

**Rancang Bangun Aplikasi Manajemen Rapat
Berbasis Web (Meenutes) dengan Fitur *AI Speech-
to-Text* untuk Notulen Otomatis pada Organisasi
Mahasiswa**

PROYEK AKHIR

Disusun Oleh:

Shifa Humaira Syahrin Tanaz
3312301047

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya
Teknik Informatika



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
2026**

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa karena atas rahmat dan hidayah-Nya, saya dapat menyelesaikan proposal akhir berjudul “Rancang Bangun Aplikasi Manajemen Rapat Berbasis Web (Meenutes) dengan Fitur AI *Speech-to-Text* untuk Notulen Otomatis pada Organisasi Mahasiswa”. Saya juga menyadari bahwa tersusunnya proposal ini tidak terlepas dari dukungan dan bimbingan dari berbagai pihak. Oleh karena itu, pada kesempatan ini saya ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada:

1. Bapak Ir. Bambang Hendrawan, S.T., M.S.M., CIPMP., CISCIP. selaku direktur Politeknik Negeri Batam.
2. Bapak Ahmad Hamim Thohari, S.S.T., M.T. selaku Kepala Jurusan Teknik Informatika Politeknik Negeri Batam.
3. Ibu Yeni Rokhayati, S.Si., M.Sc selaku Kepala Program Studi Teknik Informatika Politeknik Negeri Batam.
4. Ibu Cyntia Lasmi Andesti, selaku dosen pembimbing yang telah memberikan arahan selama penyusunan proposal akhir ini.
5. Kepada kedua orang tua saya yang selalu memberikan dukungan doa dan materi hingga saya bisa sampai ada di tahap ini.
6. Kepada rekan-rekan dan teman seperjuangan saya, terutama yang selalu memberikan semangat dan mengajarkan saya banyak hal dalam penyempurnaan proposal ini.

Penulisan laporan ini bertujuan untuk mengembangkan dan mengimplementasikan sistem manajemen rapat yang dapat mempermudah proses pencatatan notulen melalui teknologi *speech-to-text*. Diharapkan, penelitian ini dapat memberikan banyak manfaat bagi pembaca, baik sebagai referensi penelitian maupun sebagai dasar pengembangan sistem serupa. Saya sangat terbuka jikalau ada saran dan masukan yang membangun demi perbaikan di masa yang akan datang.

DAFTAR ISI

KATA PENGANTAR	2
DAFTAR ISI	3
DAFTAR TABEL	5
DAFTAR GAMBAR	6
DAFTAR LAMPIRAN	7
ABSTRAK	8
BAB I PENDAHULUAN	9
1.1. Latar Belakang	9
1.2. Rumusan Masalah	10
1.3. Tujuan	11
1.4. Batasan Masalah	11
1.5. Manfaat	12
1.5.1. Manfaat Teoritis	12
1.5.2. Manfaat Praktis	12
1.5.3. Kontribusi Terhadap SDGs	12
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek	13
BAB II TINJAUAN PUSTAKA	14
2.1. Tinjauan Solusi dan Sistem Terkait	14
2.2. Landasan Konsep dan Teknologi	16
2.3. Metode Pengembangan Produk	19
BAB III ANALISIS DAN PERANCANGAN	22
3.1. Analisis Kebutuhan	22
3.2. Perancangan	24
BAB IV IMPLEMENTASI DAN PENGUJIAN	47
4.1. Hasil Implementasi	47
4.1.1. Instalasi Sistem	47
4.1.2. Fitur Aplikasi	47
4.1.3. Alur Kerja Aplikasi	50
4.2. Pengujian	51
4.2.1. Pengujian Fungsional	51
Tabel 13. Pengujian Fungsional	51
4.2.2. UAT	52
BAB V KESIMPULAN	54

5.1 Kesimpulan	54
5.2 Saran.....	54
DAFTAR PUSTAKA	56
DAFTAR LAMPIRAN.....	58

DAFTAR TABEL

Tabel 1. Tinjauan Penelitian Sejenis.....	15
Tabel 2. Kebutuhan Fungsional	24
Tabel 3. Kebutuhan Non-Fungsional	25
Tabel 4. <i>Use Case Scenario Login</i>	26
Tabel 5. <i>Use Case Scenario</i> Mengelola Divisi Internal	27
Tabel 6. <i>Use Case Scenario</i> Mengelola Acara.....	28
Tabel 7. <i>Use Case Scenario</i> Mengelola Divisi Acara.....	29
Tabel 8. <i>Use Case Scenario</i> Tambah Rapat.....	30
Tabel 9. <i>Use Case Scenario</i> Mengonversi Suara Menjadi Teks.....	31
Tabel 10. <i>Use Case Scenario</i> Mengonversi Teks Menjadi Ringkasan	31
Tabel 11. <i>Use Case Scenario</i> Menampilkan Jadwal di Kalender	32
Tabel 12. <i>Use Case Scenario Logout</i>	33
Tabel 13. Pengujian Fungsional	51
Tabel 14. <i>User Acceptance Testing (UAT)</i>	52

DAFTAR GAMBAR

Metode <i>Agile</i> (Handayani et al, 2023)	20
Proses Bisnis Berjalan.....	23
Gambaran Umum Sistem	23
<i>Use Case Diagram</i>	25
<i>Activity Diagram Login</i>	34
<i>Activity Diagram</i> Mengelola Divisi Internal.....	35
<i>Activity Diagram</i> Mengelola Acara	36
<i>Activity Diagram</i> Mengelola Divisi Acara.....	37
<i>Activity Diagram</i> Tambah Rapat.....	38
<i>Activity Diagram</i> Mengonversi Suara Menjadi Teks.....	38
<i>Activity Diagram</i> Mengkonversi Teks Menjadi Ringkasan	39
<i>Activity Diagram</i> Menampilkan Jadwal di Kalender	40
<i>Activity Diagram Logout</i>	40
ER Diagram.....	41
Antarmuka Dashboard	43
Antarmuka Divisi Internal.....	43
Antarmuka Acara	44
Antarmuka Detail Acara	44
Antarmuka Divisi Acara	45
Antarmuka Tambah Rapat	45
Antarmuka Detail Rapat.....	46
Antarmuka Kalender	46
Implementasi <i>Login</i>	48
Implementasi Divisi Internal.....	48
Implementasi Acara	48
Implementasi Divisi Acara.....	49
Implementasi Tambah Rapat	49
Implementasi Detail Rapat.....	50
Implementasi Kalender	50
Alur Kerja Aplikasi	51

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement	58
Lampiran B. Link Produk	58
Lampiran C. Dokumen Pengujian.....	58

ABSTRAK

Proses pembuatan notulen rapat pada organisasi mahasiswa umumnya masih dilakukan secara manual sehingga proses dokumentasi rentan mengalami ketidakakuratan pencatatan serta membutuhkan durasi yang lebih panjang untuk menghasilkan notulen.. Proyek akhir ini bertujuan untuk merancang dan membangun aplikasi manajemen rapat berbasis web bernama Meenutes yang dapat membantu proses pengelolaan rapat serta menghasilkan notulen secara otomatis. Aplikasi dikembangkan menggunakan metode *Agile* dengan memanfaatkan *React.js* pada sisi *frontend*, *Flask* pada sisi *backend*, serta *SQL Server* sebagai basis data. Fitur utama aplikasi meliputi pengelolaan acara, pengelolaan divisi, penjadwalan rapat, transkripsi percakapan rapat secara real-time menggunakan *Web Speech API*, serta pembuatan ringkasan dan notulen otomatis menggunakan *Gemini 2.5 Flash*. Hasil pada implementasi menunjukkan bahwa aplikasi berhasil mengintegrasikan proses manajemen rapat dengan teknologi *Artificial Intelligence* dalam satu sistem berbasis web. Hasil pengujian fungsional menunjukkan bahwa setiap fitur pada aplikasi telah beroperasi sesuai dengan kebutuhan yang telah ditetapkan. Selain itu, pengujian usability menggunakan metode *System Usability Scale* (SUS) terhadap 25 responden memperoleh skor rata-rata sebesar 93,5 yang termasuk dalam kategori *Excellent*, *Grade A+*, dan *Acceptable*. Hasil pengujian mengindikasikan bahwa aplikasi Meenutes dapat dengan mudah digunakan, memperoleh respon positif dari para pengguna, serta mampu membantu meningkatkan efektivitas dan efisiensi proses dokumentasi rapat pada organisasi mahasiswa.

Kata Kunci: Manajemen Rapat, *Speech-to-Text*, *Web Speech API*, *Gemini 2.5 Flash*, Notulen Otomatis.

BAB I PENDAHULUAN

1.1. Latar Belakang

Organisasi kemahasiswaan menjadi wadah bagi mahasiswa untuk mengembangkan kemampuan profesional maupun keterampilan berorganisasi. Melalui berbagai program kerja yang telah direncanakan, seluruh anggota organisasi berperan dalam mendukung tercapainya visi, misi, serta tujuan yang telah ditetapkan. Selain itu, berbagai keputusan dalam organisasi umumnya dihasilkan melalui musyawarah mufakat sehingga dapat menumbuhkan sikap tanggung jawab, toleransi, kesabaran, serta kemampuan bekerja sama antaranggota (Pertiwi et al., 2021).

Dalam pelaksanaan kegiatan organisasi, salah satu aktivitas yang penting adalah rapat. Rapat digunakan sebagai sarana koordinasi, penyampaian informasi, pemecahan masalah, serta pengambilan keputusan yang berkaitan dengan pelaksanaan program kerja organisasi. Agar hasil rapat dapat terdokumentasi dengan baik, diperlukan notulen yang berfungsi sebagai catatan resmi mengenai jalannya rapat, hasil diskusi, keputusan yang diambil, serta tindak lanjut yang harus dilaksanakan oleh anggota organisasi.

Namun, proses pembuatan notulen pada organisasi mahasiswa masih banyak dilakukan secara manual oleh seorang notulis. Pendekatan tersebut berpotensi menyebabkan informasi penting terlewat, terjadi kekeliruan dalam pencatatan, serta sangat bergantung pada kemampuan notulis dalam mengikuti jalannya diskusi selama rapat. Selain itu, pembuatan notulensi yang masih bersifat individu dan manual sering kali menghasilkan notulensi yang tidak lengkap, hilang, atau tidak terdokumentasi dengan baik sehingga menyulitkan proses pencarian dan pengarsipan hasil rapat di masa mendatang (Hastriyandi et al., 2025).

Untuk mendukung pelaksanaan rapat yang lebih efektif, diperlukan sistem manajemen rapat yang mampu membantu proses perencanaan, pelaksanaan, dokumentasi, dan tindak lanjut hasil rapat. Sistem manajemen rapat berperan dalam mengorganisasikan berbagai aktivitas rapat sehingga tujuan rapat dapat tercapai dengan baik. Dalam pelaksanaannya, rapat tidak hanya memerlukan

kehadiran peserta yang memenuhi kuorum, tetapi juga dokumentasi yang akurat dan mudah diakses sebagai bahan evaluasi dan tindak lanjut keputusan yang telah disepakati (Hotman Purba et al., 2023).

Salah satu teknologi yang dapat dimanfaatkan untuk mendukung proses pembuatan notulen secara otomatis adalah teknologi *Speech-to-Text*. Teknologi ini memungkinkan suara yang dihasilkan selama rapat dikonversi menjadi teks secara real-time sehingga informasi yang disampaikan peserta rapat dapat terdokumentasi secara lebih cepat dan akurat. Dalam pengembangan aplikasi berbasis web, implementasi *Speech-to-Text* dapat dilakukan menggunakan *Web Speech API*. Menurut Swapurba dan Pratama (2023), *Web Speech API* merupakan API eksperimental yang memungkinkan pengembang mengintegrasikan fitur pengenalan suara (*Speech Recognition*) dan sintesis suara (*Speech Synthesis*) ke dalam aplikasi web. Melalui antarmuka *Speech Recognition*, sistem dapat menerima input suara dari mikrofon dan mengonversinya ke dalam bentuk teks agar dapat diolah oleh sistem.

Meskipun teknologi *Speech-to-Text* mampu mengonversi percakapan rapat menjadi teks secara otomatis, hasil transkripsi yang dihasilkan umumnya masih berupa percakapan lengkap yang panjang dan belum terstruktur sebagai notulen rapat. Oleh karena itu, diperlukan teknologi *Artificial Intelligence* yang mampu menganalisis dan merangkum hasil transkripsi menjadi informasi yang lebih ringkas dan mudah dipahami. Pendekatan yang dapat diterapkan adalah penggunaan *Large Language Model* (LLM) melalui *Gemini 2.5 Flash* yang dikembangkan oleh Google. Model ini mampu memahami konteks percakapan, mengidentifikasi informasi penting, serta menghasilkan ringkasan yang terstruktur sehingga dapat membantu proses pembuatan notulen rapat secara otomatis.

1.2. Rumusan Masalah

Berdasarkan uraian pada latar belakang, permasalahan yang menjadi fokus dalam penelitian ini dirumuskan sebagai berikut:

1. Bagaimana proses perancangan dan pengembangan aplikasi manajemen rapat berbasis web yang ditujukan untuk organisasi mahasiswa?

2. Bagaimana mengintegrasikan teknologi *Web Speech API* dan *Gemini 2.5 Flash* ke dalam aplikasi untuk menghasilkan notulen rapat secara otomatis?
3. Bagaimana membantu proses dokumentasi dan pengelolaan hasil rapat agar lebih efektif dan efisien dibandingkan metode manual?

1.3. Tujuan

Penelitian ini memiliki tujuan sebagai berikut:

1. Menghasilkan aplikasi manajemen rapat berbasis web (Meenutes) yang mampu mendukung pengelolaan kegiatan rapat di lingkungan organisasi mahasiswa.
2. Mengimplementasikan teknologi *Web Speech API* untuk menghasilkan transkripsi percakapan rapat secara otomatis serta mengintegrasikan *Gemini 2.5 Flash* untuk menghasilkan ringkasan dan notulen rapat secara otomatis.
3. Membantu meningkatkan efektivitas dan efisiensi proses dokumentasi, pengelolaan, serta pengarsipan hasil rapat dibandingkan dengan metode pencatatan manual.

