

Dokumen Ulasan: Spesifikasi Tugas Besar 3 IF2211

Bagian 1: Analisis Kualitas Dokumen

Secara umum, dokumen spesifikasi ini cukup dapat dipahami, namun terdapat beberapa area yang dapat ditingkatkan secara signifikan untuk mempermudah mahasiswa dalam mencerna informasi.

- **Penggunaan Kalimat Bertele-tele:** Beberapa bagian, terutama pada latar belakang, menggunakan kalimat yang kurang efisien.
 - **Contoh:** "Bukan sesuatu yang janggal lagi jika semakin hari tugas-tugas di Teknik Informatika Semester 4 semakin bertambah banyak."
 - **Saran Perbaikan:** Kalimat ini bisa disederhanakan menjadi, "Mahasiswa Teknik Informatika Semester 4 seringkali dihadapkan oleh tumpukan tugas dengan tenggat waktu yang berdekatan." Ini lebih langsung dan profesional. (Tidak ada 'semakin'-nya dua kali)
- **Istilah yang Kurang Didefinisikan di Awal:** Dokumen memperkenalkan konsep seperti "sistem Question and Answer" dan "kumpulan formula tertentu" tanpa memberikan definisi konkret di awal. Hal ini memaksa pembaca untuk menyimpulkan sendiri maknanya dari contoh-contoh yang diberikan kemudian, yang dapat menimbulkan ambiguitas. Sebaiknya, jelaskan secara singkat bahwa "formula" yang dimaksud adalah pola regular expression untuk mendeteksi niat (intent) pengguna.
- **Instruksi Implisit:** Pada bagian fitur, beberapa detail penting tidak dinyatakan secara eksplisit. Mereka memberikan kebebasan, namun kebebasan tersebut tidak dijelaskan lebih lanjut, ini dapat menyebabkan ambiguitas. Misalnya, pada fitur "Melihat daftar task", dokumen memberikan contoh perintah seperti "Deadline N minggu ke depan apa saja?". Dokumen tidak secara tegas menyatakan bahwa chatbot harus bisa mengekstrak nilai 'N' dari kalimat tersebut. Ini memang tersirat, namun penulisan yang lebih eksplisit akan mengurangi potensi kebingungan.

Struktur dan Kerapian Dokumen

Struktur dokumen ini fungsional, tetapi kurang rapi dan tidak memanfaatkan hierarki informasi dengan baik. Hal ini mengurangi scannability dokumen, sehingga mahasiswa sulit menemukan informasi spesifik dengan cepat.

- **Kurangnya Semantic Markup:** Dokumen ini tidak menggunakan heading (seperti H1, H2, H3) secara efektif. Bagian-bagian penting seperti "Deskripsi tugas", "Fitur-Fitur Aplikasi", dan "Spesifikasi Program" memiliki level penekanan yang sama. Penggunaan heading yang terstruktur akan menciptakan hierarki visual dan mempermudah navigasi. Kurangnya *page break* pada dokumen membuat informasi pada dokumen lebih mudah untuk terlewat. (Like, bro, minimal ada daftar isi gak sih ._.)
- **Contoh Interaksi Berbasis Gambar:** Menyajikan contoh interaksi sebagai gambar adalah praktik yang buruk. Mungkin mereka membuat ini agar teks di dalam gambar tidak dapat dicari (searchable), tidak dapat disalin (copy-paste), namun hasil yang didapatkan

dengan adanya gambar membuat penulis dokumen tersebut terkesan malas. Contoh-contoh ini seharusnya ditulis sebagai teks biasa, mungkin di dalam blok kutipan atau blok kode untuk membedakannya dari teks utama.

- **Pengelompokan Informasi yang Kurang Logis:** Ini sedikit balik lagi ke poin awal. Informasi penting tersebar di beberapa bagian. Misalnya, "Keterangan penting" dan "Lain-lain" berisi aturan-aturan krusial yang seharusnya dikelompokkan dalam bagian yang lebih deskriptif seperti "Ketentuan Pengerjaan" atau "Spesifikasi Teknis". Ini membuat mahasiswa harus membaca keseluruhan dokumen beberapa kali untuk memastikan tidak ada aturan yang terlewat.
- **Penggunaan "(dan seterusnya)" yang Ambigu:** Pada contoh fitur "Help", penggunaan frasa "... (dan seterusnya)" kurang informatif. Sebaiknya, dokumen menyatakan dengan jelas apakah mahasiswa diharapkan menampilkan semua fitur yang ada atau hanya beberapa contoh saja.

