



Artikel

Pengembangan *website* ujian tes TOEFL pada Lembaga Kursus “Plug-in”

Arya Bima Wijaya *, Yohana Tri Widayati, Satrio Agung Prakoso

Teknik Informatika, Fakultas Teknik dan Informatika, Universitas AKI Semarang, Semarang, 50139, Indonesia

Abstrak—Tes TOEFL adalah standar internasional untuk mengukur kemampuan bahasa Inggris. Penelitian ini mengembangkan sistem ujian TOEFL berbasis web di Kursus Plug-in menggunakan metode Scrum untuk mengatasi kekurangan sistem lama, seperti kontrol audio, akurasi skor, dan pencatatan hasil yang kurang detail. Data dikumpulkan melalui observasi, wawancara, dan studi pustaka. Scrum diterapkan dalam tiga *sprint* yang berfokus pada navigasi *listening*, perhitungan skor, dan pencatatan jawaban. Fitur pemutaran audio sekali, *timer*, penghitungan skor otomatis, dan tampilan hasil terperinci berhasil diimplementasikan. Pengujian menunjukkan sistem stabil dan akurat sesuai standar TOEFL. Sistem ini meningkatkan efisiensi, keadilan, dan pengalaman pengguna dalam ujian TOEFL digital. Saran pengembangan meliputi pengujian lebih luas, optimalisasi UI/UX, fitur ekspor data, dan pencatatan jawaban lebih detail untuk analisis.

Kata kunci—kontrol pemutaran audio; metode scrum; perhitungan skor; sistem ujian TOEFL; ujian berbasis web.

1. Pendahuluan

Tes TOEFL (*Test of English as a Foreign Language*) merupakan salah satu metode standar internasional yang digunakan untuk mengukur kemampuan berbahasa Inggris. Dengan meningkatnya kebutuhan akan tes ini di kalangan pelajar dan profesional, maka dibutuhkan sistem digital yang mendukung

pelaksanaan ujian secara fleksibel dan efisien.

Dalam era digital saat ini, teknologi informasi telah mengubah banyak aspek kehidupan, termasuk bidang pendidikan. Menurut Febriansyah & Voutama (2024), “Teknologi informasi telah mengubah banyak aspek kehidupan, termasuk pendidikan, di era digital yang semakin meluas. Kemajuan ini berpengaruh pada sektor pendidikan dengan dampak mudahnya akses dan pengurangan penggunaan kertas akan menjadi solusi yang relevan bagi situasi saat ini” (Febriansyah & Voutama, 2024).

Selaras dengan itu, Khazanah & Purnama (2024) juga menyatakan bahwa “proses ujian tradisional sering kali menghadapi beberapa kendala, seperti kesulitan dalam administrasi, penyusunan soal yang beragam, dan proses penilaian yang rumit serta memakan waktu” (Khazanah & Purnama, 2024).

Berbagai penelitian sebelumnya telah membahas pentingnya digitalisasi dalam pelaksanaan ujian. Sanjaya et al. (2025) menyatakan bahwa “sistem ujian tradisional yang memakan waktu dapat digantikan oleh platform digital yang lebih modern,” dan menekankan perlunya sistem ujian *online* yang meningkatkan efisiensi, transparansi, dan akuntabilitas (Pauzi & Kharisma, 2025). Sementara itu, Abdi et al. (2024) menunjukkan pentingnya keamanan dan keadilan dalam pelaksanaan ujian TOEFL berbasis web melalui implementasi algoritma pengacakan soal, seperti Fisher-Yates, untuk mencegah kecurangan dan meningkatkan keadilan antar peserta (Abdi et al., 2024).

Namun, meskipun berbagai sistem ujian *online* telah dikembangkan, masih terdapat kesenjangan antara kebutuhan pengguna dan performa sistem yang tersedia. Sistem lama yang

* Penulis korespondensi.

Alamat E-mail: aryabimaw13@gmail.com (A.B. Wijaya)

Email para penulis: ABW (aryabimaw13@gmail.com), YTW (yohana.tri@unaki.ac.id), SAP (satrio.agung@unaki.ac.id)

Digital Object Identifier 10.32815/jitika.v19i2.1189

Manuskrip dikirim 14 Agustus 2025; direvisi 15 Agustus 2025; diterima 15 Agustus 2025.

ISSN: 2580-8397(O), 0852-730X(P).

digunakan oleh Kursus Plug-in nama lembaga tempat penelitian ini dilakukan dalam pelaksanaan ujian TOEFL berbasis web menunjukkan sejumlah kendala yang signifikan (Maulana & Lubis, 2022). Salah satu masalah utama adalah ketidakmampuan sistem untuk memproses jawaban peserta setelah audio pada *section listening* diputar, sehingga peserta tidak dapat melanjutkan ujian sebagaimana mestinya (Marbun et al., 2016). Selain itu, sistem secara keliru menampilkan skor peserta dengan nilai 0, padahal dalam ujian TOEFL resmi, sangat kecil kemungkinan peserta memperoleh nilai nol. Permasalahan lain yang muncul adalah nilai akhir peserta yang identik antara satu peserta dengan lainnya, yang menimbulkan keraguan terhadap keakuratan dan keadilan proses penilaian.

Tidak hanya itu, sistem juga tidak menyediakan mekanisme pemulihan bagi peserta yang terputus dari sesi ujian karena gangguan sinyal atau koneksi internet (Oktapianti, 2024). Akibatnya, peserta yang tidak dapat mengakses kembali soal ujian harus mengulang proses dari awal atau bahkan kehilangan kesempatan menyelesaikan ujian.

Permasalahan-permasalahan tersebut menunjukkan bahwa sistem lama tidak mampu memberikan pengalaman ujian yang andal, valid, dan sesuai standar (Lubis & Irmayana, 2019). Oleh karena itu, pengembangan *website* ujian TOEFL yang baru diperlukan sebagai solusi dari berbagai keterbatasan tersebut (Diraja, 2018). Selain itu, sistem baru juga akan dirancang agar mampu menangani gangguan teknis seperti koneksi terputus, dengan fitur penyimpanan progres dan akses ulang yang fleksibel. Dengan adanya sistem baru ini, pelaksanaan ujian TOEFL di Kursus Plug-in diharapkan menjadi lebih efisien, efektif, dan terpercaya.

2. Metode

2.1. Subjek dan objek penelitian

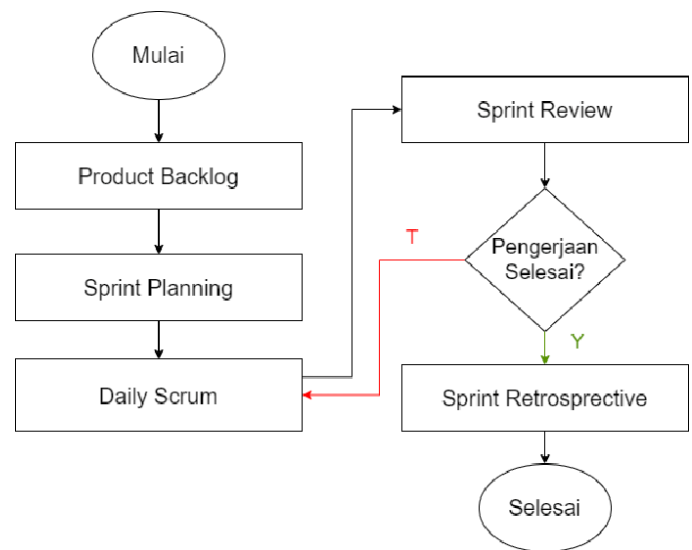
Subjek penelitian ini adalah sistem ujian TOEFL berbasis *website* yang dikembangkan untuk Lembaga Kursus Plug-in, sedangkan objek penelitian mencakup fitur dan data pelaksanaan ujian seperti soal, jawaban peserta, skor, waktu pengerjaan, serta data peserta yang terintegrasi dalam sistem.

