TRANSFORMING UNIVERSITY MENTAL HEALTH SUPPORT: AN AGENTIC AI FRAMEWORK FOR PROACTIVE INTERVENTION AND RESOURCE MANAGEMENT

BACHELOR'S THESIS

THE SUSTAINABLE DEVELOPMENT GOALS
Good Health and Well-being
Quality Education
Peace, Justice, and Strong Institutions

Written by:

GIGA HIDJRIKA AURA ADKHY
21/479228/TK/52833

INFORMATION ENGINEERING PROGRAM

DEPARTMENT OF ELECTRICAL AND INFORMATION ENGINEERING
FACULTY OF ENGINEERING UNIVERSITAS GADJAH MADA
YOGYAKARTA

2025

ENDORSEMENT PAGE

TRANSFORMING UNIVERSITY MENTAL HEALTH SUPPORT: AN AGENTIC AI FRAMEWORK FOR PROACTIVE INTERVENTION AND RESOURCE MANAGEMENT

THESIS

Proposed as A Requirement to Obtain
Undergraduate Degree (*Sarjana Teknik*)
in Department of Electrical and Information Engineering
Faculty of Engineering
Universitas Gadjah Mada

Written by:

GIGA HIDJRIKA AURA ADKHY
21/479228/TK/52833

Has been approved and endorsed

on . . . . . .

Supervisor I	Supervisor II
Dr. Bimo Sunarfri Hantono, S.T., M.Eng.	**Dr. Ir. Guntur Dharma Putra, S.T., M.Sc.**
NIP 197701312002121003	**NIP 199104132024061001**

STATEMENT

Saya yang bertanda tangan di bawah ini :

Name	: Giga Hidjrika Aura Adkhy
NIM	: 21/479228/TK/52833
Tahun terdaftar	: 2021
Program	: Sarjana Teknik
Major	: Teknologi Informasi
Faculty	: Fakultas Teknik, Universitas Gadjah Mada

Menyatakan bahwa dalam dokumen ilmiah Skripsi ini tidak terdapat bagian dari karya ilmiah lain yang telah diajukan untuk memperoleh gelar akademik di suatu lembaga Pendidikan Tinggi, dan juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang/lembaga lain, kecuali yang secara tertulis disitasi dalam dokumen ini dan disebutkan sumbernya secara lengkap dalam daftar pustaka.

Dengan demikian saya menyatakan bahwa dokumen ilmiah ini bebas dari unsur-unsur plagiasi dan apabila dokumen ilmiah Skripsi ini di kemudian hari terbukti merupakan plagiasi dari hasil karya penulis lain dan/atau dengan sengaja mengajukan karya atau pendapat yang merupakan hasil karya penulis lain, maka penulis bersedia menerima sanksi akademik dan/atau sanksi hukum yang berlaku.

Yogyakarta, 20-11-2025

Giga Hidjrika Aura Adkhy
NIM 21/479228/TK/52833

PAGE OF DEDICATION

This thesis is lovingly dedicated to my parents for their endless love and support;
to my partner, Virna Amrita, for her patience and encouragement;
and to my friends, who made this journey brighter.

PREFACE

Praise be to Allah SWT for His abundant blessings, grace, and guidance, enabling the completion of this thesis. The long journey of completing this research has been filled with twists and turns, challenges, and invaluable lessons. Throughout the preparation of this thesis, I have received tremendous guidance, assistance, and support from various parties. Therefore, I would like to express my sincere gratitude to:

1. Prof. Ir. Hanung Adi Nugroho, S.T., M.E., Ph.D., IPM., SMIEEE., as the Head of the Department of Electrical and Information Engineering, for their guidance, strategic vision, and for cultivating an academic atmosphere that enabled this research to flourish.
2. Ir. Lesnanto Multa Putranto, S.T., M.Eng., Ph.D., IPM., SMIEEE., as the Vice Head of the Department of Electrical and Information Engineering, for their encouragement, administrative support, and dedication to student development, all of which greatly enriched my academic experience.
3. Dr. Bimo Sunarfri Hantono, S.T., M.Eng., as my first thesis advisor. Thank you for the direction, guidance, and patience in steering this research to completion. Every discussion and feedback has shaped a clearer research direction amidst the complexity of ever-evolving innovation.
4. Dr. Ir. Guntur Dharma Putra, S.T., M.Sc., as my second thesis advisor. Thank you for the guidance, support, and valuable insights throughout the completion of this thesis.
5. My beloved parents, who have provided financial support and endless prayers throughout my education at Universitas Gadjah Mada. Without their sacrifices and trust, this achievement would never have been realized.
6. My partner, Virna Amrita who has always accompanied me through Discord during long nights of struggle, providing encouragement when motivation began to fade, and being a source of comfort in times of joy and hardship. Your presence has been a light in the darkness, especially when facing challenges from the volatility of the crypto world that influenced this research journey.
7. My siblings, Wisda and Ria, who have consistently prayed for and supported every step of my academic journey.
8. My close friends–Azfar, Ariq, Zakong, Diamond, Arif, Ditya, Nando, Aufa, Difta, Akhdan, Evan, Aji and others who have been companions in arms during college, sharing joys and sorrows, and serving as an incredible support system. You are my second family who made my days on campus more meaningful.
9. PT INA17, who has shown interest in and provided support for this project, validating that the research conducted has real applicative value in the industry.
10. The Sumbu Labs team–Maulana, Dzikran, Farhan, Azfar, and Virna–who have helped me in working through one of the biggest projects of my life while doing this thesis in parallel. Our collaboration in developing CAR-dano (now Ototentik) has been an unforgettable experience. You are not just colleagues, but partners in making dreams come true.