Bagian 2: Analisis Konten dan Relevansi Tugas

Bagian ini mengulas relevansi materi tugas terhadap capaian pembelajaran mata kuliah, kesesuaian beban kerja, dan potensi pembagian tugas dalam kerja kelompok.

Relevansi dengan Capaian Mata Kuliah

Tugas ini memiliki relevansi yang tinggi dengan materi dan capaian mata kuliah IF2211 Strategi Algoritma.

1. **Kesesuaian dengan Lingkup Bahasan:** Tugas ini secara eksplisit menuntut implementasi String Matching dan Regular Expression, yang merupakan salah satu dari 12 lingkup bahasan utama pada silabus. Selain itu, fitur bonus untuk rekomendasi kata (koreksi typo) mendorong mahasiswa untuk mempelajari dan mengimplementasikan algoritma berbasis dynamic programming seperti Levenshtein distance, yang juga merupakan topik inti mata kuliah.
2. **Pemenuhan Capaian Pembelajaran:**
 - **"Analyze a complex computing problem...":** Mahasiswa ditantang untuk menganalisis masalah ambigu dari pemrosesan bahasa natural dalam skala kecil. Mereka harus mengidentifikasi bagaimana perintah pengguna yang bervariasi dapat dipetakan ke dalam aksi yang spesifik, lalu memilih prinsip komputasi yang relevan (Regex, String Matching) untuk solusinya.
 - **"Design, implement, and evaluate a computing-based solution...":** Tugas ini adalah perwujudan dari capaian ini. Mahasiswa merancang arsitektur chatbot, mengimplementasikan logikanya, dan secara implisit mengevaluasi apakah solusi mereka berhasil memenuhi semua fitur yang disyaratkan.
 - **"Apply computer science theory... to produce computing-based solutions":** Ini adalah inti dari tugas besar. Mahasiswa tidak hanya belajar teori algoritma KMP, Boyer-Moore, atau Regex di kelas, tetapi langsung menerapkannya untuk membangun sebuah produk perangkat lunak yang fungsional.

Analisis Beban Tugas vs. Bobot SKS

Mata kuliah IF2211 memiliki bobot **3 SKS**. Berdasarkan standar umum, 1 SKS setara dengan sekitar 3 jam beban kerja per minggu (termasuk kuliah, tugas terstruktur, dan belajar mandiri) (*Mengacu pada definisi 1 sks (170 menit) menurut Pasal 19 Ayat (4) Peraturan Menteri Pendidikan dan Kebudayaan No. 3 Tahun 2020, serta penetapan 1 semester terdiri dari 16 minggu*). Dengan demikian, total beban kerja mata kuliah ini adalah sekitar 9-12 jam per minggu. Terdapat 3 Tugas Besar (Tubes), 3 Tugas Kecil (Tucil), UTS, dan UAS.

Menurut saya, beban tugas ini **cukup berat, berpotensi melebihi alokasi wajar** untuk satu dari tiga tugas besar.

- **Overhead Teknologi yang Signifikan:** Selain implementasi algoritma inti (String Matching, Regex, DP), mahasiswa diwajibkan membangun aplikasi berbasis web menggunakan framework seperti PHP, Flask, Django, atau JS. Ini menambahkan beban kerja yang signifikan di luar cakupan mata kuliah "Strategi Algoritma", yaitu:
 - Menyiapkan lingkungan pengembangan web.

- Membangun antarmuka pengguna (frontend).
- Mengelola routing dan logika backend.
- Menangani state percakapan.
- Opsi untuk melakukan deployment ke hosting provider.
- **Prasyarat Pengetahuan Implisit:** Meskipun prasyarat kuliah adalah Matematika Diskrit dan Algoritma & Struktur Data, tugas ini secara implisit menuntut pengetahuan dasar pengembangan web yang mungkin tidak dimiliki semua mahasiswa secara merata. Hal ini dapat menciptakan disparitas waktu pengerjaan yang besar antar kelompok.
- **Konteks Kehadiran AI:** Ini merupakan salah satu *counter argument* yang dapat dipakai, yaitu “dengan adanya AI, maka produktivitas mahasiswa juga akan naik dan tugas akan semakin gampang.” Argumen tersebut merupakan argumen yang valid dan tidak dapat dipungkiri, namun ada juga batasan LLM yang harus dipertimbangkan. Mulai dari limit chat hingga kekuatan problem solving dari LLM itu sendiri. Namun, jika diberikan *ranking*, maka argumen ini adalah salah satu argumen yang harus sangat dipertimbangkan.
- **Informasi Tambahan:** Perlu diingat kembali, bahwa per semester biasanya mata kuliah Strategi Algoritma memiliki total 3 tugas besar, 3 tugas kecil, dan 1 makalah. Ini juga memperkuat argumen bahwa spesifikasi yang diberikan melebihi alokasi wajar.

