

DOKUMENTASI TEKNIS

Ujian Multi-Sesi & Analisis IRT

Panduan Fitur Fase 6 dan Fase 7 Sistem CBT ExamPro — SMAN 1 Kabanjahe

Dibuat: 25 Maret 2026 · Developer: Erik (Guru / IT Operator)

DAFTAR ISI

1. Gambaran Umum — Apa yang Berubah?
2. Fase 6 — Ujian Multi-Sesi (Multi-Section)
 1. Apa Itu Ujian Multi-Sesi?
 2. Cara Membuat Ujian Multi-Sesi
 3. Alur Siswa Mengerjakan
 4. Cara Menghitung Nilai
 5. Monitor Real-Time
 6. Ekspor Excel
3. Fase 7 — Analisis IRT (Item Response Theory)
 1. Apa Itu IRT dan Mengapa Penting?
 2. Tiga Model IRT yang Digunakan
 3. Parameter IRT: b , a , dan c
 4. Skor Kemampuan Siswa (θ)
 5. Item Bundle: PGK dan Menjodohkan
 6. Akumulasi Data Lintas Ujian
 7. Cara Membaca Halaman Analisis IRT
4. Cara Mengaktifkan dan Menjalankan
5. Perbedaan Analisis Klasikal vs IRT

1 Gambaran Umum — Apa yang Berubah?

Pada sesi pengembangan ini, dua fitur besar ditambahkan ke sistem CBT ExamPro:

Fase 6 — Ujian Multi-Sesi

Satu paket ujian kini bisa dipecah menjadi beberapa **sesi (section)** dengan soal dan timer yang terpisah. Cocok untuk ujian yang mencakup lebih dari satu mata pelajaran atau topik besar.

Fase 7 — Analisis IRT

Setelah ujian ditutup, sistem menghitung parameter statistik lanjutan untuk setiap soal menggunakan **Item Response Theory** — metode psikometrika yang lebih akurat dari analisis klasikal biasa.

Kedua fitur ini bersifat **opsional dan kompatibel ke belakang** — ujian lama yang tidak menggunakan multi-sesi tetap berjalan normal.

2 Fase 6 — Ujian Multi-Sesi (Multi-Section)

2.1 Apa Itu Ujian Multi-Sesi?

Biasanya satu ujian memiliki satu set soal dengan satu timer. Dengan fitur Multi-Sesi, proktor dapat memecah ujian menjadi **beberapa bagian (section)**, misalnya:

Section	Nama	Jumlah Soal	Durasi	Jeda Antar Sesi
1	Pilihan Ganda	30 soal	45 menit	90 detik
2	Essay	5 soal	45 menit	—

Setiap section memiliki **soal sendiri, timer sendiri, dan nilai sendiri**. Soal tidak bisa dipakai di dua section sekaligus dalam ujian yang sama.

Catatan penting: Nilai akhir ujian multi-sesi adalah **rata-rata** dari nilai semua section, bukan total. Jadi jika Section 1 nilainya 80 dan Section 2 nilainya 70, nilai akhir = 75.

2.2 Cara Membuat Ujian Multi-Sesi (Proktor)

1 Buka form buat ujian

Klik "Buat Ujian" di halaman daftar ujian.

2 Aktifkan toggle "Ujian Multi-Sesi"

Ada di panel "Pengaturan Ujian". Saat diaktifkan, panel kiri akan berubah: kolom durasi total disembunyikan, diganti dengan builder section.

3 Tambahkan section satu per satu

Klik "+ Tambah Sesi", isi nama section, durasi (menit), dan jeda sebelum section berikutnya (detik). Urutan bisa digeser naik/turun.

4 Pilih soal per section

Panel kanan menampilkan tab satu per section. Klik tab "Sesi 1", pilih soal yang sesuai. Soal yang sudah dipilih di section lain akan terlihat redup dan tidak bisa dipilih ulang.

5 Atur bobot soal per section

Setiap section harus memiliki total bobot tepat 100. Gunakan "Terapkan Distribusi Bobot" untuk otomatis membagi rata.

Simpan dan aktifkan ujian

- Klik "Simpan". Sistem akan menghitung total durasi otomatis dari jumlah durasi semua section.

Perhatian: Sebelum mengaktifkan ujian (status → Active), sistem akan memeriksa bahwa setiap section memiliki total bobot tepat 100. Jika tidak, ujian tidak bisa diaktifkan.

2.3 Alur Siswa Mengerjakan

1 Masuk ujian dengan token

Sama seperti ujian biasa. Setelah memasukkan token, siswa langsung masuk ke Section 1.

2 Kerjakan Section 1

Timer Section 1 berjalan. Nomor soal di sidebar hanya menampilkan soal untuk section aktif. Di bagian atas terlihat informasi "Sesi 1/2 — Pilihan Ganda".

