=="
    },
    "node_modules/semver": {
      "version": "5.7.1",
      "resolved": "https://registry.npmjs.org/semver/-/semver-
5.7.1.tgz",
      "integrity": "sha512-
sauaDf/PZdVgrLTNYHRtpXa1iRiKcaebiKQ1BJdpQlWH2lCvexQdX55snPFyK7Qzpudq
bCI0qXFfOasHdyNDGQ==",
      "bin": {
        "semver": "bin/semver"
      }
    },
    "node_modules/send": {
      "version": "0.18.0",
      "resolved": "https://registry.npmjs.org/send/-/send-
0.18.0.tgz",
      "integrity": "sha512-
qqWzuOjSFOuqPjFe4NOsMLafToQQwBSOEpS+FwEt3A2V3vKubTquT3vmLTQpFgMXp8Al
FWFuP1qKaJZOtPpVXg==",
      "dependencies": {
        "debug": "2.6.9",
        "depd": "2.0.0",
        "destroy": "1.2.0",
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "etag": "~1.8.1",
        "fresh": "0.5.2",
        "http-errors": "2.0.0",
        "mime": "1.6.0",
        "ms": "2.1.3",
        "on-finished": "2.4.1",
        "range-parser": "~1.2.1",
        "statuses": "2.0.1"
      },
      "engines": {
        "node": ">= 0.8.0"
      }
    },
    "node_modules/send/node_modules/ms": {
      "version": "2.1.3",
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",
      "integrity": "sha512-
6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9x
AOtHnSO/tXtF3WRTlA==",
    },
    "node_modules/serve-static": {
      "version": "1.15.0",
      "resolved": "https://registry.npmjs.org/serve-static/-/serve-
static-1.15.0.tgz",
      "integrity": "sha512-
XGuRDNjXUijsUL0vl6nSD7cwURuzEgglbOaFuZM9g3kwDXOWVTck0jLzjPzGD+TazWbb
oZYu52/9/XPdUgne9g==",
      "dependencies": {
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "parseurl": "~1.3.3",
```

```
        "send": "0.18.0"
      },
      "engines": {
        "node": ">= 0.8.0"
      }
    },
    "node_modules/setprototypeof": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/setprototypeof/-
/setprototypeof-1.2.0.tgz",
      "integrity": "sha512-
E5LDX7Wrp85Kil5bhZv46j8jOeboKq5JMmYM3gVGdGH8xFpPWXUMsNrlODCrkoxMEeNi
/XZIwuRvY4XNwYMJpw==",
    },
    "node_modules/side-channel": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/side-channel/-/side-
channel-1.0.4.tgz",
      "integrity": "sha512-
q5XPytqFEIKHkGdiMIrY10mvLRvnQh42/+GoBlFW3b2LXLE2xxJpZFdm94we0BaoV3Rw
JyGqg5wS7epxTv0Zvw==",
      "dependencies": {
        "call-bind": "^1.0.0",
        "get-intrinsic": "^1.0.2",
        "object-inspect": "^1.9.0"
      },
      "funding": {
        "url": "https://github.com/sponsors/ljharb"
      }
    },
    "node_modules/simple-update-notifier": {
      "version": "1.1.0",
      "resolved": "https://registry.npmjs.org/simple-update-
notifier/-/simple-update-notifier-1.1.0.tgz",
      "integrity": "sha512-
VpsrsJSUcJEseSbMHkrsrAVSdvVS5I96Qo1QAQ4FxQ9wXFcB+pjj7FB7/us9+GcgfW4z
iHtYMc1J0PLczb55mg==",
      "dependencies": {
        "semver": "~7.0.0"
      },
      "engines": {
        "node": ">=8.10.0"
      }
    },
    "node_modules/simple-update-notifier/node_modules/semver": {
      "version": "7.0.0",
      "resolved": "https://registry.npmjs.org/semver/-/semver-
7.0.0.tgz",
      "integrity": "sha512-
+GB6zVA9LWh6zovYQLALHwv5rb2PHGlJi3lfiqIHxR0uuwCgefcOJc59v9fv1w8GbStw
xuuqqAjI9NMAOOgq1A==",
      "bin": {
        "semver": "bin/semver.js"
      }
    },
    "node_modules/statuses": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/statuses/-/statuses-
2.0.1.tgz",
```

```
      "integrity": "sha512-
RwNA9Z/7PrK06rYLIzFMlaF+l73iwpzsqRIFgbMLbTcLD6cOao82TaWefPXQvB2fOC4A
juYSEndS7N/mTCbkdQ==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/stream-transform": {
      "version": "3.2.6",
      "resolved": "https://registry.npmjs.org/stream-transform/-
/stream-transform-3.2.6.tgz",
      "integrity": "sha512-
/pyOvaCQFqYTmrFhmMbnAEVo3SsTx1H39eUVPOtYeAgbEUc+rDo7GoP8LbHJgU83mKtz
Je/7Nq/ipaAnUOHgJQ=="
    },
    "node_modules/streamsearch": {
      "version": "1.1.0",
      "resolved": "https://registry.npmjs.org/streamsearch/-
/streamsearch-1.1.0.tgz",
      "integrity": "sha512-
Mcc5wHehp9aXz1ax6bZUyY5afg9u2rv5cqQI3mRrYkGC8rW2hM02jWuwjtL++LS5qinS
yhj2QfLyNsuc+VsExg==",
      "engines": {
        "node": ">=10.0.0"
      }
    },
    "node_modules/string_decoder": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/string_decoder/-
/string_decoder-1.1.1.tgz",
      "integrity": "sha512-
n/ShnvDi6FHbbVfviro+WojiFzv+s8MPMHBczVePfUpDJLwoLT0ht1l4YwBCbi8pJAve
EEdnkHyPyTP/mzRfwg==",
      "dependencies": {
        "safe-buffer": "~5.1.0"
      }
    },
    "node_modules/string_decoder/node_modules/safe-buffer": {
      "version": "5.1.2",
      "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-
buffer-5.1.2.tgz",
      "integrity": "sha512-
Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOI
rt8ud/wPtojys4G6+g=="
    },
    "node_modules/string-width": {
      "version": "4.2.3",
      "resolved": "https://registry.npmjs.org/string-width/-/string-
width-4.2.3.tgz",
      "integrity": "sha512-
wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/c
xKTWyWUwvSTriiZz/g==",
      "dependencies": {
        "emoji-regex": "^8.0.0",
        "is-fullwidth-code-point": "^3.0.0",
        "strip-ansi": "^6.0.1"
      },
      "engines": {
        "node": ">=8"
```

```
      }
    },
    "node_modules/strip-ansi": {
      "version": "6.0.1",
      "resolved": "https://registry.npmjs.org/strip-ansi/-/strip-
ansi-6.0.1.tgz",
      "integrity": "sha512-
Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/
0ooI7KrPuUSztUdU5A==",
      "dependencies": {
        "ansi-regex": "^5.0.1"
      },
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/strip-bom": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/strip-bom/-/strip-bom-
2.0.0.tgz",
      "integrity": "sha512-
kwrX1y7czp1E69n2ajbG65mIo9dqvJ+8aBQXOGVxqwvNbsXdFM6Lq37dLAY3mknUwru8
CfcCbfOLL/gMo+fi3g==",
      "dependencies": {
        "is-utf8": "^0.2.0"
      },
      "engines": {
        "node": ">=0.10.0"
      }
    },
    "node_modules/supports-color": {
      "version": "7.2.0",
      "resolved": "https://registry.npmjs.org/supports-color/-
/supports-color-7.2.0.tgz",
      "integrity": "sha512-
qpCAvRl9stuOHveKsn7HncJRvv501qIacKzQlO/+Lwxc9+0q2wLyv4Dfvt80/DPn2pqO
BsJdDiogXGR9+OvwRw==",
      "dependencies": {
        "has-flag": "^4.0.0"
      },
      "engines": {
        "node": ">=8"
      }
    },
    "node_modules/to-regex-range": {
      "version": "5.0.1",
      "resolved": "https://registry.npmjs.org/to-regex-range/-/to-
regex-range-5.0.1.tgz",
      "integrity": "sha512-
65P7iz6X5yEr1cwcgvQxbbIw7Uk3gOy5dIdtZ4rDveLqhrdJP+Li/Hx6tyK0NEb+2GCy
neCMJiGqrADCSNk8sQ==",
      "dependencies": {
        "is-number": "^7.0.0"
      },
      "engines": {
        "node": ">=8.0"
      }
    },
    "node_modules/toidentifier": {
```

```
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/toidentifier/-
/toidentifier-1.0.1.tgz",
      "integrity": "sha512-
o5sSPKEkg/DIQNmH43V0/uerLrpzVedkUh8tGNvaeXpfpuwjKenlSox/2O/BTlZUtEe+
JG7s5YhEz608PlAHRA==",
      "engines": {
        "node": ">=0.6"
      }
    },
    "node_modules/touch": {
      "version": "3.1.0",
      "resolved": "https://registry.npmjs.org/touch/-/touch-
3.1.0.tgz",
      "integrity": "sha512-
WBx8Uy5TLtOSRtIq+M03/sKDrXCLHxwDcquSP2c43Le03/9serjQBIztjRz6FkJez9D/
hleyAXTBGLwwZUw9lA==",
      "dependencies": {
        "nopt": "~1.0.10"
      },
      "bin": {
        "nodetouch": "bin/nodetouch.js"
      }
    },
    "node_modules/type-is": {
      "version": "1.6.18",
      "resolved": "https://registry.npmjs.org/type-is/-/type-is-
1.6.18.tgz",
      "integrity": "sha512-
TkRKr9sUTxEH8MdfuCSP7VizJyzRNMjj2J2do2Jr3Kym598JVdEksuzPQCnlFPW4ky9Q
+iA+ma9BGm06XQBy8g==",
      "dependencies": {
        "media-typer": "0.3.0",
        "mime-types": "~2.1.24"
      },
      "engines": {
        "node": ">= 0.6"
      }
    },
    "node_modules/typedarray": {
      "version": "0.0.6",
      "resolved": "https://registry.npmjs.org/typedarray/-
/typedarray-0.0.6.tgz",
      "integrity": "sha512-
/aCDEGatGvZ2BIk+HmLf4ifCJFwvKFNb9/JeZPMulfgFracn9QFcAf5GO8B/mweUjSob
lS5In0cWhqpfs/5PQA=="
    },
    "node_modules/undefsafe": {
      "version": "2.0.5",
      "resolved": "https://registry.npmjs.org/undefsafe/-/undefsafe-
2.0.5.tgz",
      "integrity": "sha512-
WxONCrssBM8TSPRqN5EmsjVrsv4A8X12J4ArBiiayv3DyyG3ZlIg6yysuuSYdZsVz3TK
cTg2fd//Ujd4CHV1iA=="
    },
    "node_modules/unpipe": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/unpipe/-/unpipe-
1.0.0.tgz",
```

```
      "integrity": "sha512-
pjy2bYhSsufwWlKwPc+l3cN7+wuJlK6uz0YdJEOlQDbl6jo/YlPi4mb8agUkVC8BF7V8
NuzeyPNqRksA3hztKQ==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/util-deprecate": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/util-deprecate/-/util-
deprecate-1.0.2.tgz",
      "integrity": "sha512-
EPD5q1uXyFxJpCrLnCc1nHnq3gOa6DZBocAIiI2TaSCA7VCJ1UJDMagCzIkXNsUYfD1d
aK//LTEQ8xiIbrHtcw=="
    },
    "node_modules/utils-merge": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/utils-merge/-/utils-
merge-1.0.1.tgz",
      "integrity": "sha512-
pMZTvIkT1d+TFGvDOqodOclx0QWkkgi6Tdoa8gC8ffGAAqz9pzPTZWAybbsHHoED/ztM
tkv/VoYTYyShUn81hA==",
      "engines": {
        "node": ">= 0.4.0"
      }
    },
    "node_modules/vary": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/vary/-/vary-
1.1.2.tgz",
      "integrity": "sha512-
BNGbWLfd0eUPabhkXUVm0j8uuvREyTh5ovRa/dyow/BqAbZJyC+5fU+IzQOzmAKzYqYR
AISoRhdQr3eIZ/PXqg==",
      "engines": {
        "node": ">= 0.8"
      }
    },
    "node_modules/wrap-ansi": {
      "version": "7.0.0",
      "resolved": "https://registry.npmjs.org/wrap-ansi/-/wrap-ansi-
7.0.0.tgz",
      "integrity": "sha512-
YVGIj2kamLSTxw6NsZjoBxfSwsn0ycdesmc4p+Q21c5zPuZ1pl+NfxVdxPtdHvmNVOQ6
XSYG4AUtyt/Fi7D16Q==",
      "dependencies": {
        "ansi-styles": "^4.0.0",
        "string-width": "^4.1.0",
        "strip-ansi": "^6.0.0"
      },
      "engines": {
        "node": ">=10"
      },
      "funding": {
        "url": "https://github.com/chalk/wrap-ansi?sponsor=1"
      }
    },
    "node_modules/wrappy": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-
```

```
1.0.2.tgz",
      "integrity": "sha512-
l4Sp/DRseor9wL6EvV2+TuQn63dMkPjZ/sp9XkghTEbV9KlPS1xUsZ3u7/IQO4wxtcFB
4bgpQPRcR3QCvezPcQ=="
    },
    "node_modules/xtend": {
      "version": "4.0.2",
      "resolved": "https://registry.npmjs.org/xtend/-/xtend-
4.0.2.tgz",
      "integrity": "sha512-
LKYU1iAXJXUgAXn9URjiu+MWhyUXHsvfp7mcuYm9dSUKK0/CjtrUwFAxD82/mCWbtLsG
jFIad0wIsod4zrTAEQ==",
      "engines": {
        "node": ">=0.4"
      }
    },
    "node_modules/y18n": {
      "version": "5.0.8",
      "resolved": "https://registry.npmjs.org/y18n/-/y18n-
5.0.8.tgz",
      "integrity": "sha512-
0pfFzegeDWJHJIAmTLRP2DwHjdF5s7jo9tuztdQxAhINCdvS+3nGINqPd00AphqJR/0L
hANUS6/+7SCb98YOfA==",
      "engines": {
        "node": ">=10"
      }
    },
    "node_modules/yargs": {
      "version": "17.1.1",
      "resolved": "https://registry.npmjs.org/yargs/-/yargs-
17.1.1.tgz",
      "integrity": "sha512-
c2k48R0PwKIqKhPMWjeiF6y2xY/gPMUlro0sgxqXpbOIohWiLNXWslsootttv7E1e73Q
PAMQSg5FeySbVcpsPQ==",
      "dependencies": {
        "cliui": "^7.0.2",
        "escalade": "^3.1.1",
        "get-caller-file": "^2.0.5",
        "require-directory": "^2.1.1",
        "string-width": "^4.2.0",
        "y18n": "^5.0.5",
        "yargs-parser": "^20.2.2"
      },
      "engines": {
        "node": ">=12"
      }
    },
    "node_modules/yargs-parser": {
      "version": "20.2.9",
      "resolved": "https://registry.npmjs.org/yargs-parser/-/yargs-
parser-20.2.9.tgz",
      "integrity": "sha512-
y11nGElTIV+CT3Zv9t7VKl+Q3hTQoT9a1Qzezhhl6Rp21gJ/IVTW7Z3y9EWXhuUBC2Sh
nf+DX0antecpAwSP8w==",
      "engines": {
        "node": ">=10"
      }
    }
  },
```

```
  "dependencies": {
    "@streamparser/json": {
      "version": "0.0.6",
      "resolved": "https://registry.npmjs.org/@streamparser/json/-
/json-0.0.6.tgz",
      "integrity": "sha512-
vL9EVn/v+OhZ+Wcs6O4iKE9EUpwHUqHmCtNUMWjqp+6dr85+XPOSGTEsqYNq1Vn04uk9
SWlOVmx9J48ggJVT2Q=="
    },
    "abbrev": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/abbrev/-/abbrev-
1.1.1.tgz",
      "integrity": "sha512-
nne9/IiQ/hzIhY6pdDnbBtz7DjPTKrY00P/zvPSm5pOFkl6xuGrGnXn/VtTNNfNtAfZ9
/1RtehkszU9qcTii0Q=="
    },
    "accepts": {
      "version": "1.3.8",
      "resolved": "https://registry.npmjs.org/accepts/-/accepts-
1.3.8.tgz",
      "integrity": "sha512-
PYAthTa2m2VKxuvSD3DPC/Gy+U+sOA1LAuT8mkmRuvw+NACSaeXEQ+NHcVF7rONl6qca
xV3Uuemwawk+7+SJLw==",
      "requires": {
        "mime-types": "~2.1.34",
        "negotiator": "0.6.3"
      }
    },
    "ansi-regex": {
      "version": "5.0.1",
      "resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-
regex-5.0.1.tgz",
      "integrity": "sha512-
quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMW
v37qOOb9pdJlMUEKFQ=="
    },
    "ansi-styles": {
      "version": "4.3.0",
      "resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-
styles-4.3.0.tgz",
      "integrity": "sha512-
zbB9rCJAT1rbjiVDb2hqKFHNYLxgtk8NURxZ3IZwD3F6NtxbXZQCnnSi1Lkx+IDohdPl
Fp222wVALIheZJQSEg==",
      "requires": {
        "color-convert": "^2.0.1"
      }
    },
    "anymatch": {
      "version": "3.1.3",
      "resolved": "https://registry.npmjs.org/anymatch/-/anymatch-
3.1.3.tgz",
      "integrity": "sha512-
KMReFUr0B4t+D+OBkjR3KYqvocp2XaSzO55UcB6mgQMd3KbcE+mWTyvVV7D/zsdEbNnV
6acZUutkiHQXvTr1Rw==",
      "requires": {
        "normalize-path": "^3.0.0",
        "picomatch": "^2.0.4"
      }
```

```
    },
    "append-field": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/append-field/-/append-
field-1.0.0.tgz",
      "integrity": "sha512-
klpgFSWLW1ZEs8svjfb7g4qWY0YS5imI82dTg+QahUvJ8YqAY0P10Uk8tTyh9ZGuYEZE
MaeJYCF5BFuX552hsw=="
    },
    "array-flatten": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/array-flatten/-/array-
flatten-1.1.1.tgz",
      "integrity": "sha512-
PCVAQswWemu6UdxsDFFX/+gVeYqKAod3D3UVm91jHwynguOwAvYPhx8nNlM++NqRcK6C
xxpUafjmhIdKiHibqg=="
    },
    "asap": {
      "version": "2.0.6",
      "resolved": "https://registry.npmjs.org/asap/-/asap-
2.0.6.tgz",
      "integrity": "sha512-
BSHWgDSAiKs50o2Re8ppvp3seVHXSRM44cdSsT9FfNEUUZLOGWVCsiWaRPWM1Znn+mqZ
1OfVZ3z3DWEzSp7hRA=="
    },
    "async": {
      "version": "3.2.4",
      "resolved": "https://registry.npmjs.org/async/-/async-
3.2.4.tgz",
      "integrity": "sha512-
iAB+JbDEGXhyIUavoDl9WP/Jj106Kz9DEn1DPgYw5ruDn0e3Wgi3sKFm55sASdGBNOQB
8F59d9qQ7deqrHA8wQ=="
    },
    "axios": {
      "version": "0.21.4",
      "resolved": "https://registry.npmjs.org/axios/-/axios-
0.21.4.tgz",
      "integrity": "sha512-
ut5vewkiu8jjGBdqpM44XxjuCjq9LAKeHVmoVfHVzy8eHgxxq8SbAVQNovDA8mVi05kP
0Ea/n/UzcSHcTJQfNg==",
      "requires": {
        "follow-redirects": "^1.14.0"
      }
    },
    "balanced-match": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/balanced-match/-
/balanced-match-1.0.2.tgz",
      "integrity": "sha512-
3oSeUO0TMV67hN1AmbXsK4yaqU7tjiHlbxRDZOpH0KW9+CeX4bRAaX0Anxt0tx2MrpRp
WwQaPwIlISEJhYU5Pw=="
    },
    "binary-extensions": {
      "version": "2.2.0",
      "resolved": "https://registry.npmjs.org/binary-extensions/-
/binary-extensions-2.2.0.tgz",
      "integrity": "sha512-
jDctJ/IVQbZoJykoeHbhXpOlNBqGNcwXJKJog42E5HDPUwQTSdjCHdihjj0DlnheQ7bl
bT6dHOafNAiS8ooQKA=="
```