1.4. Batasan Masalah

Agar penelitian tetap berfokus pada tujuan yang telah ditetapkan, ruang lingkup penelitian dibatasi pada beberapa aspek berikut:

1. Sistem yang dikembangkan berupa aplikasi manajemen rapat berbasis web yang ditujukan untuk organisasi mahasiswa.
2. Fitur *Speech-to-Text* pada aplikasi diimplementasikan dengan memanfaatkan *Web Speech API* sehingga percakapan rapat dapat ditranskripsikan secara *real-time*.
3. Pembuatan ringkasan dan notulen otomatis dilakukan menggunakan *Gemini 2.5 Flash* berdasarkan hasil transkripsi rapat.
4. Sistem berfokus pada pengelolaan data rapat, transkripsi percakapan, dan pembuatan notulen otomatis, serta tidak mencakup fitur konferensi video atau komunikasi daring.

5. Hasil transkripsi dan ringkasan yang dihasilkan sistem dipengaruhi oleh kualitas suara, koneksi internet, dan kemampuan teknologi yang digunakan.

1.5. Manfaat

1.5.1. Manfaat Teoritis

Proyek akhir ini diharapkan dapat menjadi salah satu referensi dalam pengembangan ilmu pengetahuan, khususnya pada penerapan teknologi *Artificial Intelligence* (AI) dalam bidang manajemen rapat. Hasil proyek akhir ini diharapkan dapat dimanfaatkan sebagai acuan dalam penelitian lanjutan maupun pengembangan sistem notulen rapat otomatis yang menerapkan teknologi *Speech-to-Text* dan *Large Language Model* (LLM).

1.5.2. Manfaat Praktis

1. Membantu organisasi mahasiswa dalam mengelola data dan dokumentasi rapat secara lebih terstruktur.
2. Mempermudah proses pencatatan hasil rapat melalui fitur transkripsi otomatis menggunakan teknologi *Speech-to-Text*.
3. Membantu menghasilkan ringkasan dan notulen rapat secara otomatis sehingga mengurangi beban kerja notulis dan mempercepat proses dokumentasi.
4. Memudahkan penyimpanan, pencarian, dan pengarsipan hasil rapat untuk kebutuhan evaluasi maupun tindak lanjut kegiatan organisasi.

1.5.3. Kontribusi Terhadap SDGs

Pengembangan aplikasi Meenutes selaras dengan upaya pencapaian *Sustainable Development Goals* (SDGs), khususnya SDG 9 (*Industry, Innovation and Infrastructure*). Integrasi teknologi *Artificial Intelligence* (AI) pada aplikasi ini diharapkan mampu meningkatkan efektivitas pengelolaan informasi dan dokumentasi rapat. Di samping itu, penelitian ini juga mendukung SDG 4 (*Quality Education*) dengan menghadirkan teknologi yang dapat membantu organisasi kemahasiswaan dalam mengelola kegiatan secara lebih efisien dan inovatif.

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek Akhir ini dilaksanakan secara individu dengan tetap melibatkan kolaborasi dan sinergi bersama beberapa pihak yang berperan dalam proses pengembangan sistem. Kolaborasi dilakukan untuk memastikan bahwa solusi yang dikembangkan sesuai dengan kebutuhan pengguna serta memenuhi standar akademik yang ditetapkan.

Kolaborasi pertama dilakukan bersama dosen pembimbing yang memberikan bimbingan, masukan, serta evaluasi pada setiap tahapan pelaksanaan proyek.. Bimbingan yang diberikan mencakup penentuan ruang lingkup proyek, perancangan sistem, implementasi teknologi yang digunakan, hingga penyusunan laporan Proyek Akhir.

Selain itu, kolaborasi juga dilakukan dengan mahasiswa sebagai calon pengguna sistem sekaligus responden. Mahasiswa memberikan informasi terkait proses pelaksanaan rapat yang selama ini dilakukan, kendala yang dihadapi dalam pembuatan notulen rapat, serta kebutuhan yang diharapkan dari sistem yang dikembangkan. Masukan tersebut digunakan sebagai dasar dalam analisis kebutuhan dan pengembangan fitur pada aplikasi Meenutes.

Melalui kolaborasi dan sinergi tersebut, diharapkan aplikasi Meenutes yang dikembangkan dapat memberikan solusi yang sesuai dengan kebutuhan pengguna, khususnya dalam membantu proses dokumentasi dan pengelolaan hasil rapat pada organisasi mahasiswa.

BAB II TINJAUAN PUSTAKA

2.1. Tinjauan Solusi dan Sistem Terkait

Bagian ini berisi Beberapa penelitian dan implementasi sistem telah dilakukan untuk mendukung proses transkripsi suara, pembuatan notulen otomatis, serta peringkasan teks menggunakan teknologi *Artificial Intelligence* (AI). Penelitian-penelitian tersebut memanfaatkan berbagai pendekatan, seperti *Speech Recognition*, *Speech-to-Text*, dan *Text Summarization* untuk meningkatkan efisiensi proses dokumentasi informasi. Tinjauan terhadap penelitian sejenis dilakukan untuk memahami teknologi yang digunakan, fitur yang tersedia, serta kelebihan dan keterbatasan dari masing-masing solusi.

Penelitian oleh Swapurba dan Pratama (2023) mengimplementasikan *Web Speech API* pada aplikasi berbasis web untuk mengubah input suara menjadi teks. Penelitian ini menunjukkan bahwa teknologi *Speech Recognition* dapat dimanfaatkan sebagai alternatif input manual pada aplikasi web. Namun, penelitian tersebut berfokus pada pengenalan suara untuk pengisian data dan belum diterapkan pada kebutuhan dokumentasi rapat.

Lubis dkk. (2024) mengembangkan aplikasi *Speech-to-Text Online* yang memanfaatkan algoritma *Deep Learning* dengan pendekatan *Recurrent Neural Network* (RNN). Berdasarkan hasil penelitian tersebut, teknologi *Speech Recognition* mampu mengonversi suara menjadi teks secara otomatis sehingga proses pencatatan serta penyusunan dokumen menjadi lebih efisien. Namun, penelitian tersebut hanya berfokus pada proses transkripsi dan belum mendukung pengelolaan rapat maupun penyusunan notulen secara otomatis.

Rifaldi (2025) mengembangkan fitur *Speech Recognition* pada aplikasi notulensi menggunakan *Whisper AI* dan *Speaker Diarization*. Sistem yang dikembangkan mampu menghasilkan transkripsi percakapan rapat secara otomatis serta mengidentifikasi pembicara yang terlibat dalam diskusi. Penelitian ini menunjukkan bahwa teknologi kecerdasan buatan dapat membantu proses dokumentasi rapat secara lebih efektif. Namun, sistem tersebut belum menyediakan fitur manajemen rapat yang terintegrasi.

Penelitian yang dilakukan oleh Karo dkk. (2024) menerapkan metode *Maximum Marginal Relevance* (MMR) sebagai pendekatan dalam proses peringkasan teks secara otomatis. Hasil penelitian menunjukkan bahwa teknologi *text summarization* mampu menyederhanakan informasi dalam jumlah besar menjadi ringkasan yang lebih ringkas tanpa menghilangkan inti pembahasan. Oleh karena itu, teknologi ini berpotensi diterapkan pada sistem notulen otomatis untuk menghasilkan ringkasan dari hasil transkripsi rapat.

Berdasarkan beberapa penelitian tersebut, dapat disimpulkan bahwa teknologi *Speech Recognition* dan *Text Summarization* telah banyak digunakan untuk membantu proses dokumentasi informasi secara otomatis. Namun, sebagian besar penelitian masih berfokus pada salah satu aspek tertentu, seperti transkripsi suara atau peringkasan teks. Oleh karena itu, pada proyek ini dikembangkan aplikasi Meenutes yang mengintegrasikan fitur manajemen rapat, transkripsi percakapan menggunakan *Web Speech API*, serta pembuatan ringkasan dan notulen otomatis menggunakan *Gemini 2.5 Flash* dalam satu aplikasi berbasis web. Perbandingan penelitian sejenis dapat dilihat pada Tabel 2.1.

Tabel 1. Tinjauan Penelitian Sejenis

Peneliti/ Tahun	Masalah	Algoritma	Dataset	Framework	Performa	Kelebihan	Keterbatasan
Swapurba & Pratama (2023)	Input suara pada aplikasi web	<i>Speech Recognition (Web Speech API)</i>	Input email dan nomor invoice	<i>Web Speech API, JavaScript</i>	Berhasil mengenali input numerik	Implementasi <i>Speech-to-Text</i> langsung pada browser	Kurang akurat mengenali alamat email
Lubis dkk. (2024)	<i>Speech-to-Text Online</i>	<i>Deep Learning, RNN</i>	Data suara pengguna	<i>Web Speech API</i>	Menghasilkan transkripsi suara ke teks	Mendukung konversi suara menjadi teks secara otomatis	Akurasi dipengaruhi kualitas suara dan aksen
Rifaldi (2025)	Notulensi rapat otomatis	<i>Whisper AI, Speaker Diarization</i>	Rekaman rapat perusahaan	<i>Hugging Face, Whisper AI</i>	Transkripsi dan identifikasi pembicara	Mendukung notulensi otomatis dan pemisahan pembicara	Fokus pada transkripsi, belum menyediakan manajemen rapat
Karo dkk. (2024)	Peringkasan ulasan aplikasi	<i>Maximum Marginal Relevance (MMR)</i>	461 ulasan aplikasi <i>Digital Library System</i>	<i>Python, NLTK</i>	Precision 30,51%, Recall 56,25%, F1-Score 39,56%	Mampu menghasilkan ringkasan otomatis dari banyak dokumen	Menggunakan metode <i>extractive summarization</i>
Proyek ini	Notulen rapat otomatis pada organisasi	<i>Speech-to-Text dan LLM Summarization</i>	Data percakapan rapat organisasi mahasiswa	<i>Web Speech API, Gemini 2.5 Flash, Flask</i>	Target >85%	Mengintegrasikan manajemen rapat, transkripsi, dan ringkasan otomatis dalam	Bergantung pada kualitas suara dan koneksi internet

	mahasiswa					satu aplikasi	
--	-----------	--	--	--	--	---------------	--

2.2. Landasan Konsep dan Teknologi

Bagian ini memuat konsep-konsep dasar yang mendukung Proyek. Teori yang ditulis bukan teori umum yang terlalu luas, tetapi yang berkaitan langsung dengan topik dan studi kasus yang dikerjakan.

1. Definisi Organisasi Mahasiswa

Organisasi kemahasiswaan berfungsi sebagai media bagi mahasiswa untuk mengembangkan berbagai kompetensi, seperti kepemimpinan, komunikasi, kerja sama tim, dan tanggung jawab melalui pelaksanaan program kerja organisasi. Selain mendukung pengembangan kemampuan nonakademik, organisasi kemahasiswaan mendorong mahasiswa untuk berpartisipasi dalam proses pengambilan keputusan secara musyawarah serta membangun kerja sama antaranggota. Dengan demikian, organisasi kemahasiswaan berkontribusi dalam membentuk karakter dan profesionalitas mahasiswa (Pertiwi et al., 2021).

2. Definisi Rapat

Rapat adalah kegiatan yang mempertemukan anggota dalam suatu organisasi, perusahaan, atau instansi untuk membahas suatu permasalahan, bertukar pendapat, serta menentukan keputusan berdasarkan hasil kesepakatan bersama.. Rapat juga menjadi media komunikasi kelompok yang digunakan untuk menyatukan pemikiran serta mencapai tujuan yang telah ditetapkan bersama (Jumiyati & Azlina, 2021).

3. Definisi Notulen

Notulen rapat merupakan catatan penting yang memuat jalannya suatu kegiatan rapat mulai dari awal hingga akhir. Notulen digunakan sebagai dokumen yang mencatat jalannya rapat, hasil pembahasan, keputusan yang telah disepakati, serta berbagai informasi penting yang dapat dijadikan acuan dalam proses evaluasi dan pelaksanaan tindak lanjut (Asrin, 2024).

4. Definisi Sistem Manajemen Rapat

Sistem manajemen rapat merupakan sistem yang digunakan untuk membantu proses perencanaan, pelaksanaan, dokumentasi, dan tindak

lanjut hasil rapat agar dapat berjalan secara lebih efektif dan terorganisasi. Sistem ini berperan dalam mengelola berbagai aktivitas rapat, mulai dari penyusunan agenda, pencatatan hasil rapat, hingga pengelolaan dokumentasi yang diperlukan sebagai bahan evaluasi dan tindak lanjut keputusan yang telah disepakati. Dengan adanya sistem manajemen rapat, proses pengelolaan rapat dapat dilakukan secara lebih terstruktur sehingga tujuan rapat dapat tercapai dengan baik (Hotman Purba et al., 2023).

5. Definisi *Speech Recognition* dan *Speech-to-Text*

Speech Recognition merupakan teknologi pengenalan suara yang berfungsi mengonversi ucapan manusia menjadi data digital sehingga dapat diproses oleh sistem komputer. Salah satu penerapan yang paling umum dari teknologi ini adalah *Speech-to-Text*, yaitu proses mengubah suara atau ucapan menjadi bentuk teks secara otomatis. Teknologi ini banyak digunakan dalam berbagai aplikasi seperti *Voice-to-Text*, penerjemahan bahasa, dan sistem transkripsi otomatis. Dalam implementasinya, sistem *Speech Recognition* melakukan proses identifikasi ucapan, pemrosesan suara, serta mentranskripsikan hasil ucapan menjadi teks yang dapat digunakan oleh aplikasi (Ferdiansyah & Aditya, 2024).