3 Submit Section 1

Klik "Kumpulkan". Nilai section langsung dihitung. Siswa tidak bisa kembali ke Section 1 setelah ini.

4 Layar jeda (Istirahat)

Muncul countdown sesuai jeda yang diatur proktor (misal 90 detik). Ada tombol "Lanjutkan Sekarang" jika siswa ingin langsung ke section berikutnya.

5 Kerjakan Section 2, 3, dst.

Soal dan timer section berikutnya dimuat tanpa reload halaman. Timer dimulai dari nol untuk setiap section.

6 Section terakhir selesai → Halaman Hasil

Setelah submit section terakhir, siswa langsung diarahkan ke halaman hasil. Nilai akhir = rata-rata semua section.

2.4 Cara Menghitung Nilai

Nilai tiap section dihitung sama seperti ujian biasa (skor soal sesuai bobot masing-masing). Nilai akhir dihitung sebagai berikut:

$$\text{Nilai Akhir} = (\text{Nilai Section 1} + \text{Nilai Section 2} + \dots + \text{Nilai Section N}) / N$$

Contoh ujian 3 section:

Section	Nama	Nilai Section
1	PG Biologi	85

2	PG Fisika	70
3	Essay Kimia	90
Nilai Akhir		$(85+70+90)/3 = 81.67$

2.5 Monitor Real-Time (Proktor)

Halaman Monitor menampilkan kolom tambahan "**Sesi**" untuk ujian multi-sesi, menunjukkan section mana yang sedang dikerjakan oleh setiap siswa, contoh:

Nama Siswa	Status	Dijawab	Sesi	Nilai Sementara
Budi Santoso	Mengerjakan	28/30	1/2 — PG	82.50
Sari Dewi	Mengerjakan	4/5	2/2 — Essay	75.00

Saat proktor menambah waktu dari halaman Monitor, waktu ditambahkan ke **section yang sedang aktif** (bukan ke seluruh ujian).

2.6 Ekspor Excel

File Excel hasil ujian multi-sesi memiliki kolom tambahan untuk nilai setiap section:

No	Nama	NIS	Kelas	Sesi 1 — PG	Sesi 2 — Essay	Nilai Akhir	Status
1	Budi Santoso	12345	X IPA 1	85.00	90.00	87.50	Dikumpulkan

3 Fase 7 — Analisis IRT (Item Response Theory)

3.1 Apa Itu IRT dan Mengapa Penting?

Selama ini, kualitas soal diukur menggunakan **analisis klasikal** (Tingkat Kesukaran dan Daya Pembeda). Analisis ini sudah cukup baik, tetapi memiliki keterbatasan: hasilnya bergantung pada kemampuan kelompok siswa yang mengerjakan ujian itu.

Artinya, soal yang sama bisa terlihat "mudah" jika dikerjakan siswa pintar, dan "sulit" jika dikerjakan siswa lemah.

IRT (Item Response Theory) mengatasi hal ini dengan memisahkan:

- **Kemampuan siswa (θ)** — seberapa pintar siswa terlepas dari soal mana yang dikerjakan
- **Karakteristik soal (b , a , c)** — seberapa sulit soal, seberapa baik soal membedakan siswa, dan berapa peluang menebak

Manfaat utama IRT: Parameter soal (b , a , c) bersifat *invariant* — nilainya tidak berubah meskipun diujikan ke kelompok siswa yang berbeda. Ini memungkinkan perbandingan kualitas soal lintas ujian secara adil.

3.2 Tiga Model IRT yang Digunakan

Sistem memilih model secara otomatis berdasarkan jumlah responden per soal:

Model	Kapan Digunakan	Parameter	Catatan
1PL (Rasch)	$n < 50$ responden	b saja ($a=1, c=0$)	Paling sederhana, stabil untuk data kecil
2PL	$50 - 149$ responden	b dan a ($c=0$)	Lebih akurat, memperhitungkan diskriminasi soal
3PL	≥ 150 responden	$b, a, \text{ dan } c$	Paling lengkap, memperhitungkan faktor tebakan

Seiring bertambahnya data (soal dipakai di banyak ujian), model bisa **naik kelas otomatis** dari 1PL ke 2PL atau 3PL.

3.3 Parameter IRT: b , a , dan c

Parameter b — Kesulitan (Difficulty)

Nilai b menunjukkan tingkat kesulitan soal pada skala θ . Semakin besar b , semakin sulit soal tersebut.

