# PERANCANGAN SISTEM BANK SOAL TERDISTRIBUSI MULTI SEKOLAH (STUDI KASUS SMK NEGERI ROWOKANGKUNG)

**SKRIPSI**

**Oleh:**

**BRAMAN TYAMAHENDRAWAN NIM. 2141720097**

****

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2025**

# HALAMAN PENGESAHAN

**PERANCANGAN SISTEM BANK SOAL TERDISTRIBUSI MULTI SEKOLAH (STUDI KASUS SMK NEGERI ROWOKANGKUNG)**

Disusun oleh:

**BRAMAN TYAMAHENDRAWAN NIM. 2141720097**

Laporan Akhir ini telah diuji pada tanggal 2 Juli 2025

Disetujui oleh:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Pembimbing Utama | : | Gunawan Budi Prasetyo, ST., MMT., Ph.D.  NIP. 197704242008121001 | ........................... |
|  |  |  |  |  |
| 2. | Pembimbing Pendamping | : | Indra Dharma Wijaya, ST., M.MT.  NIP. 197305102008011010 | ........................... |
|  |  |  |  |  |
| 3. | Penguji Utama | : | .......................................................  ....................................................... | ........................... |
|  |  |  |  |  |
| 4. | Penguji Pendamping | : | .......................................................  ....................................................... | ........................... |

Mengetahui,

|  |  |
| --- | --- |
| Ketua Jurusan  Teknologi Informasi | Ketua Program Studi  Teknik Informatika |
| Prof. Dr. Eng. Rosa Andrie Asmara, ST., MT. | Dr. Ely Setyo Astuti, S.T., M.T. |
| NIP. 198010102005011001 | |  | | --- | | NIP. 197605152009122001 | |

# PERNYATAAN

Dengan ini saya menyatakan bahwa pada Skripsi ini tidak terdapat karya, baik seluruh maupun sebagian, yang sudah pernah diajukan untuk memperoleh gelar akademik di Perguruan Tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini serta disebutkan dalam daftar sitasi/pustaka.

|  |  |
| --- | --- |
|  | Malang, 2 Juli 2025  Braman Tyamahendrawan. |

# KATA PENGANTAR

Puji Syukur kami panjatkan kehadirat Allah SWT/Tuhan YME atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “PERANCANGAN SISTEM BANK SOAL TERDISTRIBUSI MULTI SEKOLAH (STUDI KASUS SMK NEGERI ROWOKANGKUNG)”. Skripsi ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Kami menyadari bahwasannya dengan tanpa adanya dukungan dan kerja sama dari berbagai pihak, kegiatan laporan akhir ini tidak akan dapat berjalan baik. Untuk itu, kami ingin menyampaikan rasa terima kasih kepada:

1. Bapak Prof. Dr. Eng. Rosa Andrie Asmara, ST., MT., selaku Ketua Jurusan Teknologi Informasi
2. Ibu Dr. Ely Setyo Astuti, S.T., M.T., selaku Ketua Program Studi Teknik Informatika
3. Bapak Gunawan Budi Prasetyo, ST., MMT., Ph.D., selaku Dosen Pembimbing I atas seluruh bantuan dan dukungannya selama penulis menyelesaikan masa studi.
4. Bapak Indra Dharma Wijaya, ST., M.MT., selaku Dosen Pembimbing II atas seluruh bantuan dan dukungannya selama penulis menyelesaikan masa studi.
5. Seluruh Dosen Program Studi D4 Teknik Informatika.
6. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Laporan Akhir dari awal hingga akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan akhir ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan laporan ini. Semoga laporan ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, 2 Juli 2025

Penulis

# DAFTAR ISI

[SAMPUL i](#_Toc202325223)

[HALAMAN PENGESAHAN ii](#_Toc202325224)

[PERNYATAAN iii](#_Toc202325225)

[KATA PENGANTAR iv](#_Toc202325226)

[DAFTAR ISI vi](#_Toc202325227)

[DAFTAR GAMBAR ix](#_Toc202325228)

[DAFTAR TABEL i](#_Toc202325229)

[BAB I. PENDAHULUAN 1](#_Toc202325230)

[1.1 Latar Belakang 1](#_Toc202325231)

[1.2 Rumusan Masalah 1](#_Toc202325232)

[1.3 Batasan Masalah 1](#_Toc202325233)

[1.4 Tujuan 2](#_Toc202325234)

[1.5 Manfaat 2](#_Toc202325235)

[BAB II. LANDASAN TEORI 3](#_Toc202325236)

[2.1 Big Data 3](#_Toc202325237)

[2.2 Hadoop 4](#_Toc202325238)

[2.3 Virtual Box 6](#_Toc202325239)

[2.4 Spring Boot 7](#_Toc202325240)

[2.5 React 9](#_Toc202325241)

[2.6 HBase 10](#_Toc202325242)

[BAB III. METODOLOGI PENELITIAN 12](#_Toc202325243)

[3.1 Analisis Kebutuhan 12](#_Toc202325244)

[3.2 Deskripsi Sistem 14](#_Toc202325245)

[3.3 Metode Pengembangan 16](#_Toc202325246)

[3.3.1 Analysis 17](#_Toc202325247)

[3.3.2 Design 17](#_Toc202325248)

[3.3.3 Implementation 21](#_Toc202325249)

[3.3.4 Testing 21](#_Toc202325250)

[3.3.5 Deployment 22](#_Toc202325251)

[3.3.6 Maintenence 22](#_Toc202325252)

[3.4 Uji Coba Sistem 22](#_Toc202325253)

[BAB IV ANALISIS DAN PERANCANGAN SISTEM 24](#_Toc202325254)

[4.1 Deskripsi Umum Sistem 24](#_Toc202325256)

[4.2 Analisis Kebutuhan 24](#_Toc202325257)

[4.2.1 Kebutuhan Perangkat Keras 24](#_Toc202325258)

[6.1.2 Kebutuhan Perangkat Lunak 26](#_Toc202325259)

[4.3 Perancagan Detail Sistem 27](#_Toc202325263)

[4.3.1 Perancangan File System Hadoop 27](#_Toc202325264)

[4.3.2 Perancangan Database NoSQL HBase 30](#_Toc202325265)

[4.3.3 Perancangan Alur Aplikasi 33](#_Toc202325266)

[BAB V IMPLEMENTASI SISTEM 38](#_Toc202325267)

[5.1 Implementasi Sistem 38](#_Toc202325268)

[5.1.1 Implementasi Antarmuka Administrator 38](#_Toc202325269)

[5.1.2 Implementasi Antarmuka Operator 42](#_Toc202325270)

[5.1.3 Implementasi Antarmuka Siswa 53](#_Toc202325271)

[BAB VI HASIL DAN PEMBAHASAN 56](#_Toc202325272)

[6.1 Pengujian UAT Sistem 56](#_Toc202325273)

[6.1.1 Testing sistem kepada user (Guru SMK Negeri Rowokangkung) 56](#_Toc202325274)

[6.1.2 Pengambilan Data Soal Ujian 56](#_Toc202325275)

[6.1.3 Pengambilan Data Siswa 57](#_Toc202325276)

[6.2 Pengujian Sistem Pelaksanaan Ujian 58](#_Toc202325277)

[6.2.1 Pengaturan Bank Soal Ujian 58](#_Toc202325278)

[6.2.2 Pengaturan User 59](#_Toc202325279)

[6.2.3 Pengaturan Ujian 59](#_Toc202325280)

[6.3 Hasil Data Pengujian 60](#_Toc202325281)

[6.3.1 Hasil Data Ujian Siswa 60](#_Toc202325282)

[BAB VII SARAN DAN KESIMPULAN 62](#_Toc202325283)

[7.1 Kesimpulan 62](#_Toc202325284)

[7.2 Saran 62](#_Toc202325285)

[DAFTAR PUSTAKA 64](#_Toc202325286)

# DAFTAR GAMBAR

[Gambar 3. 1 SDLC Waterfall 16](#_Toc202323410)

[Gambar 3. 2 Desain Sistem 19](#_Toc202323411)

[Gambar 4. 1 Struktur DataNode 29](#_Toc202323426)

[Gambar 4. 2 Alur Masuk Data HDFS 30](#_Toc202323427)

[Gambar 4. 3 Struktur HBase 33](#_Toc202323428)

[Gambar 4. 4 Alur Sistem Administrator 34](#_Toc202323429)

[Gambar 4. 5 Alur Sistem Operator Sekolah 35](#_Toc202323430)

[Gambar 4. 6 Alur Sistem Guru 36](#_Toc202323431)

[Gambar 4. 7 Alur Sistem Siswa 37](#_Toc202323432)

[Gambar 5. 1 Halaman Login 38](#_Toc202323444)

[Gambar 5. 2 Halaman Beranda 39](#_Toc202323445)

[Gambar 5. 3 Bidang Keahlian 40](#_Toc202323446)

[Gambar 5. 4 Program Keahlian 40](#_Toc202323447)

[Gambar 5. 5 Konsentrasi Keahlian 41](#_Toc202323448)

[Gambar 5. 6 Halaman Pengguna 42](#_Toc202323449)

[Gambar 5. 7 Bidang Keahlian Sekolah 42](#_Toc202323450)

[Gambar 5. 8 Program Keahlian Sekolah 43](#_Toc202323451)

[Gambar 5. 9 Konsentrasi Keahlian Sekolah 44](#_Toc202323452)

[Gambar 5. 10 Tahun Ajaran 44](#_Toc202323453)

[Gambar 5. 11 Semester 45](#_Toc202323454)

[Gambar 5. 12 Kelas 45](#_Toc202323455)

[Gambar 5. 13 Mata Pelajaran 46](#_Toc202323456)

[Gambar 5. 14 Modul 46](#_Toc202323457)

[Gambar 5. 15 Modul Filter 47](#_Toc202323458)

[Gambar 5. 16 Elemen 47](#_Toc202323459)

[Gambar 5. 17 Elemen Filter 48](#_Toc202323460)

[Gambar 5. 18 ACP 48](#_Toc202323461)

[Gambar 5. 19 ACP filter 49](#_Toc202323462)

[Gambar 5. 20 ATP 49](#_Toc202323463)

[Gambar 5. 21 ATP Filter 50](#_Toc202323464)

[Gambar 5. 22 Taksonomi 50](#_Toc202323465)

[Gambar 5. 23 Soal Ujian 51](#_Toc202323466)

[Gambar 5. 24 Bank Soal 51](#_Toc202323467)

[Gambar 5. 25 Setting Ujian 52](#_Toc202323468)

[Gambar 5. 26 Report Nilai Siswa 52](#_Toc202323469)

[Gambar 5. 27 Pengguna Sekolah 53](#_Toc202323470)

[Gambar 5. 28 Ujian Siswa 53](#_Toc202323471)

[Gambar 5. 29 Riwayat Ujian 54](#_Toc202323472)

[Gambar 5. 30 Pelaksanaan Ujian 54](#_Toc202323473)

[Gambar 5. 31 Penyelesaian Ujian 55](#_Toc202323474)

[Gambar 6. 1 Dokumentasi UAT User 56](#_Toc202323475)

[Gambar 6. 2 Dokumentasi Soal Ujian 57](#_Toc202323476)

[Gambar 6. 3 Data Siswa 58](#_Toc202323477)

[Gambar 6. 4 Data Bank Soal 59](#_Toc202323478)

[Gambar 6. 5 Data User 59](#_Toc202323479)

[Gambar 6. 6 Data Ujian 60](#_Toc202323480)

[Gambar 6. 7 Data Siswa Ujian 60](#_Toc202323481)

[Gambar 6. 8 Data Pelanggaran Siswa 61](#_Toc202323482)

# DAFTAR TABEL

[Tabel 3. 1 Deskripsi Konsep Aplikasi 14](#_Toc202323609)

# BAB I. PENDAHULUAN

* 1. **Latar Belakang**

Pendidikan modern menghadapi tantangan dalam mengelola evaluasi pembelajaran secara efektif dan efisien. Evaluasi pembelajaran yang dilakukan secara konvensional sering kali menemui berbagai kendala, seperti keterbatasan dalam pengelolaan soal, kesulitan dalam standarisasi ujian, serta lambatnya proses analisis hasil ujian n, kecepatan pemrosesan data yang tinggi agar informasi dapat diakses secara real-time, serta pengelolaan keamanan data untuk melindungi informasi sensitif, termasuk hasil ujian dan informasi pribadi siswa yang harus dijaga kerahasiaannya (Kaiiali et al., 2016; Merceedi & Sabry, 2021).

Oleh karena itu, Big Data memerlukan teknologi pendukung yang andal dan terintegrasi untuk memenuhi kebutuhan pengelolaan data yang kompleks dan dinamis dalam sistem pendidikan modern, seperti pemanfaatan sistem berbasis Hadoop, HBase, dan analitik berbasis kecerdasan buatan (AI) (Wang et al., 2020).

* 1. **Rumusan Masalah**

Berdasarkan latar belakang yang ada, maka rumusan masalah yang dapat diambil sebagai berikut:

1. Bagaimana merancang sistem ujian yang memanfaatkan bank soal terdistribusi berbasis Hadoop?
2. Bagaimana sistem ujian tersebut dapat menggunakan soal-soal dari bank soal?
3. Bagaimana sistem ujian bisa ada fitur *cheating detection*?
   1. **Batasan Masalah**

Pada penelitian ini, terdapat kebutuhan untuk membatasi masalah agar

tujuan penelitian dapat tercapai. Batasan masalah yang akan dibahas dalam

penelitian ini meliputi:

1. Sistem dirancang sebagai platform ujian digital yang terintegrasi dengan bank soal terdistribusi, di mana setiap sekolah hanya dapat mengakses soal miliknya sendiri untuk menjaga kerahasiaan data.
2. Sistem menyediakan fitur simulasi ujian yang dirancang untuk membantu siswa mempersiapkan diri secara mandiri.
3. Data yang digunakan dalam penelitian ini adalah data akademik dari SMK Negeri 1 Rowokangkung sebagai studi kasus.
   1. **Tujuan**

Tujuan penelitian ini adalah merancang sistem ujian yang terintegrasi dengan bank soal terdistribusi berbasis Hadoop dan HBase untuk SMK Negeri 1 Rowokangkung, serta mengimplementasikan mekanisme distribusi soal dari bank soal ke dalam sistem ujian tersebut.

* 1. **Manfaat**

Penelitian ini diharapkan dapat mempermudah pengelolaan soal ujian secara digital dan memastikan soal yang digunakan sesuai dengan standar evaluasi pembelajaran. Selain itu, penelitian ini mendukung pelaksanaan ujian berbasis komputer secara efisien di SMK Negeri 1 Rowokangkung. Sistem yang dikembangkan juga bermanfaat dalam menjaga kerahasiaan soal ujian serta meningkatkan transparansi dan keandalan proses evaluasi pembelajaran melalui teknologi Hadoop dan HBase.

# BAB II. LANDASAN TEORI

1. **Big Data**

Big Data mengacu pada kumpulan data dalam jumlah besar yang tidak dapat dikelola secara efisien menggunakan teknologi konvensional karena keterbatasan kapasitas penyimpanan, kecepatan pemrosesan, dan fleksibilitas data (Oracle Corporation, 2024).

Konsep Big Data pertama kali muncul pada awal tahun 2000-an untuk menjelaskan pertumbuhan eksponensial data digital yang dipengaruhi oleh meningkatnya penggunaan teknologi di berbagai sektor, termasuk pendidikan (Marr, 2021).

Big Data memiliki lima karakteristik utama yang dikenal sebagai 5V, yaitu Volume, Velocity, Variety, Veracity, dan Value (Gandomi & Haider, 2015).

1. Volume merujuk pada jumlah data yang sangat besar yang terus bertambah dari berbagai sumber, seperti data hasil ujian dari ribuan sekolah yang dikumpulkan untuk analisis Pendidikan (Williamson, 2020).
2. Velocity menggambarkan kecepatan data yang dihasilkan dan diproses dalam waktu nyata, misalnya pembaruan data soal ujian dan hasil ujian siswa yang dapat dilakukan secara langsung melalui sistem digital (Oracle Corporation, 2024).
3. Variety menunjukkan keberagaman format data, baik yang terstruktur seperti metadata soal ujian maupun yang tidak terstruktur seperti teks soal, gambar, atau video yang digunakan dalam sistem pembelajaran modern (Gandomi & Haider, 2015).
4. Veracity menekankan pentingnya menjaga keakuratan dan keandalan data untuk memastikan hasil analisis yang valid dan bebas dari kesalahan, terutama dalam sistem evaluasi pendidikan yang berbasis data besar (Merceedi & Sabry, 2021).
5. Value merujuk pada manfaat atau wawasan yang dapat diperoleh dari data untuk meningkatkan efisiensi pengelolaan pendidikan dan pengambilan keputusan berbasis data (Gandomi & Haider, 2015).

Dalam dunia pendidikan, Big Data memberikan peluang besar untuk meningkatkan efektivitas evaluasi pembelajaran, seperti analisis pola jawaban siswa untuk mengetahui pemahaman terhadap materi, prediksi performa siswa berdasarkan riwayat ujian sebelumnya, serta identifikasi tingkat kesulitan soal agar ujian dapat lebih terkalibrasi dengan baik (Chen, 2019).

Dengan menggunakan Big Data, institusi pendidikan dapat menghasilkan wawasan yang lebih mendalam untuk meningkatkan kualitas pembelajaran dan merancang strategi pengajaran yang lebih efektif (Chen, 2019).

Namun, implementasi Big Data dalam pendidikan juga menghadapi berbagai tantangan, seperti kebutuhan kapasitas penyimpanan yang besar untuk menyimpan data dari berbagai institusi pendidikan, kecepatan pemrosesan data yang tinggi agar informasi dapat diakses secara real-time, serta pengelolaan keamanan data untuk melindungi informasi sensitif, termasuk hasil ujian dan informasi pribadi siswa yang harus dijaga kerahasiaannya (Kaiiali et al., 2016; Merceedi & Sabry, 2021).

Oleh karena itu, Big Data memerlukan teknologi pendukung yang andal dan terintegrasi untuk memenuhi kebutuhan pengelolaan data yang kompleks dan dinamis dalam sistem pendidikan modern, seperti pemanfaatan sistem berbasis Hadoop, HBase, dan analitik berbasis kecerdasan buatan (AI) (Wang et al., 2020).

1. **Hadoop**

Hadoop adalah kerangka kerja open-source yang dikembangkan oleh Apache Software Foundation untuk mengatasi tantangan pengelolaan Big Data, yang sering kali mencakup volume data yang besar, kebutuhan pemrosesan cepat, dan keberagaman format data (Apache Hadoop, 2020).

Hadoop dirancang untuk menyimpan dan memproses data dalam skala besar secara terdistribusi di banyak komputer, memungkinkan efisiensi yang lebih tinggi dibandingkan sistem tradisional yang menggunakan penyimpanan terpusat (Gandomi & Haider, 2015).

Hadoop mampu menangani data dengan toleransi kegagalan yang tinggi, menjadikannya solusi yang andal untuk berbagai aplikasi Big Data, termasuk di bidang Pendidikan (Dean & Ghemawat, 2008).

Hadoop terdiri dari beberapa komponen inti yang saling mendukung, yaitu:

1. Hadoop Distributed File System (HDFS):

Sebagai sistem file terdistribusi, HDFS menyimpan data dalam blok-blok besar yang tersebar di beberapa node. HDFS dirancang untuk memastikan ketersediaan data meskipun terjadi kegagalan pada salah satu node, dengan replikasi data ke beberapa lokasi dalam cluster (Borthakur, 2007).

Hal ini menjadikannya sangat andal dalam pengelolaan data skala besar, termasuk data ujian yang dihasilkan dari berbagai sekolah (Merceedi & Sabry, 2021).

1. MapReduce:

Model pemrograman ini memungkinkan pemrosesan data dalam jumlah besar secara paralel di berbagai node dalam cluster Hadoop (Dean & Ghemawat, 2008).

Dengan memisahkan pemrosesan data menjadi dua tahap utama, yaitu "Map" dan "Reduce," teknologi ini mendukung analisis data dalam jumlah besar dengan efisiensi tinggi. MapReduce sangat cocok untuk pekerjaan seperti analisis pola jawaban siswa atau penghitungan statistik hasil ujian (Dean & Ghemawat, n.d.).

1. YARN (Yet Another Resource Negotiator):

Komponen ini bertanggung jawab untuk mengelola sumber daya dalam cluster Hadoop, termasuk penjadwalan pekerjaan dan alokasi sumber daya (Apache Hadoop, 2020).

YARN memungkinkan beberapa aplikasi dijalankan secara paralel di atas Hadoop, menjadikannya elemen penting dalam pengelolaan data ujian yang membutuhkan efisiensi tinggi dalam pemrosesan dan penyimpanan data (Apache Hadoop, 2020).

Selain komponen inti, Hadoop memiliki berbagai komponen pendukung dalam ekosistemnya, salah satunya adalah Apache HBase (George, 2011). HBase merupakan database NoSQL yang dirancang untuk menangani data tidak terstruktur dan mendukung akses real-time terhadap data besar (George, 2011).

Dalam konteks pengelolaan data ujian, HBase memungkinkan penyimpanan data soal ujian dan hasil ujian dalam berbagai format, seperti teks, gambar, dan metadata soal (George, 2011). HBase juga mendukung integrasi dengan HDFS untuk menyediakan akses data yang lebih cepat dan fleksibel dalam skenario analisis Big Data (Wang et al., 2020).

Hadoop telah menjadi solusi pilihan untuk berbagai implementasi Big Data karena skalabilitasnya yang tinggi dan kemampuannya dalam menangani data dengan volume besar (Apache Hadoop, 2020). Dalam penelitian ini, Hadoop digunakan sebagai infrastruktur utama untuk menyimpan dan mengelola data soal ujian secara terdistribusi (Wang et al., 2020).

Dengan Hadoop Distributed File System (HDFS) sebagai basis penyimpanan dan Apache HBase sebagai database NoSQL, sistem yang dirancang mampu menangani data ujian dalam skala besar dengan efisiensi tinggi, kecepatan akses real-time, dan tingkat keamanan yang tinggi (George, 2011).