2.2. Metode pengumpulan data

Penelitian ini bertujuan memperoleh data terkait kebutuhan dan pengembangan sistem ujian TOEFL berbasis web dengan tiga metode pengumpulan data, yaitu observasi tidak langsung, wawancara, dan studi pustaka. Observasi dilakukan melalui analisis dokumentasi dan sistem TOEFL yang ada untuk mempelajari alur ujian mulai dari *registrasi*, *listening*, *structure*, *reading*, hingga penyajian nilai. Wawancara dilakukan dengan 3 *pengajar TOEFL*, 1 *staf administrasi*, dan 25 *peserta ujian* untuk menggali kebutuhan, kendala, serta harapan terhadap sistem yang lebih efisien. Studi pustaka dilakukan dengan menelaah literatur terkait pengembangan sistem informasi, metode ujian TOEFL, dan teknologi ujian berbasis web dari berbagai sumber seperti buku, jurnal, dan penelitian sebelumnya.

2.3. Sumber dan jenis data

Penelitian ini menggunakan sumber data yang diperoleh dari responden dan dokumen, baik statistik maupun non-statistik, sesuai dengan subjek dan tujuan penelitian. Dengan



Gambar 1. Metode Scrum

pendekatan kualitatif berperspektif *emic*, peneliti mengutamakan pandangan informan yang diperoleh melalui wawancara atau observasi. Sumber data terdiri dari dua jenis: (1) Data primer, diperoleh langsung dari responden melalui wawancara untuk mengetahui kebutuhan fungsional dan non-fungsional sistem ujian TOEFL berbasis web; dan (2) Data sekunder, berasal dari jurnal, buku, artikel, dan dokumentasi sistem serupa sebagai landasan teori, perbandingan, serta acuan pemilihan teknologi dan metode pengembangan.

2.4. Metode pengembangan sistem

Metodologi pengembangan sistem adalah pendekatan terstruktur yang digunakan dalam merancang dan membangun sistem informasi. Pada penelitian ini, pengembangan *website* ujian TOEFL dilakukan dengan menggunakan metode Scrum (Wijaya et al., 2024), salah satu kerangka kerja *Agile Development* yang iteratif, adaptif, dan responsif terhadap perubahan kebutuhan pengguna (Noveandini et al., 2023).

Pemilihan metode Scrum didasarkan pada kebutuhan pengembangan sistem yang kompleks, dinamis, dan membutuhkan respons cepat terhadap umpan balik pengguna. Sistem ujian TOEFL memiliki sejumlah fitur penting seperti pengacakan soal, *timer*, tampilan soal *listening* berbasis audio, serta penilaian otomatis. Fitur-fitur tersebut memerlukan pengujian dan evaluasi berkala agar dapat berjalan sesuai harapan dan mudah digunakan oleh peserta maupun administrator (Noveandini et al., 2023).

Scrum merupakan kerangka kerja yang sangat cocok digunakan dalam pengembangan sistem berbasis web maupun *mobile* karena sifatnya yang adaptif, kolaboratif, dan mampu menghasilkan sistem secara bertahap serta terukur melalui *sprint-sprint* tertentu.

Pada penelitian ini, kebutuhan sistem dituangkan dalam bentuk *user story*, yaitu deskripsi singkat dari sudut pandang pengguna yang menjelaskan fitur atau fungsi yang diinginkan. Format yang digunakan mengikuti pola: "Sebagai [tipe pengguna], saya ingin [tujuan] sehingga [manfaat]". Contohnya: "Sebagai peserta ujian, saya ingin mendengar *audio listening*

hanya satu kali agar suasana ujian menyerupai TOEFL resmi". Pendekatan ini memudahkan tim dalam memahami ekspektasi pengguna, mengurutkan prioritas pengembangan, dan memastikan bahwa setiap fitur yang dibuat memiliki manfaat langsung bagi pengguna akhir.

Sebagaimana ditunjukkan pada Gambar 1, metode Scrum yang diterapkan dalam penelitian ini digunakan untuk mengatur alur pengembangan sistem ujian TOEFL secara bertahap dan terstruktur (Nurmasani et al., 2024). Dengan menggunakan Scrum, pengembangan sistem ujian TOEFL dilakukan dalam tahapan-tahapan berikut:

- a. *Product backlog*
Seluruh kebutuhan sistem dirangkum dalam daftar *product backlog* berdasarkan hasil analisis masalah dari sistem TOEFL yang sebelumnya digunakan di Kursus Plug-in. Kebutuhan tersebut meliputi *input* soal *listening* dan *reading*, hasil ujian otomatis, dan laporan skor.
- b. *Sprint planning*
Tahap ini digunakan untuk merencanakan fitur-fitur yang akan dikembangkan dalam satu periode sprint. Sebagai contoh, pengembangan diawali dari fungsi *audio listening*, lalu berlanjut ke fitur laporan skor di *sprint* berikutnya.
- c. *Sprint* (pengembangan)
Proses pengembangan dilakukan dalam waktu yang telah ditentukan (misalnya dua minggu per sprint). Dalam tahap ini, dilakukan pembangunan fitur-fitur sistem menggunakan *Framework* Laravel dan MySQL sebagai basis data.
- d. *Daily Scrum*
Evaluasi dilakukan secara rutin untuk memantau progres pekerjaan harian, mengatasi hambatan yang dihadapi seperti kesalahan pemutaran audio atau validasi jawaban, serta memastikan setiap bagian berjalan sesuai rencana.
- e. *Sprint review*
Setelah satu *sprint* selesai, dilakukan peninjauan terhadap fitur yang sudah dikembangkan. Contohnya adalah pengujian fitur soal *listening* yang hanya dapat diputar sekali dan otomatis berpindah soal setelah waktu habis.
- f. *Sprint retrospective*
Evaluasi dilakukan terhadap proses pengembangan sebelumnya untuk mengidentifikasi kekurangan atau tantangan yang muncul serta menyusun langkah perbaikan untuk *sprint* berikutnya.

Dengan penggunaan *user story* dalam Scrum, pengembangan sistem menjadi lebih terarah pada kebutuhan nyata pengguna, meminimalkan risiko fitur yang tidak relevan, serta memastikan keberterimaan sistem saat diimplementasikan di Lembaga Kursus Plug-in. Penggunaan Scrum dalam proyek ini memungkinkan sistem ujian TOEFL dikembangkan secara bertahap dan terukur. Hal ini sangat penting mengingat sistem ujian TOEFL memerlukan validitas dan reliabilitas tinggi dalam menyajikan soal, mengelola waktu, serta menghitung skor peserta. Dengan metode Scrum, setiap fitur dapat diuji secara menyeluruh dan disesuaikan berdasarkan masukan pengguna dari lembaga Kursus Plug-in sebelum sistem diluncurkan secara penuh (Nurmasani et al., 2024).

3. Hasil

3.1. Analisis sistem

Analisis sistem adalah tahap penting dalam pengembangan sistem informasi. Pada tahap ini, kebutuhan pengguna diidentifikasi serta spesifikasi fungsionalitas dan teknis sistem yang akan dikembangkan. Hasil analisis ini berfungsi sebagai referensi untuk proses desain dan implementasi sistem.