Rentang b	Kategori	Artinya
$b \leq -1.5$	Sangat Mudah	Hanya siswa dengan kemampuan sangat rendah yang mungkin gagal
$-1.5 < b \leq -0.5$	Mudah	Sebagian besar siswa bisa menjawab benar
$-0.5 < b \leq 0.5$	Sedang	Hanya siswa kemampuan rata-rata ke atas yang bisa menjawab

$0.5 < b \leq 1.5$	Sulit	Hanya siswa berkemampuan tinggi yang bisa menjawab
$b > 1.5$	Sangat Sulit	Sangat sedikit siswa yang berhasil menjawab benar

Perbedaan b vs TK klasikal: TK klasikal = proporsi siswa yang menjawab benar (dipengaruhi kemampuan kelompok). Parameter b IRT tidak dipengaruhi kemampuan kelompok — lebih objektif untuk membandingkan soal lintas kelas atau angkatan.

Parameter a — Diskriminasi (Discrimination)

Nilai a menunjukkan seberapa baik soal membedakan siswa yang mampu dari yang tidak. Semakin besar a, semakin "tajam" soal dalam memisahkan siswa pandai dan siswa lemah.

Rentang a	Kategori
$a < 0.5$	Lemah — soal hampir tidak membedakan siswa
$0.5 \leq a < 1.0$	Sedang — diskriminasi cukup
$1.0 \leq a < 2.0$	Baik — ideal untuk soal ujian
$a \geq 2.0$	Sangat Baik — soal sangat tajam

Parameter c — Tebakan (Pseudo-Guessing)

Parameter c hanya ada di model 3PL. Nilainya menunjukkan peluang minimum siswa dengan kemampuan sangat rendah sekalipun bisa menjawab benar (karena menebak).

Contoh: $c = 0.20$ artinya bahkan siswa paling lemah pun memiliki ~20% peluang menjawab benar karena menebak. Ini wajar untuk soal PG 5 pilihan (peluang tebak acak = $1/5 = 20\%$).

Kurva ICC (Item Characteristic Curve)

ICC adalah grafik yang menunjukkan hubungan antara kemampuan siswa (θ) dengan probabilitas menjawab benar. Sumbu X adalah kemampuan siswa, sumbu Y adalah probabilitas benar (0–100%).



3.4 Skor Kemampuan Siswa (θ) — Skala 0–1000

Selain parameter soal, IRT juga menghasilkan **estimasi kemampuan (θ)** untuk setiap siswa. Nilai θ mentah berada pada skala sekitar -4 sampai $+4$.

Agar lebih mudah dipahami, θ dikonversi ke **skala 0–1000**:

θ Mentah	Skor IRT (0–1000)	Interpretasi
+2.0	700	Kemampuan tinggi
+1.0	600	Di atas rata-rata
0.0	500	Rata-rata
-1.0	400	Di bawah rata-rata
-2.0	300	Kemampuan rendah

Skor IRT ini **dapat dibandingkan lintas ujian** dalam mata pelajaran yang sama, karena didasarkan pada karakteristik soal yang sama.

3.5 Item Bundle: PGK dan Menjodohkan

Untuk soal **PG Kompleks** dan **Menjodohkan**, sistem menggunakan pendekatan *item bundle* — setiap opsi atau pasangan dianalisis sebagai sub-item IRT tersendiri.

PG Kompleks (PGK)

Misalkan soal PGK dengan kunci jawaban A, C, D. Sistem akan menganalisis 3 sub-item:

- Sub A: apakah siswa memilih opsi A? (seharusnya: ya)
- Sub C: apakah siswa memilih opsi C? (seharusnya: ya)
- Sub D: apakah siswa memilih opsi D? (seharusnya: ya)

Setiap sub-item menghasilkan parameter b, a, c sendiri — sehingga bisa diketahui opsi mana yang paling sulit untuk "ditemukan" oleh siswa.

Menjodohkan

Setiap pasangan premis-respons dianalisis terpisah. Jika ada 4 pasangan, akan ada 4 sub-item, dan bisa diketahui pasangan mana yang paling sering salah dipasangkan.

3.6 Akumulasi Data Lintas Ujian

Salah satu keunggulan implementasi IRT di sistem ini adalah **akumulasi data**: setiap kali soal digunakan dalam ujian baru, estimasi parameternya menjadi semakin akurat.

Ujian pertama — 30 siswa

- 1 Model 1PL digunakan ($n < 50$). Parameter b dihitung, disimpan ke bank soal.
- 2 **Ujian kedua — 40 siswa lagi (total 70)**
Model naik ke 2PL ($n \geq 50$). Parameter b dan a dihitung dari 70 data, menggantikan estimasi lama.
- 3 **Ujian ketiga — 100 siswa lagi (total 170)**
Model naik ke 3PL ($n \geq 150$). Parameter b, a, dan c dihitung dari 170 data — paling akurat.

Data respons biner per soal tersimpan di database (`answers.irt_u`) sehingga tidak perlu mengulang proses skoring dari awal setiap kali kalibrasi ulang.

