



## Rancang Bangun Aplikasi Computer-based Test (CBT) serta Optimasinya Menggunakan Cache dan Queue Job

Rahmat Siswanto<sup>1</sup> dan Dwi Senjaya A. Morang<sup>2</sup>

<sup>1</sup>Universitas Muhammadiyah Palopo, Program Studi Rekayasa Perangkat Lunak, email: rahmat@umpalopo.ac.id

<sup>2</sup>Universitas Handayani Makassar, Program Studi Sistem Komputer, email: dwi.senjaya15@gmail.com

### [1] Abstrak

*Pelaksanaan ujian di sekolah yang masih menggunakan kertas memerlukan waktu, tenaga dan biaya yang besar dalam penerapannya, terutama dalam aktifitas pencetakan, penggandaan dan distribusi soal ujian kepada siswa yang akan mengikuti ujian. Oleh karena itu, diperlukan sebuah system yang dapat memudahkan pengisian, penggandaan dan distribusi soal dengan biaya yang kecil. Berdasarkan permasalahan tersebut, dibangunlah sebuah aplikasi CBT berbasis web yang dapat diakses dimanapun dan kapanpun serta dapat bekerja pada banyak platform (multi-platform). Sistem ini dibangun dengan metode prototyping untuk memudahkan pengembang dan pelanggan dalam pengembangan aplikasi mulai dari purwarupa awal hingga tahap produksi. Pada penerapannya, aplikasi ini diharapkan mampu mengakomodir kegiatan ujian untuk seluruh siswa yang ada di sekolah disaat bersamaan. Berdasarkan hal tersebut, aplikasi yang dibuat harus memiliki kehandalan dan keakuratan mulai dari proses pelaksanaan ujian hingga menampilkan hasil. Peneliti menerapkan sistem cache dan queue job pada aplikasi untuk mendapatkan hasil maksimal. Dari hasil pengujian menggunakan Apache Benchmark, diperoleh hasil yang menunjukkan bahwa aplikasi CBT berbasis web ini sanggup mengakomodir kegiatan ujian di sekolah dengan jumlah 737 siswa dalam satu sesi ujian.*

**Kata kunci:** CBT, queue job, prototyping, cache

### [2] Abstract

*Examinations in schools that still use paper require a great deal of time, effort and money to implement, especially in the activities of printing, duplicating and distributing exam questions to students who will take the exam. Therefore, we need a system that can facilitate the filling, duplication and distribution of questions at a small cost. Based on these problems, a web-based CBT application was built that can be accessed anywhere and anytime and can work on multiple platforms (multi-platform). This system was built using the prototyping method to make it easier for developers and users to develop applications from the initial prototype to the production stage. In practice, this application is expected to be able to accommodate exam activities for all students in the school at the same time. Based on this, the application that is made must have reliability and accuracy starting from the process of carrying out the exam to displaying the results. Researchers apply a cache and queue job system to the application to get maximum results. From the test results using the Apache Benchmark, the results showed that this web-based CBT application was able to accommodate exam activities in schools with a total of 737 students in one exam session.*

**Keywords:** CBT, queue job, prototyping, cache

---

## 1. Pendahuluan

Pelaksanaan ujian di sekolah umumnya masih menggunakan sistem *paper-based test*. Pelaksanaan ujian yang masih menggunakan kertas memerlukan waktu, tenaga dan biaya yang besar dalam penerapannya, terutama dalam aktifitas pencetakan, penggandaan dan distribusi soal ujian kepada siswa yang akan mengikuti ujian. Tantangan yang dihadapi dalam pengembangan aplikasi CBT berbasis online adalah keterbatasan infrastruktur dan sistem yang optimal untuk dapat mengakomodir ratusan hingga ribuan data secara bersamaan.

Sebelumnya telah ada penelitian pengembangan aplikasi CBT untuk Sekolah Menengah Atas (SMA) oleh Karfindo K dan Mustafa F [1] yang menerapkan ilmu statistika untuk menganalisa kemampuan siswa. Terdapat juga penelitian pembuatan aplikasi CBT di SMP Negeri 6 Cimahi oleh Utami R [2] yang juga merupakan aplikasi CBT berbasis web. Selain itu, terdapat penelitian perancangan aplikasi CBT Psikotes berbasis web oleh Hemawati S dan Chernovita S. Agustina D juga melakukan penelitian perancangan aplikasi CBT berbasis web di SMPN 2 Kuta, Bandung [3]. Penelitian-penelitian tersebut belum terdapat pengujian dalam hal kehandalan sehingga tidak diketahui apakah sistem tersebut mampu mengakomodir sebanyak 737 siswa dalam satu sesi ujian.

Penelitian yang dilakukan oleh penulis dilaksanakan di SMAN 6 Palopo. Berdasarkan hasil observasi yang telah dilakukan, SMAN 6 Palopo memiliki 737 siswa dan 52 guru. Selama ini, sekolah melaksanakan ujian menggunakan sistem *paper-based test* sehingga biaya yang dikeluarkan sangat besar untuk setiap pelaksanaan ujian untuk aktifitas pencetakan soal, penggandaan soal, hingga distribusi soal. Pengelolaan ujian mulai dari pembuatan jadwal hingga proses rekapitulasi hasil ujian juga masih di proses secara manual sehingga memerlukan waktu yang banyak.

Oleh karena itu, penulis merancang dan membangun aplikasi CBT yang dapat menyelesaikan permasalahan di sekolah terkait proses pelaksanaan ujian. Aplikasi yang dibangun merupakan aplikasi berbasis web yang dapat diakses dimanapun dan kapanpun oleh guru dan siswa. Sistem ini menggunakan *framework* Bahasa PHP yaitu Laravel versi 8 dan basis data MySQL. Selanjutnya, penulis menerapkan metode *caching* dan *queue job* untuk mengoptimalkan kinerja server dan aplikasi dalam mengakomodir ratusan hingga ribuan data dalam satu sesi ujian.

Berdasarkan uraian diatas, penulis membuat penelitian dengan judul “Rancang Bangun Aplikasi *Computer-based Test (CBT)* serta Optimasinya Menggunakan *Cache* dan *Queue Job*” yang dapat membantu sekolah dalam menerapkan ujian secara online.

## 2. Tinjauan Pustaka

### 2.1 Metode Penelitian

Pada penelitian ini, penulis menggunakan metode *prototyping* dalam perancangan dan pembuatan aplikasi serta penerapan *cache* dan *queue job* untuk mengoptimalkan performa aplikasi CBT.