Kebutuhan sistem dalam penelitian ini diperoleh melalui observasi tidak langsung dilakukan dengan menganalisis dokumentasi dan sistem yang sudah ada serta wawancara dengan pihak terkait, seperti admin dan pengelola. Selain itu, proses analisis dilakukan dengan mengacu pada pendekatan metodologi Scrum.

3.1.1. Identifikasi masalah

Sebelum membuat atau melakukan pengembangan terhadap suatu sistem, maka perlu dilakukan identifikasi masalah di mana hal ini bertujuan untuk membantu mengembangkan sistem dalam menentukan strategi dan rancangan sistem yang sesuai dengan keinginan dan kebutuhan dari pengguna sistem tersebut.

Berikut merupakan beberapa masalah yang dihadapi pada sistem ujian TOEFL di Lembaga kursus Plug-in:

- a. Setelah audio pada *section listening* diputar tidak dapat melanjutkan ujian.
- b. Sistem secara keliru menampilkan skor peserta dengan nilai 0.
- c. Nilai akhir peserta yang identik antara satu peserta dengan lainnya.

3.1.2. Analisis masalah

Dalam pengembangan sistem ujian TOEFL berbasis *website*, identifikasi dan analisis terhadap permasalahan pada sistem lama sangat penting dilakukan untuk memastikan sistem baru yang dikembangkan dapat menyelesaikan hambatan-hambatan tersebut. Berdasarkan hasil observasi dan pengujian, ditemukan beberapa permasalahan utama yang mengganggu kelancaran pelaksanaan ujian dan keakuratan hasil evaluasi. Berikut adalah analisis dari masing-masing permasalahan:

1. Setelah audio pada *section listening* diputar tidak dapat melanjutkan ujian.
Masalah ini menunjukkan bahwa terdapat kesalahan dalam manajemen alur soal atau fungsi pemutaran audio pada *section listening*. Sistem gagal memberikan transisi otomatis ke soal berikutnya setelah audio selesai diputar. Hal ini dapat disebabkan oleh:
 - a. Ketidaksesuaian logika antara kontrol audio dengan navigasi soal.
 - b. Tidak adanya *event listener* atau fungsi validasi yang menangani status pemutaran.
 - c. Kurangnya pengujian pada *section listening* secara menyeluruh.
2. Sistem secara keliru menampilkan skor peserta dengan nilai 0.
Skor 0 yang muncul meskipun peserta telah menyelesaikan

Tabel 1. *Product backlog*

No	Nama <i>Backlog</i>	Deskripsi	Prioritas
1	Navigasi dan Kontrol <i>Listening</i>	Menyempurnakan alur soal <i>listening</i> agar peserta dapat melanjutkan soal setelah audio selesai.	Tinggi
2	Logika Penghitungan Skor	Memperbaiki sistem evaluasi agar skor peserta dihitung berdasarkan jumlah jawaban benar.	Tinggi
3	Log Hasil Pengerjaan Soal	Menambahkan fitur pencatatan jumlah jawaban benar untuk keperluan audit dan validasi skor.	Tinggi

Tabel 2. *Sprint planning*

Sprint	Nama <i>Backlog</i>	Deskripsi Tugas	Estimasi Waktu
<i>Sprint</i> 1	Navigasi dan Kontrol <i>Listening</i>	Mendesain ulang alur soal <i>listening</i> setelah audio diputar & Membatasi audio hanya satu kali diputar.	1 minggu
<i>Sprint</i> 2	Logika Penghitungan Skor	Membaca jawaban peserta dari basis data secara akurat & Mengembangkan logika perhitungan skor berdasarkan jumlah jawaban benar.	1 minggu
<i>Sprint</i> 3	Log Hasil Pengerjaan Soal	Mencatat jumlah soal benar setiap peserta pada setiap <i>section</i> .	1 minggu

ujian menunjukkan kegagalan pada proses perhitungan skor atau penyimpanan data jawaban. Kemungkinan penyebab antara lain:

- Data jawaban peserta tidak terbaca dengan benar oleh sistem karena kesalahan *output*.
 - Perhitungan skor dilakukan sebelum data lengkap masuk ke sistem.
 - Skema penyimpanan basis data tidak berhasil menangkap hasil dengan benar (misalnya *field* kosong, *error* pada pengolahan *backend*).
3. Nilai akhir peserta yang identik antara satu peserta dengan lainnya.
Kesamaan skor antar peserta secara berulang menunjukkan potensi kegagalan pada sistem pengacakan soal atau fungsi penilaian. Beberapa kemungkinan penyebabnya:
- Fungsi perhitungan skor menggunakan nilai *default* atau kesalahan perulangan logika yang menghasilkan nilai tetap.
 - Kesalahan pada penyimpanan nilai ke dalam basis data yang menyebabkan data tertimpa atau disalin antar peserta.

Melalui analisis masalah ini, sistem baru yang dikembangkan diharapkan mampu menyelesaikan ketiga permasalahan inti dengan:

- Menyempurnakan alur navigasi dan kontrol pada *section listening*.
- Menjamin keakuratan logika penghitungan skor.
- Menambahkan fitur log hasil pengerjaan soal.

3.2. Implementasi Scrum

Pada titik ini, metode Scrum digunakan untuk mengembangkan sistem ujian TOEFL berbasis web. Metode ini dipilih karena merupakan salah satu yang paling banyak digunakan dan terorganisir untuk pengembangan perangkat lunak secara iteratif. Dalam penelitian ini, implementasi Scrum mencakup dokumentasi pekerjaan Scrum yang dilakukan secara berurutan, mulai dari tahap awal Scrum hingga tiga proses *sprint* yang digunakan untuk menyempurnakan fitur (Tumbade et al., 2024).

3.3. *Product backlog*

Pada saat ini, peneliti melakukan wawancara untuk menentukan kebutuhan sistem. Selanjutnya, peneliti membaca literatur yang relevan dengan penelitian ini. Ini termasuk mempelajari literatur tentang penelitian sebelumnya, diagram UML, ERD, basis data, dan metode Scrum Hasil wawancara dengan narasumber kemudian dianalisis dan dirangkum sebagai daftar kebutuhan sistem, yang dituangkan dalam Tabel 1. Tabel ini menunjukkan *product backlog* yang diperlukan beserta deskripsi dan prioritas masing-masing item, sehingga memudahkan tim pengembang dalam merencanakan tahapan pengembangan sistem.

3.3.1. *Sprint planning*

Setelah produk *backlog* diperoleh, peneliti merencanakan *sprint* untuk menentukan produk *backlog* mana yang akan dikerjakan selama *sprint*. Rencana *sprint* yang lengkap, termasuk nama *backlog*, deskripsi tugas, beserta estimasi waktu pengerjaan, ditampilkan pada Tabel 2. Tabel ini mempermudah tim pengembang dalam memprioritaskan tugas serta memantau progres pengembangan sistem secara terstruktur.

3.3.2. *Daily Scrum*

Daily Scrum adalah pertemuan singkat dan terjadwal yang dilakukan setiap hari selama *sprint* berlangsung. Tujuan utama dari kegiatan ini adalah untuk memantau progres pengembangan, mengidentifikasi hambatan, dan menyelaraskan rencana kerja harian agar *sprint* dapat diselesaikan tepat waktu.

Meskipun pengembangan sistem ujian TOEFL ini dilakukan secara individual, prinsip-prinsip dari *daily Scrum* tetap diterapkan untuk menjaga kedisiplinan dan transparansi terhadap jadwal yang telah dirancang. Evaluasi dilakukan secara mandiri melalui pencatatan harian mengenai progres fitur, kendala teknis yang ditemukan, serta rencana penyelesaian tugas untuk hari berikutnya.