```
    },
    "bluebird": {
      "version": "3.7.2",
      "resolved": "https://registry.npmjs.org/bluebird/-/bluebird-
3.7.2.tgz",
      "integrity": "sha512-
XpNj6GDQzdfW+r2Wnn7xiSAd7TM3jzkxGXBGTtWKuSXv1xUV+azxAm8jdWZN06QTQk+2
N2XB9jRDkvbmQmcRtg=="
    },
    "body-parser": {
      "version": "1.20.1",
      "resolved": "https://registry.npmjs.org/body-parser/-/body-
parser-1.20.1.tgz",
      "integrity": "sha512-
jWi7abTbYwajOytWCQc37VulmWiRae5RyTpaCyDcS5/lMdtwSz5lOpDE67srw/HYe35f
1z3fDQw+3txg7gNtWw==",
      "requires": {
        "bytes": "3.1.2",
        "content-type": "~1.0.4",
        "debug": "2.6.9",
        "depd": "2.0.0",
        "destroy": "1.2.0",
        "http-errors": "2.0.0",
        "iconv-lite": "0.4.24",
        "on-finished": "2.4.1",
        "qs": "6.11.0",
        "raw-body": "2.5.1",
        "type-is": "~1.6.18",
        "unpipe": "1.0.0"
      }
    },
    "brace-expansion": {
      "version": "1.1.11",
      "resolved": "https://registry.npmjs.org/brace-expansion/-
/brace-expansion-1.1.11.tgz",
      "integrity": "sha512-
iCuPHDFgrHX7H2vEI/5xpz07zSHB00TpugqhmYtVmMO6518mCuRMoOYFldEBl0g187uf
ozdaHgWKcYFb61qGiA==",
      "requires": {
        "balanced-match": "^1.0.0",
        "concat-map": "0.0.1"
      }
    },
    "braces": {
      "version": "3.0.2",
      "resolved": "https://registry.npmjs.org/braces/-/braces-
3.0.2.tgz",
      "integrity": "sha512-
b8um+L1RzM3WDSzvhm6gIz1yfTbBt6YTlcEKAvsmqCZZFw46z626lVj9j1yEPW33H5H+
lBQpZMP1k8l+78Ha0A==",
      "requires": {
        "fill-range": "^7.0.1"
      }
    },
    "buffer-from": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/buffer-from/-/buffer-
from-1.1.2.tgz",
      "integrity": "sha512-
```

```
E+XQCRwSbaaiChtv6k6Dwgc+bx+Bs6vuKJHHl5kox/BaKbhiXzqQOwK4cO22yElGp2OC
mjwVhT3HmxgyPGnJfQ=="
    },
    "busboy": {
      "version": "1.6.0",
      "resolved": "https://registry.npmjs.org/busboy/-/busboy-
1.6.0.tgz",
      "integrity": "sha512-
8SFQbg/0hQ9xy3UNTB0YEnsNBbWfhf7RtnzpL7TkBiTBRfrQ9Fxcnz7VJsleJpyp6rVL
vXiuORqjlHi5q+PYuA==",
      "requires": {
        "streamsearch": "^1.1.0"
      }
    },
    "bytes": {
      "version": "3.1.2",
      "resolved": "https://registry.npmjs.org/bytes/-/bytes-
3.1.2.tgz",
      "integrity": "sha512-
/Nf7TyzTx6S3yRJObOAV7956r8cr2+Oj8AC5dt8wSP3BQAoeX58NoHyCU8P8zGkNXStj
TSi6fzO6F0pBdcYbEg=="
    },
    "call-bind": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/call-bind/-/call-bind-
1.0.2.tgz",
      "integrity": "sha512-
7O+FbCihrB5WGbFYesctwmTKae6rOiIzmz1icreWJ+0aA7LJfuqhEso2T9ncpcFtzMQt
zXf2QGGueWJGTYsqrA==",
      "requires": {
        "function-bind": "^1.1.1",
        "get-intrinsic": "^1.0.2"
      }
    },
    "chalk": {
      "version": "4.1.2",
      "resolved": "https://registry.npmjs.org/chalk/-/chalk-
4.1.2.tgz",
      "integrity": "sha512-
oKnbhFyRIXpUuez8iBMmyEa4nbj4IOQyuhc/wy9kY7/WVPcwIO9VA668Pu8RkO7+0G76
SLROeyw9CpQ061i4mA==",
      "requires": {
        "ansi-styles": "^4.1.0",
        "supports-color": "^7.1.0"
      }
    },
    "chokidar": {
      "version": "3.5.3",
      "resolved": "https://registry.npmjs.org/chokidar/-/chokidar-
3.5.3.tgz",
      "integrity": "sha512-
Dr3sfKRP6oTcjf2JmUmFJfeVMvXBdegxB0iVQ5eb2V10uFJUCAS8OByZdVAyVb8xXNz3
GjjTgj9kLWsZTqE6kw==",
      "requires": {
        "anymatch": "~3.1.2",
        "braces": "~3.0.2",
        "fsevents": "~2.3.2",
        "glob-parent": "~5.1.2",
        "is-binary-path": "~2.1.0",
```

```
        "is-glob": "~4.0.1",
        "normalize-path": "~3.0.0",
        "readdirp": "~3.6.0"
      }
    },
    "cliui": {
      "version": "7.0.4",
      "resolved": "https://registry.npmjs.org/cliui/-/cliui-
7.0.4.tgz",
      "integrity": "sha512-
OcRE68cOsVMXp1Yvonl/fzkQOyjLSu/8bhPDfQt0e0/Eb283TKP20Fs2MqoPsr9SwA59
5rRCA+QMzYc9nBP+JQ==",
      "requires": {
        "string-width": "^4.2.0",
        "strip-ansi": "^6.0.0",
        "wrap-ansi": "^7.0.0"
      }
    },
    "color-convert": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/color-convert/-/color-
convert-2.0.1.tgz",
      "integrity": "sha512-
RRECPsj7iu/xb5oKYcsFHSppFNnsj/52OVTRKb4zP5onXwVF3zVmmToNcOfGC+CRDpfK
/U584fMg38ZHCaElKQ==",
      "requires": {
        "color-name": "~1.1.4"
      }
    },
    "color-name": {
      "version": "1.1.4",
      "resolved": "https://registry.npmjs.org/color-name/-/color-
name-1.1.4.tgz",
      "integrity": "sha512-
dOy+3AuW3a2wNbZHIuMZpTcgjGuLU/uBL/ubcZF9OXbDo8ff4O8yVp5Bf0efS8uEoYo5
q4Fx7dY9OgQGXgAsQA=="
    },
    "commander": {
      "version": "6.2.1",
      "resolved": "https://registry.npmjs.org/commander/-/commander-
6.2.1.tgz",
      "integrity": "sha512-
U7VdrJFnJgo4xjrHpTzu0yrHPGImdsmD95ZlgYSEajAn2JKzDhDTPG9kBTefmObL2w/n
geZnilk+OV9CG3d7UA=="
    },
    "concat-map": {
      "version": "0.0.1",
      "resolved": "https://registry.npmjs.org/concat-map/-/concat-
map-0.0.1.tgz",
      "integrity": "sha512-
/Srv4dswyQNBfohGpz9o6Yb3Gz3SrUDqBH5rTuhGR7ahtlbYKnVxw2bCFMRljaA7EXHa
XZ8wsHdodFvbkhKmqg=="
    },
    "concat-stream": {
      "version": "1.6.2",
      "resolved": "https://registry.npmjs.org/concat-stream/-
/concat-stream-1.6.2.tgz",
      "integrity": "sha512-
27HBghJxjiZtIk3Ycvn/4kbJk/1uZuJFfuPEns6LaEvpvG1f0hTea8lilrouyo9mVc2G
```

```
WdcEZ8OLoGmSADlrCw==",
      "requires": {
        "buffer-from": "^1.0.0",
        "inherits": "^2.0.3",
        "readable-stream": "^2.2.2",
        "typedarray": "^0.0.6"
      }
    },
    "content-disposition": {
      "version": "0.5.4",
      "resolved": "https://registry.npmjs.org/content-disposition/-
/content-disposition-0.5.4.tgz",
      "integrity": "sha512-
FveZTNuGw04cxlAiWbzi6zTAL/lhehaWbTtgluJh4/E95DqMwTmha3KZN1aAWA8cFIhH
zMZUvLevkw5Rqk+tSQ==",
      "requires": {
        "safe-buffer": "5.2.1"
      }
    },
    "content-type": {
      "version": "1.0.5",
      "resolved": "https://registry.npmjs.org/content-type/-
/content-type-1.0.5.tgz",
      "integrity": "sha512-
nTjqfcBFEipKdXCv4YDQWCfmcLZKm81ldF0pAopTvyrFGVbcR6P/VAAd5G7N+0tTr8Qq
iU0tFadD6FK4NtJwOA=="
    },
    "cookie": {
      "version": "0.5.0",
      "resolved": "https://registry.npmjs.org/cookie/-/cookie-
0.5.0.tgz",
      "integrity": "sha512-
YZ3GUyn/o8gfKJlnlX7g7xq4gyO6OSuhGPKaaGssGB2qgDUS0gPgtTvoyZLTt9Ab6dC4
hfc9dV5arkvc/OCmrw=="
    },
    "cookie-signature": {
      "version": "1.0.6",
      "resolved": "https://registry.npmjs.org/cookie-signature/-
/cookie-signature-1.0.6.tgz",
      "integrity": "sha512-
QADzlaHc8icV8I7vbaJXJwod9HWYp8uCqf1xa4OfNu1T7JVxQIrUgOWtHdNDtPiywmFb
iS12VjotIXLrKM3orQ=="
    },
    "core-util-is": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/core-util-is/-/core-
util-is-1.0.3.tgz",
      "integrity": "sha512-
ZQBvi1DcpJ4GDqanjucZ2Hj3wEO5pZDS89BWbkcrvdxksJorwUDDZamX9ldFkp9aw2lm
BDLgkObEA4DWNJ9FYQ=="
    },
    "csv": {
      "version": "6.3.0",
      "resolved": "https://registry.npmjs.org/csv/-/csv-6.3.0.tgz",
      "integrity": "sha512-
wXakaMNIz6qe/a15AXQ6JA9urN2lEx08J/3xpxv7Ke3GOrio300Ikbqzrlg1ML0fuEOU
LHhswvSxtN5h/72sHg==",
      "requires": {
        "csv-generate": "^4.2.6",
```

```
        "csv-parse": "^5.3.10",
        "csv-stringify": "^6.4.0",
        "stream-transform": "^3.2.6"
      }
    },
    "csv-generate": {
      "version": "4.2.6",
      "resolved": "https://registry.npmjs.org/csv-generate/-/csv-
generate-4.2.6.tgz",
      "integrity": "sha512-
VtnYqhWLcsUocA346ewFOk+rrqcoT663j11vXzD2uelXq9WguQ3QzDeVD8ISso7hhVtk
DSHcWl9psdemeiEHDA==",
    },
    "csv-parse": {
      "version": "5.3.10",
      "resolved": "https://registry.npmjs.org/csv-parse/-/csv-parse-
5.3.10.tgz",
      "integrity": "sha512-
cTXY6iy0gN5Ha/cGILeDgQE+nKiKDU2m0DjSRdJhr86BN3cM7oefBsTk2aH0LQeaYtL7
Z7HvW+jYoadmdhzeDA==",
    },
    "csv-stringify": {
      "version": "6.4.0",
      "resolved": "https://registry.npmjs.org/csv-stringify/-/csv-
stringify-6.4.0.tgz",
      "integrity": "sha512-
HQsw0QXiN5fdlO+R8/JzCZnR3Fqp8E87YVnhHlaPtNGJjt6ffbV0LpOkieIb1x6V1+xt
878IYq77SpXHWAqKkA==",
    },
    "csvtojson": {
      "version": "2.0.10",
      "resolved": "https://registry.npmjs.org/csvtojson/-/csvtojson-
2.0.10.tgz",
      "integrity": "sha512-
lUWFxGKyhraKCW8Qghz6Z0f2l/PqB1W3AO0HKJzGIQ5JRSlR651ekJDiGJbBT4sRNNv5
ddnSGVEnsxP9XRCVpQ==",
      "requires": {
        "bluebird": "^3.5.1",
        "lodash": "^4.17.3",
        "strip-bom": "^2.0.0"
      }
    },
    "debug": {
      "version": "2.6.9",
      "resolved": "https://registry.npmjs.org/debug/-/debug-
2.6.9.tgz",
      "integrity": "sha512-
bC7ElrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7A
dgFBVeAPVMNcKGsHMA==",
      "requires": {
        "ms": "2.0.0"
      }
    },
    "depd": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/depd/-/depd-
2.0.0.tgz",
      "integrity": "sha512-
g7nH6P6dyDioJogAAGprGpCtVImJhpPk/roCzdb3fIh61/s/nPsfR6onyMwkCAR/OlC3
```

```
yBC0lESvUoQEAssIrw=="
    },
    "destroy": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/destroy/-/destroy-
1.2.0.tgz",
      "integrity": "sha512-
2sJGJTaXIIaR1w4iJSNoN0hnMY7Gpc/n8D4qSCJw8QqFWXf7cuAgnEHxBpweaVcPevC2
l3KpjYCx3NypQQgaJg==",
    },
    "dezalgo": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/dezalgo/-/dezalgo-
1.0.4.tgz",
      "integrity": "sha512-
rXSP0bf+5n0Qonsb+SVVfNfIsimO4HEtmnIpPHY8Q1UCzKlQrDMfdobr8nJOOsRgWCyM
RqeSBQzmWUMq7zvVig==",
      "requires": {
        "asap": "^2.0.0",
        "wrappy": "1"
      }
    },
    "ee-first": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-
1.1.1.tgz",
      "integrity": "sha512-
WMwm9LhRUo+WUaRN+vRuETqG89IgZphVSNkdFgeb6sS/E4OrDIN7t48CAewSHXc6C8le
fD8KKfr5vY61brQlow==",
    },
    "ejs": {
      "version": "3.1.8",
      "resolved": "https://registry.npmjs.org/ejs/-/ejs-3.1.8.tgz",
      "integrity": "sha512-
/sXZeMlhS0ArkfX2Aw780gJzXSMPnKjtspYZv+f3NiKLlubezAHDU5+9xz6gd3/NhG3t
xQCo6xlglmTS+oTGEQ==",
      "requires": {
        "jake": "^10.8.5"
      }
    },
    "emoji-regex": {
      "version": "8.0.0",
      "resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-
regex-8.0.0.tgz",
      "integrity": "sha512-
MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8E
u3TdM14SawnVUmGE6A==",
    },
    "encodeurl": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-
1.0.2.tgz",
      "integrity": "sha512-
TPJXq8JqFaVYm2CWmPvnP2Iyo4ZSM7/QKcSmuMLDObfpH5fi7RUGmd/rTDf+rut/saiD
iQEeVTNgAmJEdAOx0w==",
    },
    "escalade": {
      "version": "3.1.1",
      "resolved": "https://registry.npmjs.org/escalade/-/escalade-
```

```
3.1.1.tgz",
      "integrity": "sha512-
k0er2gUkLf8O0zKJiAhmkTnJlTvINGv7ygDNPbeIsX/TJjGJZHuh9B2UxbsaEkmlEo9M
fhrSzmhIlhRlI2GXnw==",
    },
    "escape-html": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/escape-html/-/escape-
html-1.0.3.tgz",
      "integrity": "sha512-
NiSupZ4OeuGwr68lGIeym/ksIZMJodUGOSCZ/FSnTxcrekbvqrgdUxlJOMpijaKZVjAJ
rWrGs/6Jy8OMuyj9ow==",
    },
    "etag": {
      "version": "1.8.1",
      "resolved": "https://registry.npmjs.org/etag/-/etag-
1.8.1.tgz",
      "integrity": "sha512-
aIL5Fx7mawVa300al2BnEE4iNvo1qETxLrPI/o05L7z6go7fCw1J6EQmbK4FmJ2AS7kg
VF/KEZWufBfdClMcPg==",
    },
    "express": {
      "version": "4.18.2",
      "resolved": "https://registry.npmjs.org/express/-/express-
4.18.2.tgz",
      "integrity": "sha512-
5/PsL6iGPdfQ/lKM1UuielYgv3BUoJfz1aUwU9vHZ+J7gyvwdQXFEBIEIaxeGf0GIcre
ATNyBExtalisDbuMqQ==",
      "requires": {
        "accepts": "~1.3.8",
        "array-flatten": "1.1.1",
        "body-parser": "1.20.1",
        "content-disposition": "0.5.4",
        "content-type": "~1.0.4",
        "cookie": "0.5.0",
        "cookie-signature": "1.0.6",
        "debug": "2.6.9",
        "depd": "2.0.0",
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "etag": "~1.8.1",
        "finalhandler": "1.2.0",
        "fresh": "0.5.2",
        "http-errors": "2.0.0",
        "merge-descriptors": "1.0.1",
        "methods": "~1.1.2",
        "on-finished": "2.4.1",
        "parseurl": "~1.3.3",
        "path-to-regexp": "0.1.7",
        "proxy-addr": "~2.0.7",
        "qs": "6.11.0",
        "range-parser": "~1.2.1",
        "safe-buffer": "5.2.1",
        "send": "0.18.0",
        "serve-static": "1.15.0",
        "setprototypeof": "1.2.0",
        "statuses": "2.0.1",
        "type-is": "~1.6.18",
        "utils-merge": "1.0.1",
```

```
        "vary": "~1.1.2"
      }
    },
    "filelist": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/filelist/-/filelist-
1.0.4.tgz",
      "integrity": "sha512-
w1cEuf3S+DrLCQL7ET6kz+gmlJdbq9J7yXCSjK/OZCPA+qEN1WyF4ZAf0YYJa4/shHJr
a2t/d/r8SV4Ji+x+8Q==",
      "requires": {
        "minimatch": "^5.0.1"
      },
      "dependencies": {
        "brace-expansion": {
          "version": "2.0.1",
          "resolved": "https://registry.npmjs.org/brace-expansion/-
/brace-expansion-2.0.1.tgz",
          "integrity": "sha512-
XnAIvQ8eM+kC6aULx6wuQiwVsnzsi9d3WxzV3FpWTGA19F621kwdbsAcFKXgKUHZWsy+
mY6iL1sHTxWEFCytDA==",
          "requires": {
            "balanced-match": "^1.0.0"
          }
        },
        "minimatch": {
          "version": "5.1.6",
          "resolved": "https://registry.npmjs.org/minimatch/-
/minimatch-5.1.6.tgz",
          "integrity": "sha512-
lKwV/1brpG6mBUFHtb7NUmtABCb2WZZmm2wNiOA5hAb8VdCS4B3dtMWyvcoViccwAW/C
OERjXLt0zP1zXUN26g==",
          "requires": {
            "brace-expansion": "^2.0.1"
          }
        }
      }
    },
    "fill-range": {
      "version": "7.0.1",
      "resolved": "https://registry.npmjs.org/fill-range/-/fill-
range-7.0.1.tgz",
      "integrity": "sha512-
qOo9F+dMUmC2Lcb4BbVvnKJxTPjCm+RRpe4gDuGrzkL7mEVl/djYSu2OdQ2Pa302N4oq
kSg9ir6jaLWJ2USVpQ==",
      "requires": {
        "to-regex-range": "^5.0.1"
      }
    },
    "finalhandler": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/finalhandler/-
/finalhandler-1.2.0.tgz",
      "integrity": "sha512-
5uXcUVftlQMFnWC9qu/svkWv3GTd2PfUhK/3PLkYNAe7FbqJMt3515HaxE6eRL74Gdsr
iiwujiawdaB1BpEISg==",
      "requires": {
        "debug": "2.6.9",
        "encodeurl": "~1.0.2",
```