#### 2.1.1 Metode Pengembangan Sistem

Metode yang penulis gunakan dalam pengembangan aplikasi CBT adalah metode *prototyping*. Metode ini sangat cocok digunakan bagi pengembang dan pelanggan yang ingin mengembangkan sistem secara cepat dan sesuai kebutuhan pelanggannya. Dengan menggunakan

metode ini pelanggan secara tidak langsung terlibat mulai dari proses awal perancangan dan pengembangan sistem hingga pada versi final untuk tahap produksi.

Model *Prototyping* ini memiliki banyak keunggulan seperti, terjalannya komunikasi yang baik antara pengembang dan pelanggan, mempersingkat waktu pengembangan sistem, pengembang dapat bekerja lebih baik dalam mengidentifikasi kebutuhan pelanggan hingga membuat pelanggan mendapatkan gambaran terkait tahapan pengembangan sistem yang sedang dikembangkan [4].

### 2.1.2 Metode Optimasi Server dan Aplikasi

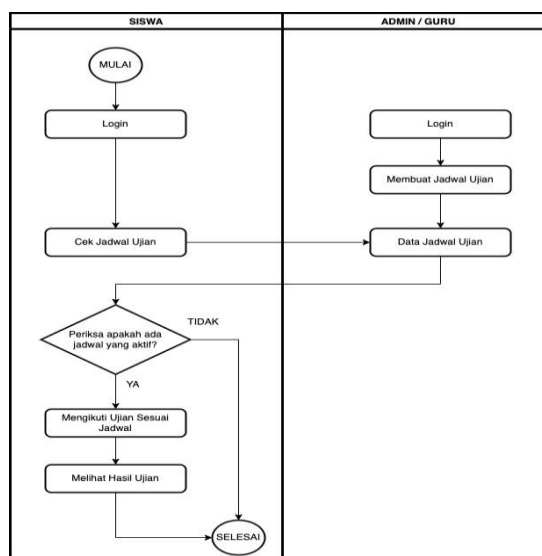
Peneliti menggunakan metode *caching* pada proses menjawab soal ujian yang dilakukan oleh siswa. Metode *caching* ini memungkinkan aplikasi untuk menyimpan data sementara selama proses ujian berlangsung. Data disimpan di *browser* melalui *local storage* masing-masing pengguna selama proses ujian berlangsung dan akan dihapus setelah ujian selesai. Data yang disimpan adalah daftar soal ujian, soal yang telah dikerjakan serta daftar jawaban siswa. Hal ini membuat beban kerja server menjadi berkurang secara drastis karena tidak ada *request* data ke server selama proses ujian berlangsung. Berbeda dengan *session storage* yang akan hilang setelah jendela ditutup, *local storage* tidak memiliki waktu *expired* sehingga data tetap aman (tetap ada) sampai data tersebut dihapus [5].

Selanjutnya, metode *queue job* diterapkan pada proses *request* ke server untuk menyimpan data hasil ujian siswa. Dengan menggunakan *queue job*, proses penyimpanan hasil ujian ke basis data akan disimpan berdasarkan urutan sesuai antrian sehingga proses penyimpanan data ke basis data tidak membebani kinerja server.

## 3. Metode Penelitian

### 3.1 Pengumpulan Data

Penelitian ini menggunakan data primer yang dikumpulkan langsung dari lokasi penelitian dengan cara wawancara ke pelanggan hingga didapatkan proses bisnis yang jelas seperti yang terlihat pada Gambar 1.



Gambar 1 Bisnis Proses Aplikasi CBT Sekolah

Secara umum, proses bisnis ujian CBT berbasis web yaitu guru membuat soal ujian untuk kelas yang diajarkannya, kemudian admin mengatur jadwal sesuai dengan jadwal yang telah disepakati oleh sekolah, selanjutnya siswa mengikuti ujian sesuai dengan jadwal yang telah ditentukan.

### 3.2 Perancangan dan Pembuatan Prototipe

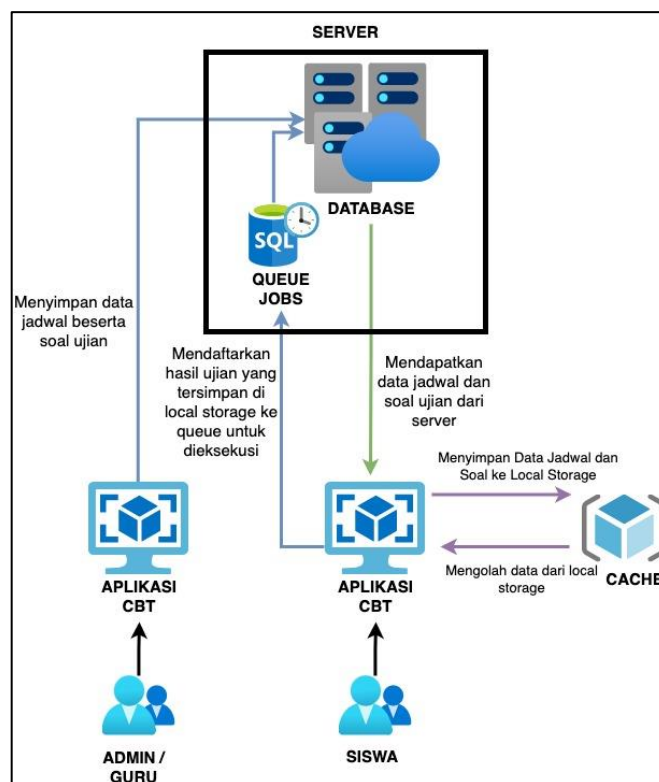
Pada tahap ini, dilakukan perangan arsitektur sistem dan *use case diagram*. Arsitektur sistem dilejaskan pada poin 3.2.1 dan *use case diagram* pada poin 3.2.2.

#### 3.2.1 Arsitektur Rancangan Sistem

Sistem ini dibangun menggunakan Bahasa PHP dengan framework Laravel 8. Sistem ini merupakan aplikasi berbasis web dengan MySQL sebagai basis datanya. Aplikasi ini menerapkan metode *caching* dengan memanfaatkan *local storage browser* serta *queue job*. Secara lengkap, arsitektur sistem dapat dilihat pada Gambar 2 sedangkan spesifikasi server dapat dilihat pada Tabel 1.