3.3.2.1. *Sprint 1*

Sprint pertama difokuskan pada penyempurnaan sistem pada *section listening*, khususnya dalam hal kontrol pemutaran

Part 1

Ended

27. ☐ A. The Computers should have been replaced much earlier.
☒ B. The timing for replacing computers is about right.
☐ C. They don't need new computers at this time.
☐ D. They don't have much time to work with new computers.
28. ☐ A. She can see the man in three days.
☐ B. She doesn't know how to help the man.
☒ C. She can come on another day, but not tomorrow.
☐ D. She is thirsty and needs some water.
29. ☐ A. The man shouldn't spend his money at the bar.
☒ B. The man shouldn't give her candy.
☐ C. The man will break expensive equipment.
☐ D. The man needs change for a vending machine.
30. ☐ A. In a warehouse.
☒ B. In the car.
☐ C. On the street.
☐ D. At a grocery store.



Gambar 2. Section listening

audio, navigasi antar *part*, dan pengintegrasian *timer*. Walaupun sistem sebelumnya telah dapat digunakan, pengembangan ulang dilakukan untuk meningkatkan kontrol dan kesesuaian alur ujian dengan standar TOEFL. Salah satu tujuan utama adalah memastikan bahwa audio hanya dapat diputar satu kali, dan peserta hanya dapat melanjutkan ke *part* berikutnya setelah menyelesaikan mendengarkan audio.

Tabel 3 menunjukkan log harian selama *sprint* 1, yang merinci aktivitas dan hasil pengembangan setiap hari. *Daily Scrum* dilakukan setiap hari selama masa *sprint* satu minggu. Progres harian digunakan untuk mengimplementasikan fitur audio satu kali putar, menyusun struktur halaman *listening* per *part*, dan menyempurnakan transisi antar *part* soal. *Timer* juga dipasang agar peserta menyelesaikan ujian dalam waktu yang ditentukan.

Pemutaran audio dikendalikan dengan menggunakan *event listener* JavaScript *onended*, yaitu fungsi yang berjalan otomatis saat audio selesai diputar. Melalui logika ini, tombol “Next” atau “Finish” yang sebelumnya tidak dapat di klik akan bisa di klik saat audio selesai diputar, dan peserta baru dapat melanjutkan ke soal berikutnya. Hal ini dilakukan untuk memastikan bahwa audio benar-benar didengarkan sebelum peserta menjawab atau berpindah *part*.

Selain itu, peserta tidak dapat memutar ulang audio, serta tidak dapat di klik tombol navigasi selama audio masih berlangsung. Hal ini dilakukan untuk menjaga integritas ujian, dan memberikan pengalaman yang mendekati ujian TOEFL sesungguhnya.

Sebagaimana ditunjukkan pada Gambar 2, mekanisme kontrol pemutaran audio memastikan bahwa tombol “Next” hanya aktif setelah audio selesai diputar, sesuai mekanisme ujian TOEFL resmi. Pemutaran audio dikontrol dengan *event*

Tabel 3. Log harian *sprint* 1

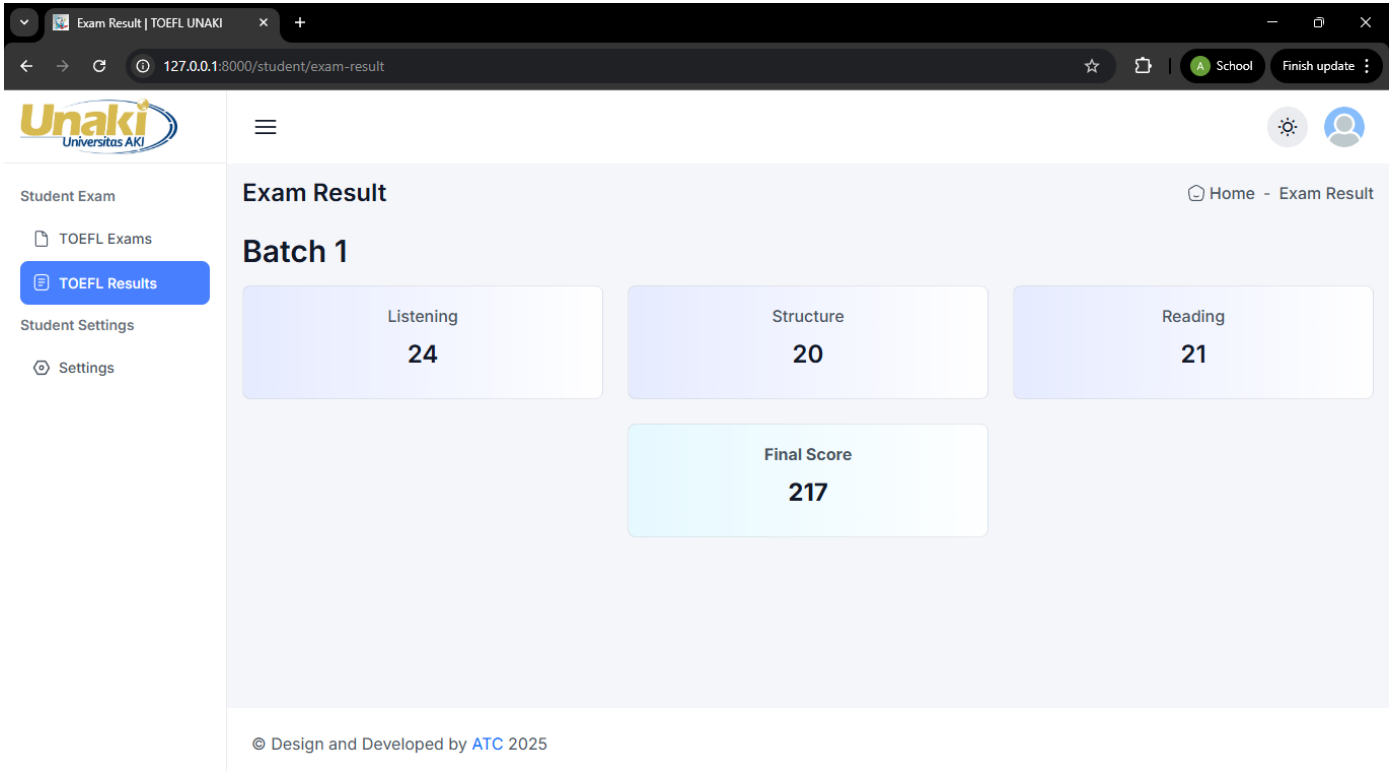
Hari	Aktivitas	Hasil
1	Perancangan ulang halaman <i>part listening</i>	Soal per <i>part</i> dipisah, tombol navigasi disiapkan.
2	Integrasi audio dan kontrol tombol <i>Play</i>	Audio hanya dapat diputar sekali, tombol dinonaktifkan setelah <i>play</i> .
3	Logika JavaScript <i>onended</i> , memunculkan tombol navigasi	Tombol navigasi hanya bisa di klik setelah audio selesai diputar.
4	Pengujian browser	Kompatibel lintas browser.

listener JavaScript *onended* (Khoirurrizal et al., 2024), sehingga tombol “Next” atau “Finish” hanya aktif setelah audio selesai diputar, mencegah peserta memutar ulang atau menggunakan tombol navigasi saat audio berjalan. Hal ini menjaga integritas ujian dan menyesuaikan mekanisme dengan ujian TOEFL resmi.