Analisis Pembagian Kerja Kelompok

Tugas ini dirancang untuk kelompok berisi 2-3 orang. Pembagian kerja dimungkinkan, namun **berisiko tidak merata dalam hal relevansi materi.**

- **Pembagian yang Mungkin:**
 - **Person A (Backend/Algorithm Specialist):** Fokus pada perancangan pola Regex, implementasi algoritma KMP/Boyer-Moore, dan logika utama untuk memproses command.
 - **Person B (Full-Stack/Integration Specialist):** Fokus pada penyiapan web server (Flask/Django), membangun API, mendesain antarmuka chat, dan menghubungkan frontend dengan backend.
 - **Person C (Data & Documentation Specialist):** Fokus pada desain skema database atau struktur file .txt, pengujian fungsionalitas, penulisan laporan, dan pembuatan video demo.
- **Risiko Ketidakmerataan:** Dalam skenario di atas, Person A adalah yang paling banyak bersinggungan langsung dengan materi inti Strategi Algoritma. Person B mungkin akan lebih banyak menghabiskan waktu untuk urusan teknis web development (tidak fokus STIMA), dan Person C pada dokumentasi. Meskipun semua anggota kelompok diwajibkan memahami keseluruhan program, porsi pembelajaran dan implementasi materi inti bisa jadi tidak seimbang. Hal ini merupakan tantangan umum dalam tugas besar berbasis proyek, namun patut menjadi perhatian.

Secara keseluruhan, tugas ini sangat baik dari segi relevansi akademis, namun memiliki beban kerja yang berat karena adanya tuntutan teknologi di luar materi inti.

Bagian 3: Analisis Kesalahan Ejaan dan Kata Tidak Baku

A. Kesalahan Ejaan (Sesuai PUEBI/EYD)

1) Penulisan istilah asing (huruf miring)

Istilah asing yang belum terserap seharusnya dicetak miring.

Contoh Salah: *Regular Expression, Question and Answer, load / save.*

Seharusnya: *regular expression, question and answer, load/save* (atau padanan Indonesia; lihat Bagian B).

Referensi: EYD V “Huruf Miring”. ([Ejaan Kemdikbud](#))

2) Tanda hubung untuk imbuhan + kata asing

Jika imbuhan Indonesia bertemu dasar asing, gunakan tanda hubung.

Contoh Salah: *Mendeploy, di-zip.*

Seharusnya: *men-deploy, di-zip.*

Referensi: EYD V “Tanda Hubung”; Seri Penyuluhan Ejaan (imbuhan + kata asing). ([Ejaan Kemdikbud](#))

3) Spasi sebelum tanda titik dua

Titik dua tidak didahului spasi.

Contoh Salah: “Batas pengumpulan : ...”, “Arsip pengumpulan :”, “Sumber : ...”.

Seharusnya: “Batas pengumpulan: ...”, “Arsip pengumpulan:”, “Sumber: ...”.

Referensi: EYD V (Penggunaan Tanda Baca: Tanda Titik Dua). ([Ejaan Kemdikbud](#))

4) Kalimat terputus/fragmen karena titik salah tempat

Contoh Salah: “Deadline Reminder Assistant. akan dibangun ...”

Seharusnya: “Deadline Reminder Assistant akan dibangun ...”

Referensi: EYD V (Tanda Baca, keterpaduan kalimat).

5) Huruf kapital pada kata ganti *Anda*

Anda wajib berhuruf awal kapital.

Contoh Salah: “... fitur ... yang **anda** buat ...”; “dapat **anda** implementasikan ...”.

Seharusnya: “... yang **Anda** buat ...”; “dapat **Anda** implementasikan ...”.

Referensi: PUEBI (Huruf Kapital: *Anda*). ([Bali Bahasa](#), [Kemdikbud Repository](#))

6) Garis miring (/) ditulis tanpa spasi kiri/kanan

Contoh Salah: “Kode Mata Kuliah / Nama Mata Kuliah”, “load / save”.

Seharusnya: “Kode mata kuliah/Nama mata kuliah”, “load/save”.