11. All parties involved in the EDU Chain Global Hackathon: Semester 3, where the UGM-AICare project successfully won 6000 USD. This achievement is proof that hard work and innovation can be rewarded, even though it sometimes became a beautiful "distraction" from the focus of thesis writing.

The journey of completing this thesis has taught me that innovation does not always follow a straight path. There are times when we are tempted to branch out, explore new ideas, and even get "lost" in hackathon after hackathon. However, each of these experiences has enriched my understanding and broadened my perspective on how technology can make a real impact on society. There were days when I felt lonely working from home, but the support from my loved ones made every challenge feel lighter.

The motivation behind choosing AI agents as the focus of my bachelor's thesis stems from a deeply personal mission: to elevate the standard of mental health services at UGM. Throughout my time as a student, I witnessed firsthand, both in myself and in my peers, how difficult it is to seek help for mental health concerns. We are often too busy, or we simply fail to prioritize our mental wellbeing until it becomes critical. Many students struggle in silence, not because help isn't available, but because the barriers to access feel too high. This realization drove me to create Aika, the AI agent in UGM-AICare, designed to provide proactive interventions and regular check-ups that meet students where they are, when they need it most.

This vision was significant enough for me to embrace the ambitious scope of this work, even knowing it would take longer to complete than a typical bachelor's thesis. I only wish the best for UGM, just as my parents and friends have always wished the best for me. This university has been the place where I met remarkable people who humbled me, challenged my perspectives, and grounded me in reality. It shaped not just my academic journey, but my character and values. If this research can contribute to making mental health support more accessible and effective for future generations of UGM students, then every late night, every challenge, and every moment of uncertainty will have been worth it.

Finally, I hope that this thesis can contribute to the advancement of knowledge, particularly in the fields of artificial intelligence and healthcare technology, and can serve as inspiration for future research. May this work bring benefits to us all, aamiin.

Yogyakarta, November 12, 2025

Giga Hidjrika Aura Adkhy

CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE AND ABBREVIATION

Abbreviations

AI	Artificial Intelligence
BDI	Belief-Desire-Intention
CBT	Cognitive Behavioral Therapy
CMA	Case Management Agent
DDDM	Data-Driven Decision-Making
DSR	Design Science Research
FNR	False Negative Rate
HEI	Higher Education Institution
HITL	Human-in-the-Loop
IA	Insights Agent
LCEL	LangChain Expression Language
LLM	Large Language Model
LMS	Learning Management System
MAS	Multi-Agent System
MoE	Mixture-of-Experts
ORM	Object-Relational Mapper
PET	Privacy-Enhancing Technology
RBAC	Role-Based Access Control
RCT	Randomized Controlled Trial
ReAct	Reasoning and Acting
RNN	Recurrent Neural Network
RQ	Research Question
STA	Safety Triage Agent
TCA	Therapeutic Coach Agent
UGM	Universitas Gadjah Mada

Nomenclature

A_t	=	Therapeutic response from the TCA at time t.
a_t	=	Action generated by a LangChain agent at step t.
Bel_t	=	The set of an agent's beliefs at time t.
C_t	=	The state of a clinical case for the CMA at time t.
$\mathcal{C}$	=	The corpus of crisis patterns used by the STA.
d_k	=	The dimension of the key vectors in a self-attention mechanism.
Des_t	=	The set of an agent's desires or goals at time t.

f_{AGENT}	=	The core decision function of a specified agent (STA, TCA, CMA, IA).
G	=	The initial goal or objective for a LangChain agent.
H_{t-1}	=	The history of a conversation up to time $t-1$.
H_t	=	The history of actions and observations for an agent up to step t.
$\mathcal{I}$	=	The library of evidence-based interventions available to the TCA.
Int_t	=	The set of an agent's intentions or committed plans at time t.
K	=	The Key matrix in the self-attention mechanism.
$\mathcal{K}$	=	The set of privacy constraints (e.g., k-anonymity) for the IA.
k	=	The anonymity threshold in k-anonymity.
M_t	=	A message from a student at time t.
$\mathcal{M}$	=	A set of anonymized messages used by the IA.
μ	=	An aggregate metric computed by the IA.
N	=	The set of nodes in a LangGraph state graph.
o_t	=	An observation received by a LangChain agent at step t.
$p(\cdot)$	=	The conditional probability distribution of a Large Language Model.
Q	=	The Query matrix in the self-attention mechanism.
q	=	An analytical query provided to the IA.
R_t	=	The risk level assessed by the STA at time t.
R_{avail}	=	The set of available resources for the CMA.
ρ	=	The routing function that determines the next node in a LangGraph.
S_t	=	The state of the LangGraph at time step t.
ΔS_i	=	The state update produced by node i in a LangGraph.
th_t	=	A thought or reasoning step generated by a LangChain agent at step t.
V	=	The Value matrix in the self-attention mechanism.
W	=	Represents the various weight matrices within a Transformer model.