```
        "escape-html": "~1.0.3",
        "on-finished": "2.4.1",
        "parseurl": "~1.3.3",
        "statuses": "2.0.1",
        "unpipe": "~1.0.0"
      }
    },
    "follow-redirects": {
      "version": "1.15.2",
      "resolved": "https://registry.npmjs.org/follow-redirects/-
/follow-redirects-1.15.2.tgz",
      "integrity": "sha512-
VQLG33o04KaQ8uYi2tVNbdrWp1QWxNNea+nmIB4EVM28v0hmP17z7aG1+wAkNzVq4KeX
Tq3221ye5qTJP91JwA==",
    },
    "formidable": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/formidable/-
/formidable-2.1.1.tgz",
      "integrity": "sha512-
0EcS9wCFEzLvfiks7omJ+SiYJAiD+TzK4Pcw1UlUoGnhUxDcMKjt0P7x8wEb0u6OHu8N
b98WG3nxtlF5C7bvUQ==",
      "requires": {
        "dezalgo": "^1.0.4",
        "hexoid": "^1.0.0",
        "once": "^1.4.0",
        "qs": "^6.11.0"
      }
    },
    "forwarded": {
      "version": "0.2.0",
      "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-
0.2.0.tgz",
      "integrity": "sha512-
buRG0fpBtRHSTCOASe6hD258tEubFoRLb4ZNA6NxMVHNw2gOcwHo9wyablzMzOA5z9xA
9L1KNjk/Nt6MT9aYow==",
    },
    "fresh": {
      "version": "0.5.2",
      "resolved": "https://registry.npmjs.org/fresh/-/fresh-
0.5.2.tgz",
      "integrity": "sha512-
zJ2mQYM18rEFOudeV4GShTGIQ7RbzA7ozbU9I/XBpm7kqgMywgmylMwXHxZJmkVoYkna
9d2pVXVXPdYTP9ej8Q==",
    },
    "fsevents": {
      "version": "2.3.2",
      "resolved": "https://registry.npmjs.org/fsevents/-/fsevents-
2.3.2.tgz",
      "integrity": "sha512-
xiqMQR4xAeHTuB9uWm+fFRcIOgKBMiOBP+eXiyT7jsgVCq1bkVygt00oASowB7EdtpOH
aaPgKt812P9ab+DDKA==",
      "optional": true
    },
    "function-bind": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/function-bind/-
/function-bind-1.1.1.tgz",
      "integrity": "sha512-
```

```
yIovAzMX49sF8Yl58fSCWJ5svSLuaibPxXQJFLmBObTuCr0Mf1KiPopGM9NiFjiYBCbf
aa2Fh6breQ6ANVTI0A=="
    },
    "get-caller-file": {
      "version": "2.0.5",
      "resolved": "https://registry.npmjs.org/get-caller-file/-/get-
caller-file-2.0.5.tgz",
      "integrity": "sha512-
DyFP3BM/3YHTQOCUL/w0OZHR0lpKeGrxotcHWcqNEdnltqFwXVfhEBQ94eIo34AfQpo0
rGki4cyIiftY06h2Fg==",
    },
    "get-intrinsic": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/get-intrinsic/-/get-
intrinsic-1.2.0.tgz",
      "integrity": "sha512-
L049y6nFOuom5wGyRc3/gdTLO94dySVKRACj1RmJZBQXlbTMhtNIgkWkUHq+jYmZvKf1
4EW1EoJnnjbmoHij0Q==",
      "requires": {
        "function-bind": "^1.1.1",
        "has": "^1.0.3",
        "has-symbols": "^1.0.3"
      }
    },
    "glob-parent": {
      "version": "5.1.2",
      "resolved": "https://registry.npmjs.org/glob-parent/-/glob-
parent-5.1.2.tgz",
      "integrity": "sha512-
AOIgSQCepiJYwP3ARnGx+5VnTu2HBYdzbGP45eLw1vr3zB3vZLeyed1sC9hnbcOc9/Sr
MyM5RPQrkGz4aS9Zow==",
      "requires": {
        "is-glob": "^4.0.1"
      }
    },
    "has": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/has/-/has-1.0.3.tgz",
      "integrity": "sha512-
f2dvO0VU6Oej7RkWJGrehjbzMAjFp5/VKPp5tTpWIV4JHHZK1/BxbFRtf/siA2SWTe09
caDmVtYYzWEIbBS4zw==",
      "requires": {
        "function-bind": "^1.1.1"
      }
    },
    "has-flag": {
      "version": "4.0.0",
      "resolved": "https://registry.npmjs.org/has-flag/-/has-flag-
4.0.0.tgz",
      "integrity": "sha512-
EykJT/Q1KjTWctppgIAgfSO0tKVuZUjhgMr17kqTumMl6Afv3EISleU7qZUzoXDFTAHT
DC4NOoG/ZxU3EvlMPQ==",
    },
    "has-symbols": {
      "version": "1.0.3",
      "resolved": "https://registry.npmjs.org/has-symbols/-/has-
symbols-1.0.3.tgz",
      "integrity": "sha512-
l3LCuF6MgDNwTDKkdYGEihYjt5pRPbEg46rtlmnSPlUbgmB8LOIrKJbYYFBSbnPaJexM
```

```
KtiPO8hmeRjRz2Td+A=="
    },
    "hexoid": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/hexoid/-/hexoid-
1.0.0.tgz",
      "integrity": "sha512-
QFLV0taWQOZtvIRIAdBChesmogZrtuXvVWsFHZTk2SU+anspqZ2vMnoLg7IE1+Uk16N1
9APic1BuF8bC8c2m5g=="
    },
    "http-errors": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/http-errors/-/http-
errors-2.0.0.tgz",
      "integrity": "sha512-
FtwrG/euBzaEjYeRqOgly7G0qviiXoJWnvEH2Z1plBdXgbyjv34pHTSb9zoeHMyDy33+
DWy5Wt9Wo+TURtOYSQ==",
      "requires": {
        "depd": "2.0.0",
        "inherits": "2.0.4",
        "setprototypeof": "1.2.0",
        "statuses": "2.0.1",
        "toidentifier": "1.0.1"
      }
    },
    "iconv-lite": {
      "version": "0.4.24",
      "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-
lite-0.4.24.tgz",
      "integrity": "sha512-
v3MXnZAcvnywkTUEZomIActle7RXXeedOR31wwl7VlyoXO4Qi9arvSenNQWne1TcRwhC
L1HwLI21bEqdpj8/rA==",
      "requires": {
        "safer-buffer": ">= 2.1.2 < 3"
      }
    },
    "ignore-by-default": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/ignore-by-default/-
/ignore-by-default-1.0.1.tgz",
      "integrity": "sha512-
Ius2VYcGNk7T90CppJqcIkS5ooHUZyIQK+ClZfMfMNFEF9VSE73Fq+906u/CWu92x4gz
ZMWOwfFYckPObzdEbA=="
    },
    "inherits": {
      "version": "2.0.4",
      "resolved": "https://registry.npmjs.org/inherits/-/inherits-
2.0.4.tgz",
      "integrity": "sha512-
k/vGaX4/Yla3WzyMCvTQOXYeIHvqOKtnqBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIF
eMpXKxa4dYeZIQqewQ=="
    },
    "ipaddr.js": {
      "version": "1.9.1",
      "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-
1.9.1.tgz",
      "integrity": "sha512-
0KI/607xoxSToH7GjN1FfSbLoU0+btTicjsQSWQlh/hZykN8KpmMf7uYwPW3R+akZ6R/
w18ZlXSHBYXiYUPO3g=="
```

```
    },
    "is-binary-path": {
      "version": "2.1.0",
      "resolved": "https://registry.npmjs.org/is-binary-path/-/is-
binary-path-2.1.0.tgz",
      "integrity": "sha512-
ZMERYes6pDydyuGidse7OsHxtbI7WVeUEozgR/g7rd0xUimYNlvZRE/K2MgZTjWy725I
felLeVcEM97mmtRGXw==",
      "requires": {
        "binary-extensions": "^2.0.0"
      }
    },
    "is-extglob": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/is-extglob/-/is-
extglob-2.1.1.tgz",
      "integrity": "sha512-
SbKbANkN603Vi4jEZv49LeVJMn4yGwsbzZworEoyEiutsN3nJYdbO36zfhGJ6QEDpOZI
FkDtnq5JRxmvl3jsoQ==",
    },
    "is-fullwidth-code-point": {
      "version": "3.0.0",
      "resolved": "https://registry.npmjs.org/is-fullwidth-code-
point/-/is-fullwidth-code-point-3.0.0.tgz",
      "integrity": "sha512-
zymm5+u+sCsSWyD9qNaejV3DFvhCKclKdizYaJUuHA83RLjb7nSuGnddCHGv0hk+KY7B
MAlsWeK4Ueg6EV6XQg==",
    },
    "is-glob": {
      "version": "4.0.3",
      "resolved": "https://registry.npmjs.org/is-glob/-/is-glob-
4.0.3.tgz",
      "integrity": "sha512-
xelSayHH36ZgE7ZWhli7pW34hNbNl8Ojv5KVmkJD4hBdD3th8Tfk9vYasLM+mXWOZhFk
gZfxhLSnrwRr4elSSg==",
      "requires": {
        "is-extglob": "^2.1.1"
      }
    },
    "is-number": {
      "version": "7.0.0",
      "resolved": "https://registry.npmjs.org/is-number/-/is-number-
7.0.0.tgz",
      "integrity": "sha512-
41Cifkg6e8TylSpdtTpeLVMqvSBEVzTttHvERD741+pnZ8ANv0004MRL43QKPDlK9cGv
Np6NZWZUBlbGXYxxng==",
    },
    "is-utf8": {
      "version": "0.2.1",
      "resolved": "https://registry.npmjs.org/is-utf8/-/is-utf8-
0.2.1.tgz",
      "integrity": "sha512-
rMYPYvCzsXywIsldgLaSoPlw5PfoB/ssr7hY4pLfcodrA5M/eArza1a9VmTiNIBNMjOG
r1Ow9mTyU2o69U6U9Q==",
    },
    "isarray": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/isarray/-/isarray-
1.0.0.tgz",
```

```
      "integrity": "sha512-
VLghIWNM6ELQzo7zwmcg0NmTVyWKYjvIeM83yjp0wRDTmUnrM678fQbcKBo6n2CJEF0s
zoG//ytg+TKla89ALQ=="
    },
    "jake": {
      "version": "10.8.5",
      "resolved": "https://registry.npmjs.org/jake/-/jake-
10.8.5.tgz",
      "integrity": "sha512-
sVpxYeuAhWt0OTWITwT98oyV0GsXyMlXCF+3L1SuafBVUIr/uILGRB+NqwkzhgXKvoJp
DIpQvqkUALgdmQsQxw==",
      "requires": {
        "async": "^3.2.3",
        "chalk": "^4.0.2",
        "filelist": "^1.0.1",
        "minimatch": "^3.0.4"
      }
    },
    "json-to-csv-export": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/json-to-csv-export/-
/json-to-csv-export-2.1.1.tgz",
      "integrity": "sha512-
JQLYqGfaJP4dVV0zIxH6RmHOdnJ32Qch3yDxDtWQy8KsOv73SthVRo38ANXckJxl/WUf
GYvvtOlVwPURMTWqlA=="
    },
    "json2csv": {
      "version": "6.0.0-alpha.2",
      "resolved": "https://registry.npmjs.org/json2csv/-/json2csv-
6.0.0-alpha.2.tgz",
      "integrity": "sha512-
nJ3oP6QxN8z69IT1HmrJdfVxhU1kLTBVgMfRnNZc37YEY+jZ4nU27rBGxT4vaqM/KUCa
vLRhntmTuBFqZLBUcA==",
      "requires": {
        "@streamparser/json": "^0.0.6",
        "commander": "^6.2.0",
        "lodash.get": "^4.4.2"
      }
    },
    "localtunnel": {
      "version": "2.0.2",
      "resolved": "https://registry.npmjs.org/localtunnel/-
/localtunnel-2.0.2.tgz",
      "integrity": "sha512-
n418Cn5ynvJd7m/N1d9WVJISLJF/ellZnfsLnx8WBWGzxv/ntNcFkJ1o6se5quUhCplf
LGBNL5tYHiq5WF3Nug==",
      "requires": {
        "axios": "0.21.4",
        "debug": "4.3.2",
        "openurl": "1.1.1",
        "yargs": "17.1.1"
      },
      "dependencies": {
        "debug": {
          "version": "4.3.2",
          "resolved": "https://registry.npmjs.org/debug/-/debug-
4.3.2.tgz",
          "integrity": "sha512-
mOp8wKcvj7XxC78zLgw/ZA+6TSgkoE2C/ienthhRD298T7UNwAg9diBpLRxC0mOezLl4
```

```
B0xV7M0cCO6P/O0Xhw==",
          "requires": {
            "ms": "2.1.2"
          }
        },
        "ms": {
          "version": "2.1.2",
          "resolved": "https://registry.npmjs.org/ms/-/ms-
2.1.2.tgz",
          "integrity": "sha512-
sGkPx+VjMtmA6MX27oA4FBFELFCZZ4S4XqeGOXCv68tT+jb3vk/RyaKWP0PTKyWtmLSM
0b+adUTEvbs1PEaH2w=="
        }
      }
    },
    "lodash": {
      "version": "4.17.21",
      "resolved": "https://registry.npmjs.org/lodash/-/lodash-
4.17.21.tgz",
      "integrity": "sha512-
v2kDEe57lecTulaDIuNTPy3Ry4gLGJ6Z1O3vE1krgXZNrsQ+LFTGHVxVjcXPs17LhbZV
GedAJv8XZ1tvj5FvSg==",
    },
    "lodash.get": {
      "version": "4.4.2",
      "resolved": "https://registry.npmjs.org/lodash.get/-
/lodash.get-4.4.2.tgz",
      "integrity": "sha512-
z+Uw/vLuy6gQe8cfaFWD7p0wVv8fJl3mbzXh33RS+0oW2wvUqiRXiQ69gLWSLpgB5/6s
U+r6BlQR0MBILadqTQ==",
    },
    "media-typer": {
      "version": "0.3.0",
      "resolved": "https://registry.npmjs.org/media-typer/-/media-
typer-0.3.0.tgz",
      "integrity": "sha512-
dq+qelQ9akHpcOl/gUVRTxVIOkAJ1wR3QAvb4RsVjS8oVoFjDGTc679wJYmUmknUF5Hw
MLOgb5O+a3KxfWapPQ==",
    },
    "merge-descriptors": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/merge-descriptors/-
/merge-descriptors-1.0.1.tgz",
      "integrity": "sha512-
cCi6g3/Zr1iqQi6ySbseM1Xvooa98N0w31jzUYrXPX2xqObmFGHJ0tQ5u74H3mVh7wLo
uTseZyYIq39g8cNp1w==",
    },
    "methods": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/methods/-/methods-
1.1.2.tgz",
      "integrity": "sha512-
iclAHeNqNm68zFtnZ0e+1L2yUIdvzNoauKU4WBA3VvH/vPFieF7qfRlwUZU+DA9P9bPX
IS90ulxoUoCH23sV2w==",
    },
    "mime": {
      "version": "1.6.0",
      "resolved": "https://registry.npmjs.org/mime/-/mime-
1.6.0.tgz",
```

```
      "integrity": "sha512-
x0Vn8spI+wuJ1O6S7gnbaQg8Pxh4NNHb7KSINmEWKiPE4RKOplvijn+NkmYmmRgP68mc
70j2EbeTFRsrswaQeg=="
    },
    "mime-db": {
      "version": "1.52.0",
      "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-
1.52.0.tgz",
      "integrity": "sha512-
sPU4uV7dYlvtWJxwwxHD0PuihVNiE7TyAbQ5SWxDCB9mUYvOgroQOwYQQOKPJ8CIbE+1
ETVlOoK1UC2nU3gYvg==="
    },
    "mime-types": {
      "version": "2.1.35",
      "resolved": "https://registry.npmjs.org/mime-types/-/mime-
types-2.1.35.tgz",
      "integrity": "sha512-
ZDY+bPm5zTTF+YpCrAU9nK0UgICYPT0QtT1NZWFv4s++TNkcgVaT0g6+4R2uI4MjQjzy
sHB1zxuWL50hzaeXiw==",
      "requires": {
        "mime-db": "1.52.0"
      }
    },
    "minimatch": {
      "version": "3.1.2",
      "resolved": "https://registry.npmjs.org/minimatch/-/minimatch-
3.1.2.tgz",
      "integrity": "sha512-
J7p63hRiAjw1NDEww1W7i37+ByIrOWO5XQQAzZ3VOcL0PNybwpfmV/N05zFAzwQ9USyE
cX6t3UO+K5aqBQOIHw==",
      "requires": {
        "brace-expansion": "^1.1.7"
      }
    },
    "minimist": {
      "version": "1.2.8",
      "resolved": "https://registry.npmjs.org/minimist/-/minimist-
1.2.8.tgz",
      "integrity": "sha512-
2yyAR8qBkN3YuheJanUpWC5U3bb5osDywNB8RzDVlDwDHbocAJveqqj1u8+SVD7jkWT4
yvsHCpWqqWqAxb0zCA==""
    },
    "mkdirp": {
      "version": "0.5.6",
      "resolved": "https://registry.npmjs.org/mkdirp/-/mkdirp-
0.5.6.tgz",
      "integrity": "sha512-
FP+p8RB8OWpF3YZBCrP5gtADmtXApB5AMLn+vdyA+PyxCjrCs00mjyUozssO33cwDeT3
wNGdLxJ5M//YqtHAJw==",
      "requires": {
        "minimist": "^1.2.6"
      }
    },
    "ms": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",
      "integrity": "sha512-
Tpp60P6IUJDTuOq/5Z8cdskzJujfwqfOTkrwIwj7IRISpnkJnT6SyJ4PCPnGMoFjC9dd
hal5KVIYtAt97ix05A==""
```