Pada proyek ini, teknologi *Speech-to-Text* digunakan untuk mengubah percakapan yang terjadi selama rapat menjadi teks secara otomatis sehingga dapat dimanfaatkan sebagai dasar pembuatan notulen rapat.

6. Definisi *Web Speech API*

Web Speech API adalah *Application Programming Interface* (API) yang memungkinkan aplikasi web memanfaatkan fitur pengenalan dan sintesis suara. Fitur *Speech Recognition* digunakan untuk mengubah masukan suara menjadi teks, sedangkan *Speech Synthesis* menghasilkan suara dari teks yang diberikan. Dengan memanfaatkan teknologi tersebut, interaksi pengguna dengan aplikasi dapat dilakukan melalui perintah suara tanpa bergantung sepenuhnya pada input menggunakan keyboard (Swapurba & Pratama, 2023).

Pada proyek ini, *Web Speech API* digunakan untuk melakukan proses transkripsi percakapan rapat secara *real-time* melalui fitur *Speech Recognition*. Hasil transkripsi tersebut kemudian digunakan sebagai data masukan untuk proses pembuatan notulen rapat secara otomatis.

7. Definisi *Gemini 2.5 Flash*

Gemini 2.5 Flash merupakan model *Artificial Intelligence* yang dikembangkan oleh *Google DeepMind* dan termasuk ke dalam kategori *Large Multimodal Model (LMM)*. Model ini dirancang untuk memberikan keseimbangan antara kualitas keluaran multimodal dan kecepatan pemrosesan (*low latency*). *Gemini 2.5 Flash* memiliki kemampuan untuk memahami berbagai jenis data, seperti teks dan gambar, serta mampu memproses konteks yang kompleks secara efisien. Kemampuan tersebut menjadikan *Gemini 2.5 Flash* sesuai untuk diimplementasikan pada aplikasi berbasis web yang memerlukan respons cepat dan hasil yang akurat (Akrianto et al., 2026).

8. Definisi *React.js*

React JS adalah *library* JavaScript *open source* yang berfungsi untuk mengembangkan antarmuka pengguna (*User Interface/UI*) pada aplikasi web dengan pendekatan berbasis komponen (*component-based*). *Library* ini pertama kali diperkenalkan oleh Jordan Walke pada tahun 2013 dan hingga saat ini menjadi salah satu teknologi yang banyak digunakan dalam pengembangan aplikasi *front-end*. Penerapan konsep *reusable component* memungkinkan komponen yang telah dibuat digunakan kembali pada berbagai bagian aplikasi sehingga pengembangan menjadi lebih cepat dan mudah. Selain itu, *React JS* dapat digunakan untuk membangun antarmuka pada berbagai platform, baik desktop maupun *mobile* (Sulistiyorini et al., 2022).

9. Definisi *Flask*

Flask adalah *micro-framework* berbasis Python yang memiliki karakteristik ringan, fleksibel, dan mudah dikembangkan sesuai kebutuhan. Berbeda dengan beberapa *framework* lain, *Flask* tidak menyediakan banyak *library* maupun tools bawaan sehingga pengembang

dapat memilih sendiri komponen tambahan yang diperlukan. Karakteristik tersebut membuat *Flask* mampu berjalan dengan kebutuhan sumber daya yang relatif rendah tanpa mengurangi fungsinya dalam membangun aplikasi web. Fitur-fitur yang tidak tersedia secara bawaan dapat ditambahkan melalui berbagai ekstensi pihak ketiga. Selain itu, *Flask* didukung oleh dua komponen utama, yaitu Werkzeug yang menyediakan mekanisme *routing*, *debugging*, dan *Web Server Gateway Interface* (WSGI), serta Jinja2 yang berperan sebagai *template engine* (Ngantung & Pakereng, 2021).

10. Definisi *SQL Server Management Studio*

SQL Server Management Studio (SSMS) merupakan alat resmi yang dikembangkan oleh Microsoft untuk mengelola *SQL Server*. SSMS menyediakan antarmuka grafis yang memungkinkan pengguna membuat tabel, menulis dan menjalankan *query*, memantau performa sistem, serta mengelola berbagai objek database secara lebih mudah. Selain itu, SSMS juga mendukung fitur *indexing* dan *performance monitoring* yang membantu administrator maupun pengembang dalam mengelola sistem basis data secara efektif (Aritonang et al., 2025).

2.3. Metode Pengembangan Produk

Penelitian ini mengembangkan aplikasi menggunakan metode Agile. Pendekatan ini menekankan proses pengembangan yang berlangsung secara bertahap dan berulang (*iterative*), sehingga perubahan kebutuhan yang muncul selama pengembangan dapat diakomodasi dengan lebih fleksibel tanpa mengganggu jalannya proses pengembangan.



Gambar 1. Metode *Agile* (Handayani et al, 2023)

Berikut ini tahapan yang digunakan dalam metode *Agile* (Handayani et al, 2023):

1. Perencanaan (*Planning*)

Tahap *Planning* merupakan tahap awal dalam penerapan metode Agile. Pada tahap ini dilakukan pengumpulan dan analisis kebutuhan sistem sebagai dasar penyusunan rencana pengembangan aplikasi. Kegiatan tersebut dilakukan melalui wawancara, observasi, serta analisis terhadap dokumen yang berkaitan dengan sistem yang akan dikembangkan.

2. Implementasi

Tahap implementasi merupakan proses merealisasikan hasil perancangan menjadi sebuah aplikasi melalui kegiatan pengkodean. Pada tahap ini dilakukan pengembangan berbagai fitur, pembangunan antarmuka pengguna, serta integrasi antar modul sehingga seluruh fungsi aplikasi dapat berjalan sesuai dengan kebutuhan pengguna.

3. Pengujian (*Testing*)

Tahap Testing dilakukan untuk mengevaluasi dan memastikan bahwa setiap fungsi pada aplikasi telah memenuhi kebutuhan serta spesifikasi yang telah ditentukan. Pengujian dilakukan secara menyeluruh untuk menemukan *bug*, *error*, maupun ketidaksesuaian fungsi sehingga aplikasi siap digunakan oleh pengguna.

4. Dokumentasi

Dokumentasi adalah proses untuk mendokumentasi sistem yang sudah lulus pengujian. Dokumentasi dilakukan dengan merekam langkah sistem saat dibangun serta mendokumentasikan hasil pengujian untuk pemeliharaan kedepannya.

5. *Deployment*

Tahap *Deployment* dilakukan dengan merilis aplikasi ke lingkungan operasional sehingga dapat diakses dan digunakan oleh pengguna. Sebelum proses rilis dilakukan, sistem terlebih dahulu menjalani verifikasi akhir untuk memastikan seluruh fitur berfungsi sebagaimana mestinya.

6. *Maintenance*

Maintenance dilakukan setelah aplikasi digunakan untuk menjaga stabilitas sistem melalui perbaikan *bug*, penyempurnaan fitur, serta penyesuaian terhadap kebutuhan baru yang muncul selama penggunaan aplikasi.

BAB III ANALISIS DAN PERANCANGAN

3.1. Analisis Kebutuhan

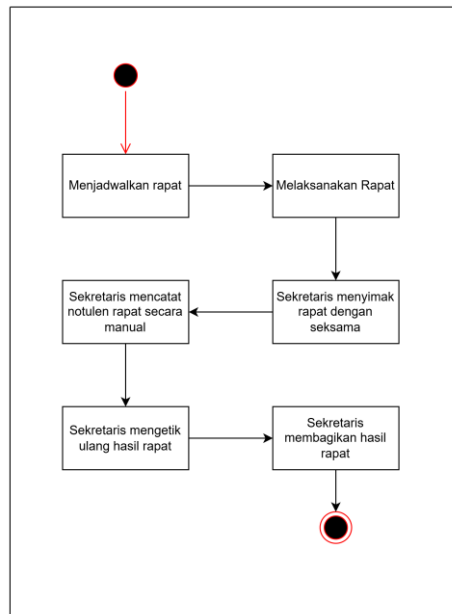
3.1.1 Proses Bisnis Berjalan

Proses pelaksanaan rapat pada organisasi mahasiswa saat ini masih dilakukan secara konvensional, terutama dalam proses dokumentasi hasil rapat. Sebelum rapat dilaksanakan, pengurus terlebih dahulu menentukan jadwal rapat sesuai dengan kebutuhan organisasi. Setelah jadwal ditetapkan, rapat dilaksanakan dengan melibatkan seluruh peserta yang berkepentingan.

Selama rapat berlangsung, sekretaris bertugas menyimak jalannya rapat untuk mencatat berbagai informasi penting seperti hasil diskusi, keputusan yang diambil, serta tindak lanjut yang harus dilaksanakan. Pencatatan notulen masih dilakukan secara manual sehingga sekretaris harus fokus mendengarkan sekaligus mencatat informasi yang disampaikan oleh peserta rapat.

Setelah rapat selesai, hasil catatan yang telah dibuat kemudian diketik ulang ke dalam format dokumen yang lebih rapi dan terstruktur. Dokumen tersebut selanjutnya dibagikan kepada peserta rapat sebagai bahan informasi dan dokumentasi hasil rapat yang telah dilaksanakan.

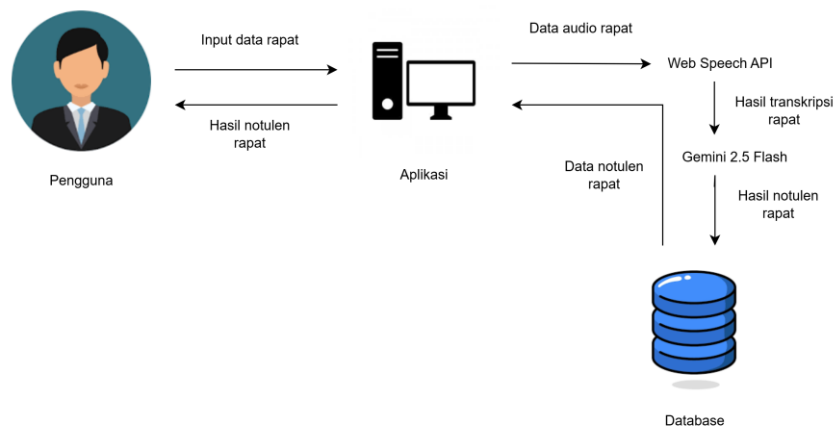
Proses bisnis yang diterapkan saat ini masih menghadapi beberapa kendala dalam pelaksanaannya, di antaranya risiko informasi yang terlewat selama proses pencatatan, ketergantungan pada kemampuan sekretaris dalam mencatat seluruh pembahasan rapat, serta waktu tambahan yang dibutuhkan untuk mengetik ulang hasil rapat. Selain itu, proses dokumentasi dan distribusi hasil rapat menjadi kurang efisien karena masih dilakukan secara manual. Proses bisnis yang diterapkan sebelum pengembangan sistem dijelaskan melalui diagram pada Gambar 2.



Gambar 2. Proses Bisnis Berjalan

3.1.2 Gambaran Umum Sistem

Gambaran umum sistem digunakan untuk memberikan ilustrasi mengenai cara kerja sistem yang akan dikembangkan. Sistem yang diusulkan pada proyek ini adalah aplikasi manajemen rapat berbasis web bernama Meenutes yang memanfaatkan teknologi *Web Speech API* untuk melakukan transkripsi percakapan rapat secara real-time serta *Gemini 2.5 Flash* untuk menghasilkan notulen rapat secara otomatis. Gambaran umum sistem yang diusulkan dapat dilihat pada Gambar 3.2.



Gambar 3. Gambaran Umum Sistem

Berdasarkan Gambar 3.2, sistem Meenutes terdiri dari beberapa komponen utama yaitu pengguna, aplikasi Meenutes, *Web Speech API*, *Gemini 2.5 Flash*, dan database. Pengguna berinteraksi dengan aplikasi untuk mengelola data rapat dan memperoleh hasil notulen rapat. Saat rapat berlangsung, aplikasi mengirimkan data audio rapat ke *Web Speech API* untuk dilakukan proses *Speech-to-Text* sehingga menghasilkan transkripsi rapat secara real-time. Hasil transkripsi tersebut kemudian diproses oleh *Gemini 2.5 Flash* untuk menghasilkan notulen rapat secara otomatis. Selanjutnya, hasil notulen disimpan ke dalam database dan dapat diakses kembali oleh pengguna melalui aplikasi Meenutes.

3.2. Perancangan

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kumpulan fitur dan fungsi yang harus dimiliki oleh aplikasi untuk memenuhi kebutuhan pengguna serta mendukung operasional sistem. Daftar fungsi yang dibuat dapat dilihat dalam tabel berikut.

Tabel 2. Kebutuhan Fungsional

FR-01	Pengguna dapat melakukan <i>login</i>
FR-02	Pengguna dapat mengelola divisi internal
FR-03	Pengguna dapat mengelola acara
FR-04	Pengguna dapat mengelola divisi acara
FR-05	Pengguna dapat menambahkan rapat baru
FR-06	Sistem dapat merekam pembahasan rapat dan mengkonversi suara menjadi teks menggunakan fitur <i>speech-to-text</i>
FR-07	Sistem dapat menghasilkan <i>summary</i> dari hasil teks yang sudah dibuat
FR-08	Sistem dapat menampilkan jadwal pada kalender
FR-09	Pengguna dapat melakukan <i>logout</i>

3.2.2 Kebutuhan Non-Fungsional

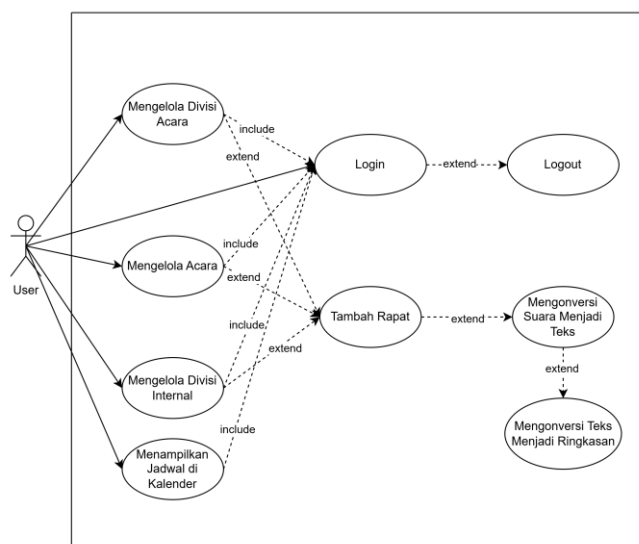
Kebutuhan nonfungsional mendeskripsikan karakteristik kualitas sistem, seperti kompatibilitas, performa, dan kemudahan penggunaan, yang mendukung kinerja aplikasi secara keseluruhan. Spesifikasi teknis yang dibuat adalah sebagai berikut.