Hadoop juga memungkinkan integrasi dengan teknologi pemrosesan data paralel seperti Apache Spark, yang dapat digunakan untuk analisis data yang lebih mendalam dan prediksi pola jawaban siswa (Zaharia et al., 2016).

1. **Virtual Box**

VirtualBox adalah perangkat lunak virtualisasi open-source yang dikembangkan oleh Oracle, memungkinkan pengguna untuk menjalankan beberapa sistem operasi secara bersamaan pada satu komputer fisik. Perangkat lunak ini menyediakan lingkungan terisolasi untuk menginstal, mengonfigurasi, dan menjalankan sistem operasi tambahan yang disebut mesin virtual (VM). VM ini berguna untuk berbagai keperluan, termasuk pengembangan, pengujian, dan implementasi teknologi berbasis server seperti Hadoop dan HBase (Oracle Corporation, 2021).

Salah satu keunggulan utama VirtualBox adalah dukungannya terhadap berbagai sistem operasi, termasuk Windows, Linux, dan macOS. Fleksibilitas ini memungkinkan pengguna untuk menginstal dan menjalankan berbagai OS sebagai mesin virtual, sesuai dengan kebutuhan mereka (Surety Systems, 2024).

VirtualBox juga menawarkan manajemen sumber daya yang fleksibel, memungkinkan pengguna untuk mengatur alokasi CPU, memori, dan ruang penyimpanan untuk setiap VM. Fitur ini memastikan bahwa setiap VM memiliki sumber daya yang cukup tanpa mengganggu kinerja sistem host (Oracle Documentation, 2020).

Selain itu, VirtualBox menyediakan fitur isolasi lingkungan, di mana setiap VM berjalan secara independen. Hal ini memastikan bahwa pengujian atau konfigurasi yang dilakukan pada satu VM tidak akan memengaruhi sistem utama atau VM lainnya (Oracle Documentation, 2020).

Dalam hal jaringan, VirtualBox menawarkan berbagai opsi konfigurasi, seperti NAT (Network Address Translation), Host-Only, dan Bridge Networking. Fitur ini memungkinkan VM untuk berkomunikasi dengan jaringan lokal maupun internet, sesuai dengan kebutuhan pengguna (Oracle Documentation, 2020).

Dalam penelitian ini, VirtualBox digunakan untuk menginstal dan mengonfigurasi Hadoop dan HBase dalam lingkungan yang terisolasi. Dengan menggunakan VirtualBox, sistem dapat diuji dan dikembangkan tanpa memengaruhi sistem utama, serta menyediakan fleksibilitas untuk menciptakan konfigurasi jaringan yang sesuai dengan kebutuhan (Codecentric AG, 2012).

Secara keseluruhan, VirtualBox menawarkan solusi virtualisasi yang kuat dan fleksibel untuk berbagai kebutuhan komputasi, mulai dari pengembangan dan pengujian perangkat lunak hingga implementasi sistem berbasis Big Data.

1. **Spring Boot**

Spring Boot, yang dikembangkan oleh Pivotal Software, adalah kerangka kerja open-source untuk membangun aplikasi berbasis Java dengan konfigurasi minimal dan efisiensi tinggi (Pivotal Software, 2024).

Framework ini dirancang untuk menyederhanakan pengembangan aplikasi berbasis Spring Framework dengan menyediakan konfigurasi otomatis, server web tersemat, dan sistem manajemen dependensi yang fleksibel (Kozak, 2023).

Salah satu fitur utama Spring Boot adalah kemampuannya dalam membuat aplikasi mandiri (standalone applications), di mana aplikasi dapat dijalankan tanpa perlu melakukan deploy ke server aplikasi eksternal. Hal ini dimungkinkan dengan adanya server web tersemat seperti Tomcat atau Jetty, yang langsung terintegrasi dalam aplikasi (Spring Framework, 2024).

Spring Boot mengikuti prinsip konvensi dibandingkan konfigurasi (convention over configuration), memungkinkan pengembang untuk langsung menggunakan pengaturan default yang optimal tanpa harus menulis konfigurasi yang panjang dan kompleks (Kozak, 2023).

Selain itu, framework ini mendukung pengembangan berbasis mikroservis dengan integrasi Spring Cloud, yang membantu dalam pengelolaan layanan terdistribusi, komunikasi antar layanan, serta penanganan kegagalan otomatis (Kozak, 2023).

Spring Boot dan Spring Cloud juga memudahkan pengembang dalam menerapkan containerization menggunakan Docker dan Kubernetes, yang semakin meningkatkan efisiensi dalam implementasi aplikasi berbasis cloud (Kozak, 2023).

Hal ini menjadikan Spring Boot sebagai solusi ideal untuk membangun aplikasi enterprise-level, termasuk dalam sistem manajemen ujian berbasis digital. Spring Boot juga mendukung berbagai model arsitektur aplikasi, seperti monolitik maupun mikroservis, dengan fitur bawaan seperti Spring Data untuk integrasi database, Spring Security untuk keamanan aplikasi, dan Spring MVC untuk pengembangan REST API (Pivotal Software, 2024).

Salah satu keunggulan utama framework ini adalah modularitas dan skalabilitasnya, yang memungkinkan sistem dapat dikembangkan dan diperluas dengan lebih fleksibel sesuai kebutuhan (Kozak, 2023).

Dalam penelitian ini, Spring Boot digunakan sebagai backend utama untuk menangani proses bisnis dalam sistem ujian digital, termasuk pengelolaan bank soal, autentikasi pengguna, serta pembuatan laporan hasil ujian (Kozak, 2023).

Backend Spring Boot akan berkomunikasi dengan database HBase untuk pengelolaan data ujian secara real-time melalui API RESTful (Spring Framework, 2024).

Dengan kombinasi Spring Boot dan HBase, sistem yang dikembangkan dapat menangani volume data besar dengan kecepatan tinggi, efisiensi penyimpanan, dan skalabilitas yang lebih baik dibandingkan sistem berbasis RDBMS tradisional (Kozak, 2023).

1. **React**

React JS adalah pustaka JavaScript yang dikembangkan oleh Facebook (sekarang Meta) untuk membangun antarmuka pengguna berbasis komponen yang dapat digunakan kembali dan mudah dikelola (React, 2024).

Salah satu fitur utama React JS adalah pendekatan Virtual DOM (Document Object Model), yang meningkatkan performa aplikasi dengan hanya memperbarui bagian yang berubah, bukan seluruh halaman, sehingga proses rendering menjadi lebih efisien dan cepat (Bawane, 2022).

React JS menggunakan pendekatan komponen (component-based architecture), yang memungkinkan pengembang untuk membangun antarmuka pengguna dari unit-unit kecil yang dapat digunakan kembali. Dengan adanya sistem ini, pengembang dapat mengembangkan UI yang modular dan fleksibel, sehingga meningkatkan efisiensi pengembangan serta skalabilitas aplikasi (Bawane, 2022).

React mendukung penggunaan JSX (JavaScript XML), yang memungkinkan pengembang untuk menulis sintaks yang mirip dengan HTML dalam kode JavaScript. JSX mempermudah pembuatan dan manipulasi elemen UI, sekaligus meningkatkan keterbacaan kode (React, 2024).

Selain itu, React mendukung pengelolaan state dan props, yang memungkinkan komunikasi antara komponen dan penyimpanan data yang lebih dinamis (Vyas, 2022).

React juga mendukung berbagai paradigma pengembangan modern seperti Single Page Application (SPA) dan Server-Side Rendering (SSR). Dalam aplikasi SPA, React memungkinkan navigasi antar halaman tanpa perlu memuat ulang seluruh halaman, menghasilkan pengalaman pengguna yang lebih responsif dan lancar (Bawane, 2022)​.

Untuk optimasi performa lebih lanjut, React dapat dikombinasikan dengan framework seperti Next.js, yang menawarkan fitur seperti prerendering dan caching untuk meningkatkan kecepatan aplikasi (Vercel, 2024).

Selain itu, ekosistem React yang luas juga memungkinkan integrasi dengan berbagai pustaka dan alat pengembang lainnya seperti Redux untuk pengelolaan state global, React Router untuk navigasi antar halaman, serta berbagai pustaka UI seperti Material-UI dan Ant Design (Bawane, 2022).

Dengan arsitektur yang fleksibel dan modular, React banyak digunakan oleh perusahaan besar seperti Facebook, Instagram, Airbnb, dan Netflix untuk membangun aplikasi web interaktif dan dinamis (React, 2024).

Dalam penelitian ini, React JS digunakan sebagai frontend untuk mengembangkan sistem ujian digital berbasis web. Dengan menggunakan arsitektur berbasis komponen dan pengelolaan state yang efisien, sistem ini dapat memberikan pengalaman pengguna yang interaktif, responsif, serta mudah dikembangkan dan diperluas sesuai dengan kebutuhan institusi pendidikan. Integrasi dengan backend berbasis Spring Boot dan HBase memungkinkan sistem ini untuk menangani data ujian dalam skala besar dengan kecepatan tinggi dan efisiensi optimal (Vyas, 2022)​.

1. **HBase**

HBase adalah sistem basis data NoSQL berorientasi kolom yang terdistribusi, toleran terhadap kesalahan, dan sangat terukur, yang dibangun di atas Apache Hadoop Distributed File System (HDFS) (Li & Chen, 2017).

HBase dirancang untuk menangani data dalam jumlah besar yang memerlukan akses cepat dan pemrosesan real-time dengan latensi rendah, menjadikannya pilihan utama dalam berbagai skenario Big Data, termasuk sistem analitik, pemrosesan data ilmiah, dan sistem informasi akademik (Mohammed et al., 2014).

Salah satu fitur utama HBase adalah kemampuannya untuk mengelola dataset besar dengan model penyimpanan berbasis kolom, yang lebih efisien dibandingkan dengan basis data relasional tradisional dalam menangani volume data besar (Naresh et al., 2017).

Dalam arsitektur HBase, data disimpan dalam bentuk families column, yang memungkinkan pencarian dan penyimpanan data secara terstruktur serta meningkatkan efisiensi retrieval data (Elashry et al., 2018).

Integrasi HBase dengan Hadoop memberikan keunggulan dalam penyimpanan data yang terdistribusi serta akses paralel dengan throughput tinggi. Hal ini sangat berguna dalam skenario yang memerlukan akses cepat terhadap data ujian atau dataset akademik berskala besar (Li & Chen, 2017).

Selain itu, HBase mendukung operasi transaksi real-time, seperti pembaruan, penambahan, dan penghapusan data, yang membuatnya cocok untuk aplikasi berbasis analitik, termasuk analisis performa ujian digital dan sistem evaluasi akademik (Mohammed et al., 2014).

Dengan mekanisme pemrosesan berbasis multi-node cluster, HBase dapat memastikan ketahanan data (fault tolerance) serta meminimalisir downtime dalam sistem yang menangani jutaan transaksi per detik (Naresh et al., 2017).

Keunggulan lain dari HBase adalah kemampuannya dalam menggunakan indeks multidimensi, yang memungkinkan sistem untuk menyusun dan mengambil data berdasarkan parameter tertentu dengan efisiensi tinggi (Elashry et al., 2018).

Selain itu, HBase menggunakan algoritma distribusi data berbasis hashing, yang memastikan bahwa data tersebar secara merata di seluruh node penyimpanan dalam cluster Hadoop. Dengan mekanisme ini, sistem berbasis HBase dapat memproses permintaan kueri dalam jumlah besar tanpa mengalami degradasi performa (Li & Chen, 2017).

Dalam konteks penelitian ini, HBase digunakan sebagai basis data utama untuk menyimpan dan mengelola bank soal ujian digital secara terdistribusi, memastikan kecepatan akses tinggi dan efisiensi penyimpanan data yang besar. Integrasi HBase dengan Hadoop memungkinkan pengolahan data soal ujian dengan volume besar secara real-time, serta memberikan fleksibilitas dalam skema penyimpanan data dibandingkan dengan sistem basis data relasional (Mohammed et al., 2014).

Oleh karena itu, dengan kombinasi HDFS dan HBase, sistem ujian digital yang dikembangkan akan memiliki keamanan, efisiensi, dan skalabilitas yang tinggi.

# BAB III. METODOLOGI PENELITIAN

* 1. **Analisis Kebutuhan**

Dalam penelitian ini akan dikembangkan sistem bank soal terdistribusi multi sekolah, yang dirancang untuk mengelola, menyimpan, dan mendistribusikan soal ujian secara efisien di SMK Negeri Rowokangkung. Sistem ini bertujuan untuk mengatasi tantangan dalam pengelolaan bank soal dan pelaksanaan ujian digital, seperti kesulitan mengakses soal antar sekolah, risiko kebocoran soal, serta perlunya sistem yang lebih aman dan efisien dalam penyimpanan serta distribusi soal (Wang et al., 2020).

Sistem yang dikembangkan harus mendukung manajemen bank soal multi sekolah, di mana setiap sekolah memiliki akses eksklusif terhadap bank soal mereka sendiri, tanpa dapat melihat atau mengubah soal milik sekolah lain. Untuk memastikan sistem berjalan dengan efisiensi tinggi, digunakan Hadoop Distributed File System (HDFS) untuk menyimpan data soal dalam jumlah besar secara terdistribusi, serta HBase sebagai basis data NoSQL untuk mengelola soal ujian (Merceedi & Sabry, 2021).

Teknologi Spring Boot akan digunakan untuk backend, menangani proses autentikasi pengguna, pengelolaan data soal, serta komunikasi dengan HBase melalui API RESTful. Sementara itu, frontend berbasis React JS akan digunakan untuk menyediakan antarmuka pengguna yang responsif dan mudah diakses oleh berbagai pengguna, seperti administrator, admin sekolah, guru, dan siswa (Bawane, 2022).

Sistem ini harus memenuhi beberapa kebutuhan utama, yaitu:

1. Manajemen Bank Soal Terdistribusi

* Guru dapat menambahkan, mengedit, dan menghapus soal berdasarkan kategori mata pelajaran dan tipe soal.
* Setiap sekolah hanya dapat mengakses dan mengelola bank soalnya sendiri untuk menjaga kerahasiaan data.

1. Simulasi Ujian

* Siswa dapat mengikuti ujian dengan sistem
* Simulasi ini membantu siswa dalam mempersiapkan diri sebelum ujian resmi.

1. Pelaksanaan Ujian Digital

* Ujian digital harus memiliki fitur cheat detection untuk menghindari kecurangan.
* Sistem harus memungkinkan guru atau operator untuk mengatur jadwal ujian.

1. Laporan Hasil Ujian

* Sistem akan menghasilkan laporan otomatis yang mencakup nilai individu siswa.
* Laporan ini akan digunakan oleh guru dan sekolah untuk mengevaluasi efektivitas metode pengajaran serta memberikan rekomendasi perbaikan bagi siswa.

Peran Pengguna dalam Sistem

Sistem ini akan memiliki empat peran utama, yaitu:

1. Administrator

* Bertanggung jawab untuk mengelola data sekolah yang terdaftar
* Bertanggung jawab untuk mengelola akun admin sekolah
* Bertanggung jawab untuk mengelola pengguna system
* Bertanggung jawab untuk konfigurasi server.

1. Admin Sekolah: Mengelola data akademik sekolah seperti tahun ajaran, semester, kelas, mata Pelajaran, soal, siswa, guru, jadwal ujian, dan laporan hasil ujian.
2. Guru: Mengelola bank soal, menyusun soal ujian, serta mengawasi jalannya ujian.
3. Siswa: Menggunakan sistem untuk mengikuti ujian dan mengerjakan ujian resmi yang telah dijadwalkan oleh guru.

Dengan arsitektur sistem yang terdistribusi dan berbasis Big Data, diharapkan sistem ini mampu meningkatkan efisiensi dalam pengelolaan ujian serta menjaga keamanan data soal ujian di SMK Negeri 1 Rowokangkung.

* 1. **Deskripsi Sistem**

Sesuai dengan latar belakang yang telah disampaikan sebelumnya, Sistem Bank Soal Terdistribusi Multi Sekolah dirancang untuk memenuhi kebutuhan pengelolaan soal ujian terdistribusi di SMK Negeri Rowokangkung.

Sistem ini bertujuan untuk memfasilitasi pengelolaan bank soal, pelaksanaan ujian digital, serta penyajian laporan hasil ujian secara otomatis. Dengan memanfaatkan teknologi Hadoop, HBase, Spring Boot, dan React, sistem ini mampu mengelola dan mengakses data soal dalam skala besar dengan efisiensi tinggi.

Sistem ini dirancang agar setiap sekolah hanya memiliki akses ke bank soal yang mereka miliki, sehingga menjaga kerahasiaan dan keamanan soal ujian. Selain itu, sistem ini menyediakan fitur pengacakan soal, laporan hasil ujian, serta sistem ujian.

Tabel 3. Deskripsi Konsep Aplikasi

|  |  |
| --- | --- |
| Judul | Perancangan Sistem Bank Soal Terdistribusi Multi Sekolah (Studi Kasus SMK Negeri Rowokangkung) |
| Jenis Aplikasi | Sistem berbasis web untuk pengelolaan dan pelaksanaan ujian |
| Pengguna | 1. Administrator 2. Admin Sekolah 3. Guru 4. Siswa |
| Hak Akses | * Administrator: Mengelola data sekolah yang terdaftar, akun pengguna, dan konfigurasi sistem. * Admin Sekolah: Mengelola data akademik sekolah, daftar siswa, guru, serta laporan ujian. * Guru: Menyusun soal ujian, menyusun kunci jawaban, dan memantau pelaksanaan ujian. * Siswa: Menggunakan sistem untuk mengikuti simulasi ujian maupun ujian sekolah. |
| Fitur Utama | * Manajemen Bank Soal berbasis Hadoop * Simulasi ujian untuk siswa. * Pelaksanaan Ujian Sekolah * Pengacakan Soal Ujian * Pelaporan hasil ujian. * Cheat Detection |
| Konten | Bank soal, pelaksanaan ujian, dan laporan hasil ujian |
| Aplikasi | Aplikasi berbasis website |
| Teknologi | * Virtual Machine * Hadoop * Hbase * Spring Boot * ReactJs |

Berdasarkan Tabel 3.1 tentang deskripsi aplikasi, maka kebutuhan sistem dikategorikan ke dalam kebutuhan perangkat lunak dan perangkat keras sebagai berikut:

* Kebutuhan Perangkat Lunak

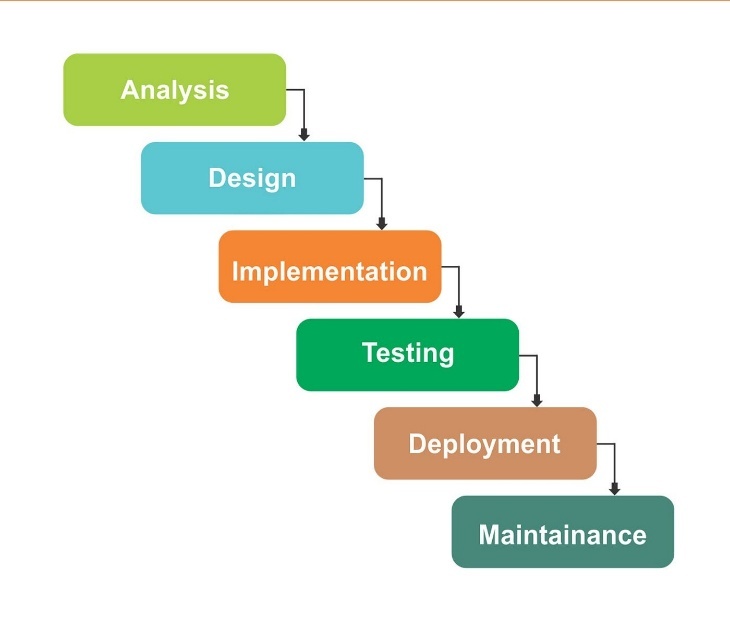
Perangkat lunak yang digunakan untuk mendukung proses pengembangan sistem ini meliputi:

* + Sistem Operasi: Windows/Linux/MacOS untuk pengembangan dan pengujian.
  + Spring Boot: Framework untuk pengembangan backend aplikasi.
  + React: Framework untuk pengembangan frontend aplikasi.
  + Hadoop: Teknologi penyimpanan dan pengelolaan data besar.
  + HBase: Database NoSQL untuk pengelolaan data terdistribusi.
  + IntelliJ IDEA: IDE untuk pengembangan backend menggunakan Java.
  + Visual Studio Code: IDE untuk pengembangan frontend berbasis React.
  + Postman: Alat pengujian API RESTful.
  + VirtualBox: Untuk menciptakan lingkungan virtual yang aman selama proses pengujian.
* Kebutuhan Perangkat Keras

Perangkat keras yang digunakan selama proses pengembangan sistem adalah:

* Server Lokal: Untuk pengujian sistem dengan spesifikasi minimal prosesor Intel i7, RAM 16 GB, dan penyimpanan SSD 500 GB.
* Laptop Pengembang: Dengan spesifikasi minimal prosesor Intel i5, RAM 8 GB, dan penyimpanan SSD 256 GB.
  1. **Metode Pengembangan**

Dalam perancangan sistem bank soal terdistribusi multi sekolah di SMK Negeri Rowokangkung, pengembangan sistem dilakukan dengan pendekatan Software Development Life Cycle (SDLC) model Waterfall. Model ini dipilih karena memiliki alur pengembangan yang terstruktur dan sesuai untuk memastikan setiap tahapan dalam pengembangan sistem dapat dilakukan dengan baik sebelum melanjutkan ke tahap berikutnya. Model Waterfall terdiri dari lima tahap utama, seperti ditunjukkan pada Gambar 3.1



Gambar 3. SDLC Waterfall

* + 1. **Analysis**

Pengembangan sistem bank soal terdistribusi multi sekolah dimulai dengan analisis kebutuhan untuk memastikan bahwa sistem dapat memenuhi persyaratan pengelolaan soal dan pelaksanaan ujian secara terintegrasi. Sistem ini dirancang untuk mendukung administrator, admin sekolah, guru, dan siswa dalam proses pengelolaan bank soal, pelaksanaan ujian, simulasi ujian, dan penyajian laporan hasil ujian.