```
    },
    "multer": {
      "version": "1.4.5-lts.1",
      "resolved": "https://registry.npmjs.org/multer/-/multer-1.4.5-
lts.1.tgz",
      "integrity": "sha512-
ywPWvcDMeH+z9gQq5qYHCCy+ethsk4goepZ45GLD63fOu0YcNecQxi64nDs3qluZB+mu
rG3/D4dJ7+dGctcCQQ==",
      "requires": {
        "append-field": "^1.0.0",
        "busboy": "^1.0.0",
        "concat-stream": "^1.5.2",
        "mkdirp": "^0.5.4",
        "object-assign": "^4.1.1",
        "type-is": "^1.6.4",
        "xtend": "^4.0.0"
      }
    },
    "negotiator": {
      "version": "0.6.3",
      "resolved": "https://registry.npmjs.org/negotiator/-
/negotiator-0.6.3.tgz",
      "integrity": "sha512-
+EUsqGPLsM+j/zdChZjsnX51g4XrHFOIXwfnCVPGlQk/k5giakcKsuxCObBRu6DSm9op
w/O6slWbJdghQM4bBg=="
    },
    "nodemon": {
      "version": "2.0.22",
      "resolved": "https://registry.npmjs.org/nodemon/-/nodemon-
2.0.22.tgz",
      "integrity": "sha512-
B8YqaKMmyuCO7BowF1Z1/mkPqLk6cs/l63Ojtd6otKjMx47Dq1utxfRxcavH1I7VSaL8
n5BUaoutadnsX3AAVQ==",
      "requires": {
        "chokidar": "^3.5.2",
        "debug": "^3.2.7",
        "ignore-by-default": "^1.0.1",
        "minimatch": "^3.1.2",
        "pstree.remy": "^1.1.8",
        "semver": "^5.7.1",
        "simple-update-notifier": "^1.0.7",
        "supports-color": "^5.5.0",
        "touch": "^3.1.0",
        "undefsafe": "^2.0.5"
      },
      "dependencies": {
        "debug": {
          "version": "3.2.7",
          "resolved": "https://registry.npmjs.org/debug/-/debug-
3.2.7.tgz",
          "integrity": "sha512-
CFjzYYAi4ThfiQvizrFQevTTXHtnCqWfe7x1AhgEscTz6ZbLbfoLRLPugTQyBth6f8ZE
RVUSyWHFD/7Wu4t1XQ==",
          "requires": {
            "ms": "^2.1.1"
          }
        },
        "has-flag": {
          "version": "3.0.0",
```

```
          "resolved": "https://registry.npmjs.org/has-flag/-/has-
flag-3.0.0.tgz",
          "integrity": "sha512-
sKJf1+ceQBr4SMkvQnBDNDtf4TXpVhVGateu0t918bl30FnbE2m4vNLX+VWe/dpjlb+H
ugGYzW7uQXH98HPEYw=="
        },
        "ms": {
          "version": "2.1.3",
          "resolved": "https://registry.npmjs.org/ms/-/ms-
2.1.3.tgz",
          "integrity": "sha512-
6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9x
AOtHnSO/tXtF3WRTlA==",
        },
        "supports-color": {
          "version": "5.5.0",
          "resolved": "https://registry.npmjs.org/supports-color/-
/supports-color-5.5.0.tgz",
          "integrity": "sha512-
QjVjwdXIt408MIiAqCX4oUKsgU2EqAGzs2Ppkm4aQYbjm+ZEWEcW4SfFNTr4uMNZma0e
y4f5lgLrkB0aX0QMow==",
          "requires": {
            "has-flag": "^3.0.0"
          }
        }
      }
    },
    "nopt": {
      "version": "1.0.10",
      "resolved": "https://registry.npmjs.org/nopt/-/nopt-
1.0.10.tgz",
      "integrity": "sha512-
NWmpvLSqUrgrAC9HCuxEvb+PSloHpqVu+FqcO4eeF2h5qYRhA7ev6KvelyQAKtegUbC6
RypJnlEOhd8vloNKYg==",
      "requires": {
        "abbrev": "1"
      }
    },
    "normalize-path": {
      "version": "3.0.0",
      "resolved": "https://registry.npmjs.org/normalize-path/-
/normalize-path-3.0.0.tgz",
      "integrity": "sha512-
6eZs5Ls3WtCisHWp9S2GUy8dqkpGi4BVSz3GaqiE6ezub0512ESztXUwUB6C6IKbQkY2
Pnb/mD4WYojCRwcwLA=="
    },
    "object-assign": {
      "version": "4.1.1",
      "resolved": "https://registry.npmjs.org/object-assign/-
/object-assign-4.1.1.tgz",
      "integrity": "sha512-
rJgTQnkUnH1sFw8yT6VSU3zD3sWmu6sZhIseY8VX+GRu3P6F7Fu+JNDoXfklElbLJSnc
3FUQHVe4cU5hj+BcUg=="
    },
    "object-inspect": {
      "version": "1.12.3",
      "resolved": "https://registry.npmjs.org/object-inspect/-
/object-inspect-1.12.3.tgz",
      "integrity": "sha512-
```

```
geUvdk7c+eizMNUDkRpW1wJwgfOiOeHbxBR/hLXK1aT6zmVSO0jsQcs7fj6MGw89jC/c
jGfLcNOrtMYtGqm81g=="
    },
    "on-finished": {
      "version": "2.4.1",
      "resolved": "https://registry.npmjs.org/on-finished/-/on-
finished-2.4.1.tgz",
      "integrity": "sha512-
oVlzkg3ENAhCk2zdv7IJwd/QUD4z2RxRwpkcGY8psCVcCYZNq4wYnVWALHM+brtuJjeP
WiYF/ClmuDr8Ch5+kg==",
      "requires": {
        "ee-first": "1.1.1"
      }
    },
    "once": {
      "version": "1.4.0",
      "resolved": "https://registry.npmjs.org/once/-/once-
1.4.0.tgz",
      "integrity": "sha512-
lNaJgI+2Q5URQBkccEKHTQOPaXdUxnZZElQTZY0MFUAuaEqe1E+Nyvgdz/aIyNi6Z9Mz
O5dv1H8n58/GELp3+w==",
      "requires": {
        "wrappy": "1"
      }
    },
    "openurl": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/openurl/-/openurl-
1.1.1.tgz",
      "integrity": "sha512-
d/gTkTb1i1GKz5k3XE3XFV/PxQ1k45zDqGP2OA7YhgsaLoqm6qRvARAZOFer1fcXritW
lGBRCu/UgeS4HAnXAA=="
    },
    "papaparse": {
      "version": "5.4.1",
      "resolved": "https://registry.npmjs.org/papaparse/-/papaparse-
5.4.1.tgz",
      "integrity": "sha512-
HipMsgJkZu8br23pW15uvo6sib6wne/4woLZPlFf3rpDyMe9ywEXUsuD7+6K9PRkJlVT
51j/sCOYDKGGS3ZJrw=="
    },
    "parseurl": {
      "version": "1.3.3",
      "resolved": "https://registry.npmjs.org/parseurl/-/parseurl-
1.3.3.tgz",
      "integrity": "sha512-
CiyeOxFT/JZyN5m0z9PfXw4SCBJ6Sygz1Dpl0wqjlhDEGGBP1GnsUVEL0p63hoG1fcj3
fHynXi9NYO4nWOL+qQ=="
    },
    "path-to-regexp": {
      "version": "0.1.7",
      "resolved": "https://registry.npmjs.org/path-to-regexp/-/path-
to-regexp-0.1.7.tgz",
      "integrity": "sha512-
5DFkuoqlv1uYQKxy8omFBeJPQcdoE07Kv2sferDCrAq1ohOU+MSDswDIbnx3YAM60qIO
nYa53wBhXW0EbMonrQ=="
    },
    "picomatch": {
      "version": "2.3.1",
```

```
      "resolved": "https://registry.npmjs.org/picomatch/-/picomatch-
2.3.1.tgz",
      "integrity": "sha512-
JU3teHTNjmE2VCGFzuY8EXzCDVwEqB2a8fsIvwaStHhAWJEeVd1o1QD80CU6+ZdEXXSL
bSsuLwJjkCBWqRQUVA==",
    },
    "process-nextick-args": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/process-nextick-args/-
/process-nextick-args-2.0.1.tgz",
      "integrity": "sha512-
3ouUOpQhtgrbOa17J7+uxOTpITYWaGP7/AhoR3+A+/1e9skrzelGi/dXzEYyvbxubEF6
Wn2ypscTKiKJFFn1ag==",
    },
    "proxy-addr": {
      "version": "2.0.7",
      "resolved": "https://registry.npmjs.org/proxy-addr/-/proxy-
addr-2.0.7.tgz",
      "integrity": "sha512-
llQsMLSUDUPT44jdrU/O37qlnifitDP+ZwrmmZcoSKyLKvtZxpyV0n2/bD/N4tBAAZ/g
JEdZU7KMraoK1+XYAg==",
      "requires": {
        "forwarded": "0.2.0",
        "ipaddr.js": "1.9.1"
      }
    },
    "pstree.remy": {
      "version": "1.1.8",
      "resolved": "https://registry.npmjs.org/pstree.remy/-
/pstree.remy-1.1.8.tgz",
      "integrity": "sha512-
77DZwxQmxKnu3aR542U+X8FypNzbfJ+C5XQDk3uWjWxn6151aIMGthWYRXTqT1E5oJvg
+ljaa2OJi+VfvCOQ8w==",
    },
    "python-shell": {
      "version": "5.0.0",
      "resolved": "https://registry.npmjs.org/python-shell/-/python-
shell-5.0.0.tgz",
      "integrity": "sha512-
RUOOOjHLhgR1MIQrCtnEqz/HJ1RMZBIN+REnpSUrfft2bXqXy69fwJASVziWExfFXsR1
bCY0TznnHooNsCo0/w==",
    },
    "qs": {
      "version": "6.11.0",
      "resolved": "https://registry.npmjs.org/qs/-/qs-6.11.0.tgz",
      "integrity": "sha512-
MvjoMCJwEarSbUYk5O+nmoSzSutSsTwF85zcHPQ9OrlFoZOYIjaqBAJIqIXjptyD5vTh
xGq52Xu/MaJzRkIk4Q==",
      "requires": {
        "side-channel": "^1.0.4"
      }
    },
    "range-parser": {
      "version": "1.2.1",
      "resolved": "https://registry.npmjs.org/range-parser/-/range-
parser-1.2.1.tgz",
      "integrity": "sha512-
Hrgsx+orqoygnmhFbKaHE6c296J+HTAQXoxEF6gNupROmmGJRoyzfG3ccAveqCBrwr/2
yxQ5BVd/GTl5agOwSg==",
```

```
    },
    "raw-body": {
      "version": "2.5.1",
      "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-
2.5.1.tgz",
      "integrity": "sha512-
qqJBtEyVgS0ZmPGdCFPWJ3FreoqvG4MVQln/kCgF7Olq95IbOp0/BWyMwbdtn4VTvkM8
Y7khCQ2Xgk/tcrCXig==",
      "requires": {
        "bytes": "3.1.2",
        "http-errors": "2.0.0",
        "iconv-lite": "0.4.24",
        "unpipe": "1.0.0"
      }
    },
    "readable-stream": {
      "version": "2.3.8",
      "resolved": "https://registry.npmjs.org/readable-stream/-
/readable-stream-2.3.8.tgz",
      "integrity": "sha512-
8p0AUk4XODgIewSi0l8Epjs+EVnWiK7NoDIEGU0HhE7+ZyY8D1IMY7odu5lRrFXGg71L
15KG8QrPmum45RTtdA==",
      "requires": {
        "core-util-is": "~1.0.0",
        "inherits": "~2.0.3",
        "isarray": "~1.0.0",
        "process-nextick-args": "~2.0.0",
        "safe-buffer": "~5.1.1",
        "string_decoder": "~1.1.1",
        "util-deprecate": "~1.0.1"
      },
      "dependencies": {
        "safe-buffer": {
          "version": "5.1.2",
          "resolved": "https://registry.npmjs.org/safe-buffer/-
/safe-buffer-5.1.2.tgz",
          "integrity": "sha512-
Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOI
rt8ud/wPtojys4G6+g=="
        }
      }
    },
    "readdirp": {
      "version": "3.6.0",
      "resolved": "https://registry.npmjs.org/readdirp/-/readdirp-
3.6.0.tgz",
      "integrity": "sha512-
hOS089on8RduqdbhvQ5Z37A0ESjsqz6qnRcffsMU3495FuTdqSm+7bhJ29JvIOsBDEEn
an5DPu9t3To9VRlMzA==",
      "requires": {
        "picomatch": "^2.2.1"
      }
    },
    "require-directory": {
      "version": "2.1.1",
      "resolved": "https://registry.npmjs.org/require-directory/-
/require-directory-2.1.1.tgz",
      "integrity": "sha512-
fGxEI7+wsG9xrvdjsrlmL22OMTTiHRwAMroiEeMgq8gzoLC/PQr7RsRDSTLUg/bZAZtF
```

```
+TVIkHc6/4RIKrui+Q=="
    },
    "safe-buffer": {
      "version": "5.2.1",
      "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-
buffer-5.2.1.tgz",
      "integrity": "sha512-
rp3So07KcdmmKbGvgaNxQSJr7bGVSVk5S9Eq1F+ppbRo70+YeaDxkw5Dd8NPN+GD6bjn
Ym2VuPuCXmpuYvmCXQ=="
    },
    "safer-buffer": {
      "version": "2.1.2",
      "resolved": "https://registry.npmjs.org/safer-buffer/-/safer-
buffer-2.1.2.tgz",
      "integrity": "sha512-
YZo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXXevb5dJI7tpyN2ADxGcQbHG7
vcyRHk0cbwqcQriUtg=="
    },
    "semver": {
      "version": "5.7.1",
      "resolved": "https://registry.npmjs.org/semver/-/semver-
5.7.1.tgz",
      "integrity": "sha512-
sauaDf/PZdVgrLTNYHRtpXa1iRiKcaebiKQ1BJdpQlWH2lCvexQdX55snPFyK7Qzpudq
bCI0qXFfOasHdyNDGQ=="
    },
    "send": {
      "version": "0.18.0",
      "resolved": "https://registry.npmjs.org/send/-/send-
0.18.0.tgz",
      "integrity": "sha512-
qqWzuOjSFOuqPjFe4NOsMLafToQQwBSOEpS+FwEt3A2V3vKubTquT3vmLTQpFgMXp8Al
FWFuP1qKaJZOtPpVXg==",
      "requires": {
        "debug": "2.6.9",
        "depd": "2.0.0",
        "destroy": "1.2.0",
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "etag": "~1.8.1",
        "fresh": "0.5.2",
        "http-errors": "2.0.0",
        "mime": "1.6.0",
        "ms": "2.1.3",
        "on-finished": "2.4.1",
        "range-parser": "~1.2.1",
        "statuses": "2.0.1"
      },
      "dependencies": {
        "ms": {
          "version": "2.1.3",
          "resolved": "https://registry.npmjs.org/ms/-/ms-
2.1.3.tgz",
          "integrity": "sha512-
6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9x
AOtHnSO/tXtF3WRTlA=="
        }
      }
    },
```

```
    "serve-static": {
      "version": "1.15.0",
      "resolved": "https://registry.npmjs.org/serve-static/-/serve-
static-1.15.0.tgz",
      "integrity": "sha512-
XGuRDNjXUijsUL0vl6nSD7cwURuzEgglbOaFuZM9g3kwDXOWVTck0jLzjPzGD+TazWbb
oZYu52/9/XPdUgne9g==",
      "requires": {
        "encodeurl": "~1.0.2",
        "escape-html": "~1.0.3",
        "parseurl": "~1.3.3",
        "send": "0.18.0"
      }
    },
    "setprototypeof": {
      "version": "1.2.0",
      "resolved": "https://registry.npmjs.org/setprototypeof/-
/setprototypeof-1.2.0.tgz",
      "integrity": "sha512-
E5LDX7Wrp85Kil5bhZv46j8jOeboKq5JMmYM3gVGdGH8xFpPWXUMsNrlODCrkoxMEeNi
/XZIwuRvY4XNwYMJpw=="
    },
    "side-channel": {
      "version": "1.0.4",
      "resolved": "https://registry.npmjs.org/side-channel/-/side-
channel-1.0.4.tgz",
      "integrity": "sha512-
q5XPytqFEIKHkGdiMIrY10mvLRvnQh42/+GoBlFW3b2LXLE2xxJpZFdm94we0BaoV3Rw
JyGqg5wS7epxTv0Zvw==",
      "requires": {
        "call-bind": "^1.0.0",
        "get-intrinsic": "^1.0.2",
        "object-inspect": "^1.9.0"
      }
    },
    "simple-update-notifier": {
      "version": "1.1.0",
      "resolved": "https://registry.npmjs.org/simple-update-
notifier/-/simple-update-notifier-1.1.0.tgz",
      "integrity": "sha512-
VpsrsJSUcJEseSbMHkrsrAVSdvVS5I96Qo1QAQ4FxQ9wXFcB+pjj7FB7/us9+GcgfW4z
iHtYMc1J0PLczb55mg==",
      "requires": {
        "semver": "~7.0.0"
      },
      "dependencies": {
        "semver": {
          "version": "7.0.0",
          "resolved": "https://registry.npmjs.org/semver/-/semver-
7.0.0.tgz",
          "integrity": "sha512-
+GB6zVA9LWh6zovYQLALHwv5rb2PHGlJi3lfiqIHxR0uuwCgefcOJc59v9fv1w8GbStw
xuuqqAjI9NMAOOgq1A=="
        }
      }
    },
    "statuses": {
      "version": "2.0.1",
      "resolved": "https://registry.npmjs.org/statuses/-/statuses-
```

```
2.0.1.tgz",
      "integrity": "sha512-
RwNA9Z/7PrK06rYLIzFMlaF+l73iwpzsqRIFgbMLbTcLD6cOao82TaWefPXQvB2fOC4A
juYSEndS7N/mTCbkdQ==="
    },
    "stream-transform": {
      "version": "3.2.6",
      "resolved": "https://registry.npmjs.org/stream-transform/-
/stream-transform-3.2.6.tgz",
      "integrity": "sha512-
/pyOvaCQFqYTmrFhmMbnAEVo3SsTx1H39eUVPOtYeAgbEUc+rDo7GoP8LbHJgU83mKtz
Je/7Nq/ipaAnUOHgJQ==""
    },
    "streamsearch": {
      "version": "1.1.0",
      "resolved": "https://registry.npmjs.org/streamsearch/-
/streamsearch-1.1.0.tgz",
      "integrity": "sha512-
Mcc5wHehp9aXz1ax6bZUyY5afg9u2rv5cqQI3mRrYkGC8rW2hM02jWuwjtL++LS5qinS
yhj2QfLyNsuc+VsExg==""
    },
    "string_decoder": {
      "version": "1.1.1",
      "resolved": "https://registry.npmjs.org/string_decoder/-
/string_decoder-1.1.1.tgz",
      "integrity": "sha512-
n/ShnvDi6FHbbVfviro+WojiFzv+s8MPMHBczVePfUpDJLwoLT0ht1l4YwBCbi8pJAve
EEdnkHyPyTP/mzRfwg==",
      "requires": {
        "safe-buffer": "~5.1.0"
      },
      "dependencies": {
        "safe-buffer": {
          "version": "5.1.2",
          "resolved": "https://registry.npmjs.org/safe-buffer/-
/safe-buffer-5.1.2.tgz",
          "integrity": "sha512-
Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOI
rt8ud/wPtojys4G6+g==""
        }
      }
    },
    "string-width": {
      "version": "4.2.3",
      "resolved": "https://registry.npmjs.org/string-width/-/string-
width-4.2.3.tgz",
      "integrity": "sha512-
wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/c
xKTWyWUwvSTriiZz/g==",
      "requires": {
        "emoji-regex": "^8.0.0",
        "is-fullwidth-code-point": "^3.0.0",
        "strip-ansi": "^6.0.1"
      }
    },
    "strip-ansi": {
      "version": "6.0.1",
      "resolved": "https://registry.npmjs.org/strip-ansi/-/strip-
ansi-6.0.1.tgz",
```