Tabel 3. Kebutuhan Non-Fungsional

NFR-01	Bahasa	Sistem menyajikan seluruh antarmuka serta informasi menggunakan bahasa Indonesia yang komunikatif sehingga seluruh anggota organisasi dapat memahami informasi yang disajikan dengan jelas.
NFR-02	<i>Kompatibility</i>	Sistem harus dapat dijalankan pada browser umum seperti <i>Microsoft edge</i> dan <i>google chrome</i>
NFR-03	<i>Skalability</i>	Sistem dirancang agar mampu mengakomodasi penambahan jumlah pengguna dan data tanpa mengurangi kualitas kinerjanya secara signifikan.

3.2.3 Diagram Use Case

Use case Diagram digunakan sebagai sarana untuk memodelkan hubungan antara aktor dan sistem dalam UML. Diagram ini memperlihatkan berbagai fungsi atau layanan yang dapat diakses oleh setiap aktor pada aplikasi. Gambar berikut menampilkan diagram *Use Case* dari aplikasi yang dikembangkan.



Gambar 4. Use Case Diagram

3.2.4 Skenario *Use Case*

Skenario *Use Case* menjelaskan urutan aktivitas yang dilakukan oleh aktor ketika menggunakan setiap fitur pada aplikasi Meenutes. Selain menggambarkan tindakan yang dilakukan aktor, skenario ini juga menunjukkan respons sistem pada setiap tahapan sehingga alur proses bisnis dapat dipahami dengan lebih jelas.

a. *Use Case Scenario Login*

Tabel 4. *Use Case Scenario Login*

<i>Use Case</i>	FR-01
<i>Nama Use Case</i>	<i>Login</i>
<i>Aktor</i>	Pengguna
<i>Deskripsi</i>	Pengguna dapat masuk ke system dengan kredensial yang valid.
<i>Kondisi Awal</i>	Pengguna sudah memiliki akun.
<i>Kondisi Akhir</i>	Pengguna berhasil masuk ke dalam aplikasi.
Alur Kejadian Normal	
Aktor	Sistem
1. Membuka halaman <i>Login</i> 3. Mengisi form <i>Login</i>	2. Tampilkan form <i>Login</i> 4. Memvalidasi kredensial pengguna 5. Mengarahkan ke halaman dashboard
Alur Kerja Alternatif	
1. Pengguna tidak mengisi data pada formulir sehingga sistem menampilkan pemberitahuan untuk melengkapi data yang masih kosong. 2. Pengguna memasukkan data akun yang tidak sesuai sehingga sistem memberikan pemberitahuan bahwa proses autentikasi gagal.	

b. *Use Case Scenario* Mengelola Divisi Internal

Tabel 5. *Use Case Scenario* Mengelola Divisi Internal

Use Case	FR-02
Nama Use Case	Mengelola Divisi Internal
Aktor	Pengguna
Deskripsi	Pengguna dapat menambahkan divisi dari internal organisasi mahasiswa.
Kondisi Awal	Pengguna telah terautentikasi.
Kondisi Akhir	Pengguna berhasil menambahkan divisi internal organisasi.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem 2. Mengakses halaman internal 4. Menekan tombol <i>Add Internal division</i> 6. Mengisi formulir 7. Menekan tombol <i>save</i> 9. Mengakses halaman <i>internal division</i> 11. Menekan tombol <i>edit internal division</i> 13. Mengisi formulir 14. Menekan tombol <i>save</i>	3. Menampilkan halaman internal 5. Menampilkan formulir 8. Menyimpan data <i>internal division</i> 10. Menampilkan halaman <i>internal division</i> 12. Menampilkan formulir 15. Menyimpan data internal division
Alur Kerja Alternatif	

1. Pengguna mengosongkan input form, sistem akan menampilkan pesan kesalahan.

c. *Use Case Scenario* Mengelola Acara

Tabel 6. *Use Case Scenario* Mengelola Acara

Use Case	FR-03
Nama Use Case	Mengelola Acara
Aktor	Pengguna
Deskripsi	Pengguna dapat menambahkan acara organisasi baru.
Kondisi Awal	Pengguna sudah terautentikasi.
Kondisi Akhir	Pengguna sudah berhasil menambahkan acara organisasi.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem 2. Mengakses halaman <i>event</i> 4. Menekan tombol <i>Add event</i> 6. Mengisi formulir 7. Menekan tombol <i>save</i> 9. Mengakses halaman detail <i>event</i> 11. Menekan tombol edit <i>event</i> 13. Mengisi formulir 14. Menekan tombol <i>save</i>	3. Menampilkan halaman <i>event</i> 5. Menampilkan formulir 8. Menyimpan data <i>event</i> 10. Menampilkan halaman detail <i>event</i> 12. Menampilkan formulir 15. Menyimpan data <i>event</i>
Alur Kerja Alternatif	

1. Pengguna mengosongkan input form, sistem akan menampilkan pesan kesalahan.

d. *Use Case Scenario* Mengelola Divisi Acara

Tabel 7. *Use Case Scenario* Mengelola Divisi Acara

Use Case	FR-04
Nama Use Case	Mengelola Divisi Acara
Aktor	Pengguna
Deskripsi	Pengguna dapat menambahkan divisi acara baru.
Kondisi Awal	Pengguna sudah terautentikasi.
Kondisi Akhir	Pengguna berhasil menambahkan divisi acara baru.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem 2. Mengakses halaman <i>event</i> 4. Mengakses halaman detail <i>event</i> 6. Menekan tombol add <i>event division</i> 8. Mengisi formulir 9. Menekan tombol <i>save</i> 11. Mengakses halaman <i>event division</i> 13. Menekan tombol edit <i>event division</i>	3. Menampilkan halaman <i>event</i> 5. Menampilkan halaman detail <i>event</i> 7. Menampilkan formulir 10. Menyimpan data <i>event division</i> 12. Menampilkan halaman <i>event division</i>

15. Mengisi formulir 16. Menekan tombol <i>save</i>	14. Menampilkan formulir 17. Menyimpan data <i>event division</i>
Alur Kerja Alternatif	
1. Pengguna mengosongkan input form, sistem akan menampilkan pesan kesalahan.	

e. *Use Case Scenario* Tambah Rapat

Tabel 8. *Use Case Scenario* Tambah Rapat

Use Case	FR-05
Nama Use Case	Tambah Rapat
Aktor	Pengguna
Deskripsi	Pengguna dapat menambahkan rapat baru.
Kondisi Awal	Pengguna sudah terautentikasi.
Kondisi Akhir	Pengguna berhasil menambah rapat baru.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem 2. Mengakses halaman tambah rapat 4. Mengisi formulir 5. Menekan tombol <i>save</i>	3. Menampilkan form input rapat 6. Memvalidasi data dan menyimpan rapat ke database 7. Menampilkan pesan rapat berhasil ditambahkan
Alur Kerja Alternatif	
1. Pengguna mengosongkan input form, sistem akan menampilkan pesan	

kesalahan.

f. *Use Case Scenario* Mengonversi Suara Menjadi Teks

Tabel 9. *Use Case Scenario* Mengonversi Suara Menjadi Teks

Use Case	FR-06
Nama Use Case	Mengonversi Suara Menjadi Teks
Aktor	Sistem
Deskripsi	Sistem dapat mengonversi suara menjadi teks secara <i>real-time</i>
Kondisi Awal	Pengguna telah memulai rapat dan fitur transkripsi telah diaktifkan.
Kondisi Akhir	Sistem berhasil menghasilkan transkripsi teks dari suara rapat secara <i>real-time</i> .
Alur Kejadian Normal	
Aktor	Sistem
1. Membuka halaman detail rapat 2. Menekan tombol mulai rekaman	3. Mengaktifkan mikrofon dan menangkap audio 4. Mengonversi suara menjadi teks secara <i>real-time</i> 5. Menyimpan teks transkripsi ke database
Alur Kerja Alternatif	
1. Jika API terganggu maka transkripsi gagal dibuat.	

g. *Use Case Scenario* Mengonversi Teks Menjadi Ringkasan

Tabel 10. *Use Case Scenario* Mengonversi Teks Menjadi Ringkasan

Use Case	FR-07
Nama Use Case	Mengonversi Teks Menjadi Ringkasan
Aktor	Sistem

Deskripsi	Sistem dapat mengonversi teks menjadi ringkasan
Kondisi Awal	Hasil transkripsi sudah selesai dibuat.
Kondisi Akhir	Sistem berhasil mengonversi teks menjadi ringkasan.
Alur Kejadian Normal	
Aktor	Sistem
1. Membuka halaman detail rapat 2. Menekan tombol buat ringkasan	3. Mengambil transkrip dan memanggil <i>Gemini API</i> 4. <i>Gemini</i> memproses dan membuat ringkasan 5. Menyimpan hasil ringkasan ke database 6. Menampilkan ringkasan di halaman detail rapat
Alur Kerja Alternatif	
1. Jika API Gemini terganggu maka ringkasan gagal dibuat	

h. *Use Case Scenario* Menampilkan Jadwal di Kalender

Tabel 11. *Use Case Scenario* Menampilkan Jadwal di Kalender

Use Case	FR-08
Nama Use Case	Menampilkan Kalender
Aktor	Sistem
Deskripsi	Sistem dapat menampilkan Kalender yang berisi data rapat.
Kondisi Awal	Data rapat sudah dibuat.
Kondisi Akhir	Sistem berhasil menampilkan Kalender.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem	

2. Mengakses halaman kalender	3. Menampilkan halaman Kalender 4. Memproses data rapat dengan <i>stored procedure</i> 5. Menampilkan daftar rapat pada kalender
Alur Kerja Alternatif	
1. Jika API terganggu maka hasil rapat gagal muncul	

i. *Use Case Scenario Logout*

Tabel 12. *Use Case Scenario Logout*

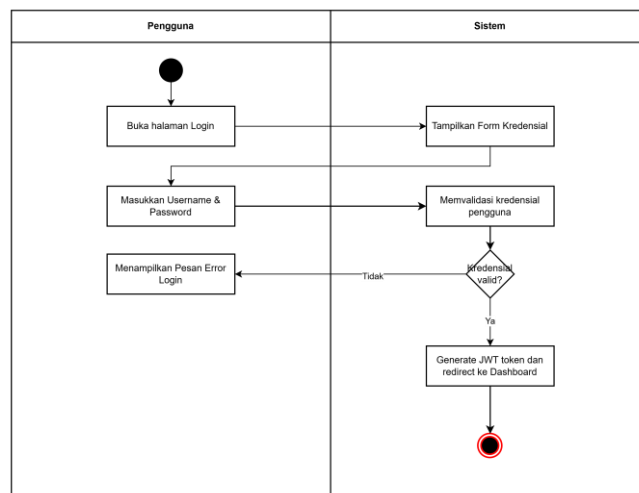
Use Case	FR-09
Nama Use Case	<i>Logout</i>
Aktor	Pengguna
Deskripsi	Pengguna dapat keluar dari sistem.
Kondisi Awal	Pengguna sudah terautentikasi.
Kondisi Akhir	Pengguna berhasil keluar dari sistem.
Alur Kejadian Normal	
Aktor	Sistem
1. Masuk ke sistem 2. Menekan tombol <i>Logout</i>	3. Menghapus JWT token dari <i>local storage</i> 4. Mengarahkan kembali ke halaman <i>Login</i>
Alur Kerja Alternatif	
1. Jika terjadi kesalaham sistem maka sistem menampilkan pesan kesalahan.	

3.2.5 Activity diagram

Activity Diagram menggambarkan alur kerja suatu proses beserta interaksi antara pengguna dan sistem hingga proses tersebut selesai. Diagram ini memperlihatkan urutan aktivitas, percabangan keputusan, serta interaksi antara pengguna dan sistem selama suatu fungsi dijalankan. Melalui *Activity Diagram*, alur kerja pada aplikasi Meenutes dapat dipahami dengan lebih jelas dan terstruktur.

1. Activity Diagram Login

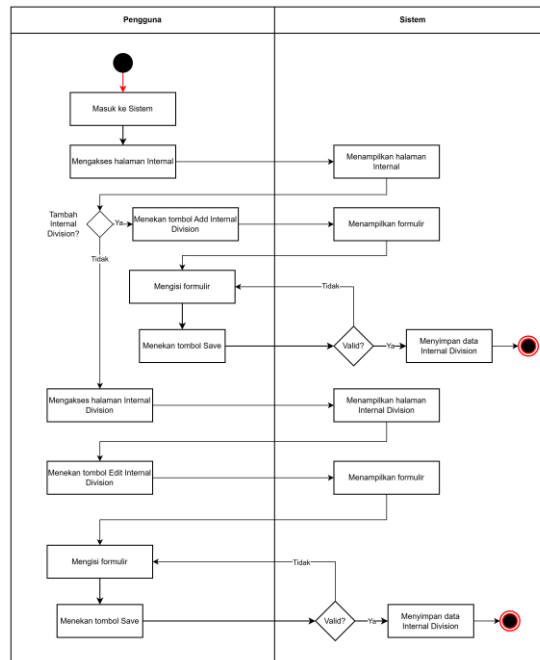
Halaman *login* berfungsi sebagai media autentikasi pengguna sebelum memperoleh akses ke aplikasi Meenutes. Pengguna diminta mengisi *username* dan *password*, kemudian menekan tombol *Login* untuk mengakses aplikasi.