INTISARI

Institusi Pendidikan Tinggi menghadapi peningkatan permintaan dukungan kesejahteraan mahasiswa namun masih bergantung pada kerangka kerja konseling reaktif yang seringkali gagal menjangkau mahasiswa sebelum krisis memuncak. Skripsi ini mengusulkan dan mengevaluasi kerangka kerja AI agentic proaktif yang dirancang untuk menjembatani kesenjangan *insight-to-action* dengan memungkinkan intervensi dini dan manajemen sumber daya berbasis data. Kami memperkenalkan *Safety Agent Suite*, arsitektur multi-agen terpisah yang mendistribusikan tanggung jawab klinis dan operasional kepada agen khusus di bawah pengawasan manusia. Sistem ini mencakup: (i) **Aika**, orkestrator Meta-Agent yang menyediakan antarmuka pengguna terpadu dan melakukan penyaringan risiko Tingkat 1 segera; (ii) **Safety Triage Agent (STA)** untuk analisis risiko percakapan Tingkat 2 yang komprehensif; (iii) **Therapeutic Coach Agent (TCA)** yang memberikan intervensi mikro terapeutik berbasis Cognitive Behavioral Therapy (CBT); (iv) **Case Management Agent (CMA)** untuk koordinasi operasional; dan (v) **Insights Agent (IA)** untuk analitik manajemen sumber daya yang menjaga privasi. Untuk menyeimbangkan responsivitas dengan kedalaman analisis, kami menggunakan arsitektur pemantauan risiko dua tingkat yang menggabungkan penyaringan segera dengan analisis percakapan mendalam untuk memungkinkan intervensi dini. Sistem multi-agen dibangun dengan LangGraph dan mencakup perlindungan untuk penggunaan alat, redaksi, dan kemampuan audit.

Kami membangun prototipe fungsional dalam platform UGM-AICare dan melakukan evaluasi berbasis skenario yang menitikberatkan secara eksklusif pada kinerja arsitektur agen: sensitivitas dan *False Negative Rate* (FNR) triase pada skenario krisis sintetis; keandalan orkestrasi melalui tingkat keberhasilan pemanggilan fungsi dan transisi state; latensi ujung-ke-ujung; verifikasi kepatuhan privasi; serta kualitas coaching melalui rubrik kepatuhan CBT dengan penilaian ahli dan validasi LLM. **Skripsi ini berfokus secara spesifik pada desain dan evaluasi kerangka multi-agen itu sendiri**—agen spesialis berbasis BDI, lapisan orkestrasi Aika, dan perilaku kolektif mereka dalam konteks percakapan kritis keselamatan. Desain basis data, komponen antarmuka pengguna, dan infrastruktur deployment didokumentasikan sebagai konteks implementasi namun bukan subjek evaluasi formal. Hasil menunjukkan kelayakan teknis keselamatan proaktif, orkestrasi agen yang andal, dan dukungan yang menjaga privasi, mengonfirmasi kapasitas sistem untuk menutup kesenjangan *insight-to-action* di bawah pengawasan manusia. Kami membahas pertimbangan etis, prinsip *privacy by design*, keterbatasan penelitian, dan kebutuhan studi klinis lapangan di masa depan dengan pengguna riil.

Kata kunci: Sistem Multi-Agen; Arsitektur BDI; Orkestrasi Agen; Triase Keselamatan; LangGraph; Human-in-the-Loop; Kesejahteraan Mahasiswa; Evaluasi Berbasis Skenario

ABSTRACT

Higher Education Institutions face rising demand for student well-being support while relying on reactive counseling frameworks that often fail to reach students before crises escalate. This thesis proposes and evaluates a proactive, agentic AI framework designed to bridge the critical 'insight-to-action' gap by enabling early intervention and data-driven resource management. We introduce the *Safety Agent Suite*, a decoupled multi-agent architecture that distributes clinical and operational responsibilities to specialized agents under human oversight. The system features: (i) **Aika**, a Meta-Agent orchestrator that provides a unified user interface and performs immediate Tier 1 risk screening; (ii) a **Safety Triage Agent (STA)** for comprehensive Tier 2 conversational risk analysis; (iii) a **Therapeutic Coach Agent (TCA)** delivering Cognitive Behavioral Therapy (CBT)-based micro-interventions; (iv) a **Case Management Agent (CMA)** for operational coordination; and (v) an **Insights Agent (IA)** for privacy-preserving resource management analytics. To balance responsiveness with depth, we employ a two-tier risk monitoring architecture that combines immediate screening with deep conversational analysis to enable early intervention. The multi-agent system is built with LangGraph and includes guardrails for tool use, redaction, and auditability.

We implement a functional prototype within the UGM-AICare platform and conduct scenario-based evaluations focused exclusively on agent architecture performance: triage sensitivity and False Negative Rate (FNR) on synthetic crisis scenarios; orchestration reliability via tool-call success and state transition behavior; end-to-end latency; privacy compliance verification; and coaching quality via CBT adherence rubrics with expert assessment and LLM validation. **This thesis focuses specifically on the design and evaluation of the multi-agent framework itself**—the BDI-based specialist agents, Aika orchestration layer, and their collective behavior in safety-critical conversational contexts. Database design, user interface components, and deployment infrastructure are documented as implementation context but are not subjects of formal evaluation. Results demonstrate the technical feasibility of proactive safety, reliable agent orchestration, and privacy-preserving support, confirming the system's capacity to close the insight-to-action gap under human-in-the-loop supervision. We discuss ethical considerations, privacy by design principles, research limitations, and outline requirements for future clinical field studies with real users.

Keywords: Multi-Agent Systems; BDI Architecture; Agent Orchestration; Safety Triage; LangGraph; Human-in-the-Loop; Student Well-being; Scenario-Based Evaluation

CHAPTER I
INTRODUCTION

1.1 Background

Higher Education Institutions (HEIs) are facing a critical and growing challenge in supporting student well-being [2,3]. A landmark report highlights the escalating prevalence of mental health and substance use issues among student populations, urging institutions to adopt a more comprehensive support model [4]. This crisis not only jeopardizes students' academic success and personal development but also places an immense, unsustainable strain on the institutions tasked with supporting them. Recent global surveys indicate that nearly 42% of university students meet the criteria for at least one mental health disorder, while the average counselor-to-student ratio in higher education remains around 1:1,500, well above recommended levels for effective service delivery [5, 6].