```
      "integrity": "sha512-
Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/
0ooI7KrPuUSztUdU5A==",
      "requires": {
        "ansi-regex": "^5.0.1"
      }
    },
    "strip-bom": {
      "version": "2.0.0",
      "resolved": "https://registry.npmjs.org/strip-bom/-/strip-bom-
2.0.0.tgz",
      "integrity": "sha512-
kwrX1y7czp1E69n2ajbG65mIo9dqvJ+8aBQXOGVxqwvNbsXdFM6Lq37dLAY3mknUwru8
CfcCbfOLL/gMo+fi3g==",
      "requires": {
        "is-utf8": "^0.2.0"
      }
    },
    "supports-color": {
      "version": "7.2.0",
      "resolved": "https://registry.npmjs.org/supports-color/-
/supports-color-7.2.0.tgz",
      "integrity": "sha512-
qpCAvRl9stuOHveKsn7HncJRvv501qIacKzQlO/+Lwxc9+0q2wLyv4Dfvt80/DPn2pqO
BsJdDiogXGR9+OvwRw==",
      "requires": {
        "has-flag": "^4.0.0"
      }
    },
    "to-regex-range": {
      "version": "5.0.1",
      "resolved": "https://registry.npmjs.org/to-regex-range/-/to-
regex-range-5.0.1.tgz",
      "integrity": "sha512-
65P7iz6X5yEr1cwcgvQxbbIw7Uk3gOy5dIdtZ4rDveLqhrdJP+Li/Hx6tyK0NEb+2GCy
neCMJiGqrADCSNk8sQ==",
      "requires": {
        "is-number": "^7.0.0"
      }
    },
    "toidentifier": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/toidentifier/-
/toidentifier-1.0.1.tgz",
      "integrity": "sha512-
o5sSPKEkg/DIQNmH43V0/uerLrpzVedkUh8tGNvaeXpfpuwjKenlSox/2O/BTlZUtEe+
JG7s5YhEz608PlAHRA=="
    },
    "touch": {
      "version": "3.1.0",
      "resolved": "https://registry.npmjs.org/touch/-/touch-
3.1.0.tgz",
      "integrity": "sha512-
WBx8Uy5TLtOSRtIq+M03/sKDrXCLHxwDcquSP2c43Le03/9serjQBIztjRz6FkJez9D/
hleyAXTBGLwwZUw9lA==",
      "requires": {
        "nopt": "~1.0.10"
      }
    },
```

```
    "type-is": {
      "version": "1.6.18",
      "resolved": "https://registry.npmjs.org/type-is/-/type-is-
1.6.18.tgz",
      "integrity": "sha512-
TkRKr9sUTxEH8MdfuCSP7VizJyzRNMjj2J2do2Jr3Kym598JVdEksuzPQCnlFPW4ky9Q
+iA+ma9BGm06XQBy8g==",
      "requires": {
        "media-typer": "0.3.0",
        "mime-types": "~2.1.24"
      }
    },
    "typedarray": {
      "version": "0.0.6",
      "resolved": "https://registry.npmjs.org/typedarray/-
/typedarray-0.0.6.tgz",
      "integrity": "sha512-
/aCDEGatGvZ2BIk+HmLf4ifCJFwvKFNb9/JeZPMulfgFracn9QFcAf5GO8B/mweUjSob
lS5In0cWhqpfs/5PQA==",
    },
    "undefsafe": {
      "version": "2.0.5",
      "resolved": "https://registry.npmjs.org/undefsafe/-/undefsafe-
2.0.5.tgz",
      "integrity": "sha512-
WxONCrssBM8TSPRqN5EmsjVrsv4A8X12J4ArBiiayv3DyyG3ZlIg6yysuuSYdZsVz3TK
cTg2fd//Ujd4CHV1iA==",
    },
    "unpipe": {
      "version": "1.0.0",
      "resolved": "https://registry.npmjs.org/unpipe/-/unpipe-
1.0.0.tgz",
      "integrity": "sha512-
pjy2bYhSsufwWlKwPc+l3cN7+wuJlK6uz0YdJEOlQDbl6jo/YlPi4mb8agUkVC8BF7V8
NuzeyPNqRksA3hztKQ==",
    },
    "util-deprecate": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/util-deprecate/-/util-
deprecate-1.0.2.tgz",
      "integrity": "sha512-
EPD5q1uXyFxJpCrLnCc1nHnq3gOa6DZBocAIiI2TaSCA7VCJ1UJDMagCzIkXNsUYfD1d
aK//LTEQ8xiIbrHtcw==",
    },
    "utils-merge": {
      "version": "1.0.1",
      "resolved": "https://registry.npmjs.org/utils-merge/-/utils-
merge-1.0.1.tgz",
      "integrity": "sha512-
pMZTvIkT1d+TFGvDOqodOclx0QWkkgi6Tdoa8gC8ffGAAqz9pzPTZWAybbsHHoED/ztM
tkv/VoYTYyShUn81hA==",
    },
    "vary": {
      "version": "1.1.2",
      "resolved": "https://registry.npmjs.org/vary/-/vary-
1.1.2.tgz",
      "integrity": "sha512-
BNGbWLfd0eUPabhkXUVm0j8uuvREyTh5ovRa/dyow/BqAbZJyC+5fU+IzQOzmAKzYqYR
AISoRhdQr3eIZ/PXqg==",
```

```
    },
    "wrap-ansi": {
      "version": "7.0.0",
      "resolved": "https://registry.npmjs.org/wrap-ansi/-/wrap-ansi-
7.0.0.tgz",
      "integrity": "sha512-
YVGIj2kamLSTxw6NsZjoBxfSwsn0ycdesmc4p+Q21c5zPuZ1pl+NfxVdxPtdHvmNVOQ6
XSYG4AUtyt/Fi7D16Q==",
      "requires": {
        "ansi-styles": "^4.0.0",
        "string-width": "^4.1.0",
        "strip-ansi": "^6.0.0"
      }
    },
    "wrappy": {
      "version": "1.0.2",
      "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-
1.0.2.tgz",
      "integrity": "sha512-
l4Sp/DRseor9wL6EvV2+TuQn63dMkPjZ/sp9XkghTEbV9KlPS1xUsZ3u7/IQO4wxtcFB
4bgpQPRcR3QCvezPcQ=="
    },
    "xtend": {
      "version": "4.0.2",
      "resolved": "https://registry.npmjs.org/xtend/-/xtend-
4.0.2.tgz",
      "integrity": "sha512-
LKYU1iAXJXUgAXn9URjiu+MWhyUXHsvfp7mcuYm9dSUKK0/CjtrUwFAxD82/mCWbtLsG
jFIad0wIsod4zrTAEQ=="
    },
    "y18n": {
      "version": "5.0.8",
      "resolved": "https://registry.npmjs.org/y18n/-/y18n-
5.0.8.tgz",
      "integrity": "sha512-
0pfFzegeDWJHJIAmTLRP2DwHjdF5s7jo9tuztdQxAhINCdvS+3nGINqPd00AphqJR/0L
hANUS6/+7SCb98YOfA=="
    },
    "yargs": {
      "version": "17.1.1",
      "resolved": "https://registry.npmjs.org/yargs/-/yargs-
17.1.1.tgz",
      "integrity": "sha512-
c2k48R0PwKIqKhPMWjeiF6y2xY/gPMUlro0sgxqXpbOIohWiLNXWslsootttv7E1e73Q
PAMQSg5FeySbVcpsPQ==",
      "requires": {
        "cliui": "^7.0.2",
        "escalade": "^3.1.1",
        "get-caller-file": "^2.0.5",
        "require-directory": "^2.1.1",
        "string-width": "^4.2.0",
        "y18n": "^5.0.5",
        "yargs-parser": "^20.2.2"
      }
    },
    "yargs-parser": {
      "version": "20.2.9",
      "resolved": "https://registry.npmjs.org/yargs-parser/-/yargs-
parser-20.2.9.tgz",
```

```
      "integrity": "sha512-
y11nGElTIV+CT3Zv9t7VKl+Q3hTQoT9a1Qzezhhl6Rp21gJ/IVTW7Z3y9EWXhuUBC2Sh
nf+DX0antecpAwSP8w=="
    }
  }
}
```

28. package.json

```
{
  "name": "product",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "demo": "node app.js",
    "start": "node app.js",
    "live": "nodemon app.js"
  },
  "author": "Shercosta",
  "license": "ISC",
  "dependencies": {
    "body-parser": "^1.20.1",
    "csv": "^6.3.0",
    "csv-generate": "^4.2.6",
    "csvtojson": "^2.0.10",
    "ejs": "^3.1.8",
    "express": "^4.18.2",
    "formidable": "^2.1.1",
    "json-to-csv-export": "^2.1.1",
    "json2csv": "^6.0.0-alpha.2",
    "localtunnel": "^2.0.2",
    "multer": "^1.4.5-lts.1",
    "nodemon": "^2.0.22",
    "papaparse": "^5.4.1",
    "python-shell": "^5.0.0"
  }
}
```

29. setup.py

```
import nltk

nltk.download("punkt")
nltk.download("averaged_perceptron_tagger")
nltk.download("wordnet")
nltk.download("omw-1.4")
nltk.download("stopwords")
```

30. slang.dic

```
{
    "1pun": "satupun",
    "7an": "tujuan",
```

```
    "Dr.": "doktor",
    "dr.": "dokter",
    "drg.": "dokter gigi",
    "Jkt": "Jakarta",
    "Jkw": "jokowi",
    "Napza": "narkoba psikotropika dan zat adiktif",
    "Nasihat": "nasehat",
    "ababil": "abg labil",
    "abis": "habis",
    "acc": "accord",
    "accord": "sesuai",
    "ad": "ada",
    "adl": "adalah",
    "adlah": "adalah",
    "adlh": "adalah",
    "administ": "administrasi",
    "adoh": "aduh",
    "afaik": "as far as i know",
    "agma": "agama",
    "aha": "tertawa",
    "ahaha": "haha",
    "ahiok": "ahok",
    "mehong": "mahal",
    "ahoax": "ahok",
    "ahokncc": "ahok",
    "aing": "saya",
    "aj": "saja",
    "aja": "saja",
    "ajak2": "ajak-ajak",
    "ajep-ajep": "dunia gemerlap",
    "ajj": "saja",
    "ak": "aku",
    "aka": "dikenal juga sebagai",
    "akherat": "akhirat",
    "akhirx": "akhirnya",
    "akika": "aku",
    "akko": "aku",
    "akkoh": "aku",
    "akku": "aku",
    "akn": "akan",
    "aktifis2": "aktivis-aktivis",
    "aktivi": "aktivis",
    "aktivis2": "aktivis-aktivis",
    "akun2": "akun-akun",
    "akuwh": "aku",
    "akyu": "aku",
    "alay": "norak",
    "aleg": "caleg",
    "alih2": "alih-alih",
    "aljasa": "asal jadi saja",
    "almaidahyg": "almaidah",
    "alow": "halo",
    "ama": "sama",
    "ambilin": "ambilkan",
    "ambl": "ambil",
    "amerik": "amarika",
    "anak2": "anak-anak",
    "analisise": "analisis",
    "ancur": "hancur",
```

```
    "angis": "nangis",
    "anjay": "anjing",
    "anjir": "anjing",
    "anjrit": "anjing",
    "ank": "anak",
    "anter": "antar",
    "ap": "apa",
    "ap2": "apa-apa",
    "apaan": "apa",
    "apasih": "apa sih",
    "ape": "apa",
    "apes": "sial",
    "aplot": "unggah",
    "aps": "apa",
    "apva": "apa",
    "aq": "saya",
    "aqu": "aku",
    "aquwh": "aku",
    "as": "amerika serikat",
    "asap": "sesegera mungkin",
    "asbun": "asal bunyi",
    "aseek": "asyik",
    "aseekk": "asyik",
    "asek": "asyik",
    "asekk": "asyik",
    "aseknya": "asyiknya",
    "asem": "asam",
    "asoy": "asyik",
    "aspal": "asli tetapi palsu",
    "astrojim": "astagfirullahaladzim",
    "astul": "asal tulis",
    "ath": "kalau begitu",
    "ato": "atau",
    "ats": "atas",
    "atuh": "kalau begitu",
    "atvsi": "asosiasi televisi swasta indonesia",
    "au": "angkatan udara",
    "au ah": "tidak mau tahu",
    "ava": "avatar",
    "awak": "saya",
    "aws": "awas",
    "ay": "sayang",
    "ayang": "sayang",
    "ayank": "sayang",
    "ayat2": "ayat-ayat",
    "ayok": "ayo",
    "b2": "berdua",
    "b4": "sebelum",
    "bacot": "banyak bicara",
    "bajir": "banjir",
    "bakalan": "akan",
    "bales": "balas",
    "bandes": "bantuan desa",
    "bangdes": "pembangunan desa",
    "bangedh": "banget",
    "bangkotan": "tua",
    "banpol": "bantuan polisi",
    "banpres": "bantuan presiden",
    "banpur": "bantuan tempur",
```

```
    "bansarkas": "bantuan sarana kesehatan",
    "baper": "bawa perasaan",
    "basbang": "basi",
    "bazis": "badan amal, zakat, infak, dan sedekah",
    "bbrp": "beberapa",
    "bcanda": "bercanda",
    "bcoz": "karena",
    "bdaya": "budaya",
    "bdg": "bandung",
    "bdk": "budak",
    "beb": "sayang",
    "bebal": "bodoh",
    "bedakin": "bedak",
    "begajulan": "nakal",
    "bejibun": "banyak",
    "beliin": "belikan",
    "belom": "belum",
    "bem": "badan eksekutif mahasiswa",
    "benarjujur": "jujur",
    "bencong": "banci",
    "bener": "benar",
    "bener2": "bener bener",
    "bentar": "sebentar",
    "ber2": "berdua",
    "ber3": "bertiga",
    "berdikari": "berdiri di atas kaki sendiri",
    "beresin": "membereskan",
    "berhenti2": "berhenti berhenti",
    "bet": "banget",
    "bete": "bosan",
    "beti": "beda tipis",
    "beud": "banget",
    "beut": "banget",
    "bg": "bagi",
    "bgd": "banget",
    "bgm": "bagaimana",
    "bgmn": "bagaimana",
    "bgs": "bagus",
    "bgst": "bangsat",
    "bgt": "banget",
    "bgtu": "begitu",
    "bhai": "bye",
    "bhkan": "bahkan",
    "bhubu": "tidur",
    "bhw": "bahwa",
    "bhy": "bahaya",
    "bijimane": "bagaimana",
    "bilanh": "bilang",
    "bimbuluh": "bimbingan dan penyuluhan",
    "bintal": "bimbingan mental",
    "bisi": "kalau-kalau",
    "bkannya": "bukannya",
    "bkl": "akan",
    "bkmt": "badan kontak majelis taklim",
    "bkn": "bukan",
    "bknnya": "bukannya",
    "bl": "beli",
    "blas": "balas",
    "blegug": "bodoh",
```

```
    "blg": "bilang",
    "blh": "boleh",
    "blm": "belum",
    "bln": "bulan",
    "bls": "balas",
    "blum": "belum",
    "bnchi": "benci",
    "bnci": "benci",
    "bnda": "benda",
    "bngung": "bingung",
    "bnran": "yang benar",
    "bntr": "sebentar",
    "bnyk": "banyak",
    "bodor": "lucu",
    "bohay": "badan aduhai",
    "bokap": "ayah",
    "bokep": "video porno",
    "boker": "buang air besar",
    "bokin": "pacar",
    "bokis": "bohong",
    "bole": "boleh",
    "boljug": "boleh juga",
    "bolot": "bodoh",
    "bonek": "bocah nekat",
    "bonyok": "ayah ibu",
    "bopi": "badan olahraga profesional indonesia",
    "boyeh": "boleh",
    "bpk": "bapak",
    "br": "baru",
    "brb": "segera kembali",
    "brg": "bareng",
    "brhadapan": "berhadapan",
    "brngkt": "berangkat",
    "bro": "saudara laki-laki",
    "brp": "berapa",
    "brsdia": "bersedia",
    "brstu": "bersatu",
    "bru": "baru",
    "brur": "saudara laki-laki",
    "bs": "bisa",
    "bsa": "bisa",
    "bsen": "bosan",
    "bsk": "besok",
    "bsr": "besar",
    "bt": "buat",
    "btol": "botol",
    "btw": "ngomong-ngomong",
    "bu_bu": "tidur",
    "buangbuang": "buang",
    "buaya": "tidak setia",
    "bubarin": "bubarkan",
    "bubbu": "tidur",
    "buber": "buka bersama",
    "bubu": "tidur",
    "bujubune": "luar biasa",
    "bumil": "ibu hamil",
    "bunuh2an": "bunuh-bunuhan",
    "buser": "buru sergap",
    "buzzerbuzzer": "buzzer",
```

```
    "bw": "bawa",
    "bwhn": "bawahan",
    "bwt": "buat",
    "byar": "bayar",
    "byk": "banyak",
    "byr": "bayar",
    "byrin": "bayarkan",
    "c8": "chat",
    "cabal": "sabar",
    "cabut": "pergi",
    "cadas": "keren",
    "caem": "cakep",
    "cakap": "berbicara",
    "calo": "makelar",
    "cama-cama": "sama-sama",
    "can": "belum",
    "cangcut": "celana dalam",
    "capcus": "pergi",
    "cape": "capek",
    "capek": "lelah",
    "caper": "cari perhatian",
    "capresss": "calon presiden",
    "caur": "jelek",
    "cayalah": "percaya",
    "cb": "coba",
    "cc": "carbon copy",
    "ce": "cewek",
    "cekak": "tidak ada uang",
    "cekal": "cegah tangkal",
    "cekidot": "coba lihat",
    "cemen": "penakut",
    "cemplungin": "cemplungkan",
    "cengengesan": "tertawa",
    "cepat2": "cepat cepat",
    "ceper": "pendek",
    "cepet": "cepat",
    "ceu": "kakak perempuan",
    "cew": "cewek",
    "cewe": "cewek",
    "chuyunk": "sayang",
    "cibuk": "sibuk",
    "cimeng": "ganja",
    "cin": "cinta",
    "cipika cipiki": "cium pipi kanan cium pipi kiri",
    "ciye": "cie",
    "ciyh": "sih",
    "cjr": "coboy junior",
    "ckck": "ck",
    "ckepp": "cakep",
    "ckp": "cukup",
    "clbk": "cinta lama bersemi kembali",
    "cm": "cuma",
    "cmiiw": "correct me if i'm wrong",
    "cmpr": "campur",
    "cmpur": "campur",
    "cnenk": "senang",
    "cong": "banci",
    "congor": "mulut",
    "conlok": "cinta lokasi",
```