Tabel 1 Spesifikasi Server

No	Komponen	Spesifikasi
1	CPU	4 Core CPU
2	Penyimpanan	SSD 80GB
3	RAM	4 GB
4	Sistem Operasi	Ubuntu 20.04 LTS (64Bit)



Gambar 2 Arsitektur Sistem CBT Sekolah

Berdasarkan informasi yang tertera pada Gambar 2, metode *caching* bekerja pada sisi *client* sedangkan metode *queue job* akan bekerja di sisi server. Kedua metode inilah yang akan

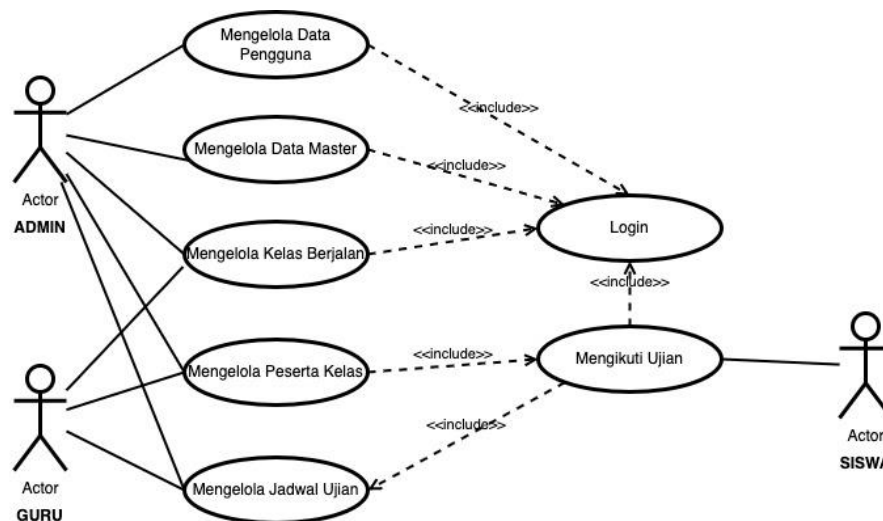
berjalan secara paralel sehingga sistem dapat mengakomodir kegiatan ujian siswa secara keseluruhan.

### 3.2.2 Usecase Diagram

Berdasarkan proses bisnis yang telah berjalan, pengembangan aplikasi ini memiliki tiga pengguna utama yaitu admin, guru dan siswa dengan deskripsi pengguna seperti yang ditampilkan pada Tabel 2. Adapun *use case diagram* dapat dilihat pada Gambar 3.

**Tabel 2 Pengguna Aplikasi**

No	Role Pengguna	Deskripsi
1	Admin	Manajemen Data Master, Manajemen Jadwal Ujian, Manajemen Hasil Ujian, Manajemen Kelas Berjalan, Manajemen Peserta Kelas, Manajemen Pengguna
2	Guru	Manajemen Jadwal Ujian, Manajemen Hasil Ujian, Manajemen Peserta Kelas
3	Siswa	Mengikuti ujian sesuai jadwal, Melihat hasil ujian



**Gambar 3 Use Case Diagram Aplikasi CBT Sekolah**

## 3.3 Uji Coba Prototipe

Uji coba prototipe dilakukan melalui tiga tahap. Tahap uji coba yang dilakukan adalah sebagai berikut:

### 3.3.1 Prototipe 1

Proses prototipe berawal dari wawancara yang dilakukan kepada pihak sekolah untuk memperoleh kebutuhan sistem yang akan dibangun. Dari hasil wawancara yang telah dilakukan diperoleh informasi terkait bisnis proses aplikasi secara umum seperti yang telah digambarkan pada Gambar 1. Selanjutnya, proses perancangan dan pembuatan sistem dilakukan sesuai dengan bisnis proses. Hasil perancangan dan pembuatan prototipe 1 diuraikan pada Tabel 3.

Tabel 3 Fitur Aplikasi Prototipe 1

No	Fitur	Admin	Guru	Siswa
1	Login	V	V	V
2	Menu Manajemen Data Master Siswa	V		
3	Menambahkan siswa	V		
4	Mengubah data siswa	V		
5	Menghapus data siswa	V		
6	Menu Manajemen Data Master Guru	V		
7	Menambahkan guru	V		
8	Mengubah data guru	V		
9	Menghapus data guru	V		
10	Menu Manajemen Data Mata Pelajaran	V		
11	Menambahkan Mata Pelajaran	V		
12	Mengubah data mata pelajaran	V		
13	Menghapus data mata pelajaran	V		
14	Menu Manajemen Kelas	V		
15	Menambahkan kelas	V		
16	Mengubah data kelas	V		
17	Menghapus data kelas	V		
18	Menu Data Pengguna	V		
19	Mengubah data pengguna	V		
20	Menghapus data pengguna	V		
21	Set Kelas Berjalan (Kelas Semester Aktif)	V		
22	Set Peserta Kelas Mata Pelajaran Kelas Semester Aktif	V		
23	Mengimport peserta kelas menggunakan excel	V		
24	Menambahkan peserta kelas menggunakan form	V		
25	Membuat Jadwal Ujian	V	V	
26	Mengedit jadwal ujian	V	V	
27	Menghapus jadwal ujian	V	V	
28	Menambahkan soal pada jadwal ujian	V	V	
29	Mengedit soal pada jadwal ujian	V	V	
30	Menghapus soal pada jadwal ujian	V	V	
31	Melihat semua hasil ujian peserta	V	V	
32	Melihat jadwal ujian yang terbuka			V
33	Mengikuti ujian sesuai jadwal yang terbuka			V
34	Melihat hasil ujian yang diikuti			V
35	Logout	V	V	V
36	Update Profil	V	V	V

### 3.3.2 Evaluasi Prototipe 1

Pada tahap ini, aplikasi diujicoba oleh beberapa perwakilan guru mata pelajaran untuk menambahkan jadwal dan soal ujian mata pelajaran masing-masing. Adapun fitur yang diujicoba yaitu halaman login, halaman kelas berjalan, halaman jadwal ujian, detail jadwal ujian, pengimputan soal, mengedit soal dan menghapus soal.

Berdasarkan hasil evaluasi prototipe 1, terdapat beberapa penyesuaian dan penambahan fitur pada aplikasi. Fitur yang ditambahkan dan disesuaikan dapat dilihat pada Tabel 4.

Tabel 4 Hasil Evaluasi Prototipe 1

No	Fitur	Admin	Guru	Siswa
1	Penambahan fitur pengimputan equation (rumus matematik)	V	V	
2	Penambahan fitur print soal ujian	V	V	
3	Penambahan fitur export berita acara ujian	V	V	
4	Penambahan fitur duplikasi jadwal / soal ujian	V	V	
5	Penambahan fitur acak soal	V	V	
6	Penambahan fitur acak opsi jawaban	V	V	
7	Perubahan poin soal yang awalnya bertipe integer menjadi double	V	V	

### 3.3.3 Prototipe 2

Pengembangan prototipe 2 merupakan implementasi dari hasil evaluasi prototipe 1 yang telah dilaksanakan sebelumnya sesuai yang diuraikan pada Tabel 3. Penulis menambahkan sebuah *library* bernama MathJax yang diintegrasikan dengan *text editor* TinyMCE sehingga pengguna dapat menambahkan *equation* pada soal maupun opsi jawaban. Untuk menambahkan *equation*, pengguna harus memasukkan kode *latex* pada *text editor*.