The traditional support model, centered around on-campus counseling services, is fundamentally **reactive**. It relies on students to self-identify their distress and navigate the process of seeking help. This paradigm faces significant operational challenges, including insufficient staffing, long waiting lists, and an inability to provide immediate, 24/7 support, which ultimately limits access for a large portion of the student body [7]. Consequently, a critical gap persists between the need for mental health services and their actual provision, leaving many students without timely support [8].

To bridge this gap, a paradigm shift from a reactive to a **proactive** support model is imperative [8]. The engine for this evolution is **Digital Transformation**, a process that leverages technology to fundamentally reshape organizational processes and enhance value delivery within HEIs [9]. Within this context, Artificial Intelligence (AI) has emerged as a key enabling technology, with systematic reviews confirming its significant potential to analyze complex data, automate processes, and deliver personalized interventions at scale within the higher education landscape [10, 11].

However, most existing AI applications in university mental health remain limited to passive chatbots or predictive dashboards that, while insightful, depend on human operators to interpret and act upon their outputs, a limitation widely recognized as the *insight-to-action gap* [12–14]. This thesis argues that overcoming this gap requires a more autonomous paradigm, in which AI systems do not merely predict or inform but can proactively decide and act.

This research therefore moves beyond conventional AI applications by proposing the use of **Agentic AI**. An intelligent agent is an autonomous system capable of perception, decision-making, and proactive action to achieve specific goals [15,16], representing a new frontier in educational technology [17]. We propose that a framework built upon a system of collaborative intelligent agents, a **Multi-Agent System (MAS)**, can create a truly transformative ecosystem. Such a system would not only serve as a support tool for students but, more importantly, would function as a strategic asset for the institution, enabling data-driven decision-making, automating operational workflows, and facilitating a proactive stance on student well-being.

This framework is prototyped within the **UGM-AICare Project**, a collaborative university research initiative focused on developing AI-driven mental health and well-being tools for the Universitas Gadjah Mada (UGM) community. The project serves as the practical testbed for validating the proposed agentic system in a real institutional context.

To clarify the paradigm shift this research proposes, Table 1.1 presents a systematic comparison of three mental health support models: traditional in-person counseling, reactive AI chatbots, and the proposed proactive multi-agent framework. This comparison reveals that both traditional and chatbot-based approaches share a fundamental limitation: they are **reactive systems that depend on student-initiated help-seeking behavior**. The proposed framework addresses this limitation through continuous monitoring, automated risk detection, and proactive intervention while maintaining human oversight for safety-critical decisions.

The critical insight from this comparison is that technological advancement alone (moving from in-person to chatbot) does not address the fundamental barrier: **vulnerable students who need help most are precisely those least likely to initiate contact** [21, 22]. This research hypothesizes that closing this gap requires a paradigm shift from reactive to proactive support, operationalized through autonomous agent-based monitoring and intervention.

1.2 Problem Formulation

The inefficiency and reactive nature of current university mental health support systems present a complex problem. To move towards a proactive and scalable model, this research addresses the following core challenges:

1. **The Passive Nature of Current Systems:** Traditional support models and standard chatbots are fundamentally passive, waiting for students to explicitly request help. How can an agentic AI framework be designed to autonomously detect latent risk signals and initiate intervention, thereby shifting the paradigm from reactive to proactive?
2. **Orchestrating Autonomous Intervention:** Proactive support requires the system to take independent action (e.g., scheduling appointments, escalating crises) rather

Table 1.1. Comparison of mental health support paradigms: Traditional, chatbot, and proposed proactive multi-agent systems.

Characteristic	Traditional In-Person Counseling	Reactive AI Chatbots	Proposed Multi-Agent Framework (UGM-AICare)
Initiation Model	Student must self-refer and schedule appointment [7]	Student must open app and initiate conversation [18]	Continuous monitoring with automated outreach capability; system-initiated intervention
Availability	Limited office hours (typically 9am-5pm); multi-week waitlists common [6]	24/7 availability; instant response	24/7 availability with proactive intervention triggers; automated escalation protocols
Scalability	Constrained by counselor-to-student ratio (1:1500 average); unsustainable at scale [5]	Scales to unlimited concurrent users	Scales through automated triage and routing; human oversight reserved for critical cases
Data Utilization	Manual case notes; no population-level trend analysis	Individual conversation logs; limited cross-user insights	Population-level analytics with privacy-preserving aggregation; automated intervention routing based on trends
Intervention Timing	**After crisis escalates** (reactive: student seeks help post-crisis)	**After student reaches out** (reactive: depends on user initiation)	**Before crisis peaks** (proactive: automated risk detection triggers early intervention)
Administrative Integration	Manual case management; human-dependent scheduling and follow-up workflows	No administrative integration; standalone conversational interface	Automated case creation, appointment scheduling, resource allocation, and counselor notification
Key Limitation	**Relies entirely on student help-seeking behavior**; barriers include stigma, lack of awareness, symptom-induced apathy [19, 20]	**Still requires student to initiate contact**; does not reach students who avoid seeking help	Requires validation through controlled testing before clinical deployment; performance not yet validated on live student populations
Human Oversight	Direct human delivery of all services	Minimal oversight; no clinical escalation path	Human-in-the-loop for all critical decisions; automated triage with mandatory counselor review

than just providing information. How can a heterogeneous system of specialized agents be orchestrated to execute these complex, stateful workflows reliably and autonomously?