Dari gambar di atas menunjukkan *button next* hanya bisa di klik saat audio selesai (*ended*) dan tidak memutar ulang audio. Dengan ini akan lebih sesuai dengan mekanisme ujian TOEFL resmi.

3.3.2.2. *Sprint* 2

Sprint kedua berfokus pada validasi serta penguatan sistem penghitungan nilai TOEFL dan penyajian hasil ujian peserta per kategori (*Listening, Structure, dan Reading*). Meskipun sistem penilaian telah berjalan dengan baik sebelumnya, *sprint* ini dilaksanakan untuk memastikan bahwa seluruh proses perhitungan skor, konversi nilai, dan tampilan hasil ujian benar-benar sesuai dengan standar yang diharapkan.



Gambar 3. Tampilan hasil penghitungan skor

Fitur utama yang ditinjau dalam *sprint* ini meliputi fungsi `convertScore()` yang digunakan untuk mengonversi jumlah jawaban benar ke skor TOEFL berdasarkan kategori, serta fungsi `finalScore()` yang menghitung nilai akhir berdasarkan ketiga kategori dengan rumus skala TOEFL Paper-Based Test (PBT). Sistem menggunakan tabel konversi khusus untuk setiap kategori dan menghasilkan skor akhir dengan formula:

FinalScore =
$$\frac{(\text{Listening} + \text{Structure} + \text{Reading}) \times 10}{\text{CategoryCount}}$$

Melalui proses ini, sistem dapat menghasilkan skor akhir dalam kisaran 310–677 yang realistis, akurat, dan sesuai praktik umum penilaian TOEFL.

Log harian selama *sprint* 2 ditampilkan pada Tabel 4, yang merinci aktivitas dan hasil pengujian setiap hari. Tabel ini menunjukkan langkah-langkah validasi dan pengujian yang dilakukan, termasuk pengujian fungsi perhitungan skor, simulasi jawaban peserta, dan penyesuaian tampilan hasil ujian.

Untuk memastikan keakuratan logika sistem penilaian, dilakukan pengujian di mana tidak ada satu pun jawaban yang benar. Tujuan pengujian ini adalah untuk melihat bagaimana sistem menangani nilai minimum dan memastikan bahwa skor yang dihasilkan tetap konsisten dengan standar konversi TOEFL PBT.

Pengujian batas atas dengan semua jawaban benar menghasilkan skor akhir 677, sesuai dengan standar maksimum skor TOEFL PBT.

Selain itu, dilakukan tes acak dengan kombinasi jumlah benar yang bervariasi di tiap kategori (misalnya *listening* 35, *structure* 25, *reading* 40). perhitungan skor dari sistem dibandingkan dengan perhitungan manual pada seluruh percobaan, dan seluruh hasilnya menunjukkan kecocokan 100%, tanpa selisih nilai. Pengujian ini dibandingkan dengan

Tabel 4. Log harian *sprint* 2

Hari	Aktivitas	Hasil
1	Telah ulang fungsi <code>convertScore()</code> dan <code>finalScore()</code>	Terverifikasi bahwa sistem menggunakan tabel konversi standar TOEFL PBT
2	Simulasi pengisian jawaban dan penghitungan skor	Skor per kategori dan <i>final score</i> sesuai ekspektasi dan tidak anomali
3	Validasi potensi nilai 0 atau skor identik antar peserta	Sistem bekerja baik, nilai berbeda sesuai <i>input</i> pengerjaan
4	Uji silang hasil sistem dengan perhitungan manual	Tidak ada selisih skor, rumus perhitungan terbukti akurat. Uji silang dilakukan pada 10 set data jawaban peserta, mencakup variasi jumlah jawaban benar dari 0 hingga 50 pada setiap kategori. Pengujian mencakup tes batas bawah (0 benar), batas atas (50 benar), dan beberapa kombinasi acak jumlah benar per kategori. Seluruh hasil perhitungan manual identik dengan hasil sistem pada setiap skenario.
5-7	Penyesuaian tampilan hasil ujian dan pengujian <i>multi-batch</i>	Nilai dapat ditampilkan per <i>batch</i> , per kategori, dan <i>final score</i> jelas

perhitungan manual, dan seluruh hasil menunjukkan kecocokan 100%. Hasil simulasi dan tampilan skor akhir dapat dilihat pada Gambar 3.

Simulasi ini menunjukkan bahwa meskipun peserta tidak memberikan jawaban yang benar, sistem tetap memberikan

Tabel 5. Standar acuan konversi skor

Jumlah Benar	Listening	Structure	Reading
0	24	20	21
1	25	20	22
2	26	21	23
3	27	22	23
4	28	23	24
5	29	25	25
6	30	26	26
7	31	27	27
8	32	29	28
9	32	31	28
10	33	33	29
11	35	35	30
12	37	36	31
13	38	37	32
14	39	38	34
15	41	40	35
16	41	40	36
17	42	41	37
18	43	42	38
19	44	43	39
20	45	44	40
21	45	45	41
22	46	46	42
23	47	47	43
24	47	48	43
25	48	49	44
26	48	50	45
27	49	51	46
28	49	52	46
29	50	53	47
30	51	54	48
31	51	55	48
32	52	56	49
33	52	57	50
34	53	58	51
35	54	60	52
36	54	61	52
37	55	63	53
38	56	65	54
39	57	67	54
40	57	68	55
41	58	-	56
42	59	-	57
43	60	-	58
44	61	-	59
45	62	-	60
46	63	-	61
47	65	-	63
48	66	-	65
49	67	-	66
50	68	-	67

nilai akhir 217, sesuai dengan standar minimum skor TOEFL *Paper-Based Test* (PBT). Hal ini menunjukkan bahwa sistem telah menggunakan tabel konversi yang benar dan tidak mengembalikan nilai 0, yang tidak realistis dalam konteks penilaian TOEFL.

Selain itu, Tabel 5 menunjukkan standar acuan konversi skor untuk setiap kategori. Misalnya, peserta yang memperoleh jumlah jawaban benar nol pada semua bagian ujian akan dikonversikan menjadi nilai 24 untuk *listening*, 20 untuk *structure*, dan 21 untuk *reading*. Tabel ini digunakan sistem

Tabel 6. Log harian *sprint* 3

Hari	Aktivitas
1	Mengidentifikasi kebutuhan untuk mencatat jumlah soal benar per <i>section</i> per peserta.
2	Menelusuri struktur relasi antar model (Exam, ExamSession, UserAnswer, dll).
3	Mengembangkan logika <i>backend</i> untuk menghitung jumlah jawaban benar setiap <i>section</i> .
4	Menyimpan dan memformat hasil ke dalam struktur <i>array</i> terstruktur (per kategori/ <i>section</i>).
5	Menyisipkan data ini ke dalam tampilan <i>exam_result</i> agar bisa ditampilkan ke pengguna.

untuk memastikan bahwa kasus ekstrem tetap menghasilkan skor valid sesuai standar TOEFL.

Berdasarkan tabel, peserta memperoleh jumlah jawaban benar sebanyak nol pada semua bagian ujian, yaitu *listening*, *structure*, dan *reading*. jumlah jawaban benar sebesar nol akan dikonversikan menjadi nilai 24 untuk *listening*, 20 untuk *structure*, dan 21 untuk *reading*.

Sistem telah berhasil menangani kasus ekstrem (semua jawaban salah) dan tetap menghasilkan skor akhir yang valid sesuai dengan logika penilaian TOEFL. Dengan merujuk pada Tabel 4, Tabel 5, dan Gambar 3, dapat disimpulkan bahwa sistem konversi dan perhitungan skor bekerja secara stabil, terukur, dan sesuai standar.