Setiap pengguna sistem memiliki peran khusus. Administrator bertugas mengelola data sekolah yang berpartisipasi, data pengguna seperti guru dan siswa, serta pengaturan akses. Admin sekolah bertanggung jawab mengelola data akademik, termasuk data siswa, guru, kelas, jadwal ujian, dan struktur mata pelajaran. Guru memiliki peran utama dalam mengelola bank soal, termasuk menambahkan, mengedit, menghapus soal, serta menentukan bobot dan tingkat kesulitan soal. Siswa dapat menggunakan sistem untuk mengikuti simulasi ujian dan ujian resmi yang telah disiapkan oleh guru.

Proses penggunaan sistem dimulai dengan pendaftaran data sekolah, pengguna, dan data akademik ke dalam sistem oleh administrator atau admin sekolah. Guru kemudian dapat mulai mengelola bank soal dengan menambahkan soal-soal baru ke sistem. Soal tersebut akan dikelompokkan berdasarkan atribut seperti mata pelajaran, tingkat kesulitan, dan jenis soal untuk memudahkan pengelolaan. Ketika ujian dijadwalkan, siswa dapat masuk ke sistem, mengakses ujian, dan menjawab soal yang diacak secara otomatis oleh sistem untuk menjaga kerahasiaan dan keadilan. Setelah ujian selesai, hasil jawaban siswa diproses secara otomatis dan tersimpan dalam database. Guru dapat mengakses laporan hasil ujian untuk melakukan evaluasi lebih lanjut.

* + 1. **Design**

Setelah analisis kebutuhan selesai dilakukan, langkah selanjutnya adalah dengan mengacu pada Work Breakdown Structure (WBS) untuk merancang sistem. WBS digunakan untuk memetakan elemen-elemen utama yang diperlukan dalam pengembangan sistem dan memastikan bahwa setiap fungsi yang dirancang sesuai dengan peran dan kebutuhan pengguna.

WORK BREAKDOWN STRUCTURE PERANCANGAN SISTEM BANK SOAL TERDISTRIBUSI MULTI SEKOLAH STUDI KASUS SMK NEGERI ROWOKANGKUNG

Siswa

Guru

Admin Sekolah

Administrator

Ujian

Capaian Pembelajaran

Tahun Ajaran

Data Sekolah

Absensi

Latihan Ujian

Tujuan Pembelajaran

Bidang Keahlian

Guru

Program Keahlian

Pembuatan Soal

Bank Soal

Modul Ajar

Materi Pembelajaran

Pembobotan Soal

Siswa

Data Pengguna

Kelas

Mata Pelajaran

Ruang Kelas

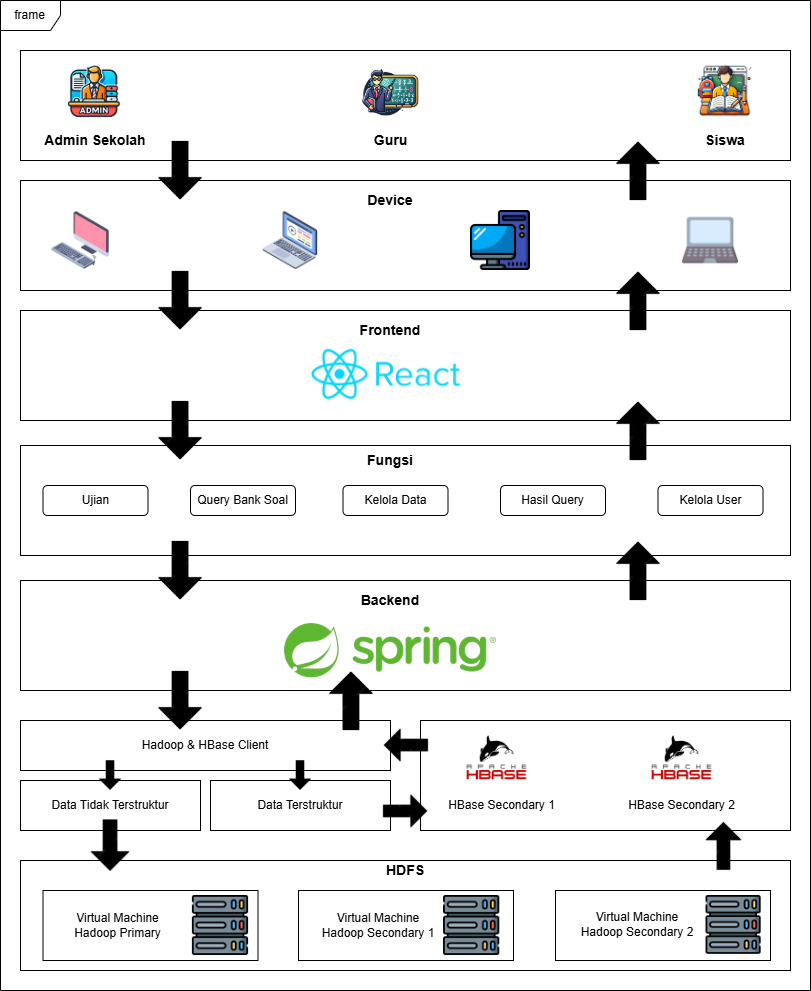
Kategori Penilaian

Absensi

Nilai Siswa

Konsentrasi Keahlian

Selain WBS, desain alur sistem berikut ini memberikan gambaran lebih rinci tentang bagaimana interaksi antar komponen, termasuk pengguna, frontend, backend, serta penyimpanan data terdistribusi, mendukung fungsionalitas sistem yang dirancang. Berikut gambar dan penjelasan dari masing-masing blok:



Gambar 3. Desain Sistem

1. Blok Pertama (Pengguna)

* Admin Sekolah: Bertanggung jawab atas manajemen data sekolah, seperti informasi guru, siswa, mata pelajaran, dan jadwal ujian.
* Guru: Berperan dalam pengelolaan bank soal, pembuatan soal ujian, serta pemantauan hasil ujian
* Siswa: Mengakses sistem untuk mengikuti simulasi ujian dan ujian resmi.

1. Blok Kedua (Device)

Setiap pengguna berinteraksi dengan sistem melalui perangkat mereka, seperti laptop atau komputer, untuk mengakses antarmuka aplikasi. Hanya laptop dan komputer yang berfungsi hal ini dikarenakan ujian diadakan di sekolah.

1. Blok Ketiga (Frontend)

Antarmuka pengguna dikembangkan menggunakan React, sebuah framework berbasis JavaScript yang memungkinkan pengalaman pengguna yang responsif dan dinamis.

1. Blok Keempat (Fungsi)

Pada blok keempat berisi fungsi dari sistem diantaranya adalah:

* Ujian, merupakan fungsi untuk melakukan latihan ujian, ujian, distribusi soal, pengacakan soal, penilaian.
* Query Bank Soal, merupakan fungsi penyimpanan butir soal dengan memanfaatkan sistem big data *Hadoop* dan juga *Hbase*.
* Kelola Data, merupakan fungsi CRUD dari setiap fitur
* Hasil Query, merupakan fungsi untuk menyimpan nilai dari hasil ujian yang telah dilakukan.
* Kelola User, merupakan fungsi untuk mengatur dan CRUD user oleh administrator dan admin sekolah

1. Blok Kelima (Backend)

Backend dikembangkan menggunakan Spring Boot, sebuah framework berbasis Java yang menangani logika bisnis dan komunikasi dengan database. Spring Boot menyediakan API untuk mendukung semua fungsi frontend, seperti autentikasi pengguna, penyimpanan data, dan pengelolaan hasil ujian.

1. Blok Keenam (Tipe data, Hadoop dan HBase client)

Sistem backend berkomunikasi dengan komponen penyimpanan data melalui Hadoop dan HBase Client untuk mengakses data terstruktur dan tidak terstruktur.

* Data Tidak Terstruktur: Termasuk file, seperti gambar atau media lainnya.
* Data Terstruktur: Soal ujian dan informasi akademik

1. HDFS (Hadoop Distributed Filesystem)

Semua data disimpan dalam sistem file terdistribusi berbasis Hadoop, dengan komponen sebagai berikut:

* Virtual Machine Hadoop Primary: Server utama yang menangani distribusi data.
* Virtual Machine Hadoop Secondary 1 & 2: Server cadangan untuk mendukung redundansi dan skalabilitas.

1. HBase

Basis data HBase digunakan untuk menyimpan data ujian secara efisien dengan latensi rendah. HBase Secondary 1 & 2 adalah replika basis data untuk memastikan ketersediaan data dan mencegah kegagalan sistem.

* + 1. **Implementation**

Hasil dari design akan diimplementasikan ke dalam sebuah kode program untuk semua fungsi ataupun modul yang akan dibangun. Adapun bahasa pemrograman yang digunakan adalah Java dan Javascript.

* + 1. **Testing**

Testing dilakukan dengan cara UAT:

* User Acceptance Testing (UAT): Pengujian sistem oleh pengguna untuk memastikan sistem telah memenuhi kebutuhan mereka.
  + 1. **Deployment**

Sistem bank soal terdistribusi diimplementasikan pada server lokal milik SMK Negeri Rowokangkung. Proses ini meliputi instalasi backend, frontend, dan database dengan konfigurasi yang sesuai untuk mendukung teknologi Hadoop, HBase, Spring Boot, dan React.

* + 1. **Maintenence**

Tahap maintenance bertujuan untuk memastikan sistem tetap berjalan dengan baik setelah implementasi. Aktivitas ini mencakup perbaikan bug yang ditemukan selama penggunaan, pembaruan sistem untuk menambah fitur atau meningkatkan performa, serta pemantauan rutin terhadap performa sistem.

* 1. **Uji Coba Sistem**

Uji coba sistem dilakukan untuk memastikan bahwa sistem yang dikembangkan dapat berfungsi dengan baik sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Proses ini bertujuan untuk mengidentifikasi kesalahan atau kekurangan dalam sistem serta mengevaluasi kinerja sistem dalam berbagai kondisi operasional. Uji coba merupakan langkah penting untuk memastikan perangkat lunak memenuhi kebutuhan pengguna dan spesifikasi teknis (Sommerville, 2011). Berikut adalah tahapan dalam uji coba sistem:

1. Persiapan Uji Coba

Pada tahap ini, dilakukan persiapan berupa pengumpulan data uji yang relevan, penyiapan lingkungan pengujian, serta penentuan skenario uji coba. Data uji disusun berdasarkan kasus-kasus yang mungkin terjadi dalam implementasi sistem. Skenario uji coba dirancang untuk mencakup pengujian normal, ekstrem, dan kondisi kesalahan (Pressman, 2010).

1. Pelaksanaan Uji Coba

Uji coba dilakukan dengan menggunakan data uji yang telah disiapkan. Setiap fitur dan fungsi sistem diuji untuk memastikan bahwa sistem mampu menangani berbagai input dengan benar. Beberapa metode pengujian yang digunakan, antara lain:

* User Acceptance Testing (UAT): Pengujian sistem oleh pengguna untuk memastikan sistem telah memenuhi kebutuhan mereka.

1. Evaluasi Hasil Uji Coba

Setelah uji coba dilakukan, hasilnya dianalisis untuk menentukan apakah sistem sudah memenuhi spesifikasi dan kebutuhan. Jika ditemukan kesalahan (bug) atau kekurangan, dilakukan perbaikan dan pengujian ulang hingga sistem berfungsi optimal. Hasil pengujian dibandingkan dengan kriteria penerimaan untuk memastikan keberhasilan pengujian (Myers et al., 2004).

Setelah seluruh perbaikan selesai, sistem akan dipersiapkan untuk implementasi penuh di SMK Negeri 1 Rowokangkung, termasuk penyusunan rencana maintenance dan dukungan teknis.

# BAB IV ANALISIS DAN PERANCANGAN SISTEM

1. 1. **Deskripsi Umum Sistem**

Penelitian ini berfokus pada perancangan “Sistem Bank Soal Terdistribusi Multi Sekolah” berbasis big data dengan studi kasus di SMK Negeri Rowokangkung. Sistem ini dirancang untuk memfasilitasi proses penyimpanan, pengelolaan, dan penggunaan bank soal serta penyelenggaraan ujian secara online.

Proses utama yang difasilitasi meliputi manajemen bank soal, pelaksanaan ujian sekolah, simulasi ujian bagi siswa, hingga pelaporan hasil ujian. Untuk mengatasi tantangan skalabilitas dan volume data soal yang besar, serta kebutuhan akan akses data yang cepat dan fleksibel, sistem ini memanfaatkan teknologi Big Data, khususnya ekosistem Hadoop untuk penyimpanan terdistribusi melalui Hadoop Distributed File System (HDFS) dan Apache HBase sebagai basis data NoSQL.

Arsitektur aplikasi ini didukung oleh Spring Boot untuk pengembangan dari sisi backend yang menangani logika pemrograman dan interaksi dengan HDFS, sementara antarmuka pengguna (frontend) dikembangkan menggunakan React JS untuk menciptakan pengalaman pengguna yang interaktif dan responsif.

* 1. **Analisis Kebutuhan**

Dalam melakukan penelitian ini, yang mencakup perancangan, pengembangan, hingga tahap pengujian “Sistem Bank Soal Terdistribusi Multi Sekolah”, diperlukan penyiapan lingkungan yang memadai. Lingkungan tersebut membutuhkan dukungan dari berbagai aspek, yang secara umum dikelompokkan menjadi dua kebutuhan pokok, yaitu kebutuhan perangkat keras dan kebutuhan perangkat lunak.

1. **Kebutuhan Perangkat Keras**

Pada bagian kebutuhan perangkat keras ini menjabarkan berbagai macam perangkat keras yang digunakan pada penelitian. Adapun spesifikasi Kebutuhan perangkat keras yang digunakan adalah sebagai berikut:

1. Laptop Server:
2. OS : Windows 11 64 Bit
3. Processor : AMD Ryzen 5 5600U with AMD Radeon Graphics
4. Ram : 16 GB
5. Jumlah : 1
6. Fungsi : Untuk menjalankan 4 VM yang terdiri dari Hadoop dan HBase
7. Laptop Client:
8. OS : Windows 11 64 Bit
9. Processor : AMD Ryzen 5 5600U with AMD Radeon Graphics
10. Ram : 16 GB
11. Jumlah : 1
12. Fungsi : Untuk menguji fungsionalitas Rest API
13. Hadoop Primary
14. OS : Ubuntu Server 24.2
15. Prosesor : 1 CPUs
16. RAM : 2 GB
17. Hardisk : 100 GB
18. Jumlah : 1
19. Fungsi : Untuk menjadi master dari Hadoop dan Hbase
20. Hadoop Secondary 1
21. OS : Ubuntu Server 24.2
22. Prosesor : 1 CPUs
23. RAM : 2 GB
24. Hardisk : 100 GB
25. Jumlah : 1
26. Fungsi : Untuk menjalankan hadoop yang ke 1 dan regionServer 1
27. Hadoop Secondary 2
28. OS : Ubuntu Server 24.2
29. Prosesor : 2 CPUs
30. RAM : 1 GB
31. Hardisk : 100 GB
32. Jumlah : 1
33. Fungsi : Untuk menjalankan hadoop yang ke 2 dan regionServer 2
34. Zookeeper 1
35. OS : Ubuntu Server 24.2
36. Processor : 2 CPUs
37. RAM : 1 GB
38. Hardisk : 25 GB
39. Jumlah : 1
40. Fungsi : Untuk mengkoordinasi HBase dengan Hadoop
41. Zookeeper 2
42. OS : Ubuntu Server 24.2
43. Processor : 2 CPUs
44. RAM : 1 GB
45. Hardisk : 25 GB
46. Jumlah : 1
47. Fungsi : Untuk mengkoordinasi HBase dengan Hadoop
48. Zookeeper 2
49. OS : Ubuntu Server 24.2
50. Processor : 2 CPUs
51. RAM : 1 GB
52. Hardisk : 25 GB
53. Jumlah : 1
54. Fungsi : Untuk mengkoordinasi HBase dengan Hadoop
55. **Kebutuhan Perangkat Lunak**

Pada bagian kebutuhan perangkat lunak ini menjabarkan berbagai macam perangkat lunak yang digunakan pada penelitian. Adapun spesifikasi Kebutuhan perangkat lunak yang digunakan adalah sebagai berikut:

1. Oracle VirtualBox

Oracle VirtualBox adalah perangkat lunak yang memungkinkan untuk menjalankan beberapa sistem operasi di komputer. VirtualBox dapat digunakan untuk penggunaan pribadi maupun Perusahaan (Corporation, 2025). Perangkat Lunak ini digunakan untuk virtualisasi machine yang memungkinkan pengguna dapat membuat dan menjalankan mesin virtual diatas sistem operasi yang berjalan.

1. Intellij IDEA

Intellij IDEA adalah Integrated Development Environment (IDE) yang dirancang untuk memaksimalkan produktivitas pengembang, khususnya untuk Java dan Kotlin. IDE ini dirancang untuk memaksimalkan produktivitas pengembang dan memiliki fokus yang kuat pada privasi dan keamanan (Intellij IDEA, 2025). Perangkat lunak ini digunakan untuk menjalankan sistem backend, berkomunikasi dengan HDFS, dan test API.

1. Visual Studio Code

Visual Studio Code (VS Code) adalah editor kode lintas platform gratis yang dikembangkan oleh Microsoft. Ini dirancang untuk memberikan pengalaman pengkodean yang efisien dan efisien bagi pengembang yang bekerja di berbagai platform, termasuk Windows, macOS, dan Linux (Microsoft, 2025). Perangkat Lunak ini digunakan untuk mengembangkan sisi antarmuka (frontend).

* 1. **Perancagan Detail Sistem**

Untuk mendukung proses pengujian, pada tahap ini dilakukan serangkaian perancangan yang mencakup perancangan topologi, database, utilitas pengujian, skenario pengujian, dan penentuan data uji yang akan digunakan.

1. **Perancangan File System Hadoop**

Arsitektur file system Hadoop dalam penelitian ini mengimplementasikan konfigurasi cluster 6-node yang terdiri dari Apache Hadoop Distributed File System (HDFS) dan Apache HBase, yang berfungsi sebagai sistem penyimpanan terdistribusi dengan ketahanan tinggi terhadap gangguan (Apache Software Foundation, 2024).

Konfigurasi cluster yang digunakan terdiri dari:

* 1 Primary Node: hadoop-primary (192.168.50.10) yang berperan sebagai NameNode dan HBase Master
* 2 Secondary Node: hadoop-secondary-1 (192.168.50.20) dan hadoop-secondary-2 (192.168.50.30) yang berfungsi sebagai DataNode dan RegionServer
* 3 ZooKeeper Node: zookeeper-1 (192.168.50.40), zookeeper-2 (192.168.50.50), dan zookeeper-3 (192.168.50.60) untuk mengatur koordinasi dan kesepakatan dalam cluster.

Proses penyimpanan data dalam HDFS memiliki alur yang terstruktur. Pertama, HDFS Client memulai proses upload file ke cluster melalui API HDFS. HDFS menggunakan arsitektur master/slave dengan NameNode sebagai server utama yang mengelola namespace file system dan mengatur akses file oleh client (Apache Software Foundation, 2024).

Client kemudian menghubungi NameNode (192.168.50.10) untuk mendapatkan informasi metadata, termasuk lokasi DataNode yang tersedia, konfigurasi ukuran block, dan faktor replikasi. Data dalam HDFS selalu disimpan dalam bentuk block dengan ukuran default 128MB, dimana setiap block memiliki identifikasi unik (Apache Software Foundation, 2024).

Data ditulis secara streaming ke DataNode yang dipilih dengan mekanisme checksum untuk menjaga keutuhan data. Setiap blok data yang ditulis akan disalin sebanyak tiga kali sesuai dengan konfigurasi default system (Apache Software Foundation, 2024).

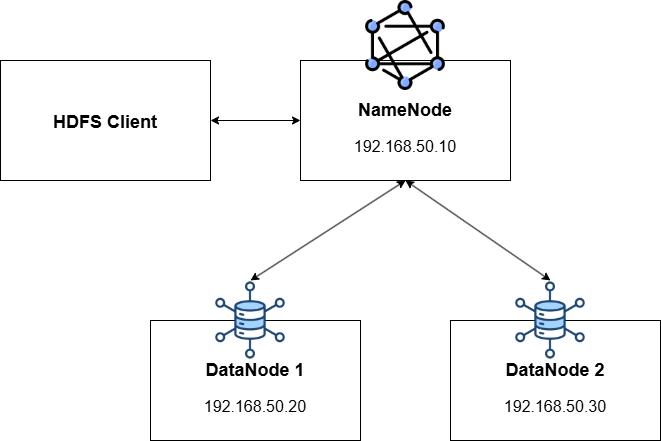
Salinan pertama ditempatkan pada node yang menulis data, salinan kedua ditempatkan pada node lain yang berada pada rak berbeda, apabila tersedia, dan salinan ketiga ditempatkan pada node lain berada pada rak yang sama dengan Salinan kedua. Sistem memberikan konfirmasi bahwa semua salinan berhasil ditulis dan metadata telah diperbarui di NameNode (Apache Software Foundation, 2024).

Implementasi 3 node ZooKeeper menggunakan algoritma konsensus Zab (ZooKeeper Atomic Broadcast) dengan konsep quorum mayoritas. Dengan 3 node, sistem dapat menoleransi kegagalan 1 node sambil mempertahankan quorum 2/3 untuk pengambilan keputusan. ZooKeeper adalah layanan terpusat untuk memelihara informasi konfigurasi, penamaan, sinkronisasi terdistribusi, dan layanan grup (Bikas Katwal, 2019).