Fitur *export* berita acara ujian menggunakan *library PHPWord* serta pada fitur cetak soal ujian, penulis menambahkan halaman HTML yang menampilkan semua soal yang telah dimasukkan kedalam soal ujian sehingga dapat diprint secara langsung. Fitur duplikasi, acak soal dan acak opsi jawaban ditambahkan kedalam sistem yang dapat diakses oleh admin dan guru sedangkan tipe data pada table poin soal diubah dari *integer* menjadi *double* sehingga memungkinkan untuk pengimputan poin pecahan.

### 3.3.4 Evaluasi Prototipe 2

Evaluasi prototipe 2 berfokus pada pengujian ulang sistem pengimputan jadwal dan soal ujian beserta fitur-fitur tambahan dan perubahan pada prototipe 2. Setelah pengujian ulang sistem pengimputan soal ujian dan fitur-fitur baru selesai, tim evaluator dari sekolah mengevaluasi menu lainnya yaitu menu kelas berjalan. Setelah proses evaluasi dilakukan, diperoleh hasil evaluasi sebagai berikut:

Tabel 5 Hasil Evaluasi Prototipe 2

No	Fitur	Admin	Guru	Siswa
1	Penambahan fitur pengimputan kelas berjalan oleh guru	V	V	
2	Penambahan fitur import kelas bejalan oleh guru	V	V	
3	Penambahan fitur penambahan peserta kelas oleh guru	V	V	
4	Penambahan fitur import peserta kelas oleh guru	V	V	
5	Penambahan fitur token untuk setiap ujian	V	V	

Berdasarkan hasil evaluasi prototipe 2, pihak sekolah sepakat untuk menambahkan fitur penambahan kelas berjalan dan peserta kelasnya pada akun guru sehingga tidak membebani kerja admin. Fitur token ditambahkan sebagai bagian dari sistem keamanan proses ujian sehingga siswa tidak dapat langsung melihat dan mengikuti ujian tanpa *token* yang diberikan oleh admin atau guru yang membuat jadwal ujian tersebut.

### 3.3.5 Prototipe 3

Setelah proses evaluasi prototipe 2 dilakukan, penulis mengembangkan prototipe 3 yang merupakan bentuk implementasi dari hasil evaluasi prototipe 2. Pada tahap ini penulis menambahkan menu input kelas berjalan dan import kelas berjalan serta input peserta kelas serta *import* peserta kelas menggunakan excel.

### 3.3.6 Evaluasi Prototipe 3

Setelah prototipe 3 selesai dikembangkan, selanjutnya dilakukan ujicoba proses ujian oleh siswa. Pengujian proses ujian diikuti oleh 203 siswa yang login serta mulai mengerjakan soal *dummy* mulai dari pukul 09.30 – 10.30. Selama sesi uji coba ujian tersebut berlangsung, semua peserta dapat mengerjakan soal dengan lancar tanpa kendala yang ditandai dengan *load average* yang kecil sebesar 0.38.

Setelah proses uji coba ujian selesai, pihak sekolah melakukan evaluasi terkait tampilan aplikasi pada siswa. Hasil evaluasi dapat dilihat pada Tabel 6.

**Tabel 6 Hasil Evaluasi Prototipe 3**

No	Fitur	Admin	Guru	Siswa
1	Penambahan informasi siswa pada halaman dashboard siswa			V

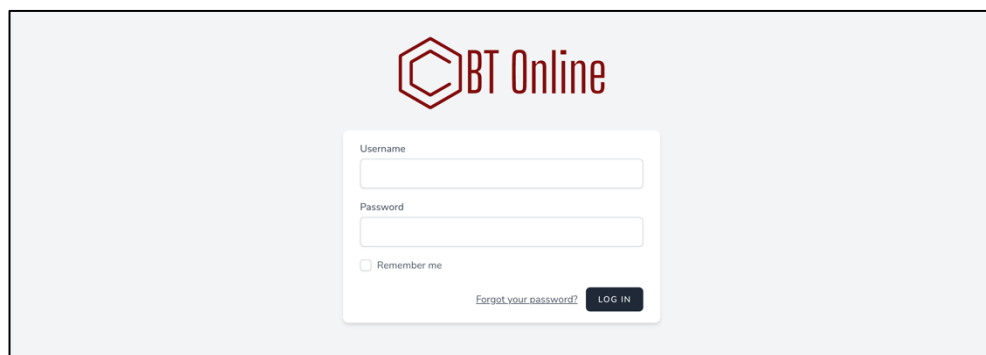
## 4. Hasil

### 4.1 Hasil Perancangan Sistem

Hasil perancangan aplikasi CBT berbasis web terdiri dari beberapa halaman aplikasi dari beberapa hak akses pengguna.

#### 4.1.1 Halaman Login

Halaman login merupakan halaman untuk memasukkan *credential* pengguna berupa *username* dan *password* agar dapat login kedalam aplikasi. Tampilan halaman login dapat dilihat pada Gambar 4.

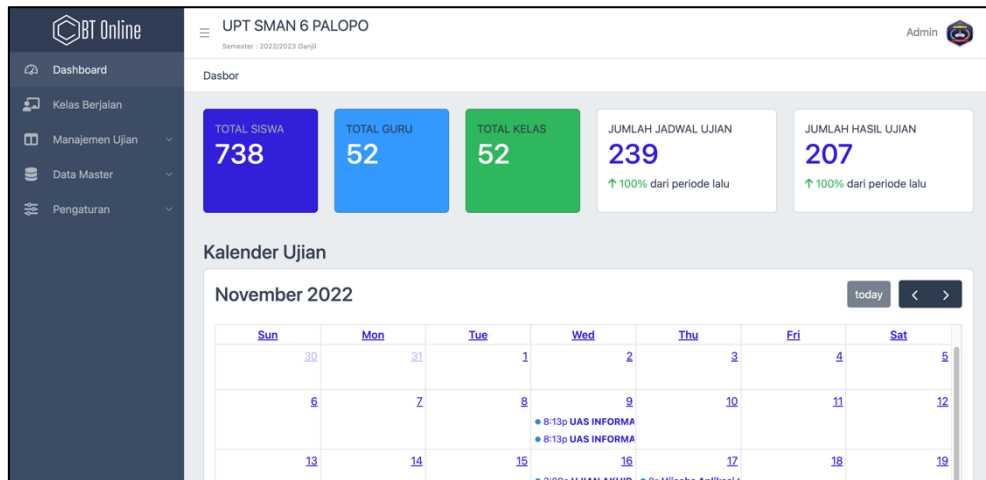


**Gambar 4 Halaman Login**

#### 4.1.2 Halaman Dashboard Admin dan Guru

Halaman dashboard seperti yang terlihat pada Gambar 5 akan ditampilkan setelah pengguna dengan hak akses guru atau admin login dengan akun masing-masing. Pada halaman dashboard akan ditampilkan jumlah siswa, jumlah guru, jumlah kelas, jumlah jadwal ujian, jumlah hasil ujian dan kalender ujian yang ditampilkan dalam bentuk kalender.