3.3.2.3. *Sprint* 3

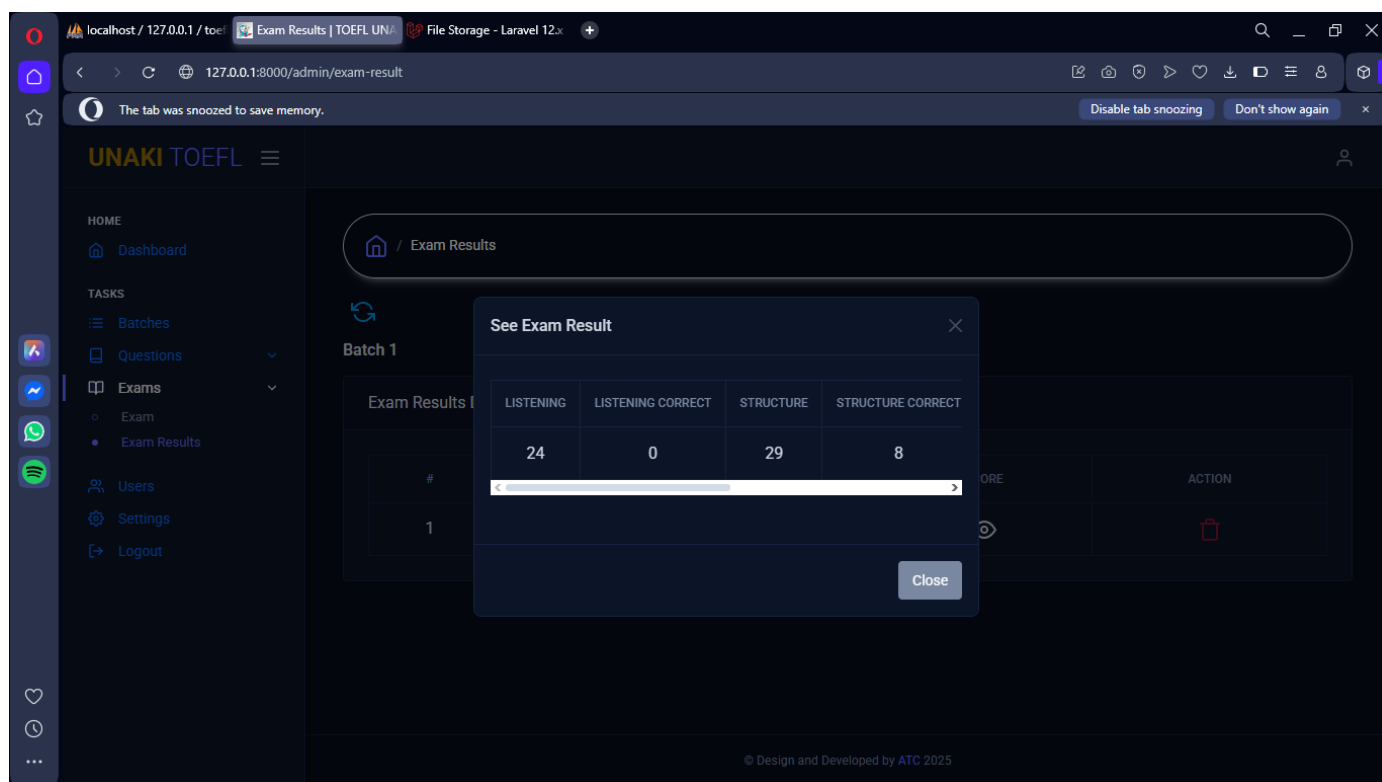
Pada *sprint* ketiga, fokus utama pengembangan adalah mencatat dan menampilkan jumlah soal yang dijawab dengan benar oleh setiap peserta ujian TOEFL pada masing-masing *section*, yaitu *listening*, *structure*, dan *reading*. Fitur ini bertujuan untuk memberikan informasi yang lebih detail kepada peserta tentang performa mereka dalam setiap bagian ujian. Selain menampilkan skor akhir yang telah dikonversi, sistem kini juga menampilkan metrik jumlah jawaban benar dan total soal pada setiap kategori. Log harian pengembangan *sprint* 3 dapat dilihat pada Tabel 6, yang merinci aktivitas pengembangan setiap hari.

Sprint ini bertujuan untuk menghitung dan mencatat jumlah jawaban benar yang diperoleh peserta untuk masing-masing *section* dalam ujian TOEFL. Selain meningkatkan transparansi hasil, fitur ini juga memberikan nilai edukatif agar peserta dapat mengetahui bagian mana yang perlu ditingkatkan.

Informasi ini memperkaya laporan hasil ujian dan memberikan wawasan yang lebih dalam kepada peserta maupun pengajar untuk melakukan evaluasi dan pembelajaran yang lebih terarah. Hasil implementasi fitur jumlah jawaban benar dapat dilihat pada Gambar 4, yang menampilkan tampilan antarmuka sistem saat menampilkan skor akhir beserta jumlah jawaban benar per *section*. Dengan adanya gambar ini, pembaca dapat memahami bagaimana sistem menyajikan informasi secara jelas dan informatif kepada pengguna.

3.3.3. *Sprint* review

Sprint review dilakukan setelah penyelesaian tiga *sprint* utama dalam pengembangan sistem ujian TOEFL berbasis web. Review ini bertujuan untuk mengevaluasi hasil kerja pengembang, memastikan fitur berjalan sesuai ekspektasi, serta



Gambar 4. Tampilan hasil jawaban benar

Tabel 7. Ringkasan *sprint review*

Kebutuhan Fitur	Sprint	Estimasi	Catatan
Mendesain ulang alur soal <i>listening</i> setelah audio diputar & Membatasi audio hanya satu kali diputar.	<i>Sprint 1</i>	1 Minggu	Fitur bekerja sesuai harapan, diuji dengan sukses, UX berjalan dengan baik.
Membaca jawaban peserta dari basis data secara akurat & Mengembangkan logika perhitungan skor berdasarkan jumlah jawaban benar.	<i>Sprint 2</i>	1 Minggu	Konversi nilai sudah sesuai standar, final <i>score</i> muncul dengan benar di halaman hasil.
Mencatat jumlah soal benar setiap peserta pada setiap <i>section</i> .	<i>Sprint 3</i>	1 Minggu	Informasi hasil lebih detail.

mengidentifikasi peluang pengembangan lanjutan.

Pada *sprint 1*, sistem berhasil menyelesaikan bagian *Listening Section* yang mencakup pemutaran audio otomatis, pengaturan timer, dan penguncian tombol navigasi selama audio belum selesai. Fitur ini diuji oleh pengguna dan berfungsi dengan baik. Penambahan logika *audio.onended* terbukti efektif untuk menjaga integritas ujian.

Pada *Sprint 2*, sistem berhasil menghitung skor TOEFL berdasarkan konversi jawaban benar ke nilai standar. Nilai dari setiap *section listening, structure, dan reading* dikonversi menggunakan tabel yang telah ditentukan, lalu dihitung *final score* menggunakan rumus standar TOEFL. Sistem juga menampilkan hasil skor akhir dengan baik di halaman hasil ujian.

Kemudian, *sprint 3* berfokus pada pencatatan jumlah soal yang dijawab benar oleh peserta pada masing-masing *section*. Hasil perhitungan ini ditampilkan dalam halaman hasil ujian dan memberikan informasi yang lebih komprehensif bagi peserta. Pengguna dapat mengetahui tidak hanya skor, tetapi juga performa riil dari segi jumlah jawaban benar dan total soal pada setiap bagian.