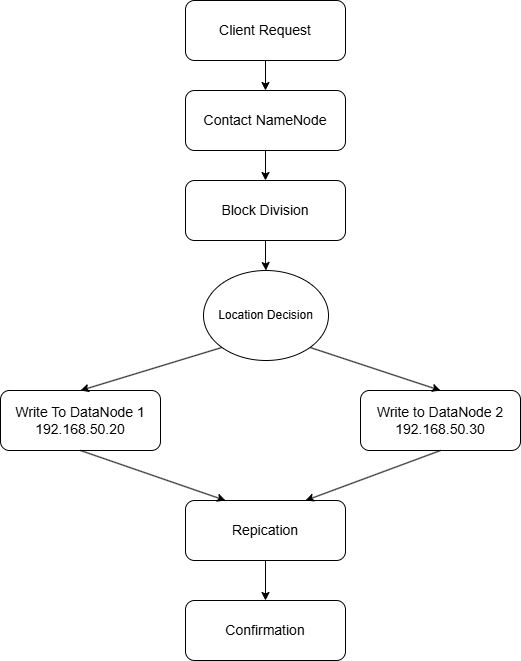
ZooKeeper menjalankan fungsi penting seperti:

* Leader Election : memilih HBase Master aktif dan menangani failover otomatis
* Configuration Management : menyimpan dan mendistribusikan konfigurasi cluster secara terpusat
* Distributed Locking: mencegah konflik operasi bersamaan pada level sistem
* Node Monitoring : memantau status kesehatan semua node dalam cluster (Noobmaster, 2021).

Melalui arsitektur cluster Hadoop 6-node ini, HDFS mampu menyediakan platform penyimpanan data dengan skalabilitas yang tinggi, ketahanan terhadap kegagalan, konsistensi yang kuat melalui koordinasi ZooKeeper, serta kemampuan akses data yang cepat. Visualisasi perancangan Hadoop ini dapat dilihat pada Gambar 4.1 yang menunjukkan struktur DataNode dan Gambar 4.2 yang menunjukkan alur masuk data HDFS.



Gambar 4. Struktur DataNode



Gambar 4. Alur Masuk Data HDFS

1. **Perancangan Database NoSQL HBase**

Arsitektur database NoSQL HBase dalam konfigurasi cluster terintegrasi dengan infrastruktur Hadoop. HBase adalah database NoSQL open-source yang terdistribusi, yang menggunakan struktur tabel berorientasi kolom dengan model data yang adaptif dan fleksibel, berbeda dengan sistem database relasional konvensional (Unogeeks, 2024).

Apache HBase biasanya dijalankan di atas HDFS dengan Apache ZooKeeper sebagai komponen penting untuk memelihara informasi konfigurasi dan mengelola seluruh cluster Apache HBase (Unogeeks, 2024).

Setiap tabel dalam HBase terdiri dari baris (row) dengan kunci unik sebagai identifikasi, yang terdiri dari beberapa column family. Setiap column family mencakup berbagai kolom dengan qualifier spesifik, dimana data disimpan dalam bentuk cell yang terdiri dari nilai dan timestamp.

Struktur tabel yang digunakan memiliki 3 skema column family:

* Detail : untuk menyimpan informasi detail record
* Main : untuk menyimpan data utama aplikasi
* Family : untuk menyimpan data relasional antar entitas

Database dalam HBase berbentuk cluster yang terdiri dari beberapa node, dimana setiap node bertanggung jawab mengelola sebagian data tabel dan mendistribusikannya secara horizontal.

Operasi write dalam HBase dimulai ketika client mengakses sistem melalui HBase Client API untuk menginisiasi proses penyimpanan data. HBase Client berkomunikasi dengan ZooKeeper untuk mendapatkan informasi lokasi HBase Master yang sedang aktif, kemudian HBase Master melakukan assignment region ke RegionServer yang sesuai berdasarkan partisi data.

RegionServer menulis data ke Write Ahead Log (WAL) terlebih dahulu untuk memastikan durabilitas data, dimana WAL berfungsi sebagai mekanisme recovery yang mencatat semua perubahan sebelum data benar-benar tersimpan (Hbase, 2025).

Selanjutnya, data disimpan sementara di MemStore yang merupakan struktur data in-memory untuk memberikan akses cepat terhadap data yang baru ditulis (Cloudera, 2012).

Ketika MemStore mencapai batas kapasitas maksimum, sistem melakukan proses flush yang mengkonversi data menjadi HFile, yaitu format file penyimpanan data terstruktur yang dioptimalkan untuk operasi read dan write dalam HBase (Cloudera, 2021).

HFile yang dihasilkan dari proses flush kemudian disimpan ke dalam HDFS (Hadoop Distributed File System). Setiap HFile akan disalin sebanyak tiga kali ke DataNode yang berbeda. Proses penyalinan ini bertujuan untuk memastikan bahwa data tetap tersedia dan tahan terhadap kegagalan, sehingga apabila terjadi kerusakan atau kehilangan data pada salah satu node penyimpanan, salinan data masih dapat diakses dari node lainnya.

Operasi read dalam HBase dimulai ketika client mengakses data melalui HBase Client API dengan mengirimkan query request ke sistem. HBase Client berkonsultasi dengan ZooKeeper untuk mendapatkan informasi lokasi region yang tepat sesuai dengan row key yang diminta, memastikan request diarahkan ke RegionServer yang bertanggung jawab.

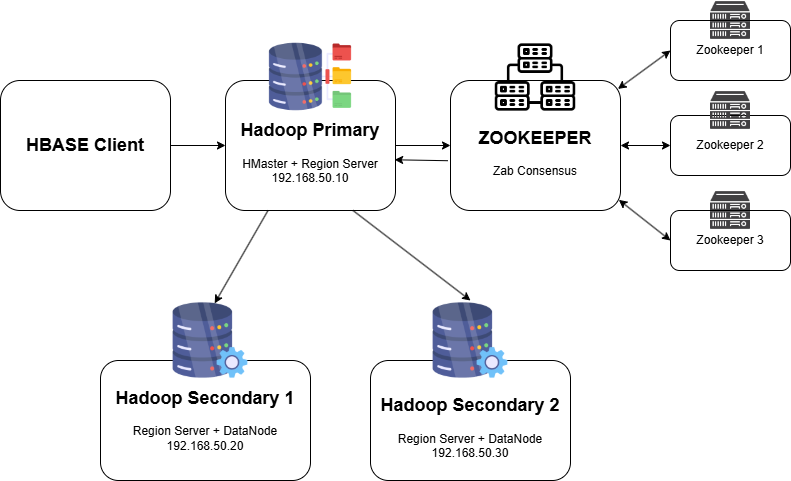
RegionServer melakukan scan operation pada dua komponen utama yaitu MemStore untuk data terbaru yang belum di-flush dan HFile untuk data yang telah tersimpan secara permanen (Cloudera, 2012)

Proses akses HFile dilakukan dari HDFS DataNode dengan menerapkan optimasi lokalitas data yang meminimalkan network latency dengan mengakses data dari node terdekat. Hasil query dari kedua sumber data tersebut kemudian digabungkan, diurutkan berdasarkan timestamp, dan dikembalikan ke client sebagai response yang lengkap dan konsisten.

Dalam implementasi cluster, pembagian peran node adalah sebagai berikut:

* hadoop-primary (192.168.50.10) : berfungsi sebagai HBase Master dan RegionServer untuk mengkoordinasikan operasi cluster dan mengelola metadata
* hadoop-secondary-1 (192.168.50.20) : berperan sebagai RegionServer untuk distribusi data dan pembagian beban
* hadoop-secondary-2 (192.168.50.30) : berperan sebagai RegionServer untuk redundansi dan pemrosesan paralel

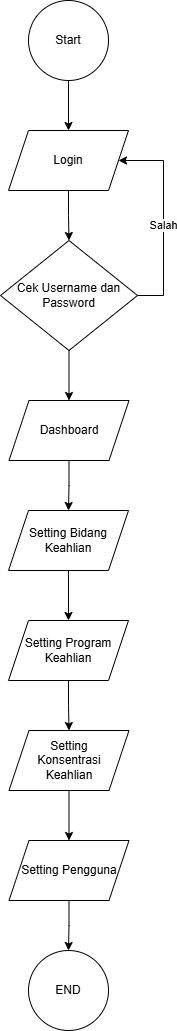
Arsitektur tabel dan database HBase ini memungkinkan skalabilitas yang tinggi, fleksibilitas format data, dan akses cepat melalui indeks terdistribusi. Pembagian peran antara Master dan RegionServer memungkinkan operasi terdistribusi yang efisien dengan ketahanan terhadap kegagalan melalui salinan data ke beberapa node dalam cluster. Visualisasi perancangan NoSQL HBase dapat dilihat pada Gambar 4.3.



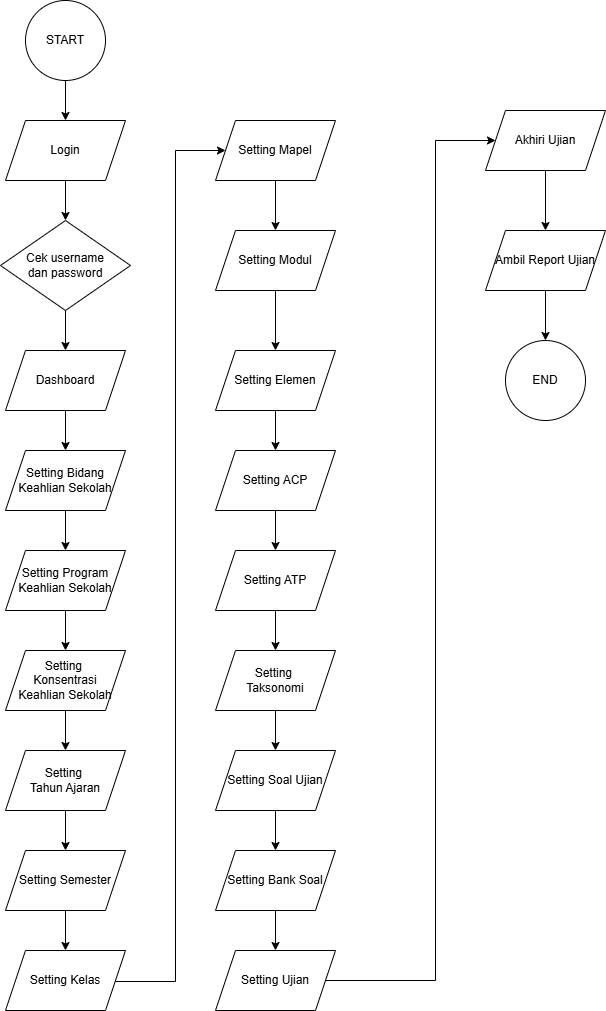
Gambar 4. Struktur HBase

1. **Perancangan Alur Aplikasi**

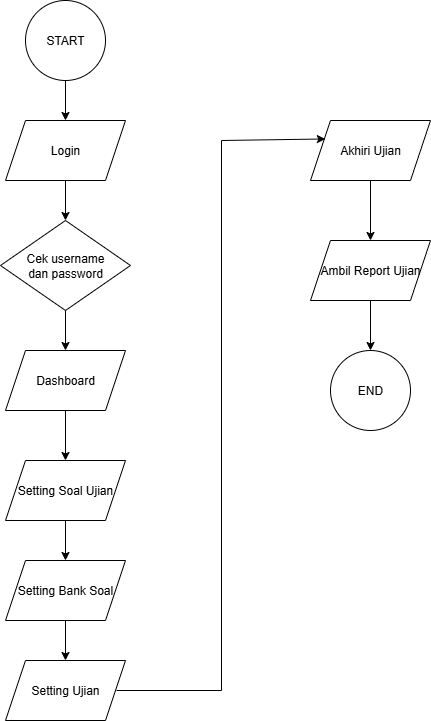
Flowchart merupakan sebuah diagram visual yang menggambarkan alur kerja suatu sistem. Flowchart bertujuan untuk memberikan pemahaman yang lebih mudah mengenai proses berjalannya sistem tersebut. Hal ini dapat dilihat dari gambar 4.4 yang memperlihatkan alur sistem khusus untuk bagian administrator. Sementara itu, gambar 4.5 menampilkan alur kerja yang berkaitan dengan Operator Sekolah.



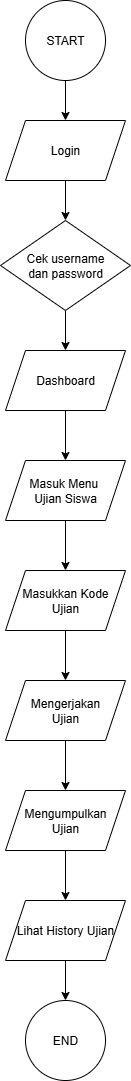
Gambar 4. Alur Sistem Administrator



Gambar 4. Alur Sistem Operator Sekolah



Gambar 4. Alur Sistem Guru



Gambar 4. Alur Sistem Siswa

# BAB V IMPLEMENTASI SISTEM

## Implementasi Sistem

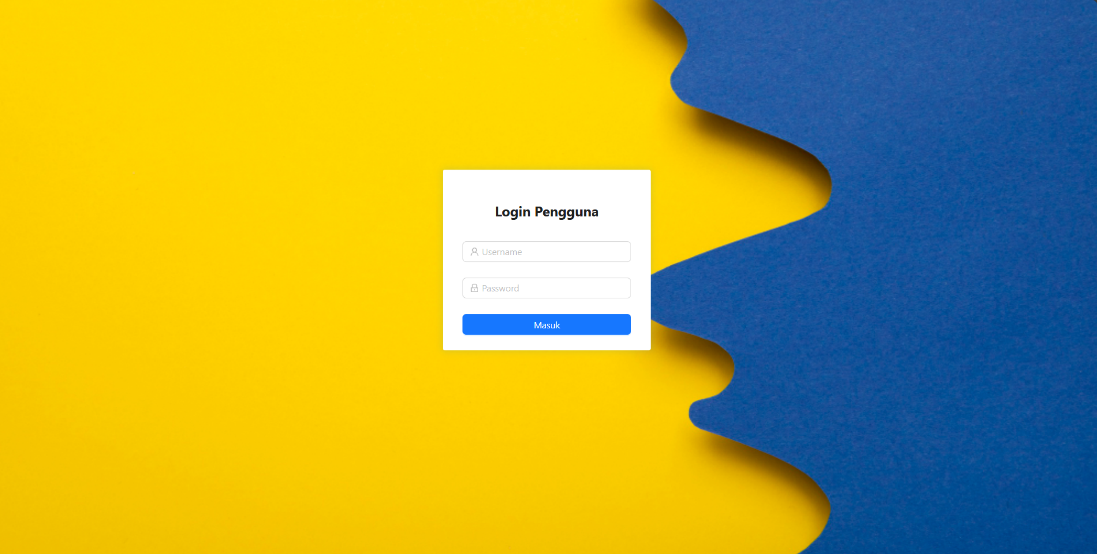
Pada tahap ini akan dilakukan tahap implementasi sistem, yaitu proses pembuatan sistem atau perangkat lunak dari tahap perancangan ke tahap coding dengan menggunakan bahasa pemrograman yang akan menghasilkan sistem atau perangkat lunak yang telah dirancang sebelumnya. Adapun hasil dari implementasi Sistem Bank Soal Bank Soal Terdistribusi Multi Sekolah (Studi Kasus SMK Negeri Rowokangkung) sebagai berikut:

### Implementasi Antarmuka Administrator

Pada tahap implementasi Antarmuka Administrator, fokusnya adalah pada penginputan Master Data. Langkah-langkahnya meliputi identifikasi entitas data yang menjadi bagian dari Master Data, pengaturan struktur dan atribut terkait.

1. Login

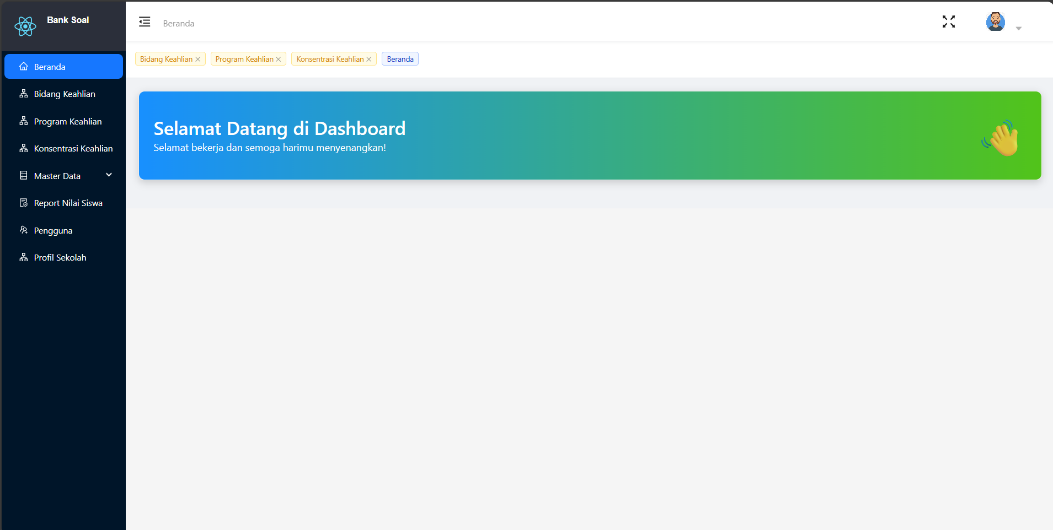
Sebelum masuk ke sistem pengguna harus melakukan proses login terlebih dahulu. Antarmuka dari proses login bisa dilihat pada gambar berikut:



Gambar 5. Halaman Login

1. Halaman Beranda

Pada halaman beranda pengguna masuk ke halaman beranda dengan sambutan Selamat Datang. Hasil antarmuka pada halaman beranda bisa dilihat pada gambar 5.2 berikut:

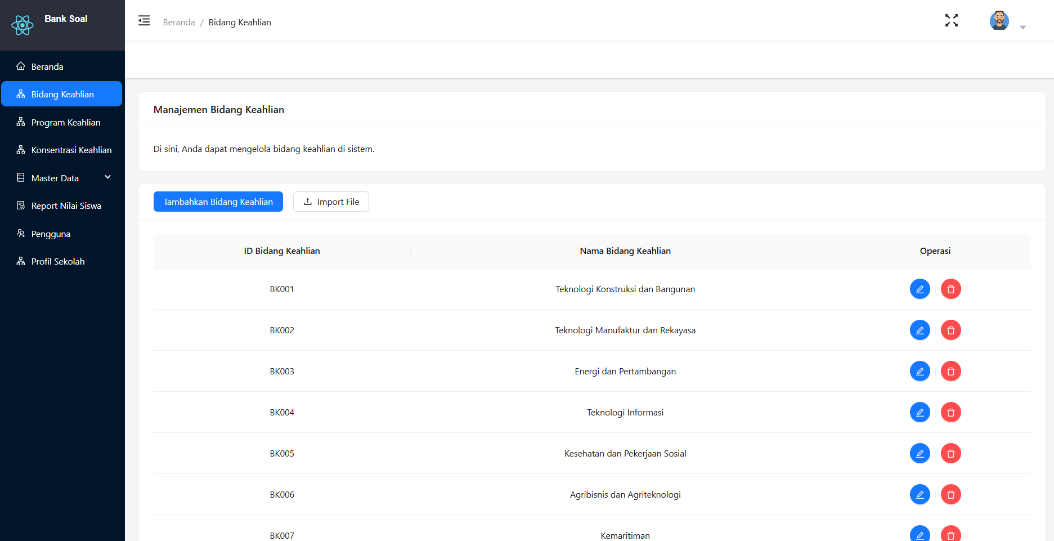


Gambar 5. Halaman Beranda

1. Bidang Keahlian

Pada halaman bidang keahlian ditampilkan data bidang keahlian yang berisi id bidang keahlian dan nama bidang keahlian, data bidang keahlian didapatkan berdasarkan KEPUTUSAN MENTERI PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI REPUBLIK INDONESIA NOMOR 244/M/2024 TENTANG SPEKTRUM KEAHLIAN DAN KONVERSI SPEKTRUM KEAHLIAN SEKOLAH MENENGAH KEJURUAN/MADRASAH ALIYAH KEJURUAN PADA KURIKULUM MERDEKA (Triyantoro, 2024).

Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data bidang keahlian. Hasil antarmuka pada halaman bidang keahlian bisa dilihat pada gambar 5.3 berikut:

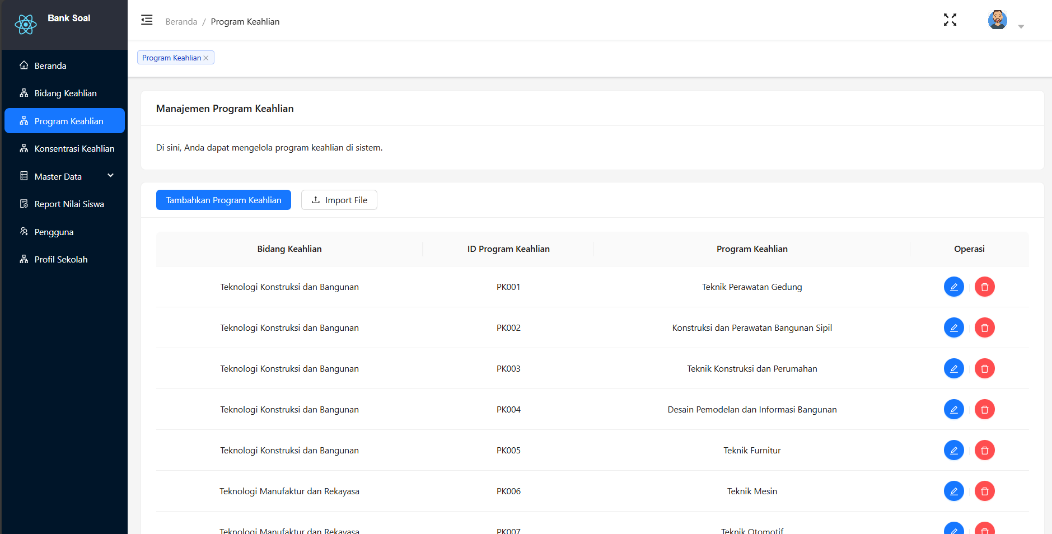


Gambar 5. Bidang Keahlian

1. Program Keahlian

Pada halaman program keahlian ditampilkan data program keahlian yang berisi id program keahlian dan nama program keahlian, data program keahlian didapatkan berdasarkan KEPUTUSAN MENTERI PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI REPUBLIK INDONESIA NOMOR 244/M/2024 TENTANG SPEKTRUM KEAHLIAN DAN KONVERSI SPEKTRUM KEAHLIAN SEKOLAH MENENGAH KEJURUAN/MADRASAH ALIYAH KEJURUAN PADA KURIKULUM MERDEKA (Triyantoro, 2024).

Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data program keahlian. Hasil antarmuka pada halaman program keahlian bisa dilihat pada gambar 5.4 berikut:

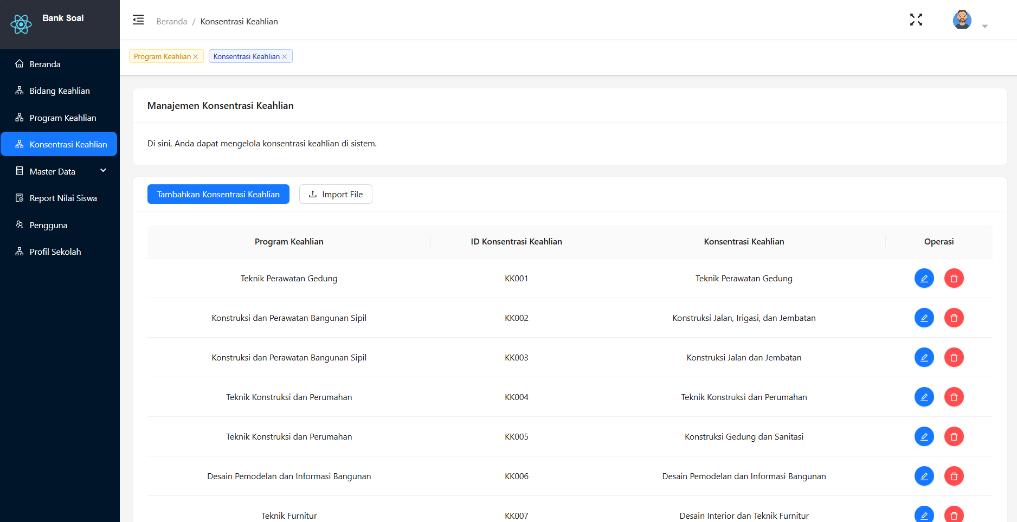


Gambar 5. Program Keahlian

1. Konsentrasi Keahlian

Pada halaman konsentrasi keahlian ditampilkan data konsentrasi keahlian yang berisi id konsentrasi keahlian dan nama konsentrasi keahlian, data konsentrasi keahlian didapatkan berdasarkan KEPUTUSAN MENTERI PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI REPUBLIK INDONESIA NOMOR 244/M/2024 TENTANG SPEKTRUM KEAHLIAN DAN KONVERSI SPEKTRUM KEAHLIAN SEKOLAH MENENGAH KEJURUAN/MADRASAH ALIYAH KEJURUAN PADA KURIKULUM MERDEKA (Triyantoro, 2024).

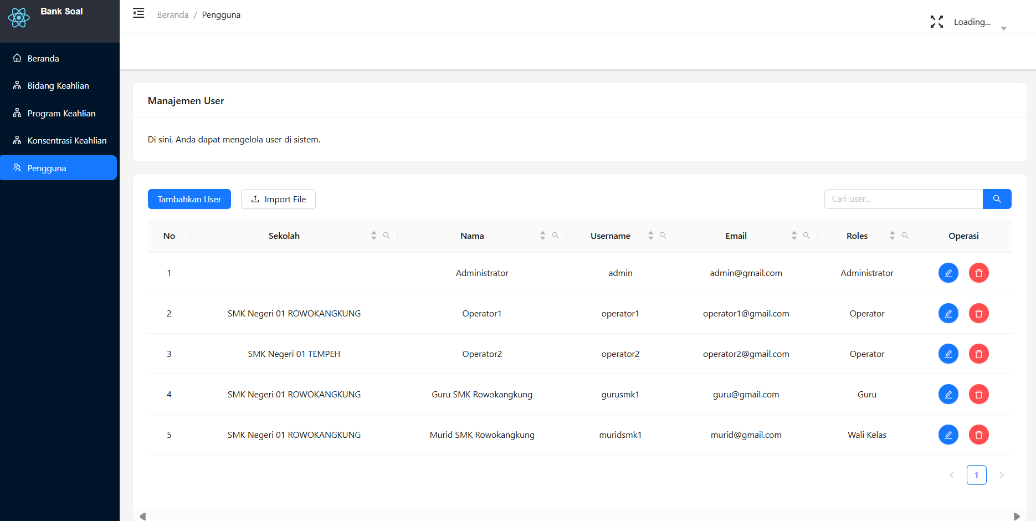
Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data konsentrasi keahlian. Hasil antarmuka pada halaman konsentrasi keahlian bisa dilihat pada gambar 5.6 berikut



Gambar 5. Konsentrasi Keahlian

1. Pengguna

Pada halaman pengguna ditampilkan data pengguna yang berisi sekolah, nama, username, email, dan roles. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data pengguna. Hasil antarmuka pada halaman pengguna bisa dilihat pada gambar 5.6 berikut



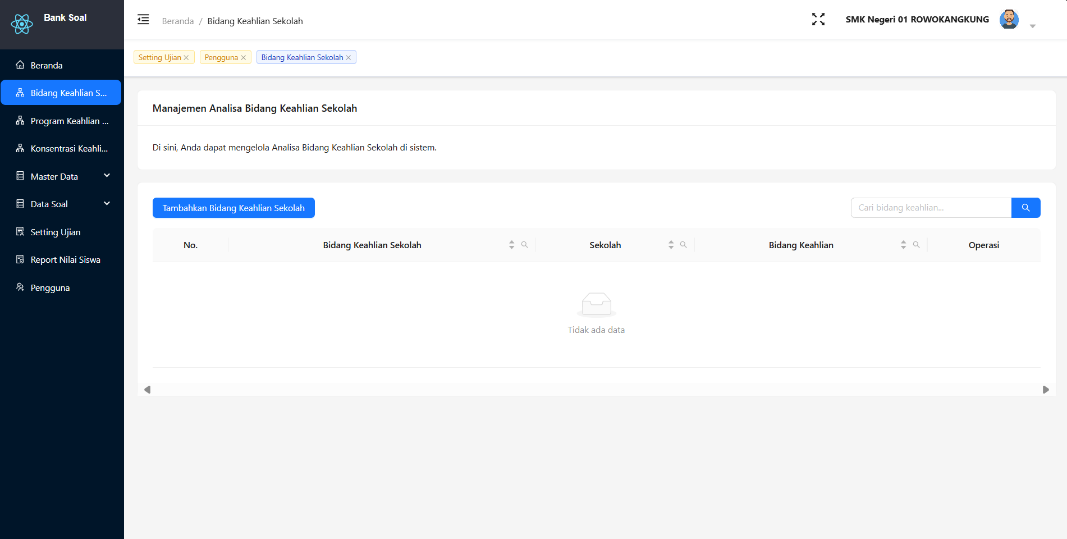
Gambar 5. Halaman Pengguna

### Implementasi Antarmuka Operator

Pada tahap implementasi Antarmuka Operator, fokusnya adalah penginputan data inti dari ujian. Langkah-langkahnya meliputi identifikasi entitas data yang menjadi bagian dari ujian, pengaturan struktur dan atribut terkait.

1. Halaman Bidang Keahlian Sekolah

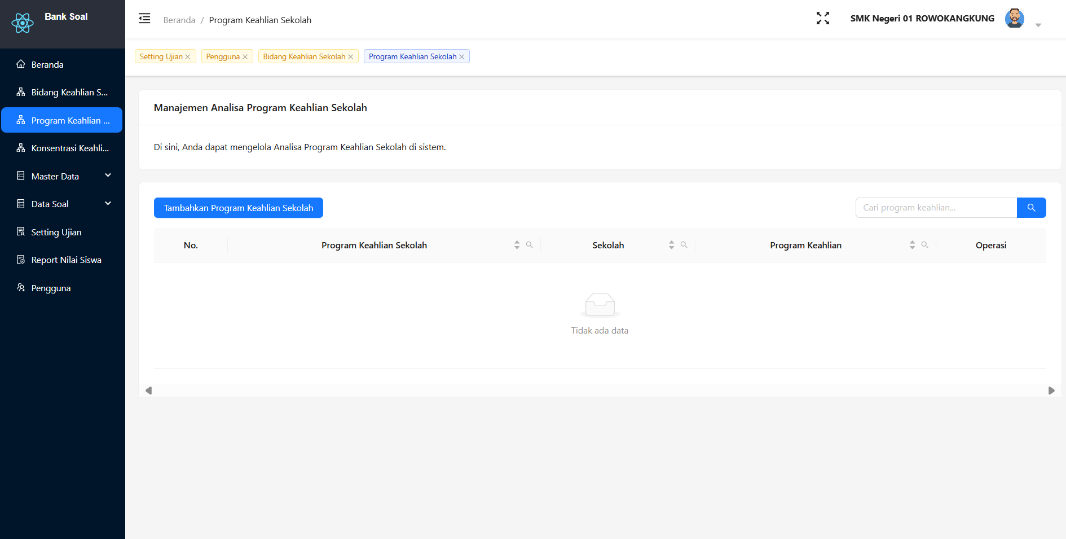
Pada halaman bidang keahlian sekolah, akan menampilkan data bidang keahlian sekolah yang berisi sekolah, nama bidang keahlian sekolah, bidang keahlian. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data bidang keahlian sekolah. Hasil antarmuka pada halaman bidang keahlian sekolah bisa dilihat pada gambar 5.7 berikut



Gambar 5. Bidang Keahlian Sekolah

1. Halaman Program Keahlian Sekolah

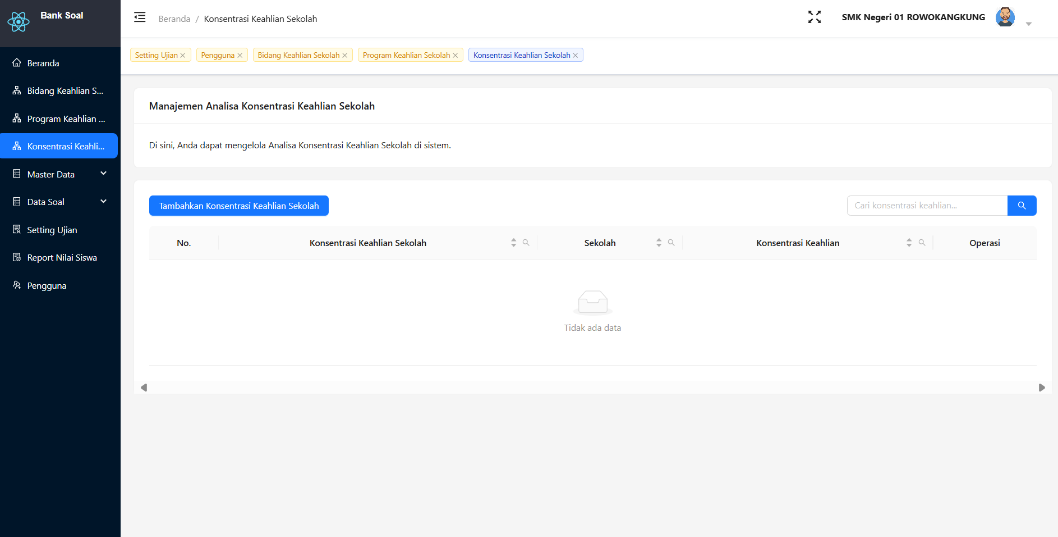
Pada halaman program keahlian sekolah, akan menampilkan data program keahlian sekolah yang berisi sekolah, nama program keahlian sekolah, program keahlian. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data program keahlian sekolah. Hasil antarmuka pada halaman program keahlian sekolah bisa dilihat pada gambar 5.8 berikut



Gambar 5. Program Keahlian Sekolah

1. Halaman Konsentrasi Keahlian Sekolah

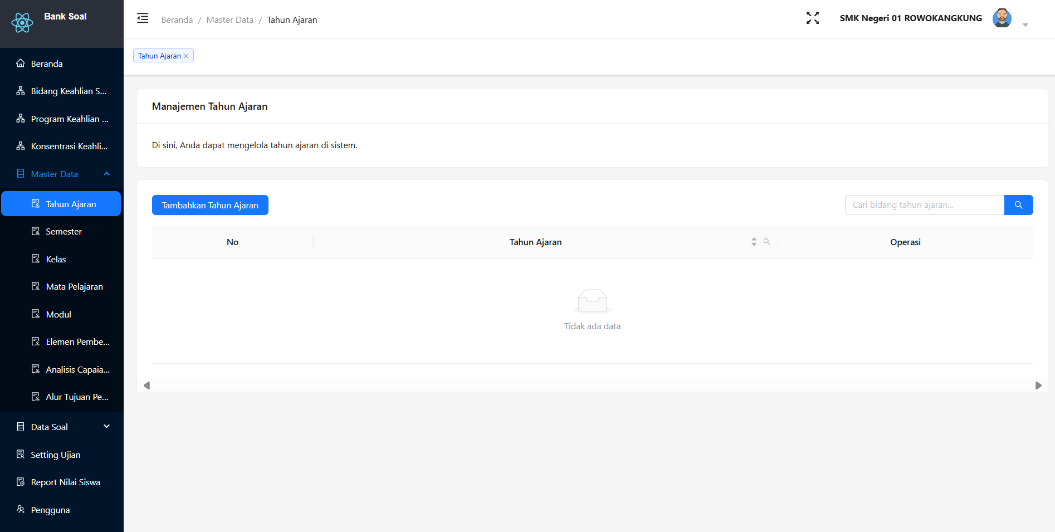
Pada halaman konsentrasi keahlian sekolah, akan menampilkan data konsentrasi keahlian sekolah yang berisi sekolah, nama konsentrasi keahlian sekolah, konsentrasi keahlian. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data konsentrasi keahlian sekolah. Hasil antarmuka pada halaman konsentrasi keahlian sekolah bisa dilihat pada gambar 5.9 berikut



Gambar 5. Konsentrasi Keahlian Sekolah

1. Halaman Master Data Tahun Ajaran

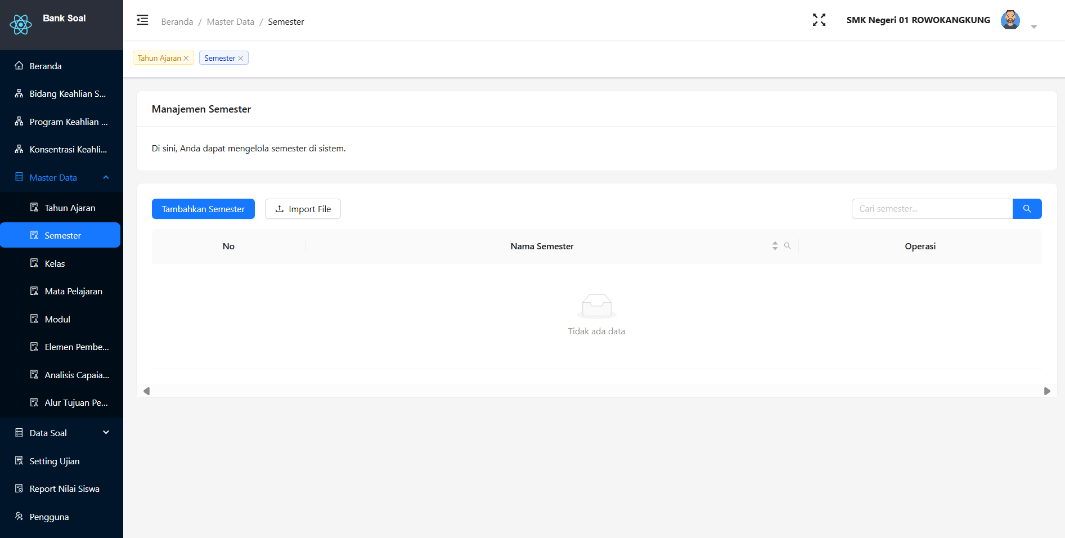
Pada halaman tahun ajaran, akan menampilkan data tahun ajaran yang berisi tahun ajaran. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data tahun ajaran. Hasil antarmuka pada halaman tahun ajaran bisa dilihat pada gambar 5.10 berikut



Gambar 5. Tahun Ajaran

1. Halaman Master Data Semester

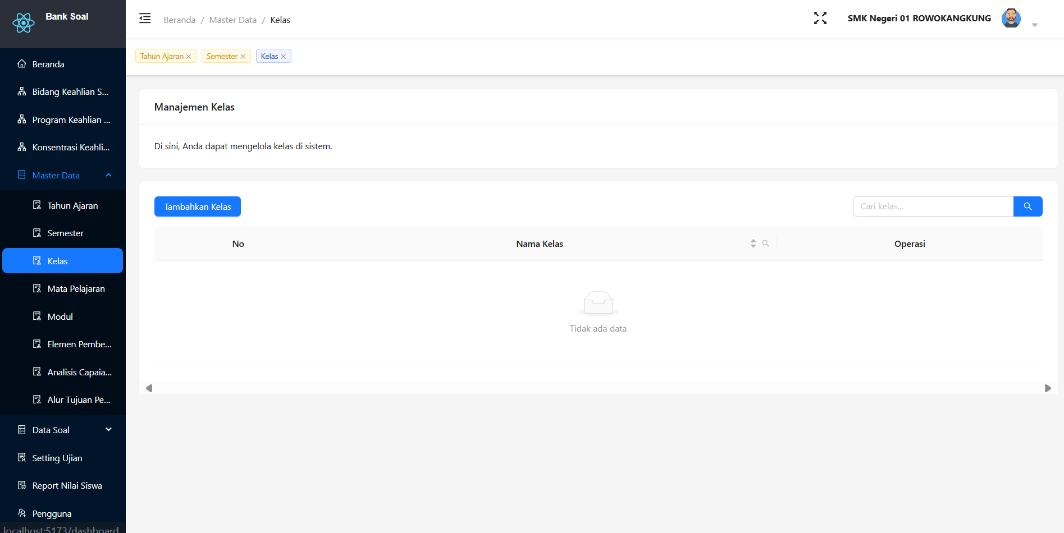
Pada halaman semester, akan menampilkan data semester yang berisi nama semester. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data semester. Hasil antarmuka pada halaman semester bisa dilihat pada gambar 5.11 berikut



Gambar 5. Semester

1. Halaman Master Data Kelas

Pada halaman kelas, akan menampilkan data kelas yang berisi nama kelas. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data kelas. Hasil antarmuka pada halaman kelas bisa dilihat pada gambar 5.12 berikut



Gambar 5. Kelas

1. Halaman Master Data Mata Pelajaran

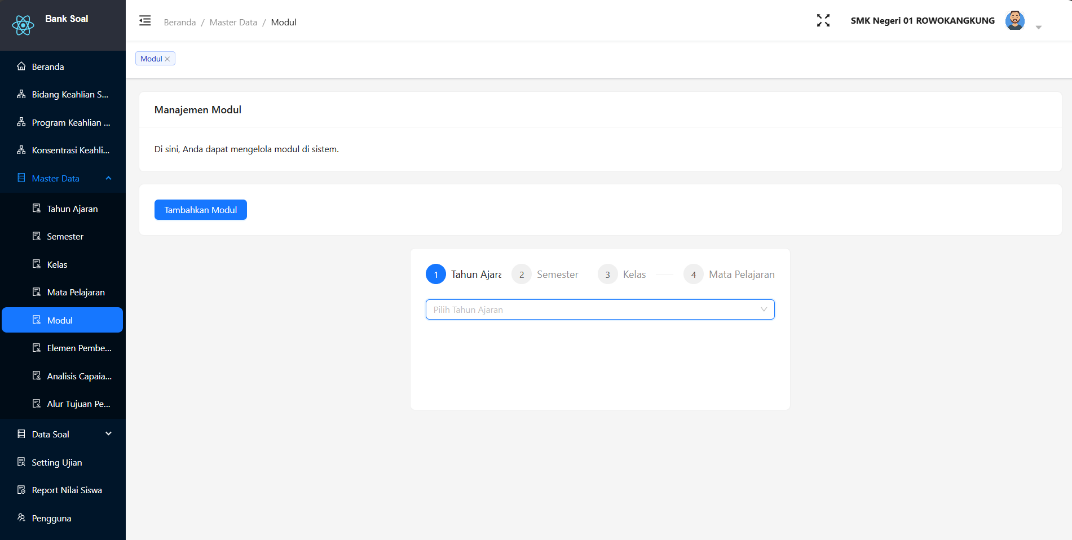
Pada halaman mata pelajaran, akan menampilkan data mata Pelajaran yang berisi tahun ajaran, kelas, dan nama mata pelajaran. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data mata pelajaran. Hasil antarmuka pada halaman mata pelajaran bisa dilihat pada gambar 5.13 berikut



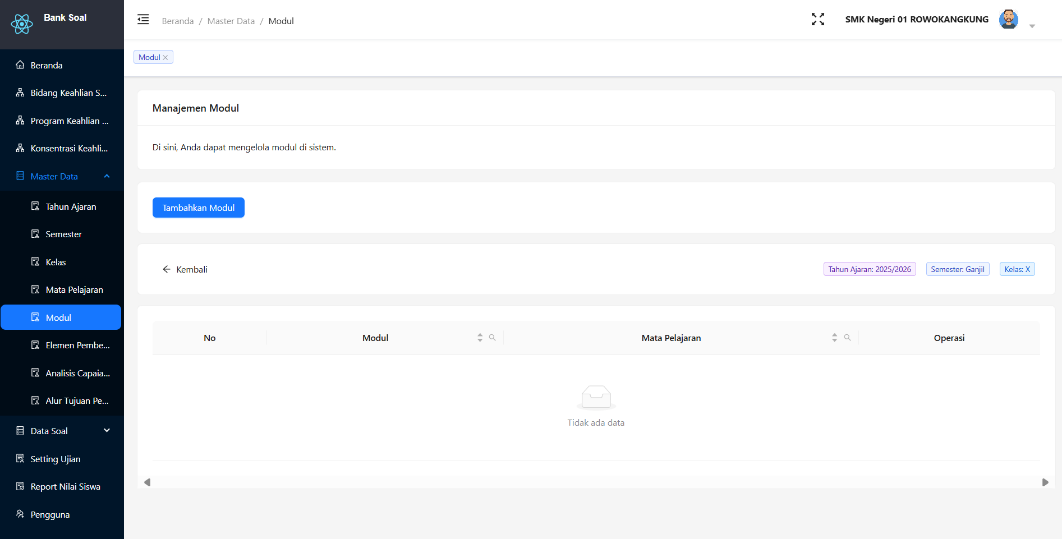
Gambar 5. Mata Pelajaran

1. Halaman Master Data Modul

Pada halaman modul, akan menampilkan data modul yang berisi mata pelajaran dan modul. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data modul. Hasil antarmuka pada halaman modul sebelum di filter bisa dilihat pada gambar 5.14 dan setelah di filter bisa dilihat pada gambar 5.15 berikut