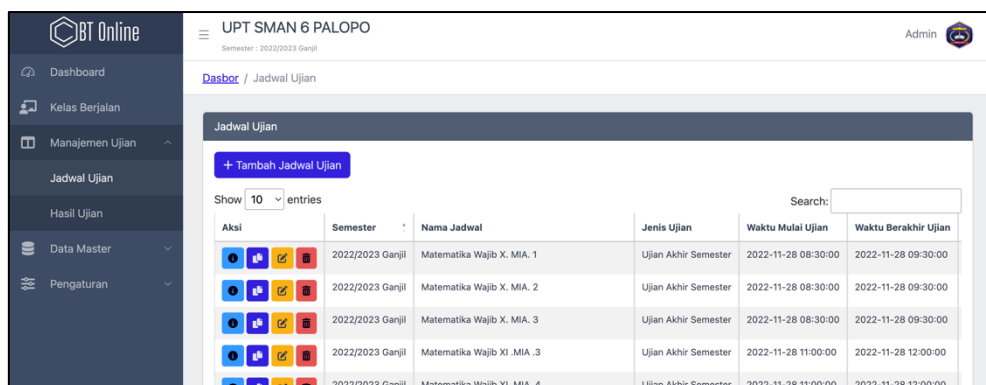




Gambar 5 Halaman Dashboard Admin

#### 4.1.3 Halaman Jadwal Ujian

Admin dan Guru dapat menambahkan jadwal ujian dan soal ujian kedalam aplikasi melalui menu Jadwal Ujian. Gambar 6 menunjukkan daftar jadwal ujian yang telah dibuat oleh admin dan guru, Gambar 7 menunjukkan menampilkan detail jadwal, sedangkan Gambar 8 menunjukkan form pengimputan soal ujian.



Gambar 6 Halaman Jadwal Ujian

Pada halaman jadwal ujian, pengguna dapat mencari jadwal berdasarkan nama jadwal, nama guru pengampu mata pelajaran, nama kelas maupun nama mata pelajaran.

Gambar 7 Detail Jadwal Ujian

Halaman detail jadwal diakses dengan menekan tombol detail pada salah satu jadwal ujian di halaman jadwal ujian. Pada halaman detail ujian akan ditampilkan informasi terkait nama nama jadwal ujian, mata pelajaran, guru mata pelajaran, waktu ujian, serta pengaturan ujian lainnya seperti yang ditampilkan pada Gambar 7. Untuk menambahkan soal, pengguna dapat menekan tombol tambah soal pilihan ganda, selanjutnya akan ditampilkan form pengimputan soal seperti yang terlihat pada Gambar 8.

Gambar 8 Form Tambah Soal

#### 4.1.4 Halaman Rekapitulasi Hasil Ujian

Halaman hasil ujian merupakan halaman untuk melihat rekapitulasi hasil dari ujian yang telah dilakukan oleh peserta ujian (siswa). Pada halaman hasil ujian, pengguna dapat melihat seluruh hasil ujian yang telah dilakukan oleh siswa dan mengesport hasilnya kedalam format excel. Halaman hasil ujian dapat dilihat pada Gambar 9.

**Ujicoba Aplikasi CBT**

Mata Pelajaran : Tes  
 Nama Guru : Fulan, S.Pd.  
 Nilai KKM : 0

Kelas : Tes  
 Semester Akademik : 2022/2023 Ganjil

Export Excel

Show 10 entries

Status	NISN/NIS	Nama Siswa	Nilai Akhir (Poin)	Nilai Akhir (Persen)	Jumlah Soal Terjawab	Jumlah Soal Benar
❌	02020157	MUH. YUSUF	40	22.86	35	8
❌	02020045	Haidir	35	20	35	7
❌	02020131	RAHMAT HIDAYAT	35	20	35	7
❌	02020268	Ririn Wulandari	55	31.43	32	11
❌	02020213	Alvin Gasong	30	17.14	35	6
❌	02020173	Siri Rahma	60	28.57	35	10
❌	02020135	Syahria	45	25.71	35	9
❌	02020235	Muh. Khiky Alfarizah	60	34.29	35	12
❌	222222	Siswa2	40	22.86	34	8
❌	02020075	AMABILIS VIRGINE	40	22.86	33	8

Showing 1 to 10 of 207 entries

Previous 1 2 3 4 5 ... 21 Next

Powered by PT. Tanorasula Tekno Kreatif © 2022

Gambar 9 Halaman Rekapitulasi Hasil Ujian

#### 4.1.5 Halaman Dashboard Siswa

Pada halaman dashboard siswa, pengguna dapat melihat daftar ujian yang telah terbuka sesuai jadwal yang telah dibuat oleh admin atau guru seperti yang terlihat pada Gambar 10.

CBT Beranda Riwayat Kelas Riwayat Hasil Ujian Siswa1

Selamat datang di CBT

Jadwal ujian akan ditampilkan ditabel [Jadwal Ujian](#) dibawah ini. Silahkan lihat detail ujian dengan cara klik tombol [Detail Ujian](#). Riwayat hasil ujian dapat dilihat pada menu [Hasil Ujian](#).