3.7 Cara Membaca Halaman Analisis IRT

Buka halaman Analisis (dari halaman Detail Ujian → tombol "Analisis Butir"). Di bagian atas terdapat dua tab:

Tab: Analisis Klasikal

Menampilkan TK (Tingkat Kesukaran), DP (Daya Pembeda), dan distribusi pilihan — fitur lama yang tidak berubah.

Tab: IRT (Item Response Theory) ✓

Menampilkan parameter IRT, kurva ICC, histogram skor siswa, dan tabel perbandingan skor IRT vs klasikal.

Yang Ada di Tab IRT:

1. **Ringkasan** — Distribusi model (berapa soal 1PL/2PL/3PL), distribusi kategori b, dan statistik ringkas.
2. **Histogram Skor IRT Siswa** — Grafik batang distribusi kemampuan siswa pada skala 0–1000. Terlihat rata-rata dan sebaran kemampuan kelas.
3. **Tabel θ vs Skor Klasikal** — Perbandingan skor IRT dengan nilai tradisional per siswa, disertai *korelasi Pearson*. Korelasi tinggi (mendekati 1) berarti kedua metode memberikan hasil serupa.
4. **Kartu Per Butir Soal** — Setiap soal ditampilkan dengan:
 - Badge model (1PL / 2PL / 3PL)
 - Nilai b dengan kategori warna
 - Nilai a (untuk 2PL/3PL)
 - Nilai c dalam % (untuk 3PL)
 - **Kurva ICC** — grafik kecil yang menunjukkan bentuk kurva probabilitas jawaban benar vs kemampuan siswa
 - Untuk bundle: tabel sub-item dengan parameter masing-masing

Jika tab IRT menampilkan pesan "Belum tersedia": Berarti ujian baru saja ditutup dan job IRT belum diproses. Pastikan `php artisan queue:work` berjalan di terminal. Setelah selesai, refresh halaman Analisis.

4 Cara Mengaktifkan dan Menjalankan

Langkah Pertama Kali (Setelah Update)

1 Jalankan migration database

Ini akan menambahkan kolom IRT ke tabel questions, exam_sessions, dan answers.

2 Build frontend

Kompilasi aset React yang baru.

```
# Di terminal project
php artisan migrate
npm run build
```

Saat Development (Lokal)

Buka 4 terminal secara bersamaan:

```
# Terminal 1 – Web server
php artisan serve

# Terminal 2 – Frontend hot reload
npm run dev

# Terminal 3 – WebSocket real-time (opsional, untuk Monitor)
php artisan reverb:start

# Terminal 4 – Queue worker (untuk IRT)
php artisan queue:work
```

Alternatif untuk development: Tambahkan `QUEUE_CONNECTION=sync` di file `.env` untuk menjalankan job IRT langsung saat ujian ditutup, tanpa perlu Terminal 4. Kekurangannya: tombol "Tutup Ujian" akan sedikit loading selama proses berlangsung.

Di Production (VPS)

Queue worker dikelola oleh Supervisor agar otomatis hidup saat server restart:

```
# /etc/supervisor/conf.d/cbtakm.conf
[program:cbtakm-worker]
command=php /var/www/cbtakm/artisan queue:work --sleep=3 --tries=1
autostart=true
```

```

autorestart=true
user=www-data

```

5 Perbedaan Analisis Klasikal vs IRT

Aspek	Analisis Klasikal	Analisis IRT
Ukuran kesulitan	TK = proporsi siswa yang benar (dipengaruhi kemampuan kelompok)	Parameter b (tidak dipengaruhi kelompok)
Ukuran diskriminasi	DP = selisih proporsi benar kelompok atas vs bawah	Parameter a (2PL/3PL) lebih presisi
Faktor tebakan	Tidak diukur	Parameter c (3PL saja)
Skor siswa	Nilai 0–100 (jumlah skor berbobot)	θ pada skala 0–1000 (bisa dibandingkan lintas ujian)
Persyaratan data	Bisa digunakan mulai 1 siswa	Minimal 5 responden per soal (makin banyak, makin akurat)
Kapan dihitung	Langsung saat ada yang submit ujian	Setelah ujian ditutup (background job)
Akumulasi lintas ujian	Tidak — hanya untuk ujian ini	Ya — makin banyak ujian, makin akurat
Penggunaan terbaik	Evaluasi cepat hasil ujian, umpan balik segera	Pengembangan bank soal jangka panjang, standarisasi ujian

Rekomendasi penggunaan: Gunakan *Analisis Klasikal* untuk evaluasi hasil ujian dan umpan balik kepada siswa. Gunakan *Analisis IRT* untuk mengembangkan bank soal yang berkualitas — identifikasi soal yang perlu diperbaiki atau diganti berdasarkan parameter IRT yang sudah terakumulasi dari banyak ujian.