Gambar 5. Activity Diagram Login

2. Activity Diagram Mengelola Divisi Internal

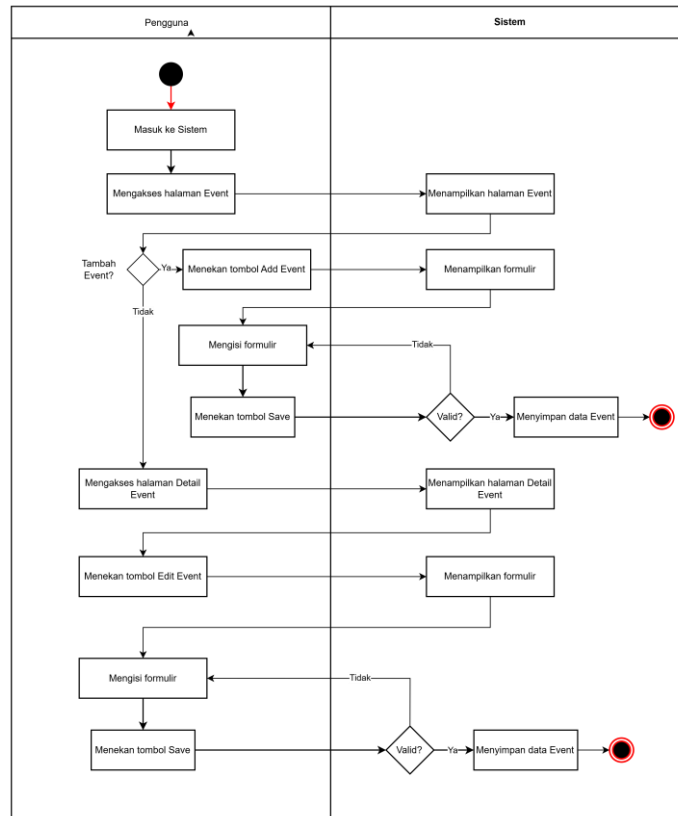
Activity Diagram Mengelola Divisi Internal menunjukkan alur pengelolaan data Divisi Internal pada aplikasi Meenutes. Melalui proses ini, pengguna dapat menambahkan maupun memperbarui data divisi menggunakan formulir yang tersedia. Sebelum data disimpan, sistem terlebih dahulu memeriksa kelengkapan dan validitas informasi yang dimasukkan.



Gambar 6. *Activity Diagram* Mengelola Divisi Internal

3. *Activity Diagram* Mengelola Acara

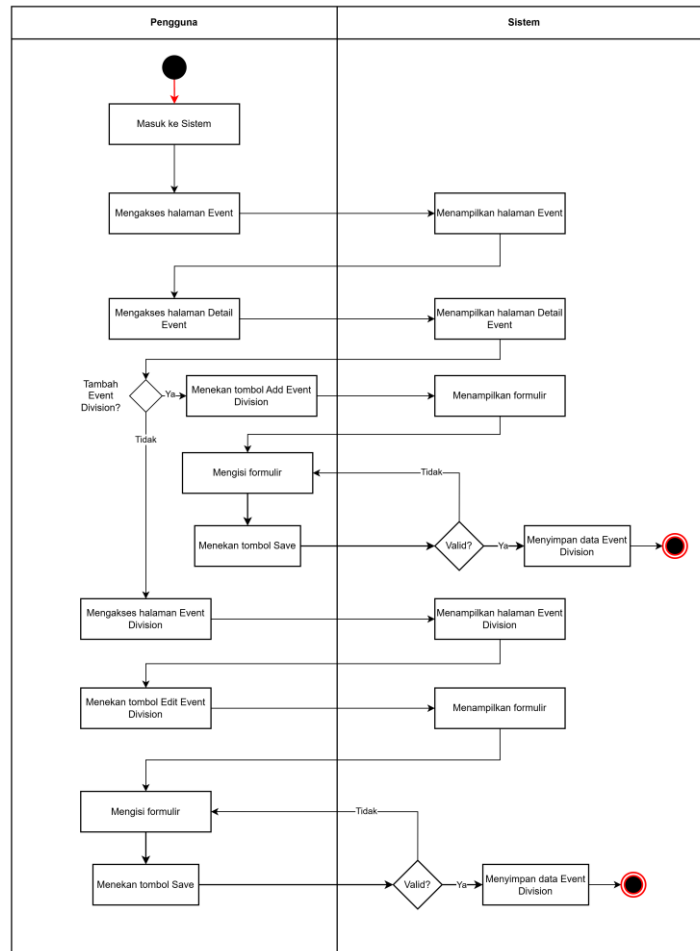
Activity Diagram Acara menggambarkan alur proses pengelolaan data *event* pada aplikasi Meenutes. Pada diagram ini pengguna dapat melakukan penambahan data event baru dan perubahan data *event* yang telah tersimpan. Sistem akan menampilkan formulir, memeriksa kelengkapan dan validitas informasi yang dimasukkan dan menyimpan data.



Gambar 7. Activity Diagram Mengelola Acara

4. Activity Diagram Mengelola Divisi Acara

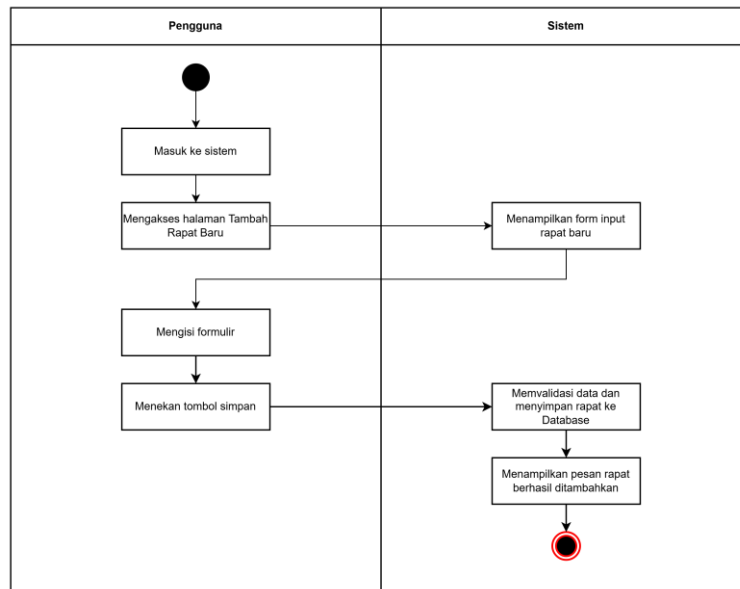
Activity Diagram Mengelola Divisi Acara menggambarkan alur proses pengelolaan data divisi pada suatu acara dalam aplikasi Meenutes. Pada diagram ini pengguna dapat menambahkan dan mengubah data divisi acara yang terkait dengan suatu acara. Sistem akan menampilkan formulir, melakukan validasi data yang dimasukkan, serta menyimpan data.



Gambar 8. *Activity Diagram Mengelola Divisi Acara*

5. *Activity Diagram Tambah Rapat*

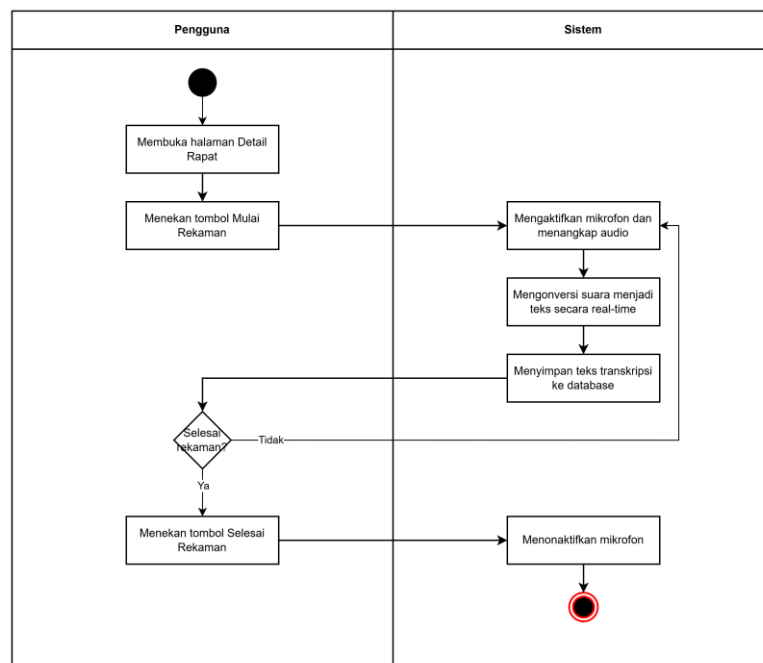
Activity Diagram Tambah Rapat menggambarkan alur proses penambahan data rapat baru pada aplikasi Meenutes. Pada proses ini pengguna mengisi formulir data rapat yang tersedia, apabila data yang dimasukkan telah memenuhi proses validasi, sistem akan menyimpannya ke basis data.



Gambar 9. *Activity Diagram* Tambah Rapat

6. *Activity Diagram* Mengonversi Suara Menjadi Teks

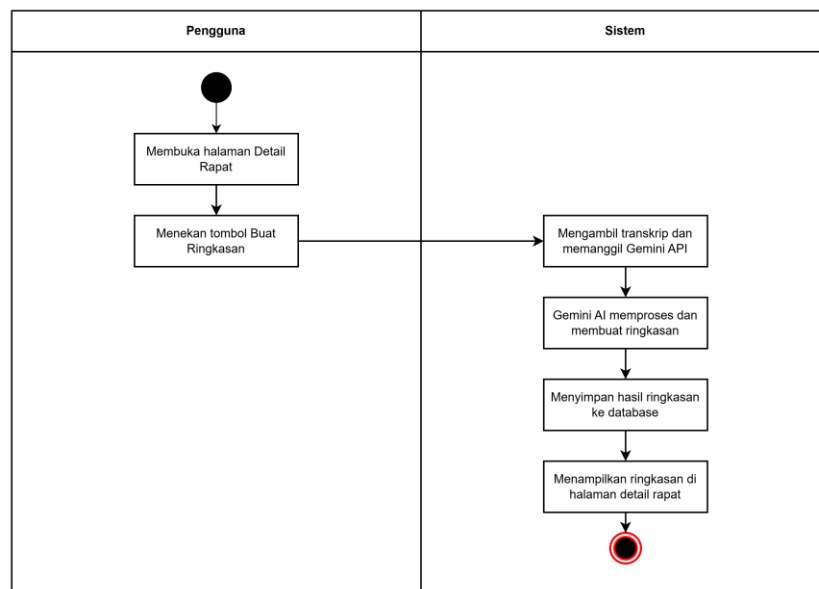
Activity Diagram Mengonversi Suara Menjadi Teks menggambarkan alur transkripsi suara pada aplikasi Meenutes. Proses dimulai dengan penerimaan input suara melalui mikrofon menggunakan *Web Speech API*, dilanjutkan dengan konversi audio menjadi teks secara *real-time*, kemudian hasil transkripsi disimpan ke dalam database sebagai dokumentasi rapat.



Gambar 10. *Activity Diagram* Mengonversi Suara Menjadi Teks

7. *Activity Diagram* Mengkonversi Teks Menjadi Ringkasan

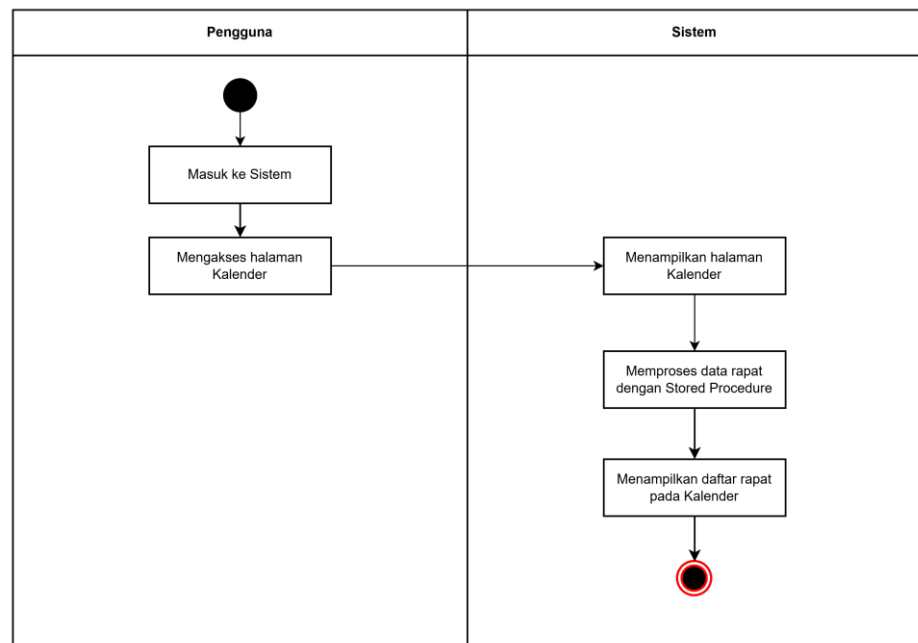
Activity Diagram Mengkonversi Teks Menjadi Ringkasan menggambarkan alur proses pembuatan ringkasan atau notulen rapat secara otomatis menggunakan *Gemini 2.5 Flash*. Sistem akan mengambil hasil transkripsi rapat yang tersedia, mengirimkannya ke layanan AI untuk diproses, kemudian menyimpan dan menampilkan hasil ringkasan kepada pengguna.