Avatar NISN/NIS: 111111  
 Nama Siswa: Siswa1

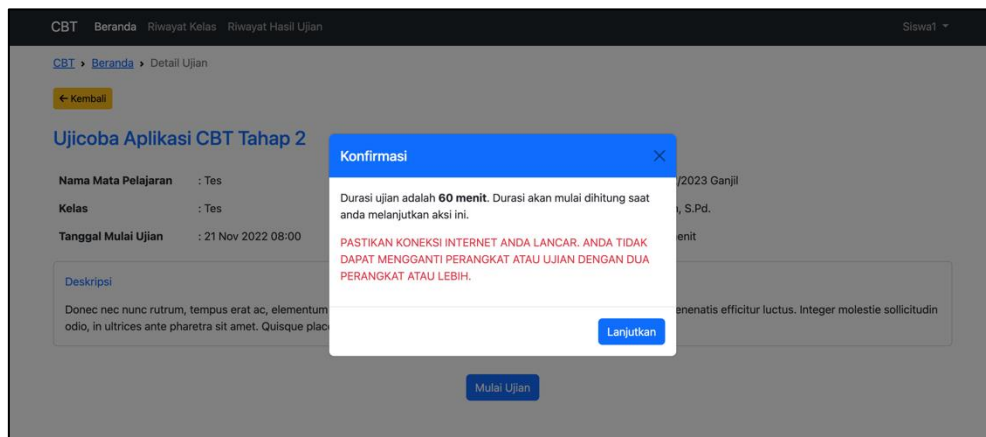
Jadwal Ujian

No.	Jadwal Pelaksanaan Ujian	Kelas	Mata Pelajaran	Guru	Jenis Ujian	Durasi Ujian	
1	Ujicoba Aplikasi CBT Tahap 2 21 Nov 2022 08:00 s/d 22 Nov 2022 12:00	Tes	Tes	Fulan, S.Pd.	Ujian Akhir Semester	60 menit	<a href="#">Detail Ujian</a>

Gambar 10 Halaman Dashboard Siswa

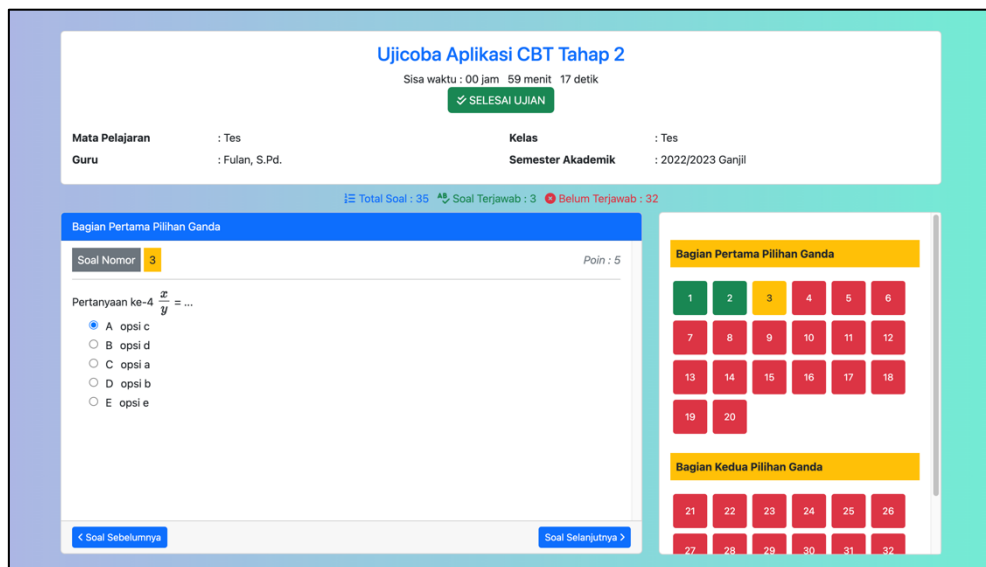
#### 4.1.6 Halaman Ujian Siswa

Untuk memulai ujian, siswa terlebih dahulu menekan tombol Detail Ujian pada daftar jadwal ujian di halaman dashboard, selanjutnya siswa memasukkan token ujian untuk dapat melihat detail jadwal ujian yang akan diikuti. Sebelum mulai mengerjakan soal, siswa akan dikonfirmasi apakah akan melanjutkan ke halaman pertanyaan seperti yang terlihat pada Gambar 11.



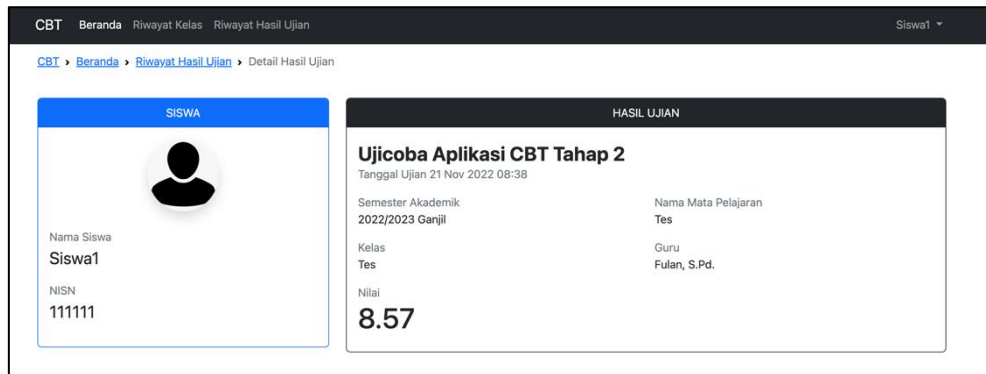
**Gambar 11 Halaman Detail Ujian dan Konfirmasi Mulai Ujian**

Setelah siswa menekan tombol lanjutkan, siswa akan diarahkan ke halaman pertanyaan untuk menjawab semua soal yang ada. Pada halaman proses pelaksanaan ujian (halaman pertanyaan) akan ditampilkan *timer* waktu mengerjakan semua soal, tombol Selesai Ujian, informasi terkait nama mata pelajaran, nama kelas, total soal, total soal yang sudah dijawab, total soal yang belum dijawab, pertanyaan dan opsi jawaban serta tombol-tombol navigasi soal seperti yang terlihat pada Gambar 12.



**Gambar 12 Halaman Proses Ujian Siswa**

Setelah proses ujian selesai, maka siswa akan diarahkan ke halaman hasil ujian. Pada halaman hasil ujian ini, siswa dapat melihat nilai hasil ujiannya apabila admin atau guru mengatur jadwal ujian untuk menampilkan nilai siswa, jika tidak, maka nilai tidak akan ditampilkan.



Gambar 13 Halaman Hasil Ujian Siswa

## 4.2 Hasil Pengujian dan Analisis

### 4.2.1 Hasil Pengujian dan Analisis *Black Box Testing*

Pengujian *black box* diuji kepada admin, guru dan siswa yang akan menggunakan aplikasi CBT online ini. Adapun fitur-fitur utama yang diuji adalah halaman login, halaman kelas berjalan, form tambah kelas berjalan, hapus kelas berjalan, halaman jadwal ujian, form tambah jadwal ujian, form ubah jadwal ujian, halaman detail jadwal ujian, form tambah soal, form edit soal, halaman hasil ujian, halaman proses ujian siswa, halaman hasil ujian siswa. Dari hasil pengujian yang dilakukan, semua fitur berjalan seperti yang seharusnya tanpa kendala sehingga dapat disimpulkan bahwa aplikasi CBT telah berjalan dengan baik dari segi fungsionalitas.