```
    "cos": "cause",
    "cow": "cowok",
    "cowwyy": "maaf",
    "coz": "karena",
    "cp": "siapa",
    "cpa": "siapa",
    "cpe": "capek",
    "cppa": "siapa",
    "cppe": "capek",
    "cr": "cara",
    "crny": "caranya",
    "cth": "contoh",
    "cucok": "cocok",
    "cuek": "tidak acuh",
    "cuex": "cuek",
    "cuman": "Cuma",
    "cumi": "Cuma miscall",
    "cungkring": "kurus tinggi",
    "cups": "culun",
    "cupu": "culun",
    "curanmor": "pencurian kendaraan bermotor",
    "curas": "pencurian dengan kekerasan",
    "curcol": "curahan hati colongan",
    "curhat": "curahan hati",
    "cust": "customer",
    "cwek": "cewek",
    "cwo": "cowo",
    "cwok": "cowok",
    "cyin": "cinta",
    "cz": "karena",
    "d": "di",
    "dagh": "sudah",
    "dah": "sudah",
    "damang": "sehat",
    "dapet": "dapat",
    "dateng": "datang",
    "de": "adik",
    "dech": "deh",
    "dek": "adik",
    "deket": "dekat",
    "demen": "suka",
    "denger": "dengar",
    "dengerin": "dengarkan",
    "deyh": "deh",
    "dg": "dengan",
    "dgn": "dengan",
    "dgr": "dengar",
    "dh": "sudah",
    "diancurin": "dihancurkan",
    "die": "dia",
    "diem2": "diam diam",
    "digosipin": "digosipkan",
    "dikti": "direktoral jendral pendidikan tinggi",
    "dimaafin": "dimaafkan",
    "dimana2": "dimana-mana",
    "dimintak": "diminta",
    "dimna": "dimana",
    "direhap": "direhab",
    "disono": "di sana",
```

```
    "diuber": "dikejar",
    "dket": "dekat",
    "dki": "daerah khusus ibukota",
    "dkit": "sedikit",
    "dkk": "dan kawan-kawan",
    "dl": "dulu",
    "dll": "dan lain-lain",
    "dlm": "dalam",
    "dlu": "dulu",
    "dmaksud": "dimaksud",
    "dmn": "dimana",
    "dn": "dan",
    "dngn": "dengan",
    "doang": "saja",
    "dodol": "bodoh",
    "doi": "dia",
    "doku": "uang",
    "donat": "donut",
    "doughnat": "donut",
    "don juan": "playboy",
    "dongok": "bodoh",
    "dongs": "dong",
    "donk": "dong",
    "donlot": "unduh",
    "dpr": "dewan perwakilan rakyat",
    "dprd": "dewan perwakilan rakyat daerah",
    "dpt": "dapat",
    "dr": "dari",
    "dri": "dari",
    "drmana": "darimana",
    "drmn": "darimana",
    "drpd": "daripada",
    "drtd": "dari tadi",
    "dsb": "dan sebagainya",
    "dst": "dan seterusnya",
    "dstu": "disitu",
    "dtg": "datang",
    "dugem": "dunia gemerlap",
    "duh": "aduh",
    "duit": "uang",
    "dulu": "dahulu",
    "dungu": "bodoh",
    "duren": "durian",
    "dy": "dia",
    "dzolim": "zalim",
    "ed": "edisi",
    "edan": "gila",
    "ego": "emosi",
    "egp": "emang gue pikirin",
    "eike": "aku",
    "eke": "aku",
    "elo": "kamu",
    "elu": "kamu",
    "emang": "memang",
    "emangnya": "memangnya",
    "emank": "memang",
    "emg": "memang",
    "emng": "memang",
    "emte": "makan teman",
```

```
    "endak": "tidak",
    "engga": "tidak",
    "enggak": "tidak",
    "entar": "nanti",
    "envy": "iri",
    "eps": "episode",
    "etc": "et cetera",
    "ex": "mantan",
    "ext": "extension",
    "fahami": "pahami",
    "fak": "fakultas",
    "fakta2": "fakta-fakta",
    "faktakta": "fakta",
    "fax": "facsimile",
    "fb": "facebook",
    "fdsi": "forum diskusi suporter indonesia",
    "fifo": "first in first out",
    "finek": "finansial ekonomi",
    "fisip": "fakultas ilmu sosial dan ilmu politik",
    "fitnag": "fitnah",
    "folbek": "follow back",
    "fpi": "front pembela islam",
    "fwd": "diterukan",
    "fyi": "sebagai informasi",
    "g": "tidak",
    "ga": "tidak",
    "gaada": "tidak ada uang",
    "gabisa": "tidak bisa",
    "gag": "tidak",
    "gahar": "sangar",
    "gaje": "tidak jelas",
    "gak": "tidak",
    "gak papa": "tidak apa-apa",
    "gamau": "tidak mau",
    "gan": "juragan",
    "gantengbs": "ganteng",
    "gantengsaya": "ganteng",
    "gapapa": "tidak apa-apa",
    "gaptek": "gagap teknologi",
    "gatau": "tidak tahu",
    "gatek": "gagap teknologi",
    "gatot": "gagal total",
    "gawe": "kerja",
    "gax": "tidak",
    "gbs": "tidak bisa",
    "gd": "gusdur",
    "gda": "tidak ada uang",
    "gebetan": "orang yang disuka",
    "geer": "gede rasa",
    "gegara": "gara-gara",
    "geje": "tidak jelas",
    "gelo": "gila",
    "gene": "begini",
    "gepeng": "gelandangan dan pengemis",
    "ghiey": "lagi",
    "ghiy": "lagi",
    "gi": "lagi",
    "gib": "gerakan indonesia beradab",
    "gile": "gila",
```

```
    "giman": "gigi mancung",
    "gimana": "bagaimana",
    "gini": "begini",
    "gingerbread": "ginger bread",
    "gino": "gigi nongol",
    "gitchu": "gitu",
    "githu": "gitu",
    "gitu": "begitu",
    "gj": "tidak jelas",
    "gk": "tidak",
    "gmana": "bagaimana",
    "gmn": "bagamana",
    "gn": "begini",
    "goblog": "goblok",
    "goblok": "bodoh",
    "gokil": "gila",
    "golongan2": "golongan-golongan",
    "golput": "golongan putih",
    "gombal": "suka merayu",
    "good": "bagus",
    "governor": "gubernur",
    "gowes": "mengayuh sepeda",
    "gpl": "tidak pakai lama",
    "gpny": "tidak punya",
    "gpp": "tidak apa-apa",
    "gr": "gede rasa",
    "gretong": "gratis",
    "gretongan": "gratisan",
    "gt": "gitu",
    "gtau": "tidak tahu",
    "gtw": "tidak tahu",
    "gua": "saya",
    "gub": "gubernur",
    "gue": "saya",
    "guna": "menggunakan",
    "guoblok": "goblok",
    "guys": "teman-teman",
    "gw": "saya",
    "gws": "cepat sembuh",
    "ha": "tertawa",
    "habis2an": "habis-habisan",
    "haghaghag": "tertawa",
    "haha": "tertawa",
    "hak2": "hak hak",
    "hakhak": "tertawa",
    "hallow": "halo",
    "ham": "hak asasi manusia",
    "handak": "bahan peledak",
    "hankam": "pertahanan dan keamanan",
    "hansip": "pertahanan sipil",
    "happy": "bahagia",
    "hardcore": "keras",
    "hehe": "he",
    "hellow": "halo",
    "helo": "halo",
    "helow": "halo",
    "heter": "pembenci",
    "heters": "pembenci",
    "hey": "hai",
```

```
    "hi": "hai",
    "hlm": "halaman",
    "hlng": "hilang",
    "hny": "hanya",
    "hnya": "hanya",
    "hoax": "isu bohong",
    "hot2nya": "hot-hotnya",
    "houm": "rumah",
    "hr": "hari",
    "hrs": "harus",
    "hrus": "harus",
    "hubad": "hubungan angkatan darat",
    "hubdar": "perhubungan darat",
    "hubla": "perhubungan laut",
    "huff": "mengeluh",
    "huft": "mengeluh",
    "hum": "rumah",
    "humas": "hubungan masyarakat",
    "humoris": "lucu",
    "humz": "rumah",
    "hurts": "pedih",
    "icecream": "ice cream",
    "icecreamsandwich": "ice cream sandwich",
    "ide2": "ide-ide",
    "idk": "saya tidak tahu",
    "idup": "hidup",
    "ikon": "lambang",
    "ilang": "hilang",
    "ilfeel": "tidak suka",
    "ilfil": "tidak suka",
    "imba": "jago sekali",
    "imho": "in my humble opinion",
    "imoet": "imut",
    "imoetz": "imut",
    "indinesia": "indonesia",
    "indo": "indonesia",
    "indon": "Indonesia",
    "info": "informasi",
    "insom": "insomnia",
    "ipb": "institut pertanian bogor",
    "irit": "hemat",
    "isengin": "bercanda",
    "ista": "nista",
    "istaa": "nista",
    "isu2": "isu-isu",
    "item": "hitam",
    "itj": "Indonesia tanpa jil",
    "ito": "itu",
    "itu2": "itu-itu",
    "itung": "hitung",
    "itungan": "hitungan",
    "iy": "iya",
    "iyala": "iya lah",
    "iye": "iya",
    "iyo": "iya",
    "ja": "saja",
    "jabar": "jawa barat",
    "jablay": "jarang dibelai",
    "jadiin": "jadikan",
```

```
    "jadul": "jaman dulu",
    "jaim": "jaga image",
    "jan": "jangan",
    "jancuk": "anjing",
    "janjinyaa": "janji",
    "jargon2": "jargon jargon",
    "jatim": "jawa timur",
    "jatohin": "menjatuhkan",
    "jawapan": "jawaban",
    "jayus": "tidak lucu",
    "jd": "jadi",
    "jdi": "jadi",
    "jdikan": "jadikan",
    "je": "saja",
    "jelas2": "jelas-jelas",
    "jem": "jam",
    "jentelmen": "berani",
    "jg": "juga",
    "jga": "juga",
    "jgk": "juga",
    "jgn": "jangan",
    "jgnkan": "jangankan",
    "ji2k": "jijik",
    "jijay": "jijik",
    "jil": "jaringan islam liberal",
    "jir": "anjing",
    "jjk": "jijik",
    "jk": "jika",
    "jkt": "jakarta",
    "jln": "jalan",
    "jng": "jangan",
    "jnj": "janji",
    "joget2": "joget-joget",
    "jomblo": "tidak punya pacar",
    "jth": "jatuh",
    "jual2": "jual-jual",
    "jubir": "juru bicara",
    "jurdil": "jujur adil",
    "jutek": "galak",
    "jwb": "jawab",
    "k": "ke",
    "ka": "kakak",
    "kab": "kabupaten",
    "kabag": "kepala bagian",
    "kabor": "kabur",
    "kacian": "kasihan",
    "kacrut": "kacau",
    "kadang2": "kadang kadang",
    "kadit": "kepala direktorat",
    "kadiv": "kepala divisi",
    "kaga": "tidak",
    "kagak": "tidak",
    "kaka": "kakak",
    "kalo": "kalau",
    "kaltim": "kalimantan timur",
    "kampret": "sialan",
    "kamtib": "keamanan dan ketertiban",
    "kamtibmas": "keamanan dan ketertiban masyarakat",
    "kamuh": "kamu",
```

```
    "kamuwh": "kamu",
    "kamyu": "kamu",
    "kan": "akan",
    "kanwil": "kantor wilayah",
    "kapolri": "Kepala Polisi Republik Indonesia",
    "kapt": "kapten",
    "karna": "karena",
    "kasubbag": "kepala subbagian",
    "kasubbid": "kepala subbidang",
    "kata2": "kata-kata",
    "katrok": "kampungan",
    "kau": "kamu",
    "kawan2": "kawan-kawan",
    "kayanya": "kayaknya",
    "kbar": "kabar",
    "kbr": "kabar",
    "kcian": "kasihan",
    "kdng": "kadang",
    "kdu": "harus",
    "ke": "kah",
    "kebranian": "keberanian",
    "keburu": "terlanjur",
    "kec": "kecamatan",
    "kedubes": "kedutaan besar",
    "kejar2": "kejar-kejar",
    "kejurnas": "kejuaraan nasional",
    "kek": "seperti",
    "kekeuh": "keras kepala",
    "keknya": "kayanya",
    "kel": "kelurahan",
    "keliatan": "kelihatan",
    "kemaren": "kemarin",
    "kemenag": "kementrian agama",
    "kemenkominfo": "kementrian komunikasi dan informatika",
    "kemrin": "kemarin",
    "keneh": "masih",
    "kentot": "senggama",
    "kepengen": "mau",
    "kepikiran": "terpikirkan",
    "kepingin": "mau",
    "kepo": "mau tahu urusan orang",
    "kepsek": "kepala sekolah",
    "kere": "tidak punya uang",
    "kesbang": "kesatuan bangsa",
    "kesian": "kasihan",
    "kesihatan": "kesehatan",
    "kesra": "kesejahteraan rakyat",
    "ketauan": "ketahuan",
    "ketrima": "diterima",
    "ketum": "ketua umum",
    "keukeuh": "keras kepala",
    "kgiatan": "kegiatan",
    "khan": "kan",
    "kibul": "bohong",
    "kibus": "kaki busuk",
    "kimpoi": "kawin",
    "kk": "kakak",
    "kl": "kalau",
    "klian": "kalian",
```

```
    "klianz": "kalian",
    "klmpk": "kelompok",
    "klo": "kalau",
    "kloter": "kelompok terbang",
    "kls": "kelas",
    "kluarga": "keluarga",
    "klw": "kalau",
    "klwrga": "keluarga",
    "km": "kamu",
    "kmaksiatan": "kemaksiatan",
    "kmari": "kemari",
    "kmna": "kemana",
    "kmps": "kampus",
    "kmpus": "kampus",
    "kmrm": "kemeramen",
    "kmrn": "kemarin",
    "kn": "kan",
    "knal": "kenal",
    "kne": "terlibat",
    "knl": "kenal",
    "knp": "kenapa",
    "knpa": "kenapa",
    "koar": "teriak",
    "kodya": "kota madya",
    "kog": "kok",
    "komdis": "komisi disiplin",
    "kompi": "komputer",
    "komsov": "komunis sovyet",
    "komtiong": "komunis Tiongkok",
    "komun": "komunitas",
    "kongkow": "kumpul bareng teman-teman",
    "konjen": "konsulat jenderal",
    "konpers": "konferensi pers",
    "kopdar": "kopi darat",
    "koq": "kok",
    "korup": "korupsi",
    "kowe": "kamu",
    "kpai": "komisi perlindungan anak indonesia",
    "kpd": "kepada",
    "kpid": "komisi penyiaran indonesia daerah",
    "kpk": "komisi pemberantas koruptor",
    "kpn": "kapan",
    "kptsan": "keputusan",
    "krenz": "keren",
    "krik": "garing",
    "krim": "kirim",
    "krm": "kirim",
    "krn": "karena",
    "ksadaran": "kesadaran",
    "kt": "kita",
    "ktauan": "ketahuan",
    "ktipu": "tipu",
    "ktk": "ketika",
    "ktmu": "ketemu",
    "ktny": "katanya",
    "ktp": "kartu tanda penduduk",
    "ktr": "kantor",
    "kudu": "harus",
    "kuper": "kurang pergaulan",
```

```
    "kuq": "kok",
    "kurng": "kurang",
    "kw": "imitasi",
    "kwn": "kawan",
    "ky": "seperti",
    "kyk": "seperti",
    "kykny": "kayanya",
    "la": "lah",
    "laka": "kecelakaan",
    "laki2": "laki-laki",
    "lam": "salam",
    "lambreta": "lambat",
    "lamp": "lampiran",
    "langi": "langit",
    "lansia": "lanjut usia",
    "lanud": "landasan udara",
    "lapas": "lembaga pemasyarakatan",
    "larang2": "larang-larang",
    "latgab": "latihan gabungan",
    "lawyer2": "lawyer lawyer",
    "lbh": "lebih",
    "lbur": "libur",
    "lebay": "berlebihan",
    "leh": "boleh",
    "lekong": "laki-laki",
    "lelet": "lambat",
    "lemot": "lemah otak",
    "lesbia": "lesbian",
    "less": "kurang",
    "lg": "lagi",
    "lgi": "lagi",
    "lgkp": "lengkap",
    "lgsg": "langsung",
    "lha": "lah",
    "lht": "lihat",
    "liat": "lihat",
    "light": "terang",
    "linmas": "perlindungan masyarakat",
    "litbang": "penelitian dan pengembangan",
    "liwat": "lewat",
    "lknt": "laknat",
    "lmyan": "lumayan",
    "lmyn": "lumayan",
    "lngkp": "lengkap",
    "lo": "kamu",
    "loch": "loh",
    "loe": "kamu",
    "lol": "tertawa",
    "lola": "lambat berfikir",
    "lom": "belum",
    "louph": "cinta",
    "loupz": "cinta",
    "low": "kalau",
    "lowh": "kamu",
    "lp": "lupa",
    "lsm": "lembaga swadaya masyarakat",
    "lu": "kamu",
    "luber": "langsung, umum, bebas, dan rahasia",
    "luchu": "lucu",
```

```
    "luchuw": "lucu",
    "luff": "cinta",
    "lum": "belum",
    "luph": "cinta",
    "luthu": "lucu",
    "luv": "love",
    "lw": "kamu",
    "lwn": "lawan",
    "lwt": "lewat",
    "lyf": "live",
    "ma": "sama",
    "maacih": "terima kasih",
    "maaciw": "terima kasih",
    "mabal": "bolos",
    "mabes": "markas besar",
    "macem": "macam",
    "macem-macem": "macam-macam",
    "macet2an": "macet-macetan",
    "macih": "masih",
    "madesu": "masa depan suram",
    "maem": "makan",
    "maen": "main",
    "magabut": "makan gaji buta",
    "mahatma": "maju sehat bersama",
    "maho": "homo",
    "majority": "mayoritas",
    "mak": "ibu",
    "mak jang": "kaget",
    "makasih": "terima kasih",
    "maksain": "memaksa",
    "malah": "bahkan",
    "malem": "malam",
    "malu2in": "memalukan",
    "mam": "makan",
    "mamz": "makan",
    "mana2": "mana-mana",
    "maneh": "kamu",
    "manen": "panen",
    "manggil": "panggil",
    "manies": "manis",
    "maniez": "manis",
    "mantabss bagus": "",
    "mantep": "mantap",
    "mao": "mau",
    "maren2": "kemarin-kemarin",
    "markus": "makelar kasus",
    "masama": "sama - sama",
    "masing2": "masing-masing",
    "masukin": "masukkan",
    "masy": "masyarakat",
    "mati2": "mati-mati",
    "maybe": "mungkin",
    "mb": "mbak",
    "mba": "mbak",
    "mbabibuta": "ceroboh",
    "mbahas": "membahas",
    "mbaur": "baur",
    "mbelain": "belain",
    "mbual": "membual",
```

```
    "mcm": "macam",
    "mdh": "mudah",
    "mdh2an": "mudah-mudahan",
    "medsos": "media sosial",
    "melayakni": "melayani",
    "melu": "ikut",
    "mencarut2": "mencarut",
    "mending": "lebih baik",
    "mendpt": "mendapat",
    "menjdi": "menjadi",
    "menklarifikasi": "mengklarifikasi",
    "menolak": "tolak",
    "menpora": "menteri pemuda dan olahraga",
    "menristek": "mentri riset, teknologi dan pendidikan",
    "mensos": "menteri sosial",
    "mepet": "dekat sekali",
    "mesjid": "masjid",
    "met": "selamat",
    "metro": "metropolitan",
    "mga": "moga",
    "mgkin": "mungkin",
    "mgkn": "mungkin",
    "mgu": "minggu",
    "mhn": "mohon",
    "migas": "minyak dan gas bumi",
    "miker": "minuman keras",
    "mikol": "minuman beralkohol",
    "milis": "mailing list",
    "miras": "minuman keras",
    "miumi": "majelis intelektual dan ulama muda indonesia",
    "mjadikan": "menjadikan",
    "mjd": "menjadi",
    "mksd": "maksud",
    "mlah": "malah",
    "mlalui": "melalui",
    "mlm": "malam",
    "mls": "malas",
    "mmbuat": "membuat",
    "mmg": "memang",
    "mmng": "memang",
    "mndukung": "mendukung",
    "mngajarkan": "mengajarkan",
    "mngkn": "mungkin",
    "mninggal": "meninggal",
    "mnipu": "tipu",
    "mnjd": "menjadi",
    "mnolak": "tolak",
    "mnrtmu": "menurutmu",
    "mnt": "minta",
    "mnyumbang": "menyumbang",
    "mo": "mau",
    "modar": "mati",
    "moga": "semoga",
    "moge": "motor gede",
    "mokad": "mati",
    "mokat": "mati",
    "moso": "masa",
    "mosok": "masa",
    "most": "paling",
```