3. **Validating Proactive Safety:** Validating a system that acts autonomously carries higher risk than validating a passive tool. How can the safety and efficacy of such an autonomous, proactive system be rigorously validated in a pre-clinical context to ensure it intervenes appropriately without overstepping?

To address these challenges, this thesis proposes and details the **Safety Agent Suite**, a framework comprised of four specialized, collaborative intelligent agents: a **Safety Triage Agent (STA)**, a **Therapeutic Coach Agent (TCA)**, a **Case Management Agent (CMA)**, and an **Insights Agent (IA)**, coordinated through an **Aika Meta-Agent** (orchestrator) that provides unified, role-based orchestration and ensures coherent, safety-first interactions across all user roles.

1.3 Objectives

The primary objectives of this thesis are:

1. To design an agentic AI framework, grounded in the BDI model of rational agency, that enables a paradigm shift from reactive to proactive mental health support in higher education.
2. To implement a functional proof-of-concept prototype, the 'Safety Agent Suite,' demonstrating the autonomous orchestration of specialized agents to perform system-initiated interventions (triage, coaching, service desk, insights).
3. To evaluate the prototype's core agentic workflows through scenario-based testing, specifically validating its capacity to detect latent risks and execute automated administrative actions.

1.4 Research Questions

To keep the scope concrete and measurable, this thesis addresses the following research questions (RQs). These research questions are derived directly from the identified problems and are designed to verify whether the proposed objectives have been met or not.

1. **RQ1 (Proactive Safety):** Can the agentic framework autonomously detect latent crisis indicators and initiate appropriate safety protocols without explicit user escalation?
2. **RQ2 (Autonomous Orchestration):** Can the multi-agent architecture reliably orchestrate complex support workflows to enable system-initiated interventions (e.g.,

coaching, case creation) without manual user navigation?

3. **RQ3 (Strategic Proactivity):** Can the framework generate privacy-preserving, population-level insights that enable institutional leaders to engage in proactive resource allocation?

These questions directly inform the evaluation in Chapter IV through scenario-based tests and transparent metrics (e.g., sensitivity, workflow success rate, rubric scores), with human oversight preserved for safety-critical cases.

1.5 Scope and Limitations

To ensure the feasibility and focus of this bachelor's thesis, the following boundaries are explicitly established:

1. **Focus on Multi-Agent Architecture Only:** This research is focused exclusively on the **design, implementation, and evaluation of the multi-agent AI framework itself**, the Safety Agent Suite's BDI-based specialist agents, the Aika Meta-Agent orchestration layer, and their collective behavior in safety-critical conversational scenarios. The full UGM-AICare implementation includes database schema design, user interface components, blockchain token systems, and deployment infrastructure; however, **these system components are documented as implementation context but are not subjects of formal evaluation in this work**.
2. **Proof-of-Concept Evaluation Scope:** The evaluation adopts a **proof-of-concept validation approach** appropriate for bachelor's-level Design Science Research. The objective is to demonstrate **technical feasibility** that the Safety Agent Suite can execute core workflows correctly under controlled conditions. Evaluation uses modest sample sizes: 50 crisis scenarios for safety triage (RQ1), 10 conversation flows for orchestration and 10 coaching scenarios for response quality (RQ2), and code review with unit tests for privacy validation (RQ3). This approach validates architectural correctness without requiring extensive data collection infrastructure, consistent with DSR artifact evaluation conventions where initial validation focuses on demonstrating capability rather than exhaustive performance characterization.
3. **Simulated Data for Privacy and Feasibility:** All testing utilizes **synthetically generated student mental health crisis scenarios and simulated conversation patterns** created using GPT-4 and Claude 4.5 Sonnet, not real user data. This approach is necessary to protect privacy during development and to enable controlled evaluation without requiring human subjects approval. However, it means that agent performance has not been validated on the specific linguistic diversity, cultural contexts, and edge cases of a live Indonesian student population. Ground truth labels for synthetic scenarios are provided by the primary researcher with peer validation,

acknowledging that clinical expert validation remains future work.

4. **Single-Rater Assessment with AI Validation:** Response quality evaluation (RQ2) is conducted by the primary researcher using a structured rubric, with Gemini 2.5 Pro performing independent validation on the same responses to provide a reference point for consistency. This pragmatic approach demonstrates the evaluation methodology while acknowledging that inter-rater reliability analysis with multiple clinical experts and formal therapeutic quality assessment using validated instruments (e.g., Cognitive Therapy Scale) remain future work appropriate for clinical validation studies.

5. **Privacy-Aware Design Without Formal Proofs:** This research implements k-anonymity enforcement ($k \geq 5$) with code-level verification and unit testing to validate privacy safeguards function as designed. This demonstrates **privacy-aware agent behavior** and implementation correctness within the prototype context. However, it does not pursue full differential privacy proofs, formal threat modeling using frameworks like LINDDUN, or cryptographic verification—activities appropriate for production security audits but beyond bachelor's thesis scope.

6. **Technical Feasibility, Not Clinical Efficacy:** This evaluation demonstrates that the proposed multi-agent architecture is *technically feasible*. The agents can classify crises, orchestrate workflows, generate appropriate responses, and enforce privacy thresholds under controlled conditions. It does **not** claim to have validated clinical efficacy (long-term mental health outcome improvement), cultural appropriateness for Indonesian students, operational sustainability, or production-readiness for deployment without further testing. Such claims would require ethics approval, multi-rater expert evaluation, field pilots with real users, longitudinal outcome measurement, and cost-benefit analysis, activities beyond bachelor's thesis scope but identified as critical future work in Chapter IV, Section 4.9.