Gambar 11. *Activity Diagram* Mengkonversi Teks Menjadi Ringkasan

8. *Activity Diagram* Menampilkan Jadwal di Kalender

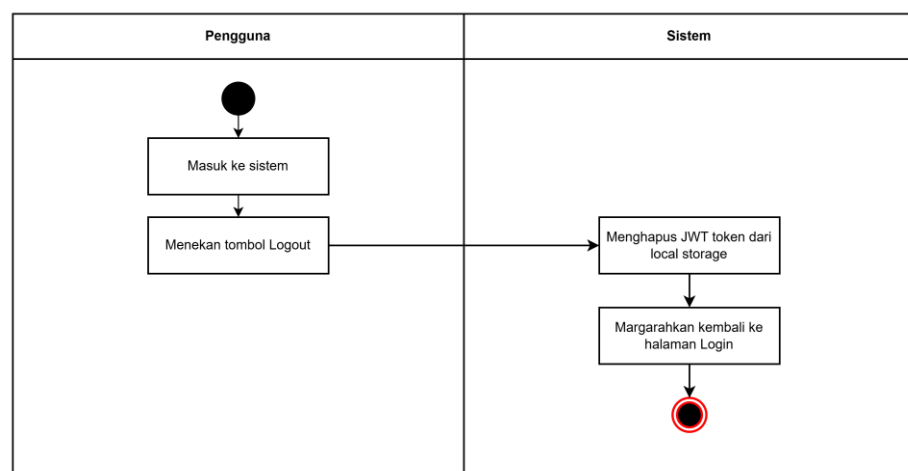
Activity Diagram Menampilkan Jadwal di Kalender menggambarkan alur proses menampilkan jadwal rapat pada fitur kalender di aplikasi Meenutes. Sistem akan mengambil data rapat yang tersimpan dalam database, memprosesnya menggunakan *stored procedure*, kemudian menampilkan daftar jadwal rapat pada halaman kalender sehingga pengguna dapat melihat jadwal rapat yang sudah tersimpan.



Gambar 12. *Activity Diagram* Menampilkan Jadwal di Kalender

9. *Activity Diagram Logout*

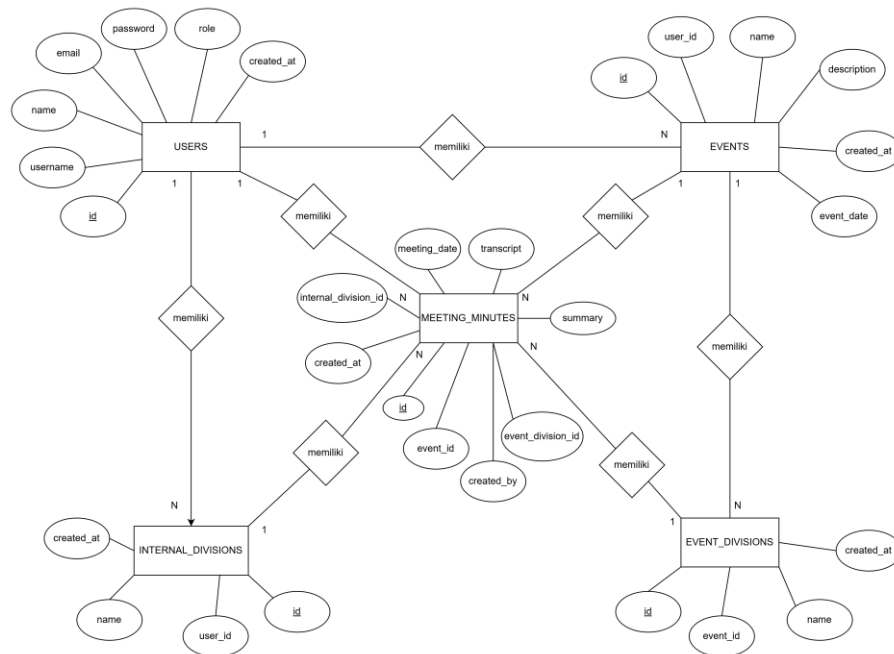
Activity Diagram Logout menjelaskan proses pengguna mengakhiri sesi pada aplikasi Meenutes. Proses logout dilakukan untuk mengakhiri sesi pengguna dengan menghapus token autentikasi yang tersimpan pada sistem sehingga pengguna tidak lagi memiliki akses ke fitur aplikasi sampai melakukan *login* kembali.



Gambar 13. *Activity Diagram Logout*

3.2.6 ER Diagram

ER Diagram menyajikan rancangan basis data yang digunakan pada aplikasi Meenutes. Diagram ini menunjukkan entitas, atribut, serta hubungan antar entitas yang digunakan untuk menyimpan data pengguna, data rapat, data divisi, dan hasil notulen rapat. ER Diagram aplikasi Meenutes ditampilkan pada Gambar 5.



Gambar 14. ER Diagram

Berdasarkan Gambar 14, terdapat lima entitas utama yang digunakan dalam sistem, yaitu *Users*, *Events*, *Internal_Divisions*, *Event_Divisions*, dan *Meeting_Minutes*. Berikut relasi yang dimiliki setiap entitas:

1. Entitas *Users*

Entitas *Users* digunakan untuk menyimpan data pengguna aplikasi. Entitas ini memiliki relasi *one-to-many* dengan entitas *Events*, *Internal_Divisions*, dan *Meeting_Minutes*. Artinya, satu pengguna dapat membuat banyak *event*, memiliki banyak divisi internal, dan membuat banyak data notulen rapat.

2. Entitas *Events*

Entitas *Events* digunakan untuk menyimpan data event atau kegiatan rapat. Entitas ini berelasi *many-to-one* dengan *Users* karena

setiap *event* dibuat oleh satu pengguna. Selain itu, entitas *Events* memiliki relasi *one-to-many* dengan *Event_Divisions* dan *Meeting_Minutes*, sehingga satu *event* dapat memiliki banyak divisi dan banyak data notulen rapat.

3. Entitas *Internal_Divisions*

Entitas *Internal_Divisions* berfungsi untuk menyimpan informasi mengenai divisi internal dalam organisasi. Entitas ini memiliki hubungan *many-to-one* dengan entitas *Users* serta hubungan *one-to-many* dengan entitas *Meeting_Minutes*. Dengan demikian, satu divisi internal dapat dikaitkan dengan banyak data notulen rapat.

4. Entitas *Event_Divisions*

Entitas *Event_Divisions* digunakan untuk menyimpan data divisi yang bertanggung jawab terhadap suatu kegiatan atau acara organisasi. Entitas ini memiliki relasi *many-to-one* dengan entitas *Users* serta relasi *one-to-many* dengan entitas *Meeting_Minutes*. Oleh karena itu, satu divisi kegiatan dapat memiliki lebih dari satu data notulen rapat.

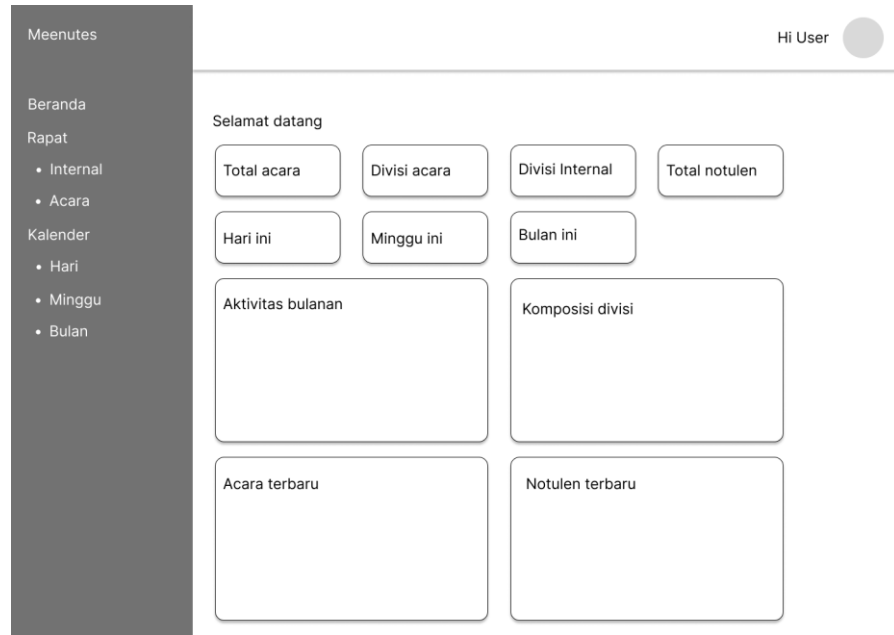
5. Entitas *Meeting_Minutes*

Entitas *Meeting_Minutes* digunakan untuk menyimpan hasil transkripsi dan ringkasan rapat. Entitas ini memiliki relasi *many-to-one* dengan *Users*, *Events*, *Internal_Divisions*, dan *Event_Divisions*. Hal ini menunjukkan bahwa setiap data notulen rapat dibuat oleh satu pengguna dan terkait dengan satu *event*, satu divisi internal, serta satu *event division*.

3.2.7 Perancangan Antarmuka (*Wireframe*)

3.2.7.1. Antarmuka *Dashboard*

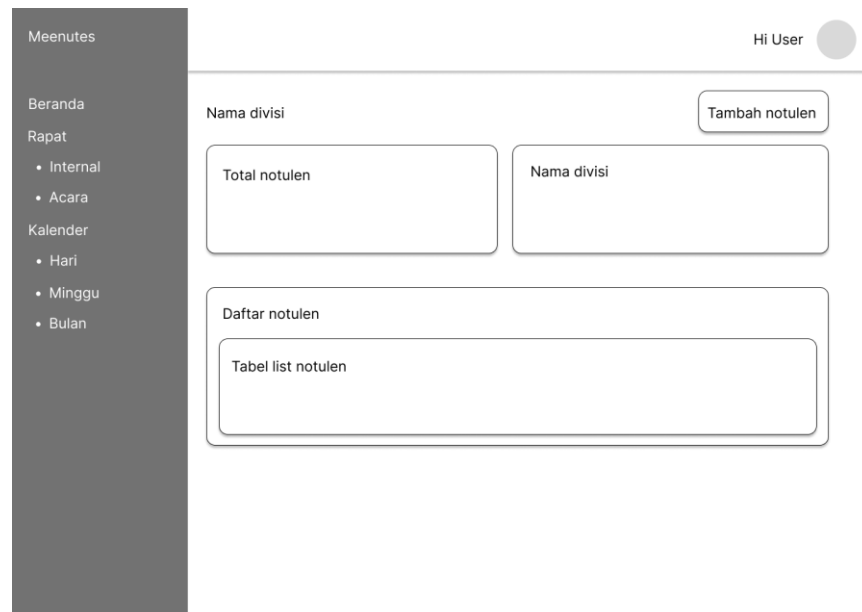
Antarmuka *dashboard* aplikasi Meenutes disajikan pada Gambar 15.



Gambar 15. Antarmuka Dashboard

3.2.7.2. Antarmuka Divisi Internal

Antarmuka divisi internal aplikasi Meenutes disajikan pada Gambar 16.



Gambar 16. Antarmuka Divisi Internal

3.2.7.3. Antarmuka Acara

Antarmuka acara aplikasi Meenutes disajikan pada Gambar 17.

The screenshot displays the Meenutes application interface. On the left is a dark sidebar with the following menu items: Meenutes, Beranda, Rapat (with sub-items: Internal, Acara), Kalender (with sub-items: Hari, Minggu, Bulan), and a list of days. The top right corner shows 'Hi User' next to a circular profile icon. The main content area is titled 'Daftar Acara' and features a 'Tambah acara' button in the top right. Below the title, there is a grid of eight rectangular boxes, each labeled 'Judul acara', arranged in three rows (three in the first two rows and two in the third). The interface uses a clean, modern design with a light gray background and rounded rectangular elements.

Gambar 17. Antarmuka Acara

3.2.7.4. Antarmuka Detail Acara

Antarmuka detail acara aplikasi Meenutes disajikan pada Gambar 18.

The screenshot displays the Meenutes application interface for event details. The sidebar and top navigation are identical to the previous screen. The main content area is titled 'Judul acara' and includes a 'Tambah notulen' button. Below the title, there are two large rectangular boxes: 'Total notulen' on the left and 'Total divisi' on the right. Further down, there is a section titled 'Daftar notulen' containing a 'Tabel list notulen' box. At the bottom, a section titled 'divisi acara' contains three smaller rectangular boxes, each labeled 'nama divisi acara'. The layout is organized and uses a consistent color scheme and rounded rectangular components.

Gambar 18. Antarmuka Detail Acara

3.2.7.5. Antarmuka Divisi Acara

Antarmuka divisi acara aplikasi Meenutes disajikan pada Gambar 19.

The screenshot shows the Meenutes application interface. On the left is a dark sidebar with the following menu items: Meenutes, Beranda, Rapat, a list containing Internal and Acara, Kalender, a list containing Hari, Minggu, and Bulan. The main content area has a header with 'Hi User' and a profile icon. Below the header, there is a 'Nama divisi' label and a 'Tambah notulen' button. The main area contains three boxes: 'Total notulen', 'Nama divisi', and 'Nama divisi acara'. Below these is a 'Daftar notulen' section containing a 'Tabel list notulen' table.

Gambar 19. Antarmuka Divisi Acara

3.2.7.6. Antarmuka Tambah Rapat

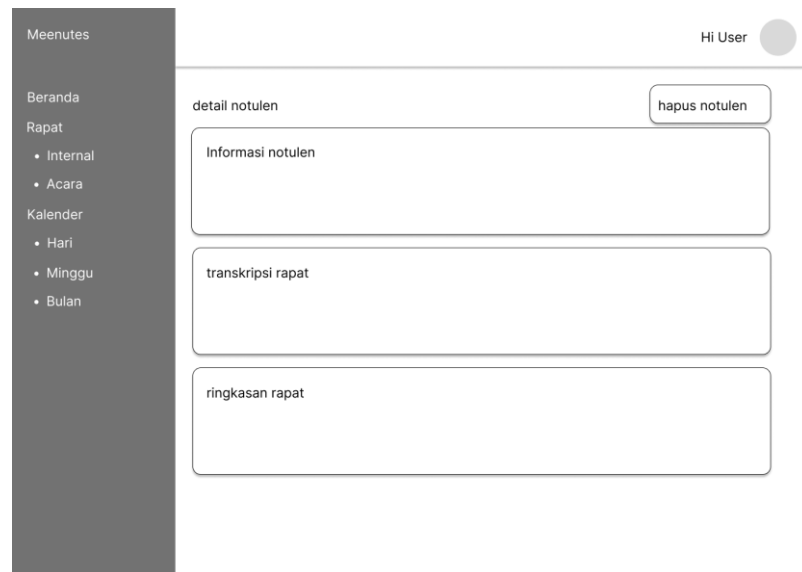
Antarmuka tambah rapat aplikasi Meenutes disajikan pada Gambar 20.