```
    "mpe": "sampai",
    "mpr": "majelis permusyawaratan rakyat",
    "mrk": "mereka",
    "msh": "masih",
    "msk": "masuk",
    "mskpn": "meskipun",
    "mslh": "masalah",
    "msng2": "masing-masing",
    "msuh": "musuh",
    "mt": "makan teman",
    "muahal": "mahal",
    "mubes": "musyawarah besar",
    "mui": "majelis ulama indonesia",
    "muker": "musyawarah kerja",
    "mulu": "melulu",
    "mumet": "pusing",
    "mumpung": "selagi",
    "muna": "munafik",
    "munas": "musyawarah nasional",
    "munaslub": "musyawarah nasional luar biasa",
    "munfik": "munafik",
    "muntaber": "muntah dan berak",
    "musda": "musyawarah daerah",
    "musti": "mesti",
    "muter2": "muter-muter",
    "muup": "maaf",
    "muupz": "maaf",
    "muuv": "maaf",
    "mw": "now watching",
    "mz": "mas",
    "n": "dan",
    "na": "nya",
    "naas": "nahas",
    "nak": "hendak",
    "nal": "kenal",
    "nan": "dan",
    "nanam": "menanam",
    "nangis": "menangis",
    "nanya": "bertanya",
    "naon": "apa",
    "napa": "kenapa",
    "napi": "narapidana",
    "napol": "narapidana politik",
    "napza": "narkotika, alkohol, psikotropika, dan zat adiktif",
    "naq": "anak",
    "narkoba": "narkotika, psikotropika, dan obat terlarang",
    "narsis": "bangga pada diri sendiri",
    "nasgor": "nasi goreng",
    "nax": "anak",
    "nda": "tidak",
    "ndak": "tidak",
    "ndiri": "sendiri",
    "ndut": "gendut",
    "ne": "ini",
    "negri": "negeri",
    "nekolim": "neokolonialisme",
    "nekolin": "neokolonialisme",
    "nelfon": "menelepon",
    "nembak": "menyatakan cinta",
```

```
    "next": "lanjut",
    "ngabis2in": "menghabiskan",
    "ngabuburit": "menunggu berbuka puasa",
    "ngakak": "tertawa",
    "ngaku": "mengaku",
    "ngambek": "marah",
    "ngambil": "mengambil",
    "ngampus": "pergi ke kampus",
    "ngamuk2": "ngamuk-ngamuk",
    "nganggur": "tidak punya pekerjaan",
    "ngantri": "mengantri",
    "ngapah": "kenapa",
    "ngapain": "sedang apa",
    "ngaret": "terlambat",
    "ngaruh": "berpengaruh",
    "ngasih": "memberikan",
    "ngawur": "berbicara sembarangan",
    "ngebandel": "berbuat bandel",
    "ngebully": "intimidasi",
    "ngeceng": "kumpul bareng-bareng",
    "ngegosip": "bergosip",
    "ngeh": "sadar",
    "ngehe": "kesal",
    "ngeklaim": "mengklaim",
    "ngekos": "tinggal di kos",
    "ngeksis": "menjadi eksis",
    "ngelamar": "melamar",
    "ngeles": "berkilah",
    "ngeliat": "melihat",
    "ngelidur": "menggigau",
    "ngemeng": "bicara terus-terusan",
    "ngentot": "senggama",
    "ngentotin": "senggama",
    "ngerampok": "merampok",
    "ngeri": "seram",
    "ngerti": "mengerti",
    "ngga": "tidak",
    "nggak": "tidak",
    "ngibul": "berbohong",
    "ngikut": "ikut",
    "ngiler": "mau",
    "nginep": "menginap",
    "ngiri": "iri",
    "ngisi": "mengisi",
    "ngisiin": "mengisikan",
    "ngmg": "bicara",
    "ngmng": "bicara",
    "ngn": "dengan",
    "ngo": "non governmental organization",
    "ngocol": "lucu",
    "ngomong": "bicara",
    "ngomongin": "membicarakan",
    "ngubek2": "mencari-cari",
    "ngumpet": "sembunyi",
    "ngumpul": "berkumpul",
    "ngurus": "mengurus",
    "ngusir2": "usir-usir",
    "nh": "nih",
    "ni": "ini",
```

```
    "nice": "bagus",
    "nie": "ini",
    "niech": "nih",
    "nih": "ini",
    "nikahkah": "nikahkan",
    "nimbrung": "ikut serta",
    "nipu": "tipu",
    "nitizen": "netizen",
    "niyh": "nih",
    "nk": "ingin",
    "nkah": "nikah",
    "nmenarik": "menarik",
    "nmr": "nomor",
    "nnt": "nanti",
    "nnti": "nanti",
    "nntn": "nonton",
    "no": "nomor",
    "nobar": "nonton bareng",
    "nongkrong": "kumpul-kumpul",
    "np": "now playing",
    "nrp": "narapidana",
    "ntap": "mantap",
    "ntar": "sebentar",
    "ntn": "nonton",
    "nu": "nahdatul ulama",
    "nuhun": "terima kasih",
    "numpuk": "bertumpuk",
    "nunggu": "menunggu",
    "nutupin": "menutupi",
    "ny": "nya",
    "nyaingin": "saing",
    "nyalahin": "salah",
    "nyampah": "membuat sampah",
    "nyampyr": "nyampur",
    "nyang": "yang",
    "nyari": "mencari",
    "nyariin": "mencari",
    "nyasar": "tersesat",
    "nyekar": "menyekar",
    "nyela": "cela",
    "nyembah": "sembah",
    "nyemot": "ambil",
    "nyerang": "serang",
    "nyiapin": "mempersiapkan",
    "nyicil": "mencicil",
    "nyiram": "menyiram",
    "nympr": "nyampur",
    "nyoblos": "mencoblos",
    "nyok": "ayo",
    "nyokap": "ibu",
    "o": "oleh",
    "o/": "oleh",
    "odmk": "orang dengan masalah kejiwaan",
    "ogah": "tidak mau",
    "oia": "oh iya",
    "ok": "ok",
    "okt": "oktober",
    "ol": "online",
    "omdo": "omong doang",
```

```
    "omg": "oh my god",
    "omong2": "ngomong-ngomong",
    "ongkir": "ongkos kirim",
    "oon": "bodoh",
    "oot": "out of topic",
    "orange": "oranye",
    "org": "orang",
    "org2": "orang-orang",
    "orglaen": "orang lain",
    "orkes": "olahraga dan kesehatan",
    "ortu": "orang tua",
    "ospek": "orientasi studi dan pengenalan kampus",
    "otda": "otonomi daerah",
    "otre": "ok",
    "otw": "on the way, sedang di jalan",
    "our": "kami",
    "p": "per",
    "p.s": "pesan",
    "p0rn": "porno",
    "pa": "apa",
    "pacal": "pacar",
    "pahe": "paket hemat",
    "pake": "pakai",
    "pakek": "pakai",
    "pala": "kepala",
    "panja": "panitia kerja",
    "pansus": "panitia khusus",
    "pantura": "pantai utara",
    "paranoid": "takut",
    "parno": "takut",
    "parpol": "partai politik",
    "parte": "partai",
    "pas2an": "pas-pasan",
    "pasutri": "pasangan suami istri",
    "patas": "cepat dan terbatas",
    "pd": "pada",
    "pdhl": "padahal",
    "pdip": "partai demokrasi Indonesia perjuangan",
    "pdskji": "perhimpunan dokter spesialis kedokteran jiwa
indonesia",
    "pdt": "padat",
    "pecahan2": "pecahan-pecahan",
    "pede": "percaya diri",
    "pelanggarab": "pelanggaran",
    "pelatda": "pemusatan latihan daerah",
    "pelatnas": "pemusatan latihan nasional",
    "pelcehn": "pelecehan",
    "pembokat": "pembantu",
    "pemda": "pemerintah daerah",
    "pemilu": "pemilihan umum",
    "pemko": "pemerintah kota",
    "pemkot": "pemerintah kota",
    "pemprov": "pemerintah provinsi",
    "pemred": "pemimpin redaksi",
    "pendukung2": "pendukung-pendukung",
    "penertiban": "tertib",
    "pengen": "ingin",
    "penjas": "pendidikan jasmani",
    "penmas": "pendidikan masyarakat",
```

```
    "perda": "peraturan daerah",
    "perek": "wanita tuna susila",
    "peremp": "perempuan",
    "pergub": "peraturan gubernur",
    "perhatiin": "perhatikan",
    "pertamax": "pertama",
    "pesenan": "pesanan",
    "pesenin": "pesankan",
    "pgang": "pegang",
    "pgi": "pergi",
    "pi": "tapi",
    "pihak2": "pihak pihak",
    "pijetin": "pijitkan",
    "pilgub": "pemilihan gubernur",
    "pilkada": "pemilihan kepala daerah",
    "pimpro": "pemimpin proyek",
    "pinter2": "pintar-pintar",
    "pisan": "sangat",
    "pisgor": "pisang goreng",
    "pk": "penjahat kelamin",
    "pke": "pakai",
    "pket": "paket",
    "pkr": "pakar",
    "pks": "partai keadilan sejahtera",
    "plg": "paling",
    "plh": "pilih",
    "pls": "tolong",
    "pm": "private message",
    "pmhaman": "pemahaman",
    "pmrnth": "pemerintah",
    "pndkung": "pendukung",
    "pndukung": "pendukung",
    "pnjilat": "penjilat",
    "pns": "pegawai negeri sipil",
    "pny": "punya",
    "poin2": "poin-poin",
    "polantas": "polisi lalu lintas",
    "polda": "kepolisian daerah",
    "polres": "kepolisian resor",
    "polwan": "polisi wanita",
    "pon": "juga",
    "ponpes": "pondok pesantren",
    "ponsel": "telepon seluler",
    "poster2": "poster-poster",
    "pp": "pulang pergi",
    "ppa": "apa",
    "presh": "segar",
    "prg": "pergi",
    "prikitiw": "cie",
    "priksa": "periksa",
    "prilaku": "perilaku",
    "prnh": "pernah",
    "pro": "profesional",
    "problem": "masalah",
    "prototype": "contoh",
    "prov": "provinsi",
    "psen": "pesan",
    "psn": "pesan",
    "psr": "pasar",
```

```
    "pst": "pasti",
    "psti": "pasti",
    "pswt": "pesawat",
    "ptemuan": "pertemuan",
    "puanas": "panas",
    "pulgar": "kasar",
    "puskesmas2": "puskesmas-puskesmas",
    "puyeng": "pusing",
    "pw": "posisi nyaman",
    "qmo": "kamu",
    "qmu": "kamu",
    "qt": "kita",
    "rakor": "rapat koordinasi",
    "rame": "ramai",
    "rang": "orang",
    "ranmor": "kendaraan bermotor",
    "raskin": "rakyat miskin",
    "rata2": "rata-rata",
    "re": "reply",
    "red": "redaksi",
    "ref": "referensi",
    "reg": "register",
    "rehab": "rehabilitasi",
    "rejeki": "rezeki",
    "rekan2": "rekan-rekan",
    "reklamuk": "reklamasi",
    "rempah2": "rempah-rempah",
    "rempong": "sulit",
    "renstra": "rencana strategis",
    "repp": "balas",
    "reskrim": "reserse kriminal",
    "restik": "reserse narkotika",
    "rhs": "rahasia",
    "rights": "benar",
    "rja": "raja",
    "rkyt": "rakyat",
    "rmbt": "rambut",
    "rmh": "rumah",
    "rofl": "rolling on the floor laughing",
    "ru": "baru",
    "rubahnn": "rubah",
    "rudal": "peluru kendali",
    "ruko": "rumah toko",
    "rusun": "rumah susun",
    "rusunawa": "rumah susun sewa",
    "rutan": "rumah tahanan",
    "ruu": "rancangan undang-undang",
    "ruz": "terus",
    "rwt": "rawat",
    "s4": "sempat",
    "sadaar": "sadar",
    "sah2": "sah-sah",
    "saia": "saya",
    "saiank": "sayang",
    "salting": "salah tingkah",
    "saltum": "salah kostum",
    "sampe": "sampai",
    "sampeyan": "anda",
    "samsek": "sama sekali",
```

```
    "sapa": "siapa",
    "sapose": "siapa",
    "satkorlak": "satuan koordinator pelaksana",
    "satpam": "satuan pengamanan",
    "say": "sayang",
    "sayan": "sayang",
    "sbaliknya": "sebaliknya",
    "sbb": "sebagai berikut",
    "sbg": "sebagai",
    "sbgn": "sebagian",
    "sbh": "sebuah",
    "sblm": "sebelum",
    "sbnr": "sebenar",
    "sbnrny": "sebenarnya",
    "sbntr": "sebentar",
    "scara": "secara",
    "scr": "secara",
    "sdg": "sedang",
    "sdgkn": "sedangkan",
    "sdh": "sudah",
    "sdkt": "sedikit",
    "sdngkn": "sedangkan",
    "se7": "setuju",
    "seabrek": "banyak",
    "sebelas dua belas": "mirip",
    "sebelom": "sebelum",
    "sebgmn": "sebagaimana",
    "sebuk": "sibuk",
    "sekjen": "sekretaris jenderal",
    "selaen": "selain",
    "selmt": "selamat",
    "semau": "asal",
    "sembako": "sembilan bahan pokok",
    "sempak": "celana dalam",
    "sempet": "sempat",
    "sempet2nya": "sempat-sempatnya",
    "sendratari": "seni drama tari",
    "seorg": "seorang",
    "sept": "september",
    "serasiii": "serasi",
    "sertijab": "serah terima jabatan",
    "sesok": "besok",
    "seting": "akting",
    "setting": "atur",
    "sgl": "segala",
    "sgrc": "support group and resource center",
    "sgt": "sangat",
    "sgtunya": "segitunya",
    "shg": "sehingga",
    "shrsnya": "seharusnya",
    "siap2": "siap-siap",
    "sidak": "inspeksi mendadak",
    "siech": "sih",
    "sieyh": "sih",
    "sigh": "sih",
    "sik": "sih",
    "sikon": "situasi dan kondisi",
    "simiskin": "miskin",
    "simpedes": "simpanan pedesaan",
```

```
    "sinetron": "sinema elektronik",
    "sip": "ok",
    "siramin": "siramkan",
    "sis": "kakak perempuan",
    "sj": "saja",
    "sjk": "sejak",
    "skali": "sekali",
    "skalian": "sekalian",
    "skl": "sekali",
    "sklh": "sekolah",
    "skrg": "sekarang",
    "skt": "sakit",
    "skul": "sekolah",
    "slalu": "selalu",
    "slama": "selama",
    "slesai": "selesai",
    "slh": "salah",
    "sll": "selalu",
    "sllu": "selalu",
    "slm": "salam",
    "slma": "selama",
    "slmt": "selamat",
    "slsai": "selesai",
    "slt": "sulit",
    "sm": "sama",
    "sma": "sama",
    "smg": "semoga",
    "smkn": "semakin",
    "smpt": "sempat",
    "smua": "semua",
    "smw": "semua",
    "smwny": "semuanya",
    "sndiri": "sendiri",
    "sng": "senang",
    "sni": "sini",
    "soft": "halus",
    "sohib": "teman",
    "sokong": "dukung",
    "soljum": "sholat jumat",
    "solve": "solusi",
    "somse": "sombong sekali",
    "songong": "sombong",
    "sorry": "maaf",
    "sory": "maaf",
    "sosbud": "sosial-budaya",
    "sosek": "sosial-ekonomi",
    "sospol": "sosial-politik",
    "sotoy": "sok tahu",
    "sowry": "maaf",
    "spa": "siapa",
    "spd": "sepeda",
    "sppa": "siapa",
    "sprti": "seperti",
    "spt": "seperti",
    "spy": "supaya",
    "srt": "surat",
    "srtfkt": "sertifikat",
    "stelah": "setelah",
    "stiap": "setiap",
```

```
    "still": "masih",
    "stlh": "setelah",
    "stop": "henti",
    "subbag": "subbagian",
    "suk": "masuk",
    "sumbangin": "sumbangkan",
    "sumpek": "sempit",
    "sy": "saya",
    "sya": "saya",
    "syg": "sayang",
    "syp": "siapa",
    "t4": "tempat",
    "tabanas": "tabungan pembangunan nasional",
    "tahi": "kotoran",
    "tai": "kotoran",
    "taik": "tahi",
    "tajir": "kaya",
    "tak": "tidak",
    "tar": "nanti",
    "tarik2": "tarik-tarik",
    "tau": "tahu",
    "tauk": "tahu",
    "taun": "tahun",
    "taw": "tahu",
    "tawh": "tahu",
    "tayangan2": "tayangan-tayangan",
    "td": "tadi",
    "tdi": "tadi",
    "tdk": "tidak",
    "te2p": "tetap",
    "teh": "kakak perempuan",
    "teken": "tekan",
    "tekor": "rugi",
    "telat": "terlambat",
    "telkom": "telekomunikasi",
    "telmi": "telat berpikir",
    "telp": "telepon",
    "teman2": "teman-teman",
    "temen": "teman",
    "temen2": "teman-teman",
    "tenang2": "tenang tenang",
    "tengil": "menyebalkan",
    "tengok": "menjenguk",
    "tepar": "terkapar",
    "terbitin": "terbitkan",
    "tetp": "tetap",
    "teve": "televisi",
    "tggu": "tunggu",
    "tgl": "tanggal",
    "tgok": "tengok",
    "tgu": "tunggu",
    "thanks": "terima kasih",
    "thankz": "terima kasih",
    "thd": "terhadap",
    "thdp": "terhadap",
    "thn": "tahun",
    "thx": "thanks",
    "tilang": "bukti pelanggaran",
    "tilep": "tilap",
```

```
    "tioghoa": "tionghoa",
    "tipi": "TV",
    "tipiwan": "TvOne",
    "tirani": "tiran",
    "tk": "tidak",
    "tkg": "tukang",
    "tks": "terima kasih",
    "tl": "timeline",
    "tll": "terlalu",
    "tlp": "telepon",
    "tlpn": "telepon",
    "tls": "tulis",
    "tman": "teman",
    "tmbah": "tambah",
    "tmbh": "tambah",
    "tmen2": "teman-teman",
    "tmn": "teman",
    "tmn2": "teman-teman",
    "tmpah": "tumpah",
    "tmph": "tumpah",
    "tmpt": "tempat",
    "tnda": "tanda",
    "tngu": "tunggu",
    "tnh": "tanah",
    "tni": "tentara nasional indonesia",
    "tnp": "tanpa",
    "tntg": "tentang",
    "tnya": "tanya",
    "tnyta": "ternyata",
    "togel": "toto gelap",
    "tokai": "tai",
    "tokoh2": "tokoh-tokoh",
    "toserba": "toko serba ada",
    "toys": "main",
    "tp": "tapi",
    "tpi": "tetapi",
    "tq": "terima kasih",
    "trdhulu": "terdahulu",
    "tren": "trend",
    "trending": "trend",
    "trfkir": "terfikir",
    "trgntg": "tergantung",
    "trh": "terhadap",
    "trhdp": "terhadap",
    "trima": "terima kasih",
    "trims": "terima kasih",
    "trimz": "terima kasih",
    "trkhr": "terakhir",
    "trlalu": "terlalu",
    "trllu": "terlalu",
    "trm": "terima",
    "trmsk": "termasuk",
    "trnyt": "ternyata",
    "trs": "terus",
    "trus": "terus",
    "trutama": "terutama",
    "trz": "terus",
    "ts": "penulis",
    "tsangka": "tersangka",
```