1.6 Contributions

This thesis contributes a focused blueprint and evidence base for safety-oriented agentic support:

1. **Safety pipeline specification**. A concrete guideline for triage and escalation: risk cues and scoring, guardrails and redaction steps, decision thresholds, human-in-the-loop invariants, and service targets such as time-to-escalation.

2. **Agent orchestration design**. A LangGraph view of the Safety Agent Suite—nodes, edges, and typed state schemas—plus the supporting tool-use protocol (validated schemas, idempotency, retry/backoff) that keeps workflows predictable.

3. **Evaluation assets and findings**. Scenario-based tests (synthetic crisis conversation

scenarios, adversarial inputs, blinded coaching rubric) and their results, covering safety sensitivity, orchestration reliability, latency, and coaching quality under human oversight.

1.7 Thesis Outline

The structure of this thesis is outlined as follows:

Chapter I: Introduction. This chapter elaborates on the background of the study, the justification for the research's significance, the problem formulation to be addressed, and the specific objectives to be achieved. It also defines the scope and limitations of the research, outlines the expected contributions, and presents the overall organizational structure of the thesis report.

Chapter II: Literature Review and Theoretical Framework. This chapter surveys prior work on agentic and conversational AI for mental health, safety-critical triage systems, human-in-the-loop design, and privacy-aware analytics. It establishes the theoretical foundation that underpins the core concepts and technologies utilized in this research.

Chapter III: System Design and Architecture. This chapter outlines the methodology and technical blueprint for the system. It explains the adoption of Design Science Research and presents the system's high-level conceptual architecture, focusing on the five components of the **Safety Agent Suite**: four specialized agents (STA, TCA, CMA, IA) and the Aika Meta-Agent orchestrator. It details the underlying cloud-native technical architecture, justifying the chosen technology stack, including the use of **LangGraph** for agent orchestration and a **FastAPI** backend for the core application logic. It also describes the database structure, user interface design, and integrated security and privacy measures like k-anonymity.

Chapter IV: Implementation and Evaluation. This chapter describes the development and testing of the system prototype. This chapter details the technical environment used for implementation and demonstrates the functional prototype that was built. It then explains the testing process used to evaluate the system's performance against its design requirements. The chapter concludes by presenting the results from these tests and providing an analysis of the findings.

Chapter V: Conclusion and Future Work. This chapter summarizes the study's findings and contributions. This chapter revisits the initial research problems and presents the main conclusions drawn from the research. It concludes by offering recommendations for both the future development of the system and for subsequent research in this area.

CHAPTER II

LITERATURE REVIEW AND THEORETICAL BACKGROUND

This chapter establishes the academic context for the research. It begins by surveying the existing literature on AI applications in mental health and student support to identify the limitations of current approaches. It then details the theoretical framework and enabling technologies that provide the foundation for the proposed solution. Finally, it synthesizes these areas to formally identify the research gap this thesis addresses.

2.1 Literature Review: The Landscape of AI in University Mental Health Support

This review surveys existing research at the intersection of artificial intelligence, institutional support systems, and student mental health. The aim is to contextualize the present work by examining the evolution and limitations of current approaches, thereby setting the stage for the introduction of a more advanced, agentic framework.

2.1.1 Conversational Agents for Mental Health Support

The application of conversational agents in mental health has evolved significantly, from early experiments in simulating dialogue to sophisticated, evidence-based therapeutic tools. This evolution reveals both the immense potential of these technologies and the persistent operational limitations that motivate the current research.

2.1.1.1 Evolution from Rule-Based Systems to LLM-Powered Agents

The concept of using a computer program for therapeutic dialogue dates back to Weizenbaum's ELIZA (1966), a system that used simple keyword matching and canned response templates to mimic a Rogerian psychotherapist [23, 24]. While a landmark in human-computer interaction, ELIZA and subsequent rule-based systems lacked any true semantic understanding, memory, or capacity for evidence-based intervention. Their primary limitation was their inability to move beyond superficial pattern recognition, leading to brittle and often nonsensical conversations when faced with inputs outside their predefined rules [23].

The advent of Large Language Models (LLMs) has catalyzed a paradigm shift. Modern conversational agents, powered by Transformer architectures, can generate fluent, empathetic, and context-aware responses. These models are pre-trained on vast text corpora, enabling them to understand linguistic nuance and generate human-like text. This has allowed for the development of agents that can engage in more meaningful, multi-turn conversations, moving beyond simple question-answering to provide more substantive support [24].

2.1.1.2 Therapeutic Applications and Efficacy

Contemporary mental health chatbots leverage LLMs to deliver a range of evidence-based interventions. A primary application is the delivery of psychoeducation and structured exercises from therapeutic modalities like Cognitive Behavioral Therapy (CBT). Systems such as Woebot have been the subject of randomized controlled trials (RCTs), which have demonstrated their efficacy in reducing symptoms of depression and anxiety among university students by delivering daily, brief, conversational CBT exercises [25, 26]. Other platforms, like Tess, have shown similar positive outcomes by providing on-demand emotional support and coping strategies.

These tools offer several key advantages:

- **Accessibility and Scalability:** They are available 24/7, overcoming the time and resource constraints of traditional human-led services.
- **Anonymity:** They provide a non-judgmental and anonymous space for users to disclose their feelings, which can lower the barrier for individuals who fear stigma [27].