The screenshot shows the Meenutes application interface for the 'Tambah Rapat' section. The sidebar is identical to the previous screenshot. The main content area has a header with 'Hi User' and a profile icon. Below the header, there is a 'Tambah notulen' label and a 'Tambah notulen' button. The main area contains a form titled 'Informasi notulen' with two input fields: 'judul notulen' and 'tanggal rapat'. Below the form is a 'simpan notulen' button.

Gambar 20. Antarmuka Tambah Rapat

3.2.7.7. Antarmuka Detail Rapat

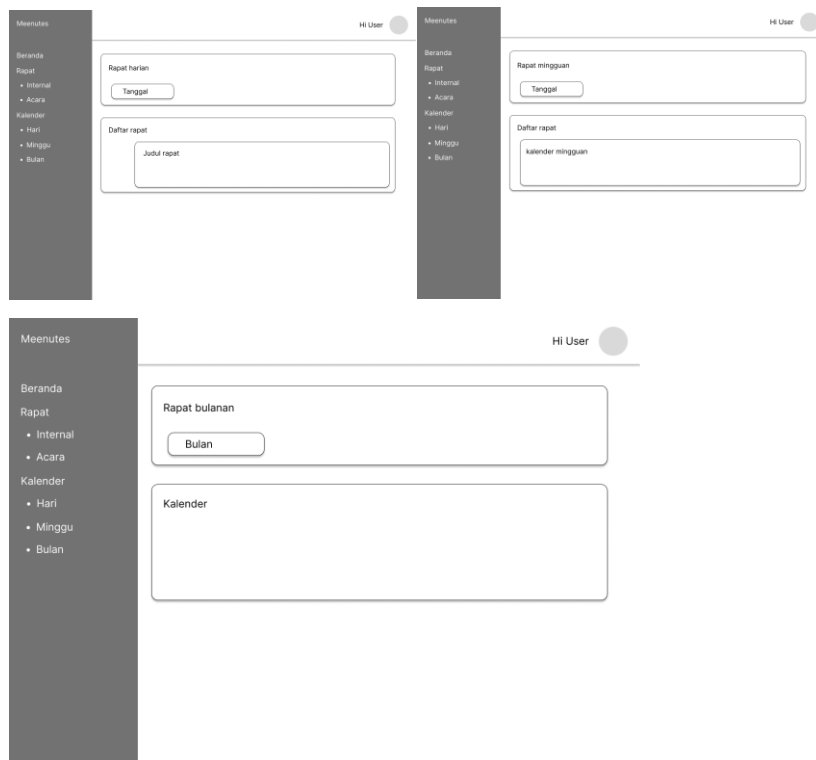
Antarmuka detail rapat aplikasi Meenutes disajikan pada Gambar 21.



Gambar 21. Antarmuka Detail Rapat

3.2.7.8. Antarmuka Kalender

Antarmuka kalender aplikasi Meenutes disajikan pada Gambar 22.



Gambar 22. Antarmuka Kalender

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

4.1.1. Instalasi Sistem

Proses instalasi sistem dilakukan untuk mempersiapkan lingkungan yang dibutuhkan agar aplikasi Meenutes dapat berjalan dengan baik. Instalasi dilakukan pada sisi *frontend*, *backend*, basis data, serta layanan pendukung yang digunakan dalam aplikasi.

Tahap pertama adalah instalasi perangkat lunak pendukung, yaitu *Node.js* untuk menjalankan aplikasi *frontend* berbasis *React.js*, *Python* untuk menjalankan backend berbasis *Flask*, serta *SQL Server Management Studio* (SSMS) sebagai sistem manajemen basis data. Setelah itu dilakukan instalasi seluruh dependensi yang dibutuhkan oleh *frontend* dan *backend* sesuai dengan konfigurasi proyek.

Tahap berikutnya adalah konfigurasi basis data *SQL Server* yang meliputi pembuatan database, tabel, dan *stored procedure* yang digunakan oleh aplikasi. Selanjutnya dilakukan konfigurasi koneksi database pada *backend* agar aplikasi dapat melakukan proses penyimpanan dan pengambilan data.

Setelah konfigurasi database selesai, dilakukan konfigurasi layanan AI yang digunakan dalam sistem. *Web Speech API* digunakan untuk proses *Speech-to-Text* secara real-time, sedangkan *Gemini 2.5 Flash* dikonfigurasi menggunakan API Key untuk menghasilkan ringkasan dan notulen rapat secara otomatis.

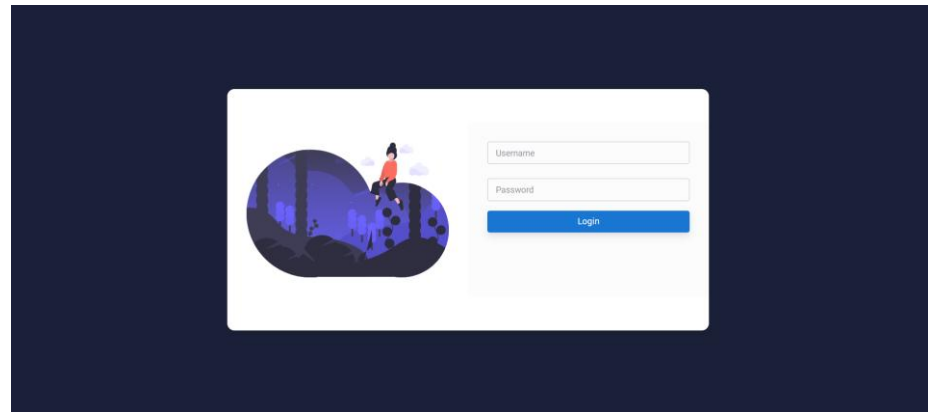
Tahap terakhir adalah menjalankan aplikasi frontend dan backend pada lingkungan pengembangan untuk memastikan seluruh komponen sistem dapat saling terhubung dan berfungsi dengan baik. Setelah proses instalasi dan konfigurasi berhasil dilakukan, aplikasi Meenutes siap digunakan untuk proses pengelolaan rapat dan pembuatan notulen otomatis.

4.1.2. Fitur Aplikasi

Berikut Penjelasan dari fitur-fitur yang tersedia pada sistem.

4.1.2.1. Login

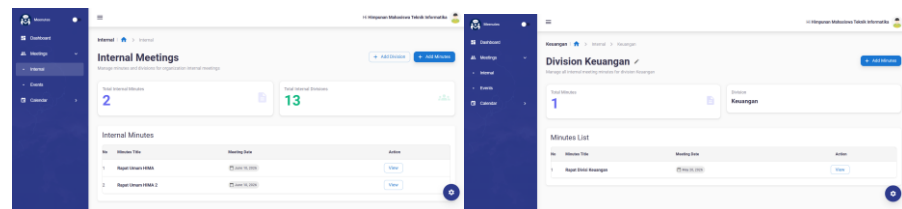
Pada fitur *login*, pengguna dapat memasukkan *username* dan *password* untuk masuk ke dalam aplikasi.



Gambar 23. Implementasi *Login*

4.1.2.2. Kelola Divisi Internal

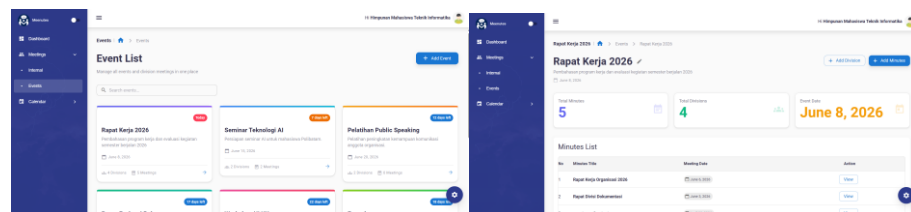
Pada fitur Kelola divisi internal, pengguna dapat mengelola divisi internal organisasi dengan cara menambahkan divisi internal baru dan mengedit data divisi.



Gambar 24. Implementasi Divisi Internal

4.1.2.3. Kelola Acara

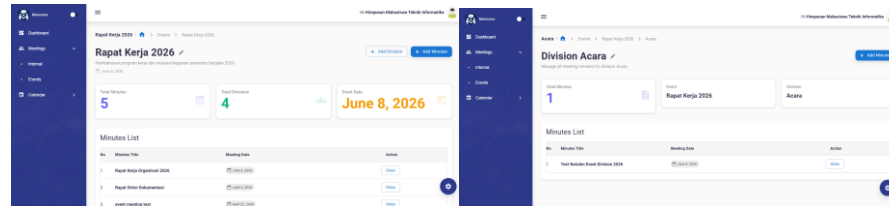
Pada fitur Kelola acara, pengguna dapat mengelola acara yang diadakan organisasi dengan menambahkan acara baru dan mengedit data acara.



Gambar 25. Implementasi Acara

4.1.2.4. Kelola Divisi Acara

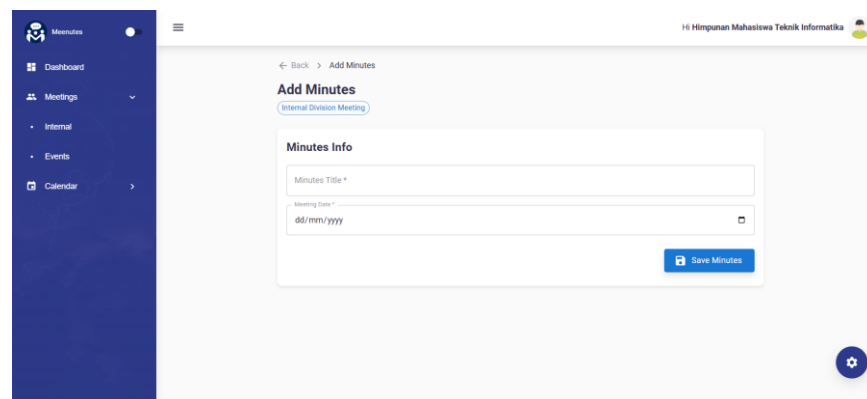
Pada fitur Kelola divisi acara, pengguna dapat mengelola divisi yang ada di dalam acara organisasi dengan menambahkan divisi baru dan mengedit data divisi.



Gambar 26. Implementasi Divisi Acara

4.1.2.5. Tambah Rapat

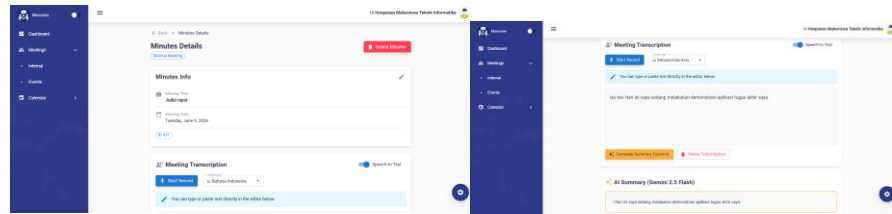
Pada fitur tambah rapat, pengguna dapat menambahkan rapat baru sesuai dengan asal rapatnya. Data sumber rapat akan tersimpan secara otomatis dengan mengambil id dari url sebelumnya, sehingga rapat internal dan rapat acara dapat dibedakan. Pengguna dapat mengisi judul rapat tanggal rapat diadakan.



Gambar 27. Implementasi Tambah Rapat

4.1.2.6. Detail Rapat

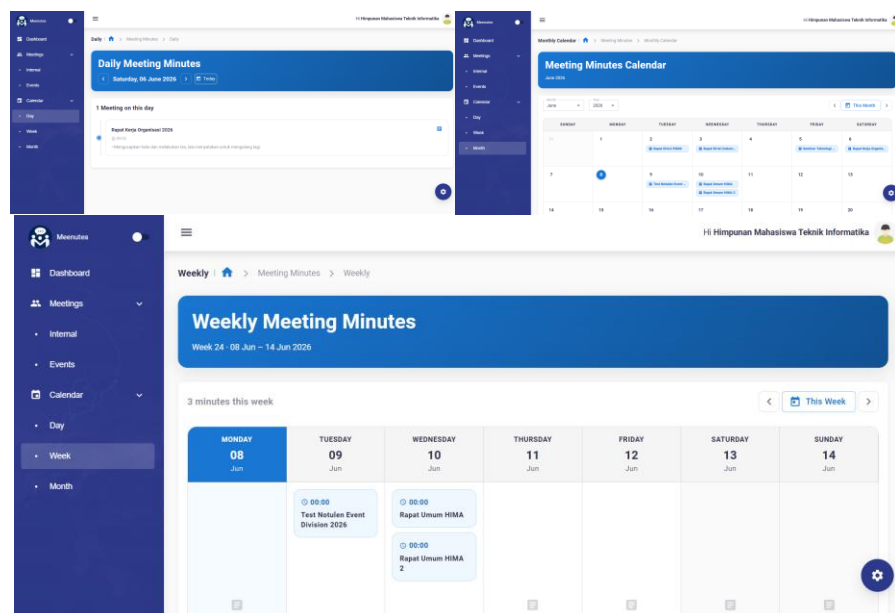
Pada fitur detail rapat, pengguna dapat melihat informasi rapat yang sudah tersimpan sebelumnya dan bisa mulai melakukan perekaman untuk mendapatkan transkripsi secara *real-time*, hasil dari transkripsi juga akan tersimpan secara otomatis secara *real-time*. Saat rapat sudah selesai, pengguna dapat menekan tombol generate *summary* untuk mendapatkan hasil ringkasan dari transkripsi.