### 4.2.2 Hasil Uji Coba Apache Benchmarking Tool

Penulis menggunakan Apache Benchmarking Tool untuk menguji kehandalan sistem dalam menangani request yang masuk ke server. Sebelum memulai uji coba, penulis terlebih dahulu mengestimasi jumlah *concurrent user* menggunakan persamaan sebagai berikut [6]:

$$CCU = AU / 10 \quad (1)$$

Berdasarkan persamaan diatas, CCU adalah *concurrent users* dan AU adalah *active users* sehingga, diperoleh CCU sebesar 73,7 siswa yang dibulatkan menjadi 74 siswa dari total pengguna aktif sebesar 737 siswa. Hasil persamaan tersebut dimasukkan kedalam perintah sebagai berikut:

```
ab -n 737 -c 74 https://[Address]
```

Parameter n merupakan total *request* yang mengarah ke server sedangkan parameter c merupakan total *concurrent user* yang mengakses bersamaan dalam satu waktu (ms) [7]. Berdasarkan hasil pengujian yang dilakukan, diperoleh hasil seperti yang ditunjukkan pada Gambar 14.

```

Concurrency Level:      74
Time taken for tests:   11.168 seconds
Complete requests:     737
Failed requests:       0
Total transferred:     837969 bytes
HTML transferred:     0 bytes
Requests per second:   65.99 [#/sec] (mean)
Time per request:      1121.337 [ms] (mean)
Time per request:      15.153 [ms] (mean, across all concurrent requests)
Transfer rate:         73.28 [Kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:    2   502 224.5   513   1553
Processing: 67   592 724.0   350   3097
Waiting:    30   480 752.2   215   3054
Total:      70  1093 736.0   883   3680

Percentage of the requests served within a certain time (ms)
 50%    883
 66%    974
 75%   1060
 80%   1101
 90%   1855
 95%   3150
 98%   3444
 99%   3608
100%   3680 (longest request)

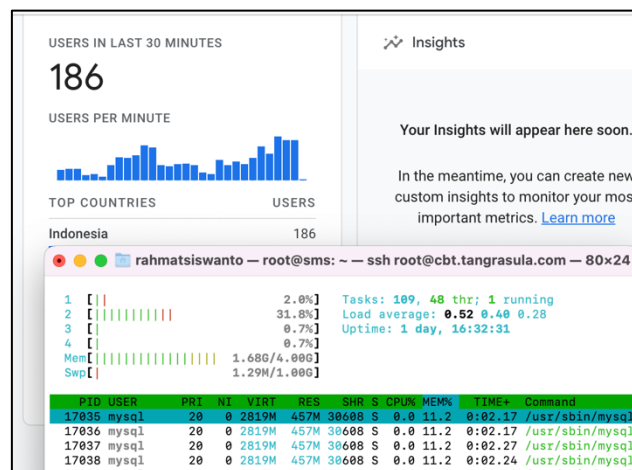
```

**Gambar 14 Hasil Pengujian Apache Benchmarking Tool**

Gambar 14 memperlihatkan jumlah *request* per detik yang mampu dieksekusi sebanyak 65,99 *request* per detik, rata-rata waktu per *request* sebesar 1.1121 ms, rata-rata waktu koneksi sebesar 1.093 ms dan waktu maksimum koneksi sebesar 3.680 ms yang terjadi apabila total *request* yang sebanyak 737 dieksekusi dalam waktu bersamaan (ms).

#### 4.2.3 Hasil Uji Coba Ujian (Simulasi Ujian) Tahap 1

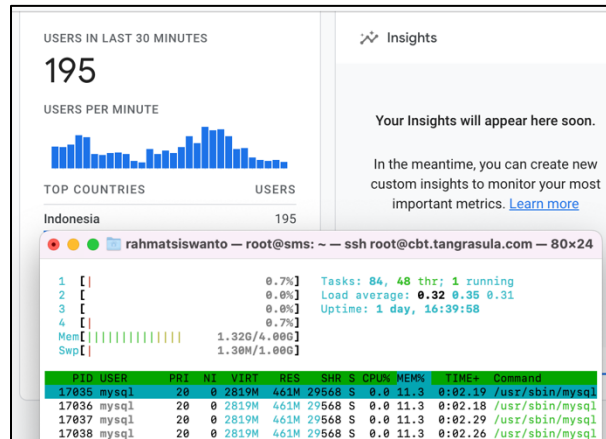
Setelah ujicoba menggunakan Apache Benchmarking Tool dilakukan, pihak sekolah menginstruksikan kepada siswa kelas XII yang berjumlah 235 siswa untuk mengikuti proses simulasi dan uji coba aplikasi CBT mulai pukul 09.30 – 10.30. Berdasarkan data yang diperoleh, siswa yang mengikuti ujian pada sesi uji coba sebanyak 205 siswa. Dari hasil pemantauan aktivitas server, diperoleh peak penggunaan processor adalah 33.94% dari total 4 core CPU yang digunakan, sedangkan peak penggunaan RAM adalah 1.68GB dari 4GB RAM dengan load average 0.52 seperti yang terlihat pada Gambar 15.



**Gambar 15 Peak Penggunaan Processor dan RAM Pada Uji Coba atau Simulasi Ujian**

Pada Gambar 15 terlihat jumlah pengguna yang aktif pada 30 menit terakhir berjumlah 186 siswa, angka ini merupakan angka yang ditampilkan dari *google analytics*. Jumlah pengguna yang tampil bisa saja lebih apabila diambil dari 60 menit terakhir. Angka penggunaan processor

dan RAM turun pada saat request ke server sedikit atau sesi ujian telah berakhir seperti yang terlihat pada gambar 15.



Gambar 16 Pengguna Aktif dalam 30 Menit Terakhir Setelah Simulasi Ujian

Berdasarkan Gambar 16, load average turun menjadi 0.32, penggunaan processor turun menjadi 1,4% dan penggunaan RAM turun menjadi 1.32GB. Dari hasil pengujian ini dapat diperoleh kesimpulan bahwa aplikasi CBT siswa telah berjalan dengan sangat baik untuk mengakomodir ujian lebih dari 200 siswa.