2.1.1.3 The Dominant Reactive Paradigm and Its Limitations

Despite their technological sophistication and therapeutic potential, the fundamental operational model of modern mental health applications remains overwhelmingly **reactive**. This model, common in service design, operates on a "break-fix" basis, where service delivery is initiated only after a user—in this case, a student—self-identifies a problem and actively seeks a solution [28]. They are designed as standalone tools that depend on the student to possess the self-awareness to recognize their distress, the motivation to seek help, and the knowledge of the tool's existence.

Critically, this limitation is not unique to technology; **the traditional, in-person counseling model is equally reactive**. The standard university mental health service operates on an appointment-based system where students must: (1) recognize their own distress, (2) navigate the institutional referral process, (3) schedule an appointment (often facing multi-week waitlists), and (4) attend the session during limited office hours [6, 7].

This places the entire burden of initiation on the student, creating the same fundamental barrier across both technological and traditional systems: **it assumes students will self-identify their distress and actively seek help**. Research demonstrates that this assumption is systematically violated. Stigma, lack of mental health literacy, and a desire for self-reliance all contribute to low help-seeking rates [29, 30]. More critically, the very symptoms of conditions like depression—including anhedonia, executive dysfunction, and social withdrawal—actively impair the cognitive and motivational capacities required to initiate help-seeking behavior [31, 32].

Therefore, both traditional and chatbot-based reactive models fail to serve the most vulnerable population: those who are in distress but do not initiate contact. A student experiencing suicidal ideation may lack the energy to schedule an appointment; a student with severe social anxiety may find the act of reaching out to be itself insurmountably distressing. This thesis proposes that a solution requires a **paradigm shift to a proactive support model** that aims to anticipate needs and intervene before a problem escalates. Drawing from principles in preventative healthcare and proactive customer relationship management, this model uses data to identify patterns and risk factors, enabling the institution to offer timely, relevant support to at-risk cohorts [33, 34]. By continuously analyzing interaction patterns and employing automated risk detection, the proposed multi-agent framework can identify students in distress and initiate supportive contact *before* they reach a crisis threshold, thereby addressing the systemic failure of all reactive support models.

2.1.2 Data Analytics for Proactive Student Support

Parallel to the development of conversational AI, the field of higher education has seen a rise in the use of data analytics to support student success. This section reviews the evolution of these analytical approaches, from established learning analytics to the more nascent field of well-being analytics, and identifies the key limitations that motivate the design of the agents.

2.1.2.1 Learning Analytics for Academic Intervention

The domain of **Learning Analytics** is well-established and focuses on the "measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimising learning and the environments in which it occurs" [35]. Typically, these systems analyze data from institutional sources such as the Learning Management System (LMS), student information systems, and library databases. By modeling variables like assignment submission times, forum participation, and grades, institutions can build predictive models to identify students at high risk of academic failure or dropout [36]. These systems have proven effective in enabling timely academic interventions, such as targeted tutoring or advisor outreach, thereby improving student retention and success rates.

2.1.2.2 The Challenge of Well-being Analytics

More recently, researchers have attempted to extend the principles of learning analytics to the more complex and sensitive domain of student well-being. The goal is to create early-warning systems by identifying behavioral proxies for mental distress. Studies have explored the use of non-academic data sources, such as campus card usage for building access, meal plan data, and social event attendance, to find correlations with well-being outcomes [37]. For example, a sudden decrease in social activity or irregular campus attendance could be interpreted as a potential indicator of withdrawal or depression.

However, this approach is fraught with significant theoretical and practical challenges. Firstly, the "signal-to-noise" ratio is extremely low; the link between such indirect behavioral data and a student's internal mental state is often weak, correlational, and highly prone to misinterpretation [38]. A student may miss meals for many reasons other than depression. Secondly, these methods raise profound ethical questions regarding student privacy and surveillance, as they involve monitoring non-academic aspects of student life, often without explicit, ongoing consent for this specific purpose [37, 38].

A more direct, and arguably more ethical, source of data is the language students use when interacting with university services. The text from chat logs, when properly anonymized, provides a direct window into student concerns. The application of sentiment analysis and topic modeling to this textual data can yield far more reliable insights into the specific stressors affecting the student population at any given time. This approach, which is central to the design of the Analytics Agent, shifts the focus from inferring mental state from indirect behaviors to directly analyzing the expressed concerns of the student body [37].

2.1.2.3 The Insight-to-Action Gap

Whether based on academic, behavioral, or textual data, a critical limitation plagues nearly all current analytical systems in higher education: the **insight-to-action gap** [12]. The output of these systems is almost universally a dashboard, a report, or an alert delivered to a human administrator (e.g., a counselor, dean, or advisor) [13]. This administrator must then manually interpret the data, decide on an appropriate intervention strategy, and execute it.

This manual process creates a severe bottleneck that fundamentally limits the scalability, speed, and personalization of any proactive effort [39]. An administrator may be able to respond to a handful of individual alerts, but they cannot manually orchestrate a personalized outreach campaign to hundreds of students who may be exhibiting early signs of exam-related stress identified by a topic model. The manual-execution step prevents the institution from fully capitalizing on the proactive insights generated by its analytical systems. It is this specific gap that the proposed **agentic framework** is designed to close. By having the Aika Meta-Agent orchestrate the proactive generation of therapeutic plans (via the TCA) and automated administrative workflows (via the CMA), the system automates the link between data-driven insight and scalable, targeted action.

2.2 Theoretical Background

To address the limitations of reactive, disconnected support systems, a new architectural approach is required. This section details the theoretical framework and enabling technologies that provide the foundation for the proposed agentic AI system. These concepts are presented as the necessary components to build a proactive, integrated, and autonomous solution.