Ringkasan hasil pengembangan dan implementasi fitur dari ketiga *sprint* dapat dilihat pada Tabel 7, yang menunjukkan kebutuhan fitur, *sprint* terkait, estimasi waktu, serta catatan hasil implementasi. Dari tabel ini, pembaca dapat memahami progres setiap *sprint*, fungsionalitas yang diuji, serta peningkatan pengalaman pengguna yang dicapai.

Selama *sprint* berlangsung, komunikasi antar tim dilakukan secara rutin melalui *daily Scrum* untuk memantau perkembangan harian. Setiap hambatan teknis yang ditemukan segera dibahas dan ditangani. Pengujian dilakukan setelah setiap *sprint* untuk memastikan bahwa fitur berjalan stabil dan sesuai dengan *user story* yang telah disepakati.

Secara keseluruhan, semua fitur inti yang direncanakan telah berhasil diimplementasikan dengan baik. Sistem bekerja stabil dan dapat digunakan oleh peserta untuk melakukan ujian TOEFL secara daring. *Feedback* dari pengguna selama *review* menunjukkan kepuasan atas kelengkapan fitur dan kemudahan penggunaan.

Tabel 8 menunjukkan perkembangan tiap *sprint* dalam proyek pengembangan sistem ujian TOEFL. Setiap *sprint* menghasilkan fitur yang diuji, serta mencakup penambahan fungsionalitas dan peningkatan pengalaman pengguna.

3.3.4. *Sprint retrospective*

Pada tahap *sprint Retrospective*, tim merefleksikan hasil kerja

selama tiga sprint. Beberapa hal positif yang dicapai antara lain:

- Sprint 1*: Fitur *audio*, *timer*, dan tombol navigasi berfungsi baik, namun dokumentasi dan uji coba awal perlu ditingkatkan.
- Sprint 2*: Konversi skor TOEFL berjalan akurat, tetapi perlu pengujian untuk kasus ekstrem dan dokumentasi tabel nilai.
- Sprint 3*: Penambahan data soal benar/salah memperjelas hasil peserta, meskipun kode *controller* perlu dirapikan dan UI admin ditambahkan ke *backlog*.

Langkah perbaikan meliputi: pembuatan cek list uji coba, dokumentasi perubahan, penambahan pengujian ekstrem, *refactoring* kode, dan perencanaan fitur lanjutan.

Hasil refleksi ini dapat dilihat secara terstruktur pada Tabel 8, yang merangkum kebutuhan fitur, keterangan implementasi, dan evaluasi selama *sprint retrospective*. Dari tabel ini, pembaca dapat memahami fitur mana yang sudah berjalan baik, fitur yang membutuhkan dokumentasi tambahan, serta fitur yang perlu diperbaiki atau ditambahkan pengujian lanjutan.

Tabel di atas menjabarkan berbagai kebutuhan fitur yang telah diidentifikasi dan dikembangkan selama proses Scrum, lengkap dengan keterangan implementasinya serta evaluasi dalam sesi *sprint retrospective*. Melalui tabel ini, dapat dilihat bahwa sebagian besar fitur telah berhasil diterapkan sesuai dengan kebutuhan, meskipun beberapa fitur masih memerlukan penyempurnaan dari sisi dokumentasi dan pengujian lanjutan. Evaluasi ini menjadi acuan penting untuk iterasi dan perbaikan sistem pada *sprint* selanjutnya.

4. Pembahasan

Hasil pengembangan sistem ujian TOEFL berbasis web pada Lembaga Kursus Plug-in menunjukkan bahwa penerapan metode Scrum efektif dalam menyelesaikan permasalahan yang ada pada sistem sebelumnya. Proses pengembangan yang bersifat iteratif memudahkan tim untuk melakukan perbaikan berdasarkan umpan balik yang diperoleh pada setiap *sprint*, sehingga kualitas sistem meningkat secara bertahap.

Penerapan kontrol *audio* dengan fitur *once play* melalui *event listener* onended mampu mencegah peserta mengulang audio secara tidak sah, sehingga integritas ujian dapat terjaga. Selain itu, logika perhitungan skor yang mengacu pada tabel konversi TOEFL PBT menghasilkan penilaian yang lebih akurat dan menghindari kesalahan nilai nol atau skor identik antar peserta. Fitur *score reporting* yang rinci juga mempermudah instruktur dalam melakukan evaluasi per bagian, seperti *Listening*, *Structure*, dan *Reading*.

Temuan ini sejalan dengan penelitian sebelumnya yang menunjukkan bahwa sistem ujian daring yang dibangun dengan pendekatan iteratif cenderung lebih stabil dan *user-friendly*. Secara praktis, sistem ini dapat menjadi referensi bagi pengembang lain yang ingin membangun platform ujian berbasis web, khususnya dalam menjaga validitas dan reliabilitas hasil.

Meskipun pengujian sistem secara fungsional telah dilakukan secara internal melalui *daily Scrum*, uji silang perhitungan skor, dan pengujian batas ekstrem (nilai minimum dan maksimum), pengujian berbasis pengguna akhir (*User Acceptance Test / UAT*) belum dapat dilaksanakan pada tahap ini karena keterbatasan waktu dan logistik. Hasil UAT diharapkan

Tabel 8. Ringkasan *sprint retrospective*

No	Kebutuhan Fitur	Keterangan	<i>Sprint Retrospective</i>
1	Navigasi dan kontrol audio <i>Listening</i>	Audio harus berjalan otomatis, tombol "Next" aktif setelah audio selesai	Sudah berjalan baik, namun perlu dokumentasi teknis dan validasi tambahan
2	Konversi skor ke skala TOEFL	Akurasi Tinggi	Sudah dilakukan uji coba nilai ekstrem (semua jawaban salah) dengan hasil skor minimum 217 sesuai standar TOEFL PBT. Tabel konversi nilai minimum hingga maksimum telah disiapkan untuk dokumentasi.
3	Tampilkan hasil jawaban benar dan belum dijawab	Admin melihat jumlah soal benar	Sudah diterapkan, perlu perapian <i>controller</i>
4	Menyediakan informasi final <i>score</i> dan detail per kategori	Peserta melihat skor tiap kategori dan <i>final score</i>	Berfungsi baik, disarankan ditambahkan grafik visualisasi untuk peningkatan UI/UX

dapat memberikan evaluasi langsung dari peserta dan pengelola lembaga mengenai kelayakan sistem, kemudahan penggunaan, serta kesesuaian fitur dengan kebutuhan nyata. Implementasi UAT direkomendasikan pada pengembangan selanjutnya agar sistem dapat diuji lebih komprehensif sebelum digunakan secara luas.

Keterbatasan penelitian lainnya meliputi pengujian yang hanya dilakukan pada satu lembaga kursus, sehingga hasilnya belum mewakili kondisi di lokasi lain dengan variasi perangkat dan kualitas jaringan berbeda. Selain itu, aspek *user experience* belum diukur secara kuantitatif, sehingga evaluasi kepuasan pengguna masih bersifat kualitatif.