```
    "tsb": "tersebut",
    "tsk": "tersangka",
    "tst": "tahu sama tahu",
    "ttd": "tertanda",
    "ttg": "tentang",
    "ttng": "tentang",
    "ttp": "tetap",
    "tu": "itu",
    "tuch": "tuh",
    "tuh": "itu",
    "tuir": "tua",
    "tuwh": "tuh",
    "tv": "televisi",
    "tw": "tahu",
    "twit2": "tweet-tweet",
    "twt": "tweet",
    "tyap": "tiap",
    "u": "untuk",
    "u/": "untuk",
    "uasai": "kuasa",
    "ud": "sudah",
    "uda": "sudah",
    "udah": "sudah",
    "udh": "sudah",
    "ugm": "universitas gajah mada",
    "ui": "universitas indonesia",
    "ujg": "ujung",
    "ujung2nya": "ujung-ujungnya",
    "ul": "ulangan",
    "ultah": "ulang tahun",
    "unair": "universitas airlangga",
    "undang2": "undang undang",
    "undip": "universitas diponegoro",
    "univ": "universitas",
    "unyu": "menggemaskan",
    "uplot": "unggah",
    "uppu": "apa",
    "urang": "saya",
    "urg": "saya",
    "usah": "perlu",
    "ust": "ustadz",
    "ut": "untuk",
    "utang": "hutang",
    "utip": "kutip",
    "utk": "untuk",
    "utul": "ujian tertulis",
    "uu": "undang-undang",
    "valas": "valuta asing",
    "valid": "sah",
    "victory": "menang",
    "vol": "volume",
    "vote": "pilih",
    "w/": "dengan",
    "wad": "buat",
    "wadir": "wakil direktur",
    "wagub": "wakil gubernur",
    "wamil": "wajib militer",
    "war": "perang",
    "warbiyasak": "luar biasa",
```

```
    "waria": "wanita pria",
    "warkop": "warung kopi",
    "warnet": "warung internet",
    "warteg": "warung tegal",
    "wartel": "warung telekomunikasi",
    "waspadalah": "waspada",
    "wat": "buat",
    "watuk": "batuk",
    "who": "world health organization",
    "wk": "tertawa",
    "wkt": "waktu",
    "wkwk": "tertawa",
    "wkwkkw": "tertawa",
    "wtf": "what the fuck",
    "wth": "what the hell",
    "x": "tidak",
    "xan": "tak akan",
    "xew": "apa",
    "xixixi": "tertawa",
    "y": "ya",
    "ya": "iya",
    "yad": "yang akan datang",
    "yap": "iya",
    "yapz": "iya",
    "yaudah": "ya sudah",
    "yauwda": "ya sudah",
    "yawdah": "ya sudah",
    "ybs": "yang bersangkutan",
    "yg": "yang",
    "yi": "yaitu",
    "yl": "yang lain",
    "yng": "yang",
    "yo": "iya",
    "yoha": "iya",
    "yowes": "ya sudah",
    "yuk": "ayo",
    "yup": "iya",
    "ywdh": "ya sudah",
    "zonk": "bodoh"
}
```

31. spell.py

```
import json
import pandas as pd

with open('slang.dic') as f:
    slang = json.load(f)

kamus_slang = pd.read_json('slang.dic',typ = 'series')
pd.DataFrame(kamus_slang.items(), columns = ['contraction',
'original']).head()
contractions_list = list(kamus_slang.keys())

def spell_check(text):
    words = []
    for word in text.split():
```

```
        if word in contractions_list:
            words = words + kamus_slang[word].split()
        else:
            words = words + word.split()

    text_converted = " ".join(words)
    return text_converted
```

Lampiran II Dokumentasi dengan Narasumber

1. Wawancara daring dengan Bapak Sukma Erawan, S.Kom, Guru SMA Negeri 68 Jakarta.

2. Wawancara daring dengan Ibu Ajeng Yeni Ratnasari, S.Pd, Guru Pondok Pesantren Alfassalam.

3. Wawancara daring dengan Bapak Alan Syahwandi, S.Pd.I, Guru SMP Negeri 3 Bogor.

4. Wawancara luring dengan Bapak Ahmad Arfandi, S.Kom, dan Ibu Virgawaty Dewy, S.Pd, Guru SMA Negeri 65 Jakarta.

5. Wawancara luring dengan Ibu Dita Putri Bestari, S.Pd, dan Ibu Ayu Nulantika, S.Pd, Guru SMA Negeri 65 Jakarta.

6. Wawancara daring dengan Ibu Dian Mustika Dewi Priyandari, S. Sos., M.M, Dosen Universitas Suryadarma.

7. Wawancara daring dengan Bapak Zamroni Riza, S.E, Guru SMA Sapta Kharisma Jakarta.

8. Wawancara daring dengan Bapak Muhammad Lutfi, S.Pd, Guru SMA Negeri 113 Jakarta.

Lampiran III Software Requirement Specification

Software Requirements Specification

for

Easysay

Version 1.0 approved

Prepared by

119140114

Geizka Rozilia Ruicosta

1 September 2023

Table of Contents

Table of Contents ii
Revision History ii
1. Pendahuluan 1
1.1 Tujuan Penulisan Dokumen 1
1.2 Audien yang Dituju dan Pembaca yang Disarankan 1
1.3 Batasan Produk 1
1.4 Definisi dan Istilah 1
1.5 Refrensi 2
2. Deskripsi Keseluruhan 3
2.1 Deskripsi Produk 3
2.2 Fungsi Produk 3
2.3 Penggolongan Karakterik Pengguna 3
2.4 Lingkungan Operasi 3
2.5 Batasan Desain dan Implementasi 3
2.6 Dokumentasi Pengguna 4
3. Kebutuhan Antarmuka Eksternal 5
3.1 Antarmuka Pengguna (User Interfaces) 5
3.2 Antarmuka Perangkat Keras (Hardware Interface) 5
3.3 Antarmuka Perangkat Lunak (Software Interface) 5
3.4 Antarmuka Komunikasi (Communication Interface) 5
4. Kebutuhan Fungsional 6
4.1 Use Case Diagram 6
4.2 Activity Diagram 7
5. Kebutuhan Non-Fungsional 8

Revision History

Name	Date	Reason For Changes	Version

1. Pendahuluan

1.1 Tujuan Penulisan Dokumen

Dokumen ini, yang merupakan Software Requirement Specification (SRS) untuk aplikasi "Easysay," bertujuan untuk memberikan panduan rinci terkait kebutuhan fungsional dan non-fungsional dari aplikasi. Tujuan utama penulisan dokumen ini adalah:

1. Memberikan Panduan: Menyajikan panduan yang komprehensif tentang fitur dan karakteristik aplikasi "Easysay" untuk memandu tim pengembangan dalam pengembangan perangkat lunak yang efektif dan sesuai dengan kebutuhan pengguna.
2. Komunikasi dengan Pemangku Kepentingan: Memberikan wawasan yang jelas kepada pemangku kepentingan, seperti pengembang, pengguna, dan pihak terkait lainnya, tentang fungsi dan tujuan utama aplikasi.
3. Dasar Evaluasi: Menjadi dasar evaluasi kinerja dan keberhasilan aplikasi "Easysay" selama dan setelah pengembangan.

1.2 Audien yang Dituju dan Pembaca yang Disarankan

Dokumen ini ditujukan kepada berbagai pemangku kepentingan yang terlibat dalam pengembangan, penggunaan, dan pemeliharaan aplikasi "Easysay." Audien yang dituju meliputi:

1. Tim Pengembang: Mendapatkan pandangan mendalam tentang kebutuhan aplikasi dan petunjuk teknis yang diperlukan untuk implementasi dan pengembangan aplikasi kedepannya.
2. Manajemen Proyek: Memahami ruang lingkup proyek, jadwal waktu, dan sumber daya yang diperlukan untuk pengembangan aplikasi.
3. Pengguna Akhir (Pengajar): Mengetahui fungsionalitas dan cara penggunaan aplikasi untuk keperluan penilaian esai otomatis.

1.3 Batasan Produk

Batasan produk melibatkan keterangan mengenai fitur atau elemen tertentu yang tidak akan disertakan dalam aplikasi "Easysay." Batasan ini berguna untuk memberikan pemahaman tentang ruang lingkup aplikasi dan batasan yang mungkin mempengaruhi penggunaan dan implementasinya. Batasan produk untuk "Easysay" mencakup:

1. Batasan Penggunaan: Aplikasi ini dirancang untuk penilaian esai teoritis dan bukan untuk ujian berbasis pilihan ganda atau bentuk ujian lainnya.
2. Batasan Bahasa: Meskipun mendukung bahasa Indonesia dan Inggris, aplikasi ini tidak dapat menilai esai dalam bahasa lain. Aplikasi ini dapat menerapkan bahasa lainnya apabila dikembangkan lebih lanjut.

1.4 Definisi dan Istilah

Berikut adalah beberapa definisi dan istilah yang digunakan dalam dokumen ini:

1. Easysay: Nama aplikasi penilaian esai otomatis yang dibahas dalam dokumen ini.
2. Pengajar: Termasuk guru dan dosen yang menggunakan aplikasi untuk menilai esai pelajar.
3. Pelajar: Termasuk siswa dan mahasiswa yang jawabannya dinilai menggunakan aplikasi.

1.5 Refrensi

Referensi mencakup sumber daya dan dokumen lain yang digunakan sebagai acuan dalam penyusunan dokumen ini. Referensi dapat termasuk buku, artikel, atau standar industri yang relevan dengan pengembangan aplikasi dan penilaian esai otomatis.

Refrensi untuk aplikasi Easysay dapat dilihat pada Tugas Akhir yang dibuat oleh Geizka Rozilia Ruicosta dengan judul "Aplikasi Penilaian Esai Ootomatis Berbahasa Indonesia dan Inggris Menggunakan Metode *Test-Driven Development*".

2. Deskripsi Keseluruhan

2.1 Deskripsi Produk

Aplikasi "Easysay" dirancang sebagai sistem penilaian esai otomatis yang dapat membantu pengajar (guru/dosen) mengevaluasi jawaban esai pelajar (siswa/mahasiswa) secara efisien. Aplikasi ini, yang disebut sebagai "Easysay," memungkinkan pengguna untuk memeriksa sejumlah besar jawaban esai dalam waktu yang relatif singkat. Fitur kunci termasuk pemilihan bahasa (Indonesia atau Inggris), penilaian tunggal dan jamak, serta integrasi model machine learning menggunakan TF-IDF dan Cosine Similarity.

2.2 Fungsi Produk

Fungsi aplikasi “Easysay” dapat dilihat pada daftar dibawah ini:

1. Penilaian esai otomatis menggunakan model machine learning.
2. Pilihan bahasa untuk penilaian esai (Indonesia atau Inggris).
3. Penilaian tunggal: Memeriksa satu pasangan jawaban esai pengajar dan pelajar.
4. Penilaian jamak: Memungkinkan pengguna mengunggah berkas CSV dengan jawaban pelajar yang berisi nama dan jawaban.
5. Tampilan hasil penilaian dengan skor dan detail pasangan jawaban.

2.3 Penggolongan Karakterik Pengguna

Aplikasi "Easysay" ditujukan khusus untuk satu golongan pengguna, yaitu:

1. Pengajar (Guru/Dosen): Pengguna yang akan menggunakan aplikasi "Easysay" untuk memasukkan jawaban esai pengajar dan pelajar. Pengajar bertanggung jawab atas proses penilaian dan evaluasi menggunakan fitur-fitur yang disediakan oleh aplikasi.

2.4 Lingkungan Operasi

Perangkat lunak "Easysay" akan beroperasi dalam lingkungan sebagai berikut:

1. Platform: Dapat dioperasikan di sistem operasi Windows atau Linux.
2. Perangkat Keras: Memerlukan Node.js (minimal versi 16.16.0) untuk frontend dan Python (minimal versi 3.11.5) untuk backend.
3. Sistem Operasi: Kompatibel dengan Windows atau Linux.

2.5 Batasan Desain dan Implementasi

Batasan desain dan implementasi aplikasi “Easysay” ini dapat dilihat sebagai berikut:

1. Batasan Penggunaan: Aplikasi hanya ditujukan untuk menilai esai dengan jawaban benar atau teoritis.
2. Batasan Bahasa: Meskipun mendukung bahasa Indonesia dan Inggris, aplikasi tidak dapat menilai esai dalam bahasa lain.
3. Keamanan: Tidak ada kebutuhan keamanan khusus yang diimplementasikan.

2.6 Dokumentasi Pengguna

Dokumentasi pengguna untuk aplikasi "Easysay" akan tersedia melalui halaman bantuan di situs web resmi. Dokumentasi ini mencakup:

1. FAQ (Frequently Asked Questions): Sekumpulan pertanyaan umum yang sering diajukan oleh pengguna beserta jawaban rinci yang memberikan pemahaman lebih lanjut tentang berbagai aspek dan fitur aplikasi.

2. Panduan Pengguna:

    a. Cara Menggunakan Aplikasi: Tutorial langkah demi langkah yang menjelaskan cara memanfaatkan fitur-fitur utama aplikasi.
    b. Penilaian Tunggal: Panduan khusus untuk menggunakan fitur penilaian tunggal, termasuk memasukkan jawaban pengajar dan pelajar.
    c. Penilaian Jamak: Instruksi untuk menggunakan fitur penilaian jamak, termasuk langkah-langkah mengunggah berkas CSV dan spesifikasinya.

Dengan menyediakan dokumen ini secara daring, diharapkan pengguna dapat dengan mudah mengakses informasi yang mereka butuhkan dan memanfaatkan fitur-fitur aplikasi "Easysay" secara optimal.

3. Kebutuhan Antarmuka Eksternal

3.1 Antarmuka Pengguna (User Interfaces)

Antarmuka Pengguna pada aplikasi "Easysay" didesain untuk memberikan pengalaman pengguna yang intuitif dan efisien. Beberapa karakteristik antarmuka pengguna meliputi:

1. Bahasa Pilihan: Pengguna dapat memilih bahasa antarmuka, yaitu Bahasa Indonesia atau Bahasa Inggris.
2. Halaman Utama (Home): Menyediakan navigasi untuk memilih metode penilaian tunggal atau jamak, serta memilih bahasa dan mengakses halaman bantuan.
3. Penilaian Tunggal: Formulir input untuk memasukkan jawaban esai pengajar dan pelajar dengan tombol "Periksa".
4. Penilaian Jamak: Formulir untuk mengunggah berkas CSV dengan tombol "Periksa" dan opsi untuk menentukan kolom nama dan jawaban pelajar.
5. Hasil Penilaian: Menampilkan tabel pasangan nama, jawaban, dan skor esai pelajar dengan opsi untuk mengunduh dalam format PDF.

3.2 Antarmuka Perangkat Keras (Hardware Interface)

Antarmuka perangkat keras aplikasi "Easysay" mencakup karakteristik logis dan fisik antara perangkat lunak dan komponen perangkat keras sistem, seperti:

1. Sistem Operasi: Kompatibel dengan Windows dan Linux.
2. Node.js dan Python: Memerlukan perangkat keras yang mendukung Node.js (minimal versi 16.16.0) untuk frontend dan Python (minimal versi 3.11.5) untuk backend.

3.3 Antarmuka Perangkat Lunak (Software Interface)

Aplikasi “Easysay” akan berjalan menggunakan perangkat lunak Node.js dan Python dengan versi spesifik yang telah dirincikan sebelumnya.

3.4 Antarmuka Komunikasi (Communication Interface)

Antarmuka komunikasi aplikasi "Easysay" mencakup persyaratan berbagai fungsi komunikasi. Protokol komunikasi seperti HTTP digunakan untuk komunikasi antara frontend dan backend.

Dokumen ini memberikan gambaran umum mengenai antarmuka eksternal yang dibutuhkan oleh aplikasi "Easysay." Rincian desain antarmuka pengguna akan didokumentasikan secara lebih terperinci dalam spesifikasi antarmuka pengguna terpisah. Tidak ada antarmuka database yang diperlukan karena aplikasi tidak menyimpan hasil penilaian.

4. Kebutuhan Fungsional

Berikut adalah daftar kebutuhan fungsional untuk aplikasi "Easysay." Setiap kebutuhan memiliki ID unik dan penjelasan yang relevan.

ID	Deskripsi
F0	Aplikasi *website* dapat menentukan algoritma penilaian yang digunakan berdasarkan permintaan pengguna melalui tombol
F1A	Aplikasi *website* dapat menerima masukkan dua buah jawaban berupa teks melalui *input form*
F1B	Aplikasi *website* dapat melakukan penilaian otomatis dari kedua masukkan pada fungsi dengan ID F1A
F2A	Aplikasi *website* dapat menerima masukkan berkas dengan ekstensi CSV, sebuah jawaban berupa teks, nama kolom nama pelajar berupa teks, dan nama kolom jawaban pelajar berupa teks.
F2B	Aplikasi *website* dapat melakukan penilaian berdasarkan masukkan pada fungsi dengan ID F2A

4.1 Use Case Diagram

Dibawah ini dapat diperhatikan interaksi pengguna pada sistem yang terjadi pada aplikasi "Easysay".

Pada *use case diagram* tersebut, dapat diperhatikan bahwa aktor merupakan pengajar yang ingin melakukan pengecekkan jawaban esai milik pelajar. Pengguna dapat melihat hasil jawaban yang diperiksa dengan mengirimkan esai yang ingin diperiksa, kemudian memilih bahasa dan teknik pengecekkan yang diinginkan.

4.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan alur pekerjaan dari aktivitas pada suatu sistem.

Pada gambar tersebut dapat diperhatikan bahwa aplikasi dimulai dengan pengguna memilih bahasa dan metode penilaian yang akan digunakan. Kemudian berdasarkan pilihan tersebut, sistem akan menentukan masukkan yang dapat dilakukan oleh pengguna. Apabila semua masukkan telah diisi, kemudian sistem akan meminta *server* melakukan penilaian yang diminta oleh pengguna berdasarkan bahasa dan metode penilaian. Setelah *server* selesai melakukan penilaian, *server* akan mengirimkan hasil penilaian untuk ditampilkan pada aplikasi agar dapat dilihat oleh pengguna.

5. Kebutuhan Non-Fungsional

Untuk meningkatkan kenyamanan pengguna saat menggunakan aplikasi penilaian esai otomatis berbasis website ini, diperlukan beberapa kebutuhan non-fungsional yang ingin dikembangkan. Kebutuhan ini dapat meningkatkan retensi pengguna agar lebih nyaman saat menggunakan aplikasi ini. Pada tabel dibawah ini dapat dilihat kebutuhan non-fungsional yang akan dipenuhi.

ID	Deskripsi
NF1	Aplikasi *website* dapat digunakan pada berbagai perangkat, komputer dan mobile
NF2	Aplikasi *website* dapat dijalankan pada peramban web yang mendukung ES6
NF3	Aplikasi *website* memiliki tampilan yang menyesuaikan dengan ukuran layar perangkat yang digunakan (responsif)