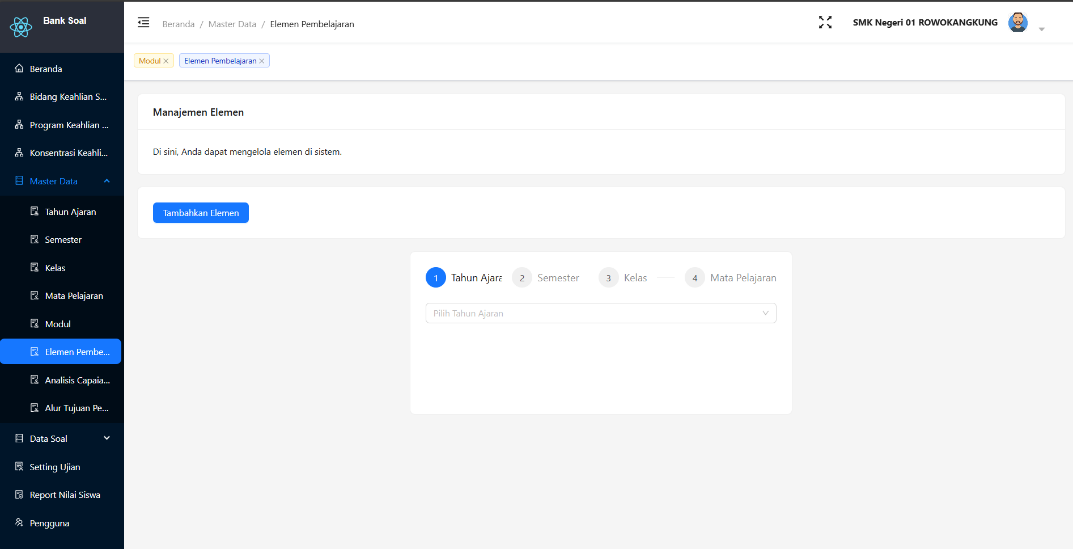
Gambar 5. Modul



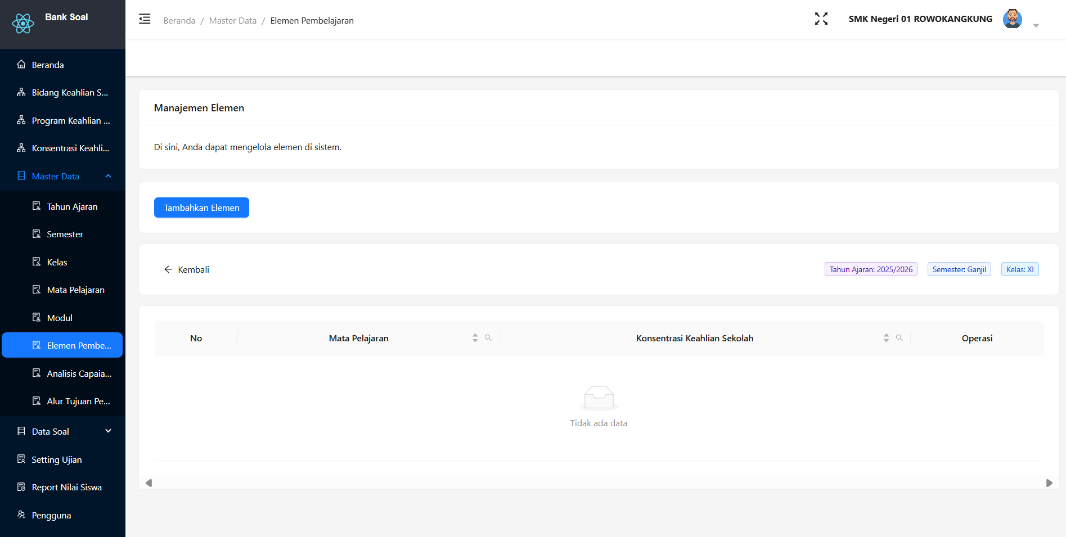
Gambar 5. Modul Filter

1. Halaman Master Data Elemen

Pada halaman elemen, akan menampilkan data elemen yang berisi tahun ajaran, smester, kelas, nama mata pelajaran, dan nama elemen. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data elemen. Hasil antarmuka pada halaman elemen sebelum di filter bisa dilihat pada gambar 5.16 dan setelah di filter bisa dilihat pada gambar 5.17 berikut



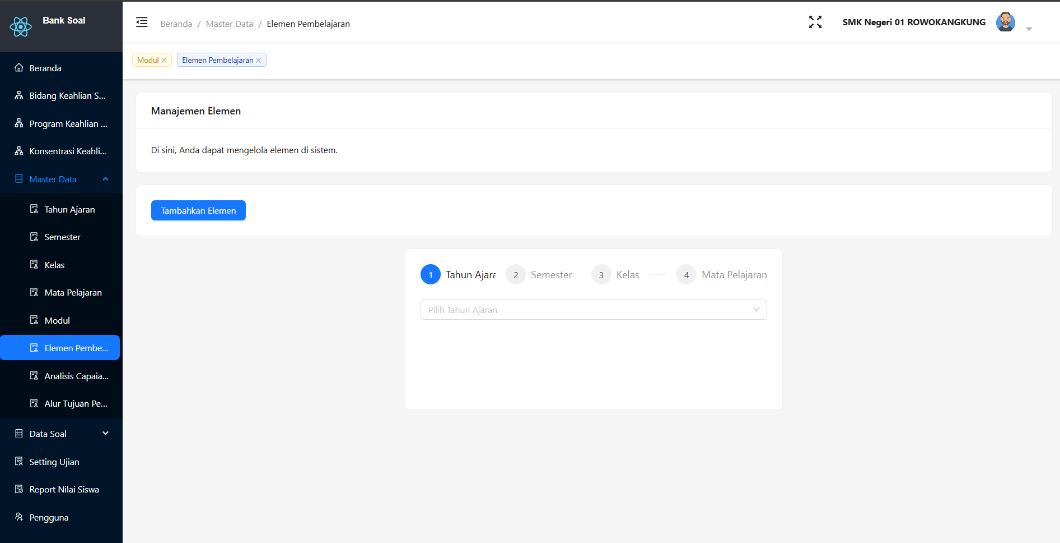
Gambar 5. Elemen



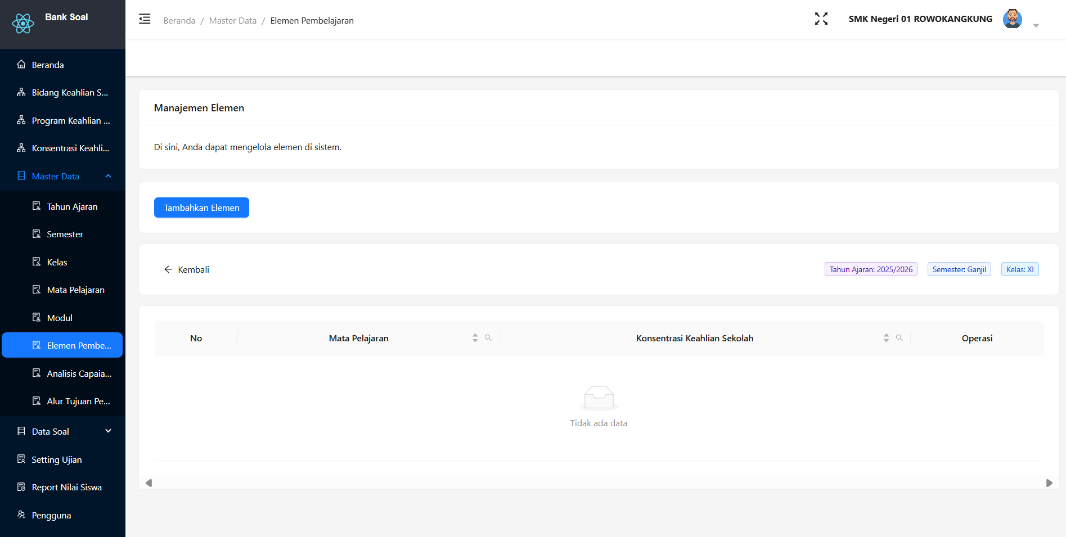
Gambar 5. Elemen Filter

1. Halaman Master Data Analisis Capaian Pembelajaran (ACP)

Pada halaman acp, akan menampilkan data acp yang berisi tahun ajaran, smester, kelas, nama mata pelajaran, elemen, nama acp. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data acp. Hasil antarmuka pada halaman acp sebelum di filter bisa dilihat pada gambar 5.18 dan setelah di filter bisa dilihat pada gambar 5.19 berikut



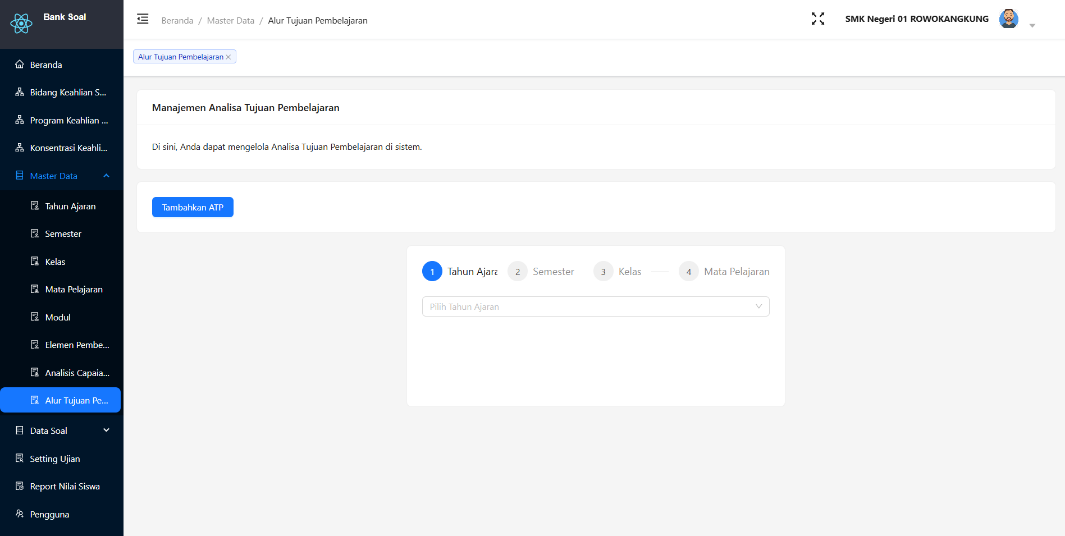
Gambar 5. ACP



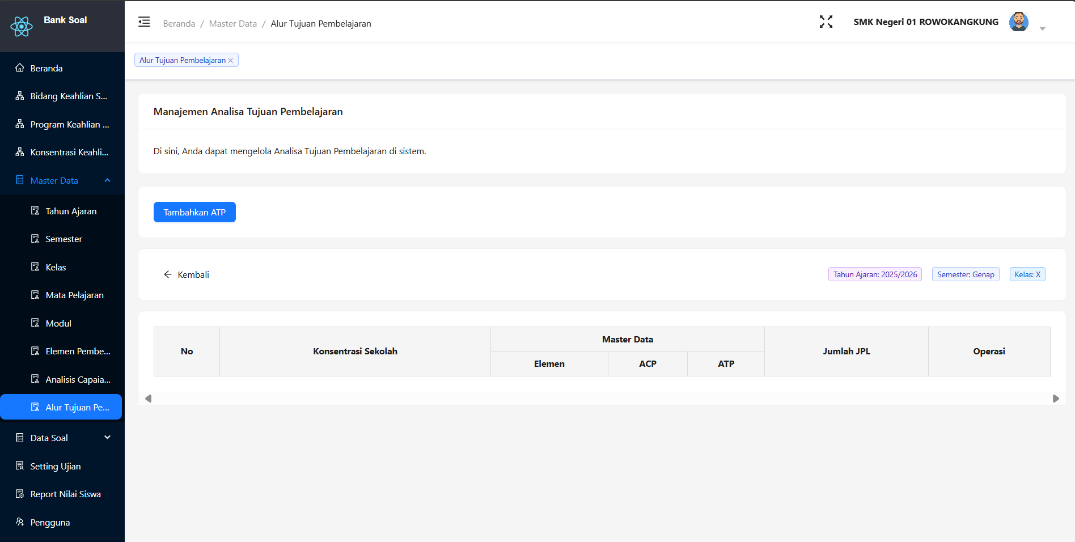
Gambar 5. ACP filter

1. Halaman Master Data Analisis Tujuan Pembelajaran (ATP)

Pada halaman atp, akan menampilkan data atp yang berisi tahun ajaran, smester, kelas, nama mata pelajaran, elemen, acp, nama atp, jumlah jpl. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data atp. Hasil antarmuka pada halaman atp sebelum di filter bisa dilihat pada gambar 5.20 dan setelah di filter bisa dilihat pada gambar 5.21 berikut



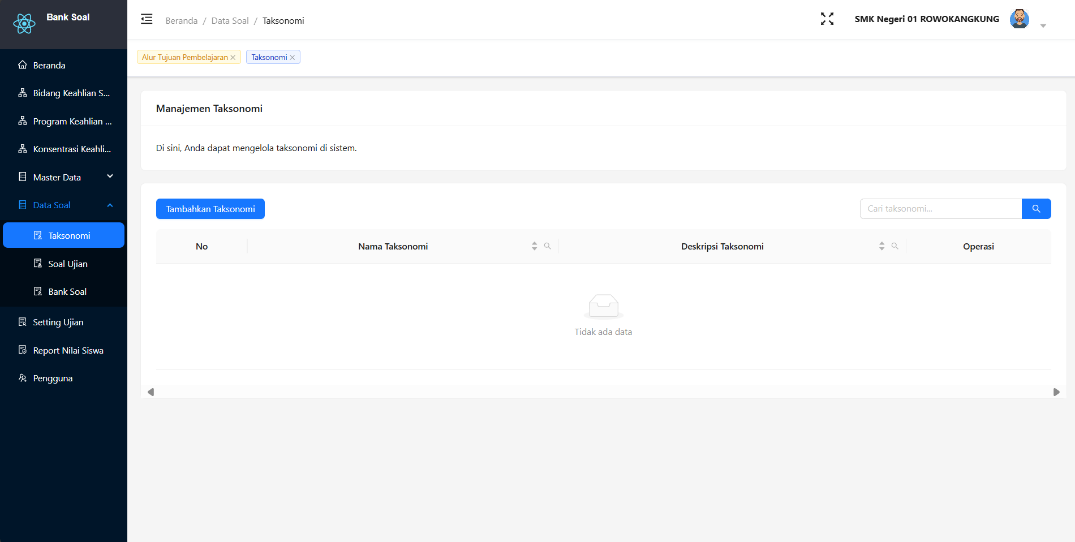
Gambar 5. ATP



Gambar 5. ATP Filter

1. Halaman Data Soal Taksonomi

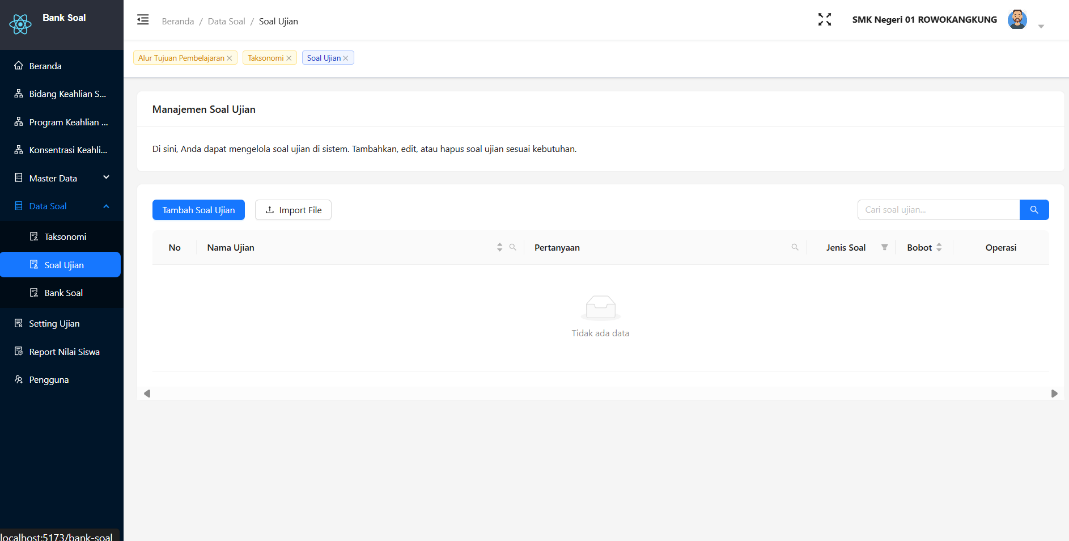
Pada halaman taksonomi, akan menampilkan data taksonomi yang berisi tingkatan taksonomi dan deskripsi taksonomi. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data taksonomi. Hasil antarmuka pada halaman taksonomi bisa dilihat pada gambar 5.22 berikut



Gambar 5. Taksonomi

1. Halaman Data Soal Ujian

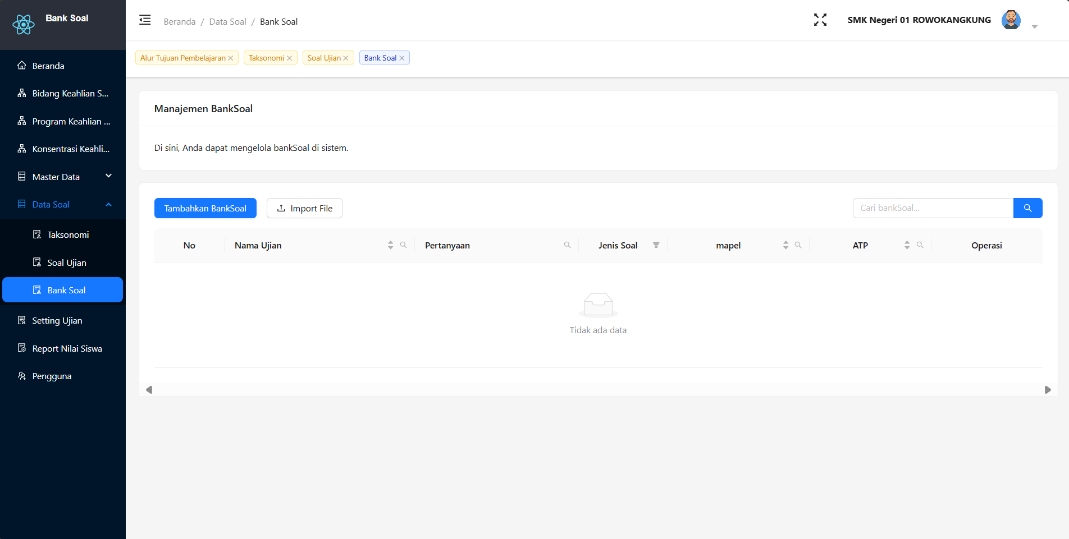
Pada halaman soal ujian, akan menampilkan data soal ujian yang berisi nama ujian, petanyaan, jenis soal, bobot. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data soal ujian. Selain itu, terdapat modal yang akan menampilkan soal beserta pilihan dan jawaban yang benar. Hasil antarmuka pada halaman soal ujian bisa dilihat pada gambar 5.23 berikut



Gambar 5. Soal Ujian

1. Halaman Data Soal Bank Soal

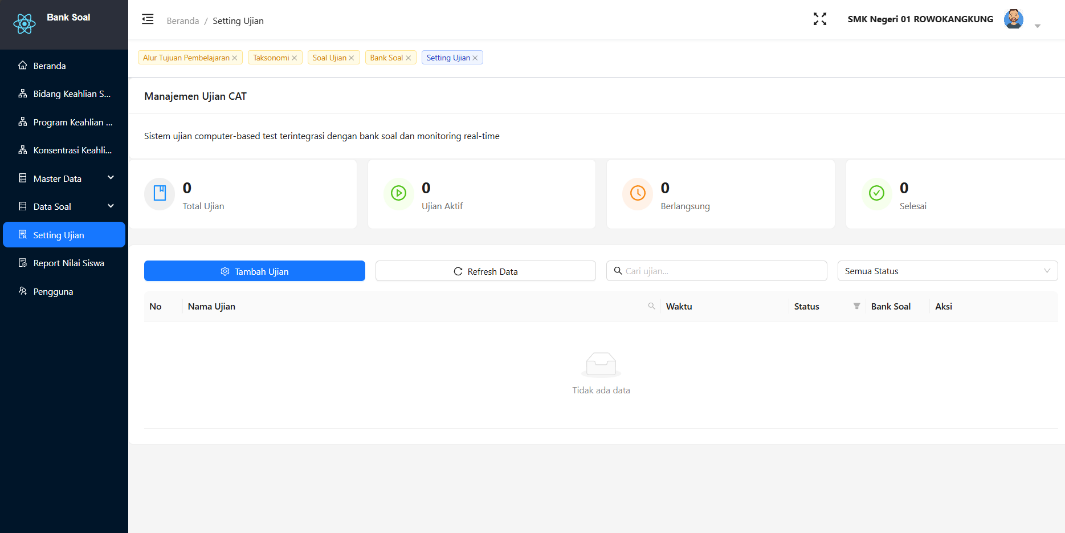
Pada halaman bank soal, akan menampilkan data bank soal yang berisi nama ujian, petanyaan, jenis soal, maple, atp. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data bank soal. Selain itu, terdapat modal yang akan menampilkan atribut lengkap data bank soal. Hasil antarmuka pada halaman bank soal bisa dilihat pada gambar 5.24 berikut