5. Kesimpulan

Berdasarkan hasil pengembangan sistem ujian TOEFL berbasis *website* menggunakan metode Scrum di Lembaga Kursus Plug-in, dapat disimpulkan bahwa penerapan metode Scrum memberikan pendekatan yang efektif dan terstruktur dalam membagi tahapan kerja menjadi *sprint-sprint* yang terukur. Melalui tiga *sprint* yang dilakukan, sistem berhasil dikembangkan secara bertahap dengan pengujian dan evaluasi yang konsisten. Sistem ini mampu menyelesaikan berbagai permasalahan yang ditemukan pada sistem sebelumnya, seperti ketidaksempurnaan navigasi pada bagian *listening*, akurasi perhitungan skor yang belum sesuai, serta belum adanya pencatatan hasil pengerjaan peserta secara terperinci. Fitur fitur penting seperti *timer*, kontrol *audio*, konversi skor otomatis, hingga pencatatan jumlah jawaban benar telah berhasil diimplementasikan sesuai kebutuhan. Secara keseluruhan, sistem yang dikembangkan telah berjalan dengan baik dan siap digunakan dalam proses simulasi atau pelatihan TOEFL secara digital.

Saran yang dapat diberikan untuk pengembangan lebih

lanjut sistem ini adalah sebagai berikut:

- Pengujian sistem sebaiknya dilakukan secara lebih luas dan berkelanjutan agar dapat menemukan *bug* atau kekurangan dari berbagai sisi penggunaan.
- Antarmuka pengguna (UI/UX) perlu dioptimalkan agar lebih ramah pengguna dan responsif di berbagai perangkat, terutama perangkat *mobile*.
- Fitur ekspor data hasil ujian ke format PDF atau Excel dapat ditambahkan untuk mempermudah dokumentasi dan pelaporan.
- Sistem dapat ditingkatkan dengan menambahkan fitur rekaman jawaban peserta secara lebih detail, seperti mencatat pilihan jawaban mana yang dipilih pada setiap soal, guna keperluan analisis lebih lanjut.

Ketersediaan data

Seluruh data penulis telah disajikan di dalam artikel.

Deklarasi konflik kepentingan

Para penulis menyatakan tidak mempunyai kepentingan finansial yang bersaing atau koneksi pribadi yang dapat dianggap mempengaruhi penelitian yang disajikan dalam artikel ini.

Kontribusi penulis

Semua penulis mendesain artikel, berkontribusi dalam penulisan konten, dan merevisi naskah artikel. Semua penulis membaca dan menyetujui naskah versi akhir.

Daftar rujukan

- Abdi, M., Amru, A., Rafiq, S. F., & Safriadi, S. (2024). Implementasi algoritma fisher-yates untuk pengacakan dinamis soal TOEFL dalam ujian online berbasis web. *Jurnal Nasional Komputasi Dan Teknologi Informasi (JNKTI)*, 7(5), 1286–1291. <https://doi.org/10.32672/jnkti.v7i5.8086>
- Diraja, Z. (2018). *Pembangunan aplikasi simulasi tes TOEFL berbasis web* [Universitas Esa Unggul]. <https://digilib.esaunggul.ac.id/public/UEU-Undergraduate-10082-jurnal.Image.Marked.pdf>
- Febriansyah, M. R., & Voutama, A. (2024). Rancang bangun sistem ujian online berbasis website menggunakan metode waterfall. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(3), 2640–2647. <https://doi.org/10.36040/jati.v8i3.9563>
- Khazanah, D. P. N., & Purnama, G. (2024). Perancangan sistem ujian online menggunakan metode pengembangan Waterfall berbasis web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(2), 2332–2339. <https://doi.org/10.36040/jati.v8i2.9475>
- Khoirurrizal, M. F., Hidayat, C. R., & Ruuhwan, R. (2024). Analisis perbandingan framework front-end Javascript Solidjs dan Vuejs pada pengembangan website interaktif. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(2). <https://doi.org/10.23960/jitet.v12i2.4106>
- Lubis, L. R., & Irmayana, A. (2019). Analisis kesulitan mahasiswa IPTS dalam menyelesaikan soal-soal TOEFL. *Jurnal Education and Development*, 7(3), 118. <https://doi.org/10.37081/ed.v7i3.1202>
- Marbun, Y. Y., Isnanto, R. R., & Martono, K. T. (2016). Pembuatan aplikasi TOEFL sebagai media pelatihan Bahasa Inggris berbasis web. *Jurnal Teknologi Dan Sistem Komputer*, 4(1), 83. <https://doi.org/10.14710/jtsiskom.4.1.2016.83-92>
- Maulana, C., & Lubis, I. A. (2022). Students' problem of taking TOEFL test at STMIK Royal Academic year 2020/2021. *Journal of Science and Social Research*, 5(1), 5. <https://doi.org/10.54314/jssr.v5i1.811>
- Noveandini, R., Wulandari, M. S., & Hakim, A. (2023). Penerapan Metode Scrum pada rancang bangun sistem informasi penjualan Toko Sepatu Rabbani Shoes. *Seminar Nasional Teknologi Informasi Dan Komunikasi STI&K (SeNTIK)*, 192–198. <https://ejournal.jak-stik.ac.id/index.php/sentik/article/view/3454>
- Nurmasani, A., Kurniawan, F. D., Hartanto, A. D., & Fajri, I. N. (2024). Penerapan Metode Scrum pada pengembangan sistem informasi pencatatan magang. *Information System Journal*, 7(01), 34–44. <https://doi.org/10.24076/infosjournal.2024v7i01.1616>
- Oktapianti, S. (2024). TOEFL score of Tadris Chemistry students at UIN Antasari in achievement of CEFR standard. *Tarbiyah: Jurnal Ilmiah Kependidikan*, 13(1), 55–63. <https://doi.org/10.18592/tarbiyah.v13i1.11417>
- Pauzi, Y., & Kharisma, I. L. (2025). Pengembangan website ujian online pada SMP Putra Perintis. *MAJU: Indonesian Journal of Community Empowerment*, 2(1), 153–165. <https://doi.org/10.62335/cqex4k29>
- Sanjaya, H. R., Situmorang, L., M Syahrul, Kalmay, L., & Pamungkas, R. W. P. (2025). Pengembangan sistem ujian online berbasis mobile dengan Agile-Scrum untuk meningkatkan efisiensi dan pengalaman pengguna. *Jurnal Kecerdasan Buatan Dan Teknologi Informasi*, 4(1), 60–69. <https://doi.org/10.69916/jkbt.v4i1.209>
- Tumbade, M. O., Hartomo, K. D., & Purnomo, H. D. (2024). Pengembangan perangkat lunak berbasis website menggunakan kombinasi Metode Scrum dan V-Model. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(3), 447–460. <https://doi.org/10.25126/jtiik.938456>
- Wijaya, R. I., Usman, M. L. L., Wibowo, F. M., & Gustalika, M. A. (2024). Rancang bangun sistem informasi tracer study berbasis website menggunakan Metode Scrum (studi kasus: Institut Teknologi Telkom Purwokerto). *Journal of Software Engineering and Multimedia (JASMED)*, 2(1), 10–20. <https://doi.org/10.20895/jasmed.v2i1.1235>



Arya Bima Wijaya saat ini sedang menempuh gelar Sarjana Komputer pada Fakultas Teknik dan Informatika, Universitas AKI (Abadi Karya Indonesia) Semarang. Fokus penelitiannya saat ini adalah pengembangan sistem informasi dan aplikasi berbasis web, khususnya pada pengembangan *website* ujian TOEFL.



Yohana Tri Widayati saat ini, merupakan dosen aktif di Fakultas Teknik dan Informatika, Universitas AKI Semarang.



Satrio Agung Prakoso merupakan salah satu dosen pengajar di Fakultas Teknik dan Informatika Universitas AKI Semarang. Saat ini beliau sedang menjabat sebagai ketua program studi Teknik Informatika pada Universitas AKI Semarang.