Gambar 28. Implementasi Detail Rapat

4.1.2.7. Kalender

Pada fitur kalender, pengguna dapat melihat jadwal rapat yang sudah dibuat berdasarkan sub menu yang dipilih. Terdapat tiga kalender yang tersedia, yaitu kalender harian, kalender mingguan dan kalender bulanan.

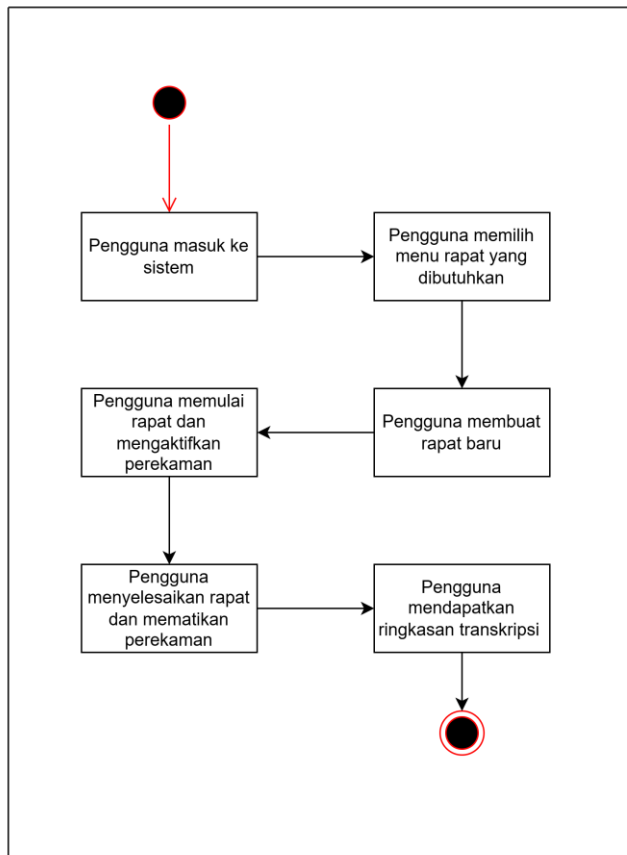


Gambar 29. Implementasi Kalender

4.1.3. Alur Kerja Aplikasi

Alur kerja aplikasi Meenutes menjelaskan urutan aktivitas pengguna saat menggunakan sistem, dimulai dari proses login hingga memperoleh hasil transkripsi dan ringkasan rapat. Setelah berhasil masuk ke aplikasi, pengguna memilih menu rapat yang akan digunakan sebelum melanjutkan ke proses berikutnya. Setelah itu, pengguna dapat membuat rapat baru dan memulai rapat dengan mengaktifkan fitur perekaman. Selama rapat berlangsung, sistem akan merekam dan mengolah percakapan yang terjadi. Setelah rapat selesai, pengguna menghentikan perekaman dan mengakhiri rapat. Sistem kemudian

memproses hasil rekaman menjadi transkripsi serta menghasilkan ringkasan rapat secara otomatis. Adapun alur kerja aplikasi dapat dilihat pada Gambar 30.



Gambar 30. Alur Kerja Aplikasi

4.2. Pengujian

4.2.1. Pengujian Fungsional

Tabel 13. Pengujian Fungsional

Kebutuhan Fungsional	Total Test case	Persentase Kelulusan
Mengelola divisi acara	5	100%
Mengelola acara	5	100%
Mengelola divisi internal	5	100%
Menampilkan kalender	3	100%

<i>Login</i>	3	100%
Tambah rapat	5	100%
<i>Logout</i>	2	100%
Mengkonversi suara menjadi teks	5	100%
Mengkonversi teks menjadi ringkasan	5	100%

Evaluasi fungsional aplikasi dilakukan menggunakan metode *Black Box Testing* dengan mengacu pada 38 *test case* yang telah disusun. Seluruh skenario pengujian berhasil dilaksanakan tanpa ditemukan *bug* maupun kegagalan fungsi. Temuan tersebut mengindikasikan bahwa setiap fitur telah diimplementasikan sesuai dengan kebutuhan fungsional serta spesifikasi sistem.

4.2.2. UAT

Tabel 14. *User Acceptance Testing* (UAT)

No	Pertanyaan	SS	S	N	ST	STS
1.	Saya berpikir akan sering menggunakan aplikasi ini.	23	3	0	0	0
2.	Saya berpikir akan sering menggunakan aplikasi ini.	0	0	0	7	18
3.	Saya merasa aplikasi ini mudah digunakan.	16	9	0	0	0
4.	Saya memerlukan bantuan orang lain untuk menggunakan aplikasi ini.	0	0	0	5	20
5.	Saya merasa fitur-fitur dalam aplikasi ini terintegrasi dengan baik.	19	6	0	0	0
6.	Saya merasa terdapat banyak ketidakkonsistenan dalam aplikasi ini.	0	0	0	8	17
7.	Saya merasa sebagian besar pengguna dapat mempelajari penggunaan aplikasi ini dengan cepat.	22	3	0	0	0
8.	Saya merasa aplikasi ini membingungkan untuk digunakan.	0	0	0	8	17

9.	Saya merasa percaya diri ketika menggunakan aplikasi ini.	16	9	0	0	0
10.	Saya perlu mempelajari banyak hal sebelum dapat menggunakan aplikasi ini.	0	0	0	8	17

Berdasarkan hasil *User Acceptance Testing* (UAT) menggunakan metode *System Usability Scale* (SUS) terhadap 25 responden, diperoleh skor SUS rata-rata sebesar 93,5. Nilai tersebut menunjukkan bahwa aplikasi Meenutes memiliki tingkat kegunaan yang sangat baik. Berdasarkan nilai tersebut maka dapat aplikasi Meenutes dapat dikategorikan *Excellent*, memiliki tingkat penerimaan *Acceptable* dan memperoleh *Grade A+*.

Berdasarkan hasil pengujian, dapat disimpulkan bahwa aplikasi Meenutes telah memenuhi aspek kemudahan penggunaan serta mendapatkan penerimaan yang baik dari pengguna. Selain itu, aplikasi mampu mendukung kemudahan penggunaan, meningkatkan efisiensi, serta memberikan pengalaman yang memuaskan dalam proses pembuatan notulen rapat.

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, serta pengujian aplikasi yang telah dilakukan, diperoleh beberapa kesimpulan sebagai berikut:

1. Aplikasi manajemen rapat berbasis web (Meenutes) berhasil dirancang dan dibangun untuk membantu pengelolaan kegiatan rapat pada organisasi mahasiswa, mulai dari pengelolaan data acara, divisi, penjadwalan rapat, hingga dokumentasi hasil rapat.
2. Teknologi *Web Speech API* berhasil diintegrasikan ke dalam aplikasi untuk mengubah percakapan rapat menjadi teks secara *real-time*, sedangkan *Gemini 2.5 Flash* berhasil digunakan untuk menghasilkan ringkasan dan notulen rapat secara otomatis berdasarkan hasil transkripsi.
3. Aplikasi Meenutes mampu membantu proses dokumentasi, pengarsipan, dan pencarian hasil rapat secara lebih efektif dibandingkan metode pencatatan manual karena seluruh data rapat tersimpan secara terpusat dalam sistem.
4. Pengujian menggunakan metode *System Usability Scale* (SUS) terhadap 25 responden menghasilkan skor rata-rata sebesar 93,5. Hasil tersebut mengklasifikasikan aplikasi ke dalam kategori *Excellent*, *Grade A+*, dan *Acceptable*. Dengan demikian, aplikasi dinilai memiliki tingkat *usability* yang tinggi dan layak digunakan oleh pengguna.

5.2 Saran

Sebagai masukan bagi pengembangan berikutnya, penelitian ini mengajukan beberapa rekomendasi sebagai berikut:

1. Pengembangan aplikasi Meenutes selanjutnya dapat dilakukan dengan menyediakan versi mobile sehingga pengguna dapat mengakses dan mengelola rapat dengan lebih fleksibel melalui perangkat *smartphone*.
2. Sistem dapat dikembangkan dengan menambahkan fitur identifikasi pembicara (*speaker diarization*) agar hasil transkripsi rapat dapat menunjukkan identitas pembicara secara otomatis.

3. Perlu dilakukan peningkatan pada fitur *Speech-to-Text* untuk meningkatkan akurasi transkripsi, terutama pada kondisi lingkungan yang bising, variasi aksen pengguna, dan kualitas koneksi internet yang tidak stabil.

DAFTAR PUSTAKA

- Al-Saqqa, S., Sawalha, S., & AbdelNabi, H. (2020). Agile software development: Methodologies and trends. *International Journal of Interactive Mobile Technologies*, 14(11).
- Handayani, H., Ayulya, A. M., Faizah, K. U., Wulan, D., & Rozan, M. F. (2023). Perancangan Sistem Informasi Inventory Barang Berbasis Web Menggunakan Metode Agile Software Development. *Jurnal Testing Dan Implementasi Sistem Informasi*, 1(1), 29–40. <https://doi.org/10.55583/jtisi.v1i1.324>
- Karo Karo, I. M., Dewi, S., & Perdana, A. (2024). Implementasi Text Summarization Pada Review Aplikasi Digital Library System Menggunakan Metode Maximum Marginal Relevance. *JEKIN - Jurnal Teknik Informatika*, 4(1), 25–31. <https://doi.org/10.58794/jekin.v4i1.671>
- Lubis, N., Siambaton, M. Z., & Aulia, R. (2024). Implementasi Algoritma Deep Learning pada Aplikasi Speech to Text Online dengan Metode Recurrent Neural Network (RNN). *Sudo Jurnal Teknik Informatika*, 3(3), 113–126. <https://doi.org/10.56211/sudo.v3i3.583>
- Purba, E. H., Supriyanti, D., & Yulianjani, A. (2023). Sistem Informasi E-Meeting Berbasis Web Sebagai Pengelolaan Rapat Terpadu. *Journal Cerita*, 9(2), 202–210. <https://doi.org/10.33050/cerita.v9i2.2879>
- Rifaldi, R., & Cahyono, A. B. (2024). Pengembangan Fitur Speech Recognition Pada Aplikasi Notulensi Menggunakan Whisper AI. *INNOVATIVE: Journal Of Social Science Research*, 4, 8555–8566.
- Hastryandi, H., Wahyuni, S., & Syahnaz, E. (2024). Rancang Bangun Aplikasi Manajemen Rapat Politeknik Negeri Sambas Berbasis Web. *Prosiding Seminar Nasional Dan Teknologi*, 15(1), 369–378. <https://doi.org/10.36499/psnst.v15i1.14868>
- Akrianto, M. I., Nugroho, A., & Hidayat, M. M. (2025). INTEGRASI GEMINI 2 . 5 FLASH API PADA FRAMEWORK NEXT . JS UNTUK OTOMASI TRANSFORMASI VISUAL PRODUK E-COMMERCE. *JITET (Jurnal Informatika Dan Teknik Elektro Terapan)*, 14(2), 678–684. <https://doi.org/10.23960/jitet.v14i2.9394>
- Aritonang, G. R. M. L., Hasan, M. A., Ridwan, M. K., Iman, M. N., Silalahi, R. C. P., & Sipayung, R. S. (2025). Optimasi Query Sql Server Dengan Teknik Indexing Dan Performance Monitoring. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(2), 3094–3099. <https://mail.ejournal.itn.ac.id/index.php/jati/article/view/13179/7375>
- Ferdiansyah, D., & Aditya, C. S. K. (2024). Implementasi Automatic Speech Recognition Bacaan Al-Qur'an Menggunakan Metode Wav2Vec 2.0 dan OpenAI-Whisper. *Jurnal Teknik Elektro Dan Komputer TRIAC*, 11(1), 0–5.
- Asrin, F. (2024). Analisis Pengujian Menggunakan User Acceptance Test (Uat) Pada Aplikasi Manajemen Notulensi Rapat Bappeda Kota Pontianak. *JSR: Jaringan Sistem Informasi Robotik*, 8(1), 34–41. <http://ojsamik.amikmitragama.ac.id/index.php/js/article/view/340%0Ahttp://ojsamik.amikmitragama.ac.id/index.php/js/article/download/340/240>

- Pertiwi, A. D., Septian, R. N., Ashifa, R., & Prihantini, P. (2021). Peran Organisasi Kemahasiswaan dalam Membangun Karakter: Urgensi Organisasi Kemahasiswaan pada Generasi Digital. *Aulad: Journal on Early Childhood*, 4(3), 107–115. <https://doi.org/10.31004/aulad.v4i3.202>
- Ngantung, R. K., & Pakereng, M. A. I. (2021). Model Pengembangan Sistem Informasi Akademik Berbasis User Centered Design Menerapkan Framework Flask Python. *Jurnal Media Informatika Budidarma*, 5(3), 1052. <https://doi.org/10.30865/mib.v5i3.3054>
- Swapurba, G., & Pratama, I. (2023). Implementasi Voice To Text Pada Invoice Checking Berbasis Web. *Intechno Journal (Information Technology Journal)*, 5(2), 65–74. <https://doi.org/10.24076/intechnojournal.2023v5i2.1391>
- Jumiyati, & Azlina, Y. (2021). *Rapat: Komunikasi Bisnis yang Efektif*. 6(3), 84–97.
- Tri Sulistyorini, Sova, E., & Ramadhan, R. (2022). Pemantauan Kasus Penyebaran Covid-19 Berbasis Website Menggunakan Framework React Js Dan Api. *Jurnal Ilmiah Multidisiplin*, 1(04), 01–13. <https://doi.org/10.56127/jukim.v1i04.137>

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement
[Dokumen Proses Pengumpulan Requirement](#)

Lampiran B. Link Produk
[Link Github Frontend](#)
[Link Github Backend](#)

Lampiran C. Dokumen Pengujian
[Dokumen Pengujian](#)