Gambar 5. Bank Soal

1. Halaman Data Setting Ujian

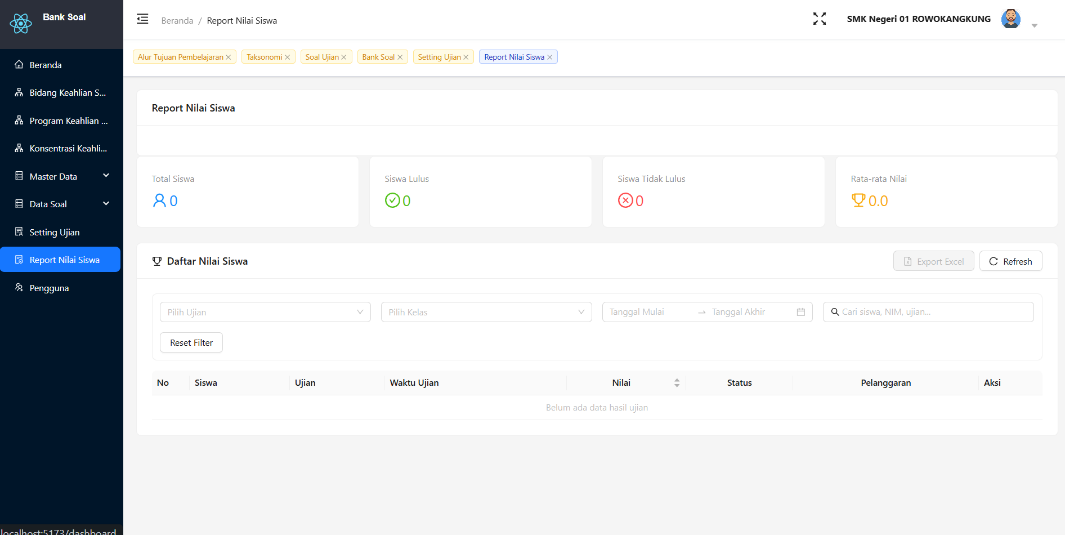
Pada halaman setting ujian, akan menampilkan data setting ujian yang berisi nama ujian, waktu, status, bank soal. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data setting ujian. Hasil antarmuka pada halaman setting ujian bisa dilihat pada gambar 5.25 berikut



Gambar 5. Setting Ujian

1. Halaman Report Nilai Siswa

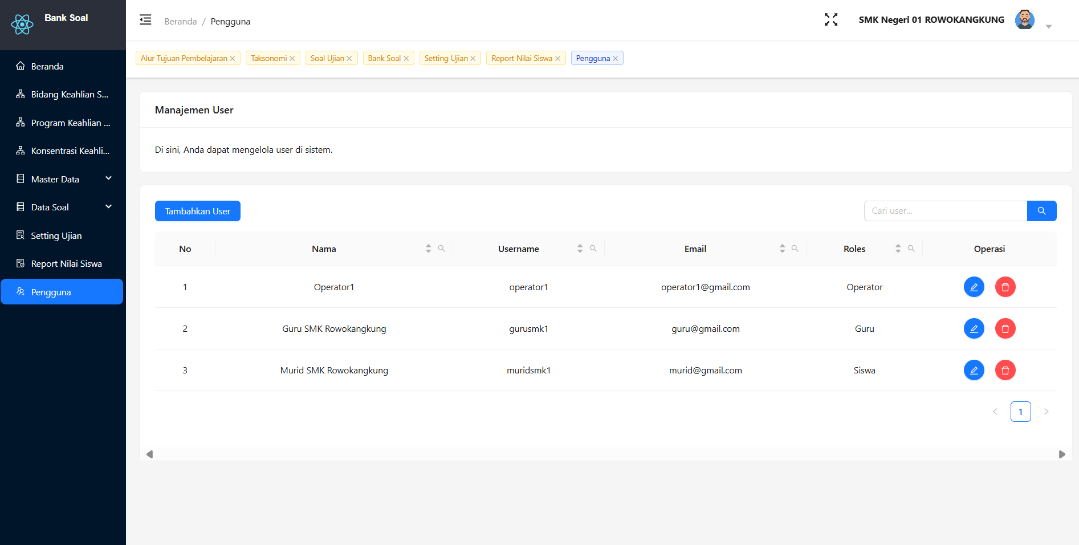
Pada halaman report nilai siswa, akan menampilkan data report nilai siswa yang berisi siswa, ujian, waktu ujian, nilai, status, pelanggaran. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data report nilai siswa. Hasil antarmuka pada halaman report nilai siswa bisa dilihat pada gambar 5.26 berikut



Gambar 5. Report Nilai Siswa

1. Halaman Pengguna (Sekolah)

Pada halaman pengguna ditampilkan data pengguna yang berisi nama, username, email, dan roles. Pada halaman ini juga terdapat tombol untuk menambah, mengedit dan menghapus data pengguna. Hasil antarmuka pada halaman pengguna bisa dilihat pada gambar 5.27 berikut



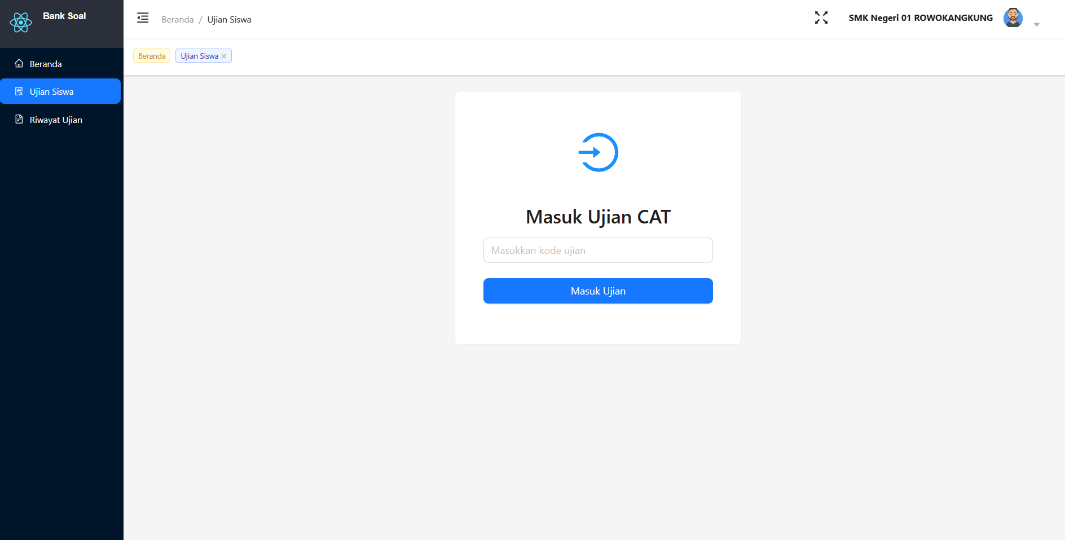
Gambar 5. Pengguna Sekolah

### Implementasi Antarmuka Siswa

Pada tahap implementasi Antarmuka Siswa, fokusnya adalah pengerjaaan ujian dan history pengerjaannya. Langkah-langkahnya meliputi identifikasi entitas data yang menjadi bagian dari ujian, pengaturan struktur dan atribut terkait.

1. Halaman Ujian Siswa

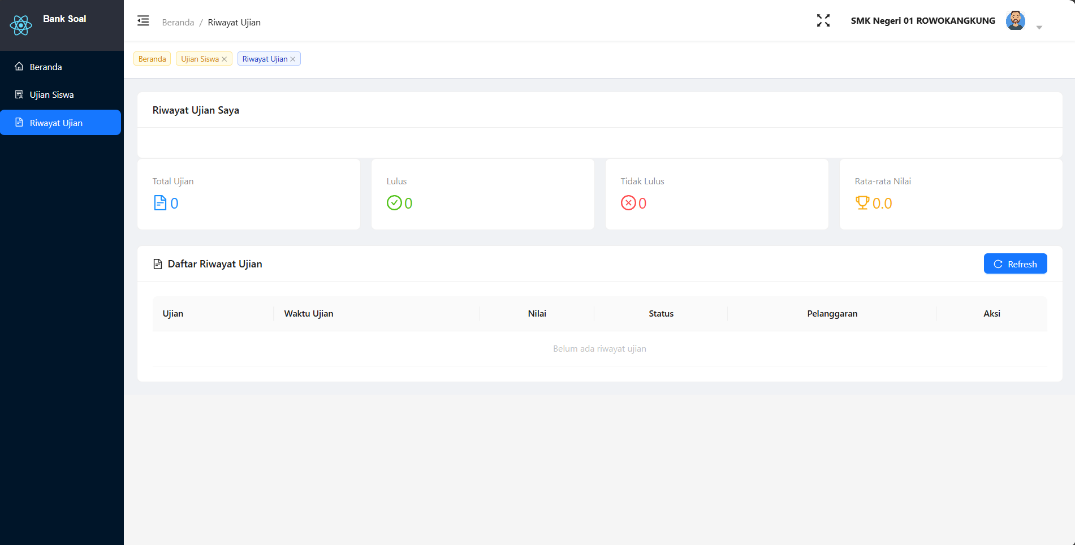
Pada halaman ujian, akan menampilkan ujian yang memasukkan kode ujian, Ketika kode ujian dimasukkan oleh siswa ujian akan dimulai. Hasil antarmuka pada halaman ujian siswa bisa dilihat pada gambar 5.28 berikut



Gambar 5. Ujian Siswa

1. Halaman Riwayat Ujian

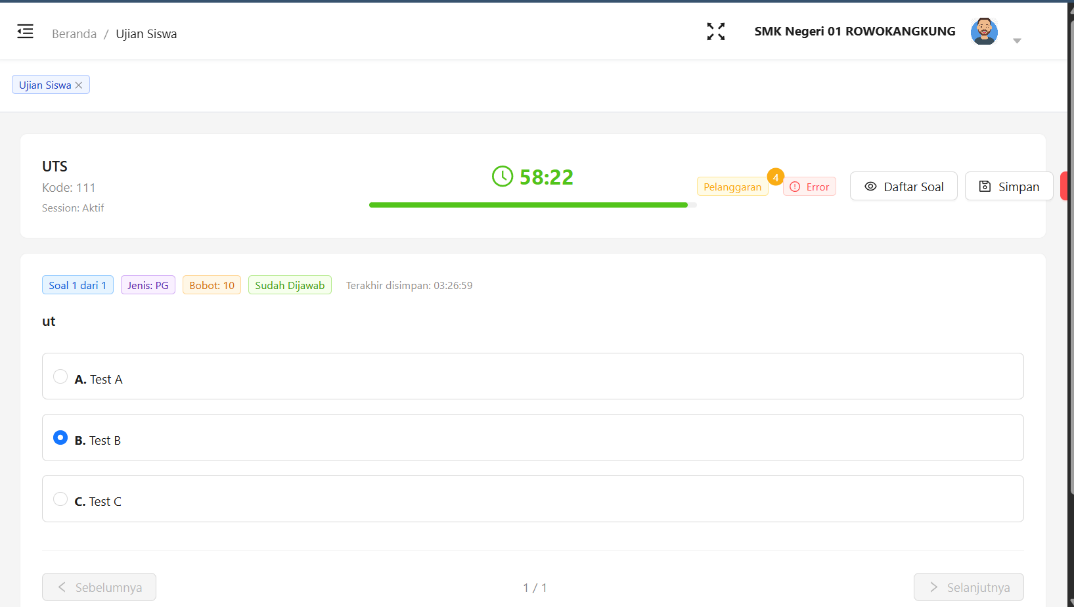
Pada halaman riwayat ujian, akan menampilkan data riwayat ujian yang berisi ujian, waktu ujian, nilai, status, pelanggaran. Hasil antarmuka pada halaman riwayat ujian bisa dilihat pada gambar 5.29 berikut



Gambar 5. Riwayat Ujian

1. Halaman Pelaksanaan Ujian Siswa

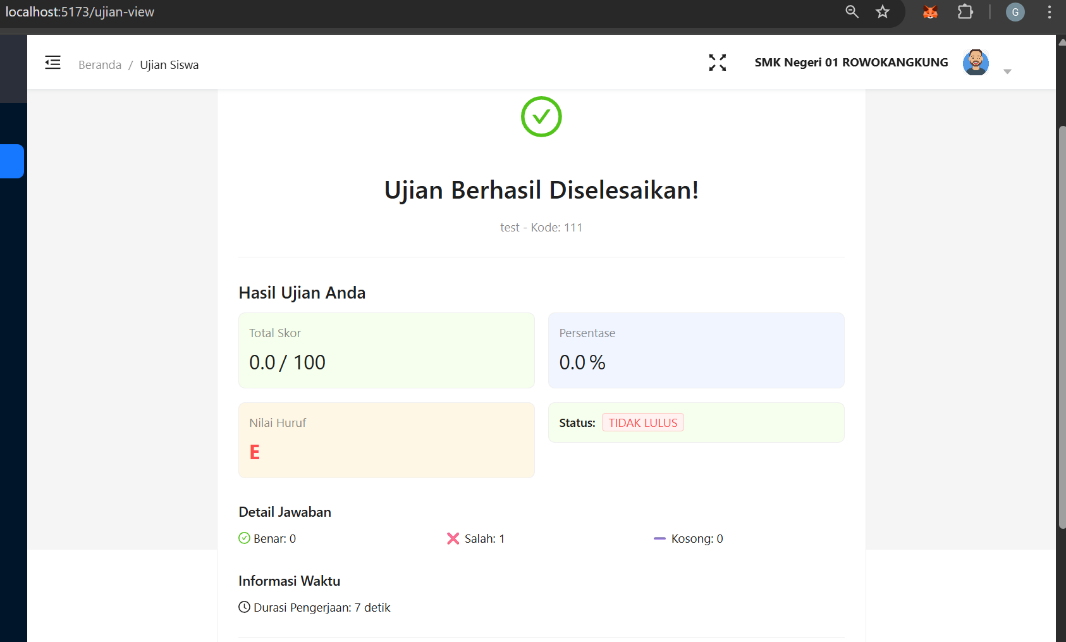
Pada halaman pelaksanaan, akan menampilkan data soal ujian, pilihan jawaban, waktu ujian, daftar soal. Hasil antarmuka pada halaman pelaksanaan ujian bisa dilihat pada gambar 5.30 berikut



Gambar 5. Pelaksanaan Ujian

1. Halaman Penyelesaian Ujian

Pada halaman penyelesaian ujian, akan menampilkan data nilai, presentase, nilai huruf, status. Hasil antarmuka pada halaman penyelesaian ujian bisa dilihat pada gambar 5.31 berikut



Gambar 5. Penyelesaian Ujian

# BAB VI HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan hasil dari pengujian yang telah dilakukan dalam Perancangan sistem ini. Mendeskripsikan bagaimana pengujian dilakukan bersama secara online di website dengan waktu fleksibel akan tetapi pelanggaran siswa tercatat

* 1. **Pengujian UAT Sistem**

Pada tahap ini dijelaskan hasil uji coba sistem dengan user yang berbeda. Pengujian UAT secara offlinemaupun online.

1. **Testing sistem kepada user (Guru SMK Negeri Rowokangkung)**

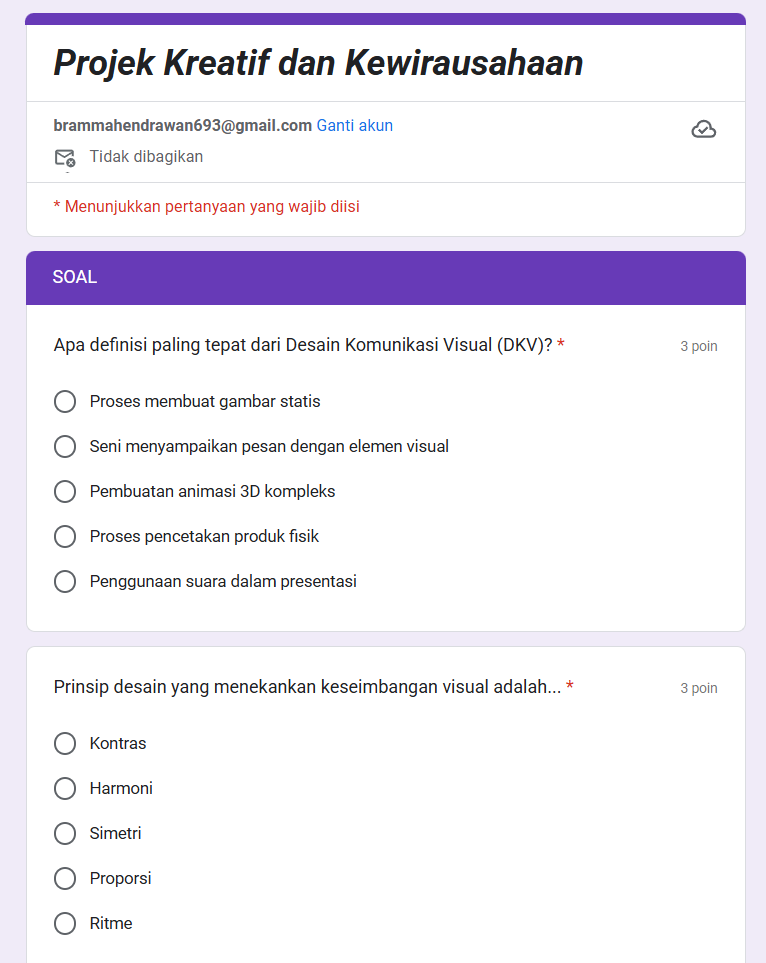
Pada tahap ini telah dilakukan presentasi sistem tentang sistem ujiannya, cara pengoperasian sistem, dan fitur fitur yang ada di sistem. Pelaksanaan UAT presentasi sistem dilaksanakan secara offline di SMK Negeri Rowokangkung dan disaksikan oleh user yaitu 3 Guru SMK Negeri Rowokangkung. Berikut adalah dokumentasi testing sistem ke user sebagaimana terlihat pada gambar 6.1



Gambar 6. Dokumentasi UAT User

1. **Pengambilan Data Soal Ujian**

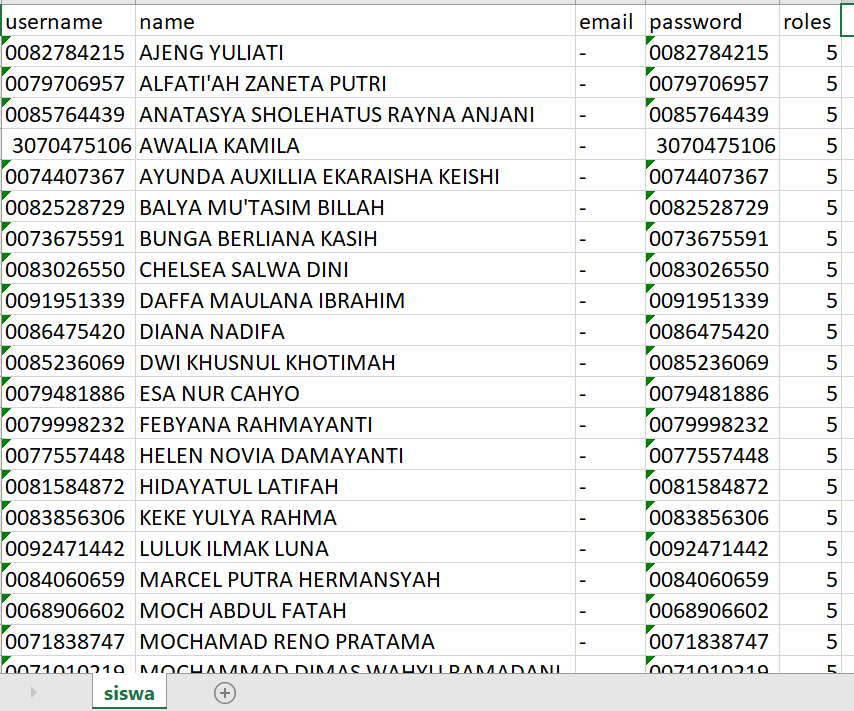
Pada tahap ini telah dilakukan pengambilan data Soal Ujian dengan mata pelajaran Projek Kewirausahaan kelas XI SMK Negeri Rowokangkung. Tujuan dari pengambilan data soal ujian ini digunakan untuk simulasi ujian pada sistem. Soal ini nantinya dimasukkan ke dalam bank soal sistem yang kemudian digunakan untuk testing ujian siswa SMK Negeri Rowokangkung. Berikut dokumentasi contoh soalnya mata pelajaran projek kewirausahaan seperti yang terlihat pada gambar 6.2 berikut.



Gambar 6. Dokumentasi Soal Ujian

1. **Pengambilan Data Siswa**

Pada tahap ini telah dilakukan pengambilan data Siswa 1 kelas 34 orang dengan jenjang kelas XI SMK Negeri Rowokangkung. Tujuan dari pengambilan data siswa ini digunakan untuk login siswa saat melaksanakan simulasi ujian pada sistem. Siswa ini nantinya masuk menggunakan nisn sebagai username dan password. Data siswa diimport dari excel secara langsung oleh sistem. Data siswa seperti yang terlihat pada gambar 6.3 berikut.