Referensi: EYD V (Tanda Garis Miring). ([Ejaan Kemdikbud](#), [Ivan Lanin](#))

7) Kata depan *di/ke* ditulis terpisah

Contoh Salah: “3 minggu **kedepan** ...”, “kejadian **dimana** ...”, “disini”.

Seharusnya: “ke depan”, “di mana”, “di sini”.

Referensi: Seri Penyuluhan Ejaan (kata depan di/ke/dari terpisah). ([Garuda Kemdikbud](#))

8) Ejaan nama bahasa/merek

Contoh Salah: “Javascript”, “Youtube”.

Seharusnya: “JavaScript”, “YouTube”.

Referensi: Kaidah ejaan nama diri (umum); konsistensi kapitalisasi.

9) Konsistensi nama jenis: *chatbot*

Bukan nama diri → huruf kecil.

Contoh Salah: “ChatBot”, “Chatbot” (tidak konsisten).

Seharusnya: “chatbot” (kecuali di awal kalimat/judul).

Referensi: PUEBI (Huruf Kapital, bukan nama diri). ([rumahpusbin.kemdikbud.go.id](#))

10) Nama diri “Ganesha” harus kapital

Contoh Salah: “logo ‘gajah’ **ganesha**.”

Seharusnya: “logo ‘gajah’ **Ganesha**.”

Referensi: PUEBI (Huruf Kapital, nama diri). ([rumahpusbin.kemdikbud.go.id](#))

11) “dan/atau” seharusnya dengan garis miring yang menempel

Dokumen memakai “link ... atau referensi ...” (OK), tetapi bila memakai “dan/atau”, garis miring menempel.

Contoh Benar (acuan): “buku **dan/atau** majalah”.

Referensi: PUEBI (Tanda Garis Miring). ([Bali Bahasa](#))

12) Penulisan format rentang lebih baik pakai “s.d.” atau tanda pisah (,)

Bila menulis “sampai dengan”, gunakan “s.d.” atau tanda pisah panjang.

Contoh Acuan: “Tanggal 5, 10 April 2013” atau “5 s.d. 10 April 2013”.

Referensi: Seri Penyuluhan Ejaan (tanda pisah untuk ‘sampai dengan’).

B. Penggunaan Kata Tidak Baku (Sesuai KBBI VI)

1) *point* → bentuk baku: *poin*.

Contoh Salah: “**Point** i sampai dengan iv ...”

Seharusnya: “**Poin** i sampai dengan iv ...”

Referensi: Rujuk entri *poin* di KBBI VI Daring. ([kbbi.kemdikbud.go.id](#))

2) *assistant* → asisten.

Contoh Salah: “daftar task ... oleh **assistant**”, “Apa yang bisa **assistant** lakukan?”

Seharusnya: “asisten”.

Referensi: KBBI VI (umum). ([kbbi.kemdikbud.go.id](#))

3) *task* → tugas.

Contoh Salah: “daftar **task** ...”, “**task** tersebut ...”

Seharusnya: “tugas”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

4) *user* → pengguna.

Contoh Salah: “... masukan **user** ...”, “... kata yang digunakan **user** ...”

Seharusnya: “pengguna”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

5) *command* → perintah.

Contoh Salah: “perbedaan **command** ...”, “**command-command** ...”

Seharusnya: “perintah”, “perintah-perintah”.

Referensi: KBBI VI & EYD V (reduplikasi). (kbbi.kemdikbud.go.id)

6) *website* → situs web.

Contoh Salah: “Deployment **website** ...”, “ke dalam **website**”.

Seharusnya: “deployment situs web” / “ke dalam situs web”.

Referensi: KBBI VI (“situs”, “web”). (kbbi.kemdikbud.go.id)

7) *library file* → pustaka (atau berkas pustaka).

Contoh Salah: “kecuali **library file** dan regex ...”

Seharusnya: “kecuali **pustaka** dan regex ...”.

Referensi: KBBI VI (“pustaka”). (kbbi.kemdikbud.go.id)

8) *regular expression* → ekspresi reguler (regex).

Contoh Salah: “bahasa yang menunjang **regular expression** ...”

Seharusnya: “ekspresi reguler (regex)”.

Referensi: KBBI VI (serapan *ekspresi*, *reguler*), EYD V (Huruf Miring). ([Ejaan Kemdikbud](http://EjaanKemdikbud))

9) *link* → tautan.

Contoh Salah: “**Link** pengumpulan ...”

Seharusnya: “**Tautan** pengumpulan ...”.

Referensi: KBBI VI (“tautan”). (kbbi.kemdikbud.go.id)

10) *list* → daftar.