#### 4.2.4 Hasil Uji Coba Ujian (Simulasi Ujian) Tahap 2

Ujicoba tahap 2 dilaksanakan 1 pekan setelah ujicoba tahap 1. Peserta yang mengikuti uji coba tahap 2 ini adalah siswa kelas XI yang berjumlah 135 siswa. Untuk mendapatkan hasil pengujian yang lebih akurat, penulis menggunakan aplikasi bernama Monitorix untuk melihat secara rinci riwayat aktivitas yang terjadi di server pada saat simulasi berlangsung. Hasil ujicoba tahap 2 yang dilakukan dapat dilihat dari Gambar 17.



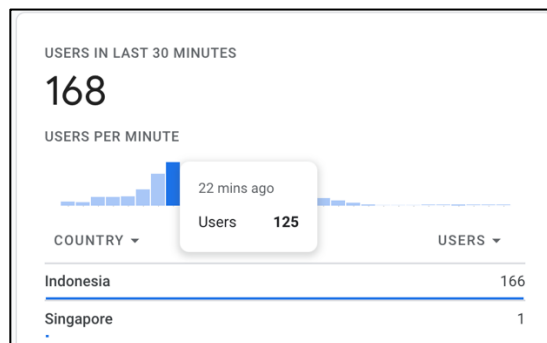
Gambar 17 Hasil Monitorix Pengujian Tahap 2

Ujicoba tahap 2 dilakukan dengan menginstruksikan siswa untuk login secara bersamaan dan selesai ujian secara bersamaan untuk menguji kemampuan sistem dalam menerima ratusan request secara bersamaan. Berdasarkan Gambar 17, terdapat *peak load average* yang mencapai

0.9 namun hal ini tidak berlangsung lama karena pada penelitian ini penulis menggunakan metode *queue job* sehingga sistem tetap berjalan lancar tidak membebani server. *Peak* ini terjadi pada saat siswa diinstruksikan untuk selesai ujian secara bersamaan, hal ini menyebabkan adanya *request* dalam jumlah yang banyak ke server.

#### 4.2.5 Hasil Uji Coba Ujian (Simulasi Ujian) Tahap 3

Ujicoba tahap 3 diikuti oleh siswa kelas X. Ujicoba tahap 3 ini diikuti oleh 158 siswa. Berdasarkan hasil monitoring dari *google analytics*, didapatkan informasi dalam satu menit, jumlah siswa aktif secara bersamaan tertinggi berada diangka 125 siswa per menit sesuai dengan yang ditampilkan pada Gambar 18. Pemantauan pada aplikasi Monitorix juga menunjukkan kinerja server yang stabil seperti yang ditunjukkan pada Gambar 19.



Gambar 18 Hasil *Google Analytics* Pengujian Tahap 3



Gambar 19 Hasil Monitorix Pengujian Tahap 3

## 5. Kesimpulan dan Saran

### 5.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian ini adalah metode *caching* dan *queue job*, implementasi aplikasi CBT sekolah berjalan dengan lancar tanpa kendala hingga dapat mengakomodir hingga 737 request berdasarkan hasil pengujian Apache Benchmarking Tool yang telah dilakukan. Berdasarkan ujicoba yang dilakukan langsung oleh *end user* atau siswa, sistem mampu mengakomodir sebanyak 235 siswa yang diambil dari jumlah siswa yang mengikuti ujian pada uji coba tahap pertama.



Karena keterbatasan waktu dan kondisi, kami tidak dapat menguji untuk seluruh siswa secara bersamaan. Namun, berdasarkan hasil pengujian dari Apache Benchmarking Tool dan uji coba pada end user dapat disimpulkan bahwa sistem ini mampu mengakomodir ujian sekolah untuk seluruh siswa yang berjumlah 737 siswa secara bersamaan. Penerapan aplikasi ini juga dapat mengurangi beban biaya yang dikeluarkan oleh pihak sekolah sehingga dana tersebut dapat dialokasikan untuk kegiatan lainnya.

## 5.2 Saran

Saran yang diberikan oleh pengguna kepada peneliti/penulis pada penelitian ini adalah ditambahkan fitur untuk tidak mengizinkan siswa untuk membuka halaman lain selain halaman ujian serta aplikasi lain pada *smartphone* siswa selama proses ujian berlangsung.

## Daftar Pustaka

- [1] K. Karfindo and F. Mustafa, "Pengembangan aplikasi Computer Based Test (CBT) untuk Sekolah Menengah Atas (SMA)," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 3, no. 1, pp. 42–48, Jan. 2017, doi: 10.26594/REGISTER.V3I1.715.
- [2] R. D. Utami, "Aplikasi Computer-Based Test Ujian pada SMP Negeri 6 Cimahi," *Jurnal Teknologi dan Informasi*, vol. 9, no. 2, pp. 151–160, Oct. 2019, doi: 10.34010/JATI.V9I2.2234.
- [3] D. A. S. Agustina, "PERANCANGAN APLIKASI COMPUTER BASED TEST (CBT) BERBASIS WEB (STUDI KASUS DI SMP NEGERI 2 KUTA - BADUNG)," *Jurnal Teknologi Informasi dan Komputer*, vol. 2, no. 1, Oct. 2017, Accessed: Nov. 22, 2022. [Online]. Available: <https://jurnal.undhirabali.ac.id/index.php/jutik/article/view/223>
- [4] A. Susanto and ; Meiryani, "System Development Method with The Prototype Method," *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, vol. 8, p. 7, 2019, Accessed: Nov. 21, 2022. [Online]. Available: [www.ijstr.org](http://www.ijstr.org)
- [5] J. Triyono, "Implementasi Localstorage pada Pemrograman Client Berbasis JSON," 2019, Accessed: Nov. 22, 2022. [Online]. Available: <http://publikasiilmiah.ums.ac.id/handle/11617/11742>
- [6] "Estimating Concurrent Users - IBM Documentation." [https://www.ibm.com/docs/en/cognos-analytics/10.2.2?topic=SSEP7J\\_10.2.2/com.ibm.swg.ba.cognos.crn\\_arch.10.2.2.doc/c\\_arch\\_estimatingconcurrentusers.htm](https://www.ibm.com/docs/en/cognos-analytics/10.2.2?topic=SSEP7J_10.2.2/com.ibm.swg.ba.cognos.crn_arch.10.2.2.doc/c_arch_estimatingconcurrentusers.htm) (accessed Dec. 24, 2022).
- [7] I. K. S. Satwika and K. N. Semadi, "PERBANDINGAN PERFORMANSI WEB SERVER APACHE DAN NGINX DENGAN MENGGUNAKAN IPV6," *Scan : Jurnal Teknologi Informasi dan Komunikasi*, vol. 15, no. 1, pp. 10–15, Feb. 2020, doi: 10.33005/SCAN.V15I1.1847.