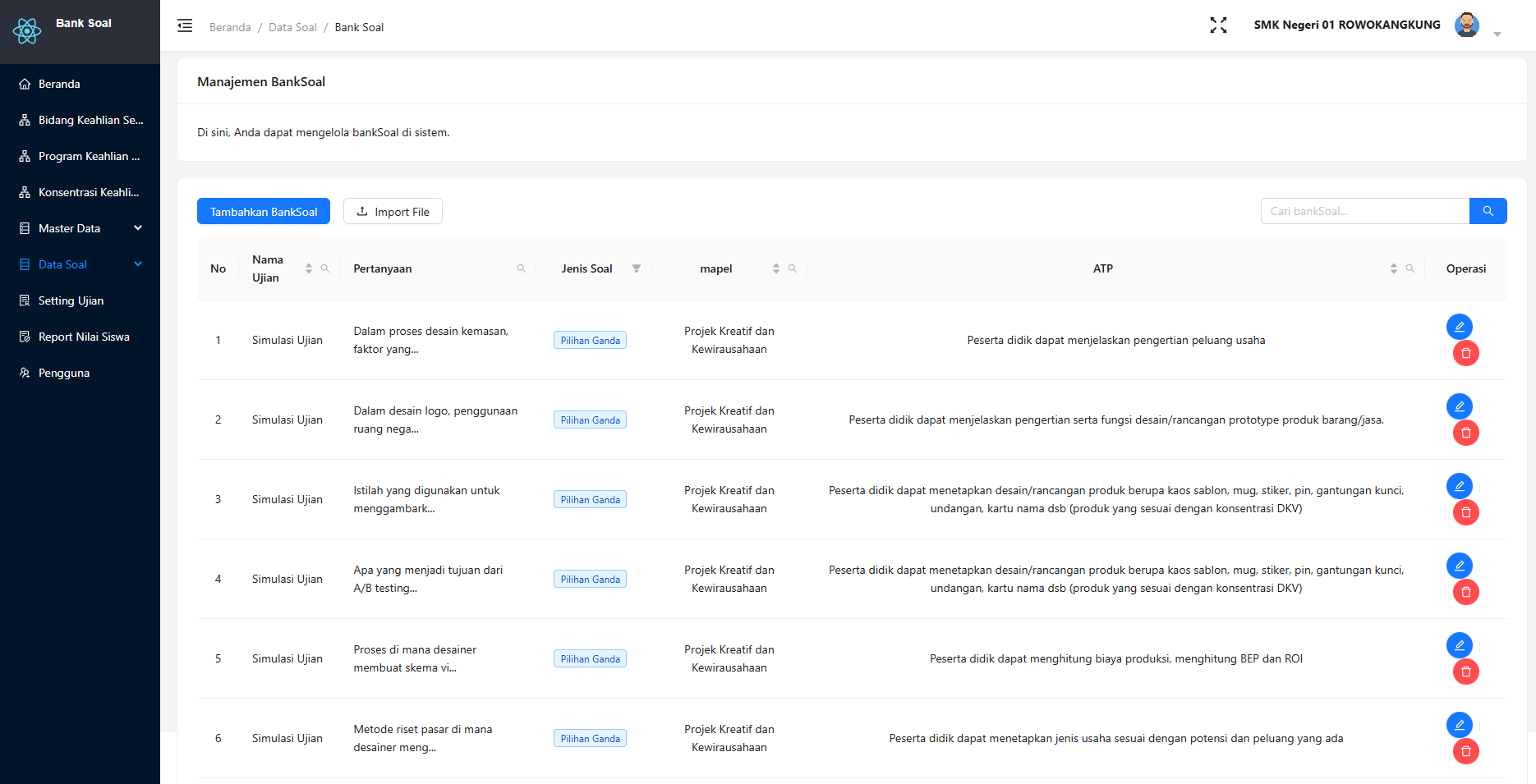
Gambar 6. Data Siswa

* 1. **Pengujian Sistem Pelaksanaan Ujian**

Pada tahap ini dijelaskan pelaksanaan uji coba sistem ujian dengan bank soal Simulasi Ujian yang diambil dari mata pelajaran projek kewirausahaan kelas XI SMK Negeri Rowokangkung.

1. **Pengaturan Bank Soal Ujian**

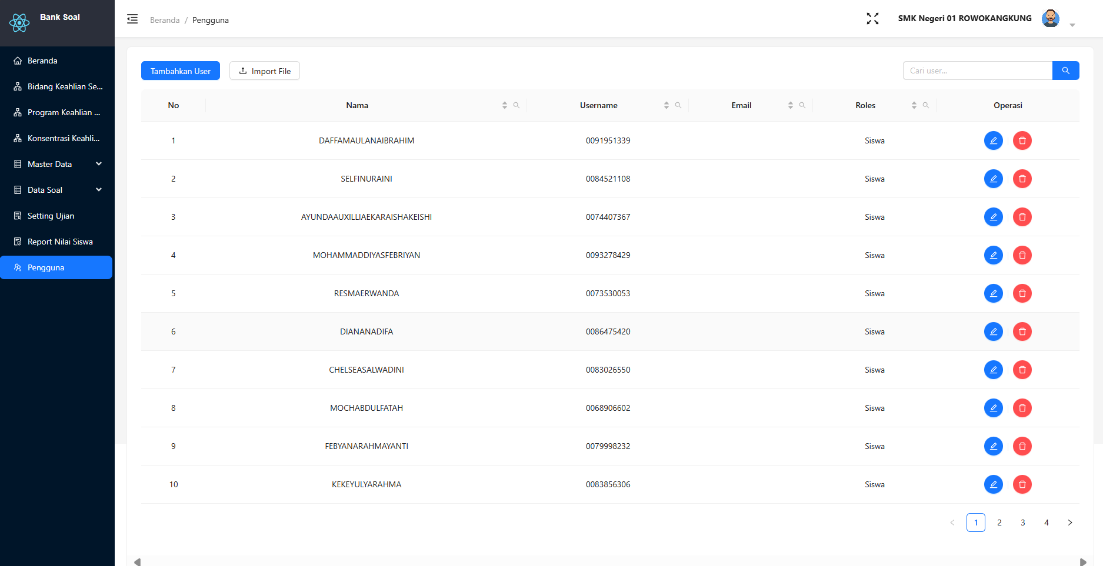
Pada tahap ini dilakukanlah pengaturan bank soal pada sistem, yaitu pengambilan soal berdasarkan nama bank soal. Setiap bank soal bisa memiliki lebih dari satu soal. Pada pengaturan kita kali ini memiliki 1 bank soal mata pelajaran projek kewirausahaan dengan total 20 soal. Data bank soal seperti yang terlihat pada gambar 6.4 berikut.



Gambar 6. Data Bank Soal

1. **Pengaturan User**

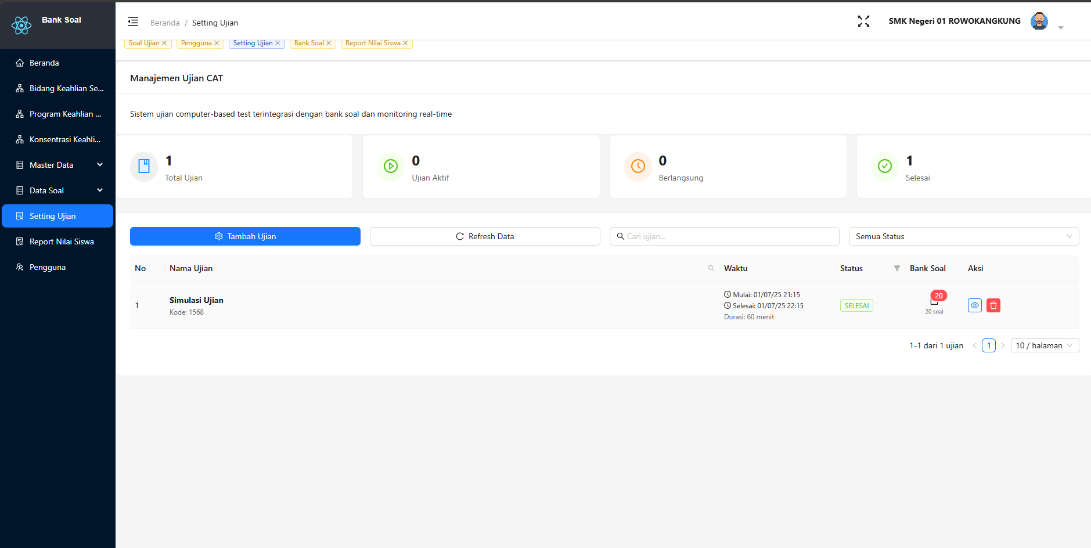
Pada tahap ini dilakukanlah pengaturan user siswa yang ikut simulasi ujian. User ini diimport dari excel ke sistem seperti pada gambar 6.3. User ini berjumlah 34 siswa yang terdiri dari 1 kelas. Data siswa seperti yang terlihat pada gambar 6.4 berikut.



Gambar 6. Data User

1. **Pengaturan Ujian**

Pada tahap ini dilakukanlah pengaturan ujian untuk melakukan simulasi ujian yang akan diakses user siswa SMK Negeri Rowokangkung secara online. Ujian menggunakan bank soal Simulasi Ujian seperti yang terlihat pada gambar 6.4. Ujian dilakukan selama 60 menit dengan kode ujian 1568. Data ujian seperti yang terlihat pada gambar 6.6 berikut



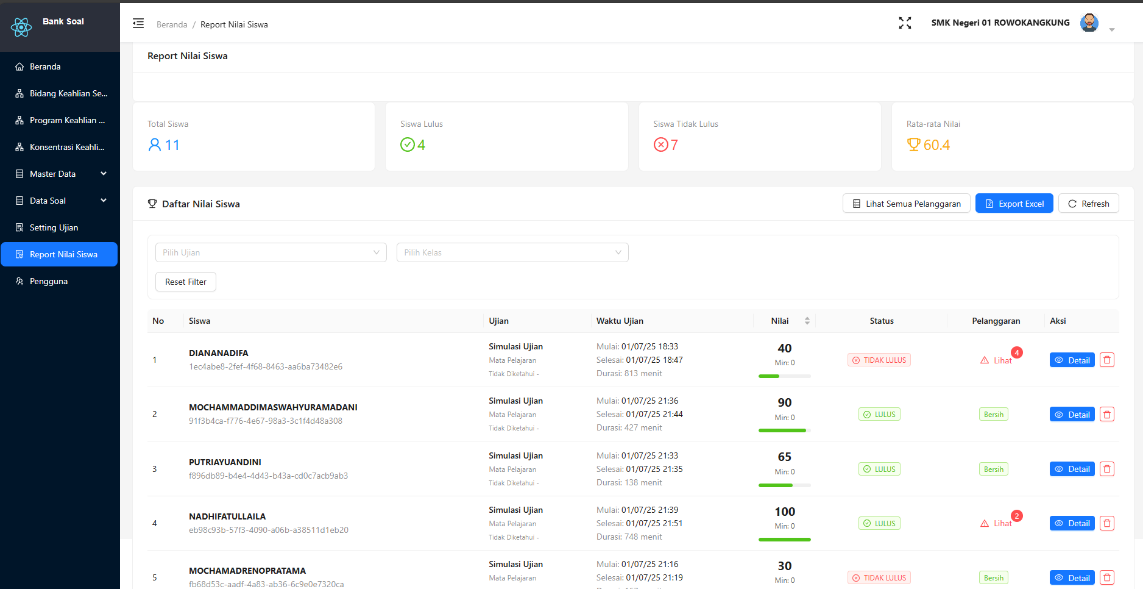
Gambar 6. Data Ujian

* 1. **Hasil Data Pengujian**

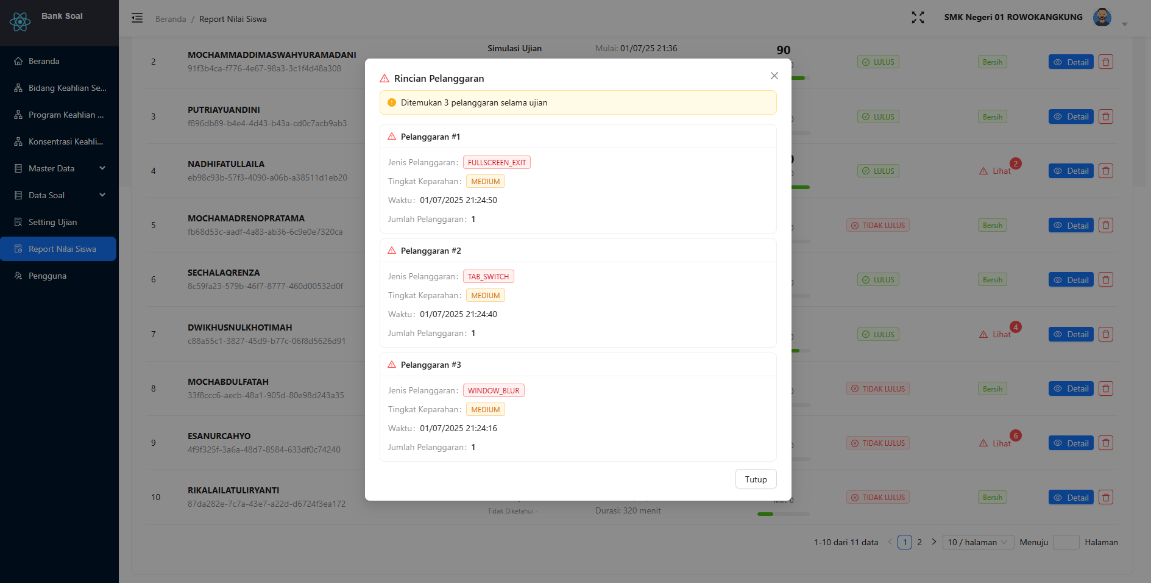
Pada tahap ini dijelaskan hasil uji coba sistem ujian dengan user banyak siswa yang melakukan ujian secara online. Data ini akan dilihat hasil siapa yang melakukan ujian secara jujur maupun tidak jujur.

1. **Hasil Data Ujian Siswa**

Pada tahap ini dilakukanlah report data siswa yang melakukan ujian. Pada uji coba kali ini diketahuilah siswa yang melakukan ujian secara jujur maupun tidak jujur. Data siswa ujian seperti yang terlihat pada gambar 6.7 berikut. Dilaporkan juga data pelanggaran persiswa yang terlihat seperti gambar 6.8 berikut.



Gambar 6. Data Siswa Ujian



Gambar 6. Data Pelanggaran Siswa

# BAB VII SARAN DAN KESIMPULAN

1. **Kesimpulan**

Berdasarkan hasil perancangan dan implementasi Sistem Bank Soal Terdistribusi Multi Sekolah di SMK Negeri Rowokangkung, dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem berhasil mengimplementasikan arsitektur Big Data dengan memanfaatkan Hadoop dan HBase untuk penyimpanan data soal ujian secara terdistribusi.
2. Penggunaan teknologi Spring Boot sebagai backend dan React sebagai frontend memberikan kemudahan dalam pengembangan sistem, penyajian antarmuka yang interaktif, serta mendukung aksesibilitas yang baik untuk berbagai peran pengguna, yaitu Administrator, Admin Sekolah, Guru, dan Siswa.
3. Fitur-fitur penting yang berhasil diimplementasikan yaitu:
   * Manajemen bank soal terdistribusi berdasarkan sekolah.
   * Fitur ujian bagi siswa.
   * Pelaksanaan ujian digital terjadwal.
   * Laporan hasil ujian.
   * Implementasi *cheating detection*.
4. Sistem ini telah diuji melalui User Acceptance Testing (UAT) oleh pihak guru SMK Negeri Rowokangkung, dan berdasarkan hasil pengujian, sistem dinyatakan telah sesuai dengan kebutuhan serta dapat digunakan dalam pelaksanaan ujian sekolah.
5. **Saran**

Adapun saran yang dapat diberikan untuk pengembangan dan penelitian selanjutnya adalah sebagai berikut:

1. Peningkatan Fitur Cheating Detection
   * Peneliti selanjutanya dapat mengembangkan fitur deteksi kecurangan lebih canggih, seperti integrasi dengan sistem monitoring berbasis webcam, analisis perilaku pengguna.
2. Analisa Data Ujian
   * Peneliti selanjutanya dapat mengembangkan fitur menganalisis hasil ujian siswa, memetakan tingkat kesulitan soal, serta memberikan rekomendasi perbaikan pembelajaran secara otomatis.

# DAFTAR PUSTAKA

Apache Software Foundation. (2024, October 9). *HDFS Architecture*. Https://Hadoop.Apache.Org/. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html

Bawane, M. (2022). A Review on Technologies used in MERN stack. *International Journal for Research in Applied Science and Engineering Technology*, *10*(1), 479–488. https://doi.org/10.22214/ijraset.2022.39868

Bikas Katwal. (2019, March 31). *Zookeeper Tutorial With Practical Example*. Https://Bikas-Katwal.Medium.Com/Zookeeper-Introduction-Designing-a-Distributed-System-Using-Zookeeper-and-Java-7f1b108e236e.

Borthakur, D. (2007). *The Hadoop Distributed File System: Architecture and Design*. http://hadoop.apache.org/core/.

Chen, C. (2019). The Computer-Based Examination in China’s Higher Education: Policies, Advantages, and Problems. *International Conference on Education, Management Science and Economics (ICEMSE 2019)*, *96*, 20–23. https://doi.org/10.2991/icemse-19.2019.4

Cloudera. (2012, June 18). *Apache HBase Write Path*. Https://Www.Cloudera.Com/Blog/Technical/Apache-Hbase-Write-Path.Html.

Cloudera. (2021, June 8). *HBase I/O Components*. Https://Docs.Cloudera.Com/Runtime/7.2.10/Configuring-Hbase/Topics/Hbase-Io-Components.Html.

Corporation, O. (2025, April 15). *VirtualBox User Manual*. Https://Www.Virtualbox.Org. https://www.virtualbox.org/wiki/Documentation

Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified Data Processing on Large Clusters. *COMMUNICATIONS OF THE ACM*, *51*(1), 107–113. https://doi.org/10.1145/1327452.1327492

Elashry, A., Shehab, A., & Riad, A. M. (2018). An Efficient Partitioning Technique in SpatialHadoop. *International Journal of Intelligent Computing and Information Science (IJICIS)*, *18*(1), 1–13. https://doi.org/10.21608/ijicis.2018.15893

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), 137–144. https://doi.org/10.1016/j.ijinfomgt.2014.10.007

George, L. (2011). *HBase: The Definitive Guide* (M. Loukides & J. Steele, Eds.; First Edition). O’Reilly Media, Inc.

Hbase, A. (2025, May 27). *Apache HBase Reference Guide*. Https://Hbase.Apache.Org/Book.Html.

Intellij IDEA. (2025, April 7). *Intellij IDEA*. Https://Www.Jetbrains.Com/Help/Idea/Discover-Intellij-Idea.Html. https://www.jetbrains.com/help/idea/discover-intellij-idea.html

Kaiiali, M., Ozkaya, A., Altun, H., Haddad, H., & Alier, M. (2016). Designing a Secure Exam Management System (SEMS) for M-Learning Environments. *IEEE Transactions on Learning Technologies*, *9*(3), 258–271. https://doi.org/10.1109/TLT.2016.2524570

Kozak, M. (2023). Analysis of the Spring Boot and Spring Cloud in developing Java cloud applications. *Journal Computer Sciences Institute (JCSI)*, *27*, 112–120. https://doi.org/10.35784/jcsi.3130

Li, Y., & Chen, R. (2017). Huge Data Analysis and Processing Platform based on Hadoop. *International Conference on Materials Science, Machinery and Energy Engineering (MSMEE 2017)*, *123*, 267–271. https://doi.org/10.2991/msmee-17.2017.56

Merceedi, K. J., & Sabry, N. A. (2021). A Comprehensive Survey for Hadoop Distributed File System. *Asian Journal of Research in Computer Science*, *11*, 46–57. https://doi.org/10.9734/ajrcos/2021/v11i230260

Microsoft. (2025, May 8). *Visual Studio Code*. Https://Code.Visualstudio.Com/. https://code.visualstudio.com/

Mohammed, E. A., Far, B. H., & Naugler, C. (2014). Applications of the MapReduce programming framework to clinical big data analysis: Current landscape and future trends. *BioData Mining*, *7*(1), 1–23. https://doi.org/10.1186/1756-0381-7-22

Myers, Glenford., Sandler, Corey., Badgett, Tom., & Thomas, Todd. (2004). *The Art of Software Testing, Second Edition* (Second Edition). John Wiley & Sons.

Naresh, P., Shekhar, G. N., Kiran Kumar, M., & Rajyalakshmi, P. (2017). Implementation of Multi-node Clusters in Column Oriented Database using HDFS. *International Journal of Engineering and Applied Computer Science (IJEACS)*, *02*(6), 186–190. https://doi.org/10.24032/ijeacs/0206/03

Noobmaster. (2021, April). *Leader Election and Data Replication in Apache ZooKeeper*. Https://Noobtomaster.Com/Apache-Zookeeper/Leader-Election-and-Data-Replication-in-Zookeeper/.

Pressman, R. S. (2010). *Software Engineering: A Practitioner’s Approach* (Seventh Edition). Raghothaman Srinivasan. www.mhhe.com/pressman.

Sommerville, Ian. (2011). *Software engineering*. Pearson.

Triyantoro. (2024). *KEPUTUSAN-MENTERI-PENDIDIKAN-KEBUDAYAAN-RISET-DAN-TEKNOLOGI*.

Unogeeks. (2024, June). *Hbase Zookeeper*. Https://Unogeeks.Com/Hbase-Zookeeper/.

Vyas, R. (2022). Comparative Analysis on Front-End Frameworks for Web Applications. *International Journal for Research in Applied Science and Engineering Technology*, *10*(7), 298–307. https://doi.org/10.22214/ijraset.2022.45260

Wang, B., Huang, W., Qin, X., Song, Y., Wang, C., & Cui, X. (2020). A Distributed Exam Item Bank System Based on Hadoop Ecosystem. *International Conference on Computer Science and Educational Informatization (CSEI)*. https://doi.org/10.1109/CSEI50228.2020.9142493

Williamson, B. (2020). Big Data in Education: The digital future of learning, policy and practice. In *Big Data in Education: The digital future of learning, policy and practice*. SAGE Publications Ltd. https://doi.org/10.4135/9781529714920

Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M. J., Ghodsi, A., Gonzalez, J., Shenker, S., & Stoica, I. (2016). Apache spark: A unified engine for big data processing. *Communications of the ACM*, *59*(11), 56–65. https://doi.org/10.1145/2934664