Contoh Salah: “**List** kata-kata penting ...”

Seharusnya: “**Daftar** kata-kata penting ...”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

11) *dynamic programming* → pemrograman dinamis.

Contoh Salah: “... Levenshtein distance ... **dynamic programming**.”

Seharusnya: “pemrograman dinamis”.

Referensi: KBBI VI (serapan *program*, *dinamis*). (kbbi.kemdikbud.go.id)

12) *metriks* → **metrik** (tanpa “s”).

Contoh Salah: “Ada berbagai **metriks** ...”

Seharusnya: “berbagai **metrik** ...”.

Referensi: KBBI VI (“metrik”). (kbbi.kemdikbud.go.id)

13) *mismatch* → **tidak cocok/ketakselarasan**.

Contoh Salah: “perintah ... **mismatch** dengan daftar kata ...”

Seharusnya: “tidak cocok/ada ketakselarasan”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

14) *typo* → **salah ketik**.

Contoh Salah: “... kesalahan kata (**typo**) ...”

Seharusnya: “salah ketik”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

15) *source code* → **kode sumber**.

Contoh Salah: “folder src berisi **source code**”

Seharusnya: “kode sumber”.

Referensi: KBBI VI (“sumber”, padanan terma TI). (kbbi.kemdikbud.go.id)

16) *QnA* → **tanya jawab**.

Contoh Salah: “lewat **QnA** ...”

Seharusnya: “lewat **tanya jawab** ...”.

Referensi: KBBI VI (“tanya jawab”). (kbbi.kemdikbud.go.id)

17) *deadline* → **tenggat**.

Contoh Salah: “Batas akhir pengumpulan ... **deadline** ...” (berbagai kemunculan).

Seharusnya: “tenggat”.

Referensi: KBBI VI (“tenggat”). (kbbi.kemdikbud.go.id)

18) *kakas* (untuk ‘tool’) kurang tepat → **perangkat/alat**.

Contoh Salah: “anda dapat menggunakan salah satu **kakas website** ...”

Seharusnya: “salah satu **perangkat/alat** pengembangan web ...”.

Referensi: KBBI VI (makna *kakas*), padanan umum TI. (kbbi.kemdikbud.go.id)

19) *deployment* (bahasa Inggris) → **pendayagunaan/penerapan/peluncuran** (pilih sesuai konteks).

Contoh Salah: “**Deployment website** harus dipertahankan ...”

Seharusnya: “**Pendayagunaan/penyebarluasan situs web** ...”.

Referensi: KBBI VI; EYD V (huruf miring untuk istilah asing). ([Ejaan Kemdikbud](http://EjaanKemdikbud))

20) *interfac/features* → **antarmuka/fitur**.

Contoh Salah: “**Interface, Features**, dan Unsur Kreativitas ...”

Seharusnya: “**Antarmuka, Fitur**, dan Unsur Kreativitas ...”.

Referensi: KBBI VI (serapan *fitur*), padanan *antarmuka*. (kbbi.kemdikbud.go.id)

21) *user --bukan programmer--* (campur kode) → *pengguna, bukan pemrogram*, (gunakan tanda pisah em).

Contoh Salah: “input dibebaskan ke **user --bukan programmer--** ...”

Seharusnya: “dibebaskan ke **pengguna, bukan pemrogram**, ...”.

Referensi: PUEBI “Tanda Pisah”. ([Ivan Lanin](#))

22) *Youtube* (ejaan merek tidak baku)

Lihat juga Bagian A (kapitalisasi merek).

Contoh Salah: “mengunggahnya ke **Youtube**.”

Seharusnya: “ke **YouTUBE**.”

Referensi: Kaidah nama diri (umum).

23) *DATE_1 until DATE_2* → *DATE_1 sampai DATE_2* (hindari bahasa Inggris di teks Indonesia).

Contoh Salah: “(DATE_1 **until** DATE_2)”

Seharusnya: “(DATE_1 **sampai** DATE_2)”.

Referensi: EYD V (Huruf Miring untuk istilah asing bila tetap dipakai). ([Ejaan Kemdikbud](#))

24) *help* → *bantuan*.

Contoh Salah: “Menampilkan opsi **help** yang difasilitasi ...”

Seharusnya: “opsi **bantuan** ...”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)

25) *keyword* → *kata kunci*.

Contoh Salah: “... satu **keyword** untuk memperbaharui ...”

Seharusnya: “satu **kata kunci** ...”.

Referensi: KBBI VI. (kbbi.kemdikbud.go.id)