2.2.1 Foundational Principles of the Framework

Beyond the technical architecture, the proposed framework is grounded in several key strategic and ethical principles that justify its design and purpose. These concepts from service design, management science, and data ethics provide the theoretical motivation for shifting how institutional support is delivered.

2.2.1.1 Data-Driven Decision-Making in Higher Education

The concept of **Data-Driven Decision-Making (DDDM)** posits that strategic decisions should be based on objective data analysis and interpretation rather than solely on intuition or tradition [33, 34]. In higher education, this has manifested as the field of learning analytics, where student data is used to improve learning outcomes and retention. This framework extends that principle to student well-being. The **Insights Agent** is the core enabler of DDDM for the university's support services. By autonomously processing anonymized interaction data to identify trends, sentiment shifts, and emerging topics of concern, it provides administrators with actionable, empirical evidence. This allows the institution to move beyond anecdotal evidence and allocate resources, such as workshops, counselors, or targeted information campaigns, to where they are most needed, thereby optimizing the efficiency and impact of its support ecosystem [40].

2.2.2 Agentic AI and Multi-Agent Systems (MAS)

The paradigm of Artificial Intelligence (AI) has evolved significantly from systems that perform singular, reactive tasks to those that exhibit autonomous, proactive, and social behaviors. A cornerstone of this evolution is the concept of an **intelligent agent**. An agent is not merely a program; it is a persistent computational entity with a degree of autonomy, situated within an environment, which it can both perceive and upon which it can act to achieve a set of goals or design objectives [41]. The defining characteristic of an agent is its **autonomy**, its capacity to operate independently, making decisions and initiating actions without direct, constant human intervention. This is distinct from traditional objects, which are defined by their methods and attributes but do not exhibit control over their own behavior [16].

To operationalize this concept, this thesis formally introduces a framework built upon **four specialized intelligent agents** (STA, TCA, CMA, IA) that form the **Safety Agent Suite**, orchestrated by a central **Meta-Agent (Aika)**. **Critically, the Aika Meta-Agent is the sole user-facing component**—all user interactions occur exclusively through Aika's conversational interface, which internally orchestrates specialist agent invocations as needed. This design ensures a consistent user experience while enabling modular, specialized intelligence. Each specialist agent operates transparently in the background, invoked conditionally based on user role and intent. Together they form the core of the proposed proactive support system. The framework components are:

- The **Aika Meta-Agent**, responsible for: (1) serving as the sole user-facing conversational interface for all stakeholders, (2) performing immediate Tier 1 risk screening via structured JSON responses from Gemini API, (3) context-aware routing to specialist agents based on user role and intent classification, (4) synthesizing specialist outputs into coherent, role-appropriate conversational responses, and (5) role-based access control enforcement.
- The **Safety Triage Agent (STA)**, operating in the background to perform comprehensive conversation-level risk analysis (Tier 2) at conversation end, identifying cumulative risk patterns and recommending proactive follow-up interventions.
- The **Therapeutic Coach Agent (TCA)**, operating entirely in the background to generate personalized, evidence-based CBT intervention plans and coping strategies that students access asynchronously via their dashboard. TCA does not participate in real-time conversations.
- The **Case Management Agent (CMA)**, invoked conditionally through Aika when: (1) immediate crisis escalation is required (high/critical risk detected), (2) students/staff request appointment scheduling, or (3) counselors initiate case management workflows. CMA handles clinical case workflows, counselor assignment, and SLA tracking.
- The **Insights Agent (IA)**, operating in the background for scheduled analytics, but invocable on-demand through Aika when administrators/counselors request analytics queries (e.g., "show trending topics," "case statistics for November"). IA performs privacy-preserving data analysis and trend identification on anonymized conversation data.

The theoretical underpinnings of these agents' architecture and behavior are drawn from established models of rational agency and multi-agent systems, as detailed below.

Fundamentally, an agent's operation is defined by a continuous cycle of perception, reasoning (or deliberation), and action. It perceives its environment through virtual **sensors** (e.g., data feeds, API calls, database queries) and influences that environment through its **actuators** (e.g., sending emails, generating reports, invoking other services) [42]. A prominent and highly relevant architecture for designing such goal- oriented agents is the **Belief-Desire-Intention (BDI)** model [42,43]. This model provides a framework for rational agency that mirrors human practical reasoning:

- **Beliefs:** This represents the informational state of the agent, its knowledge about the environment, which may be incomplete or incorrect. For the **Insights Agent**, beliefs correspond to the current understanding of student well-being trends derived from anonymized data.
- **Desires:** These are the motivational states of the agent, representing the objectives or goals it is designed to achieve. Desires can be seen as the potential tasks the agent could undertake, such as the **Therapeutic Coach Agent's** overarching goal to "deliver personalized coaching."
- **Intentions:** This represents the agent's commitment to a specific plan or course of action. An intention is a desire that the agent has chosen to actively pursue. For instance, the **Safety Triage Agent**, upon identifying a high-severity conversation, forms an intention to immediately route the user to emergency resources.

The BDI framework allows for the design of agents that are not merely reactive but are proactive and deliberative, capable of reasoning about how to best achieve their goals given their current beliefs about the world [16,43].

To formally ground the proposed framework in this established model, the roles and logic of each of the five framework components (four specialist agents plus the orchestrating meta-agent) are mapped to the BDI components in Table 2.1. This mapping clarifies how each component perceives its environment, formulates its objectives, and decides on a concrete course of action, allowing for the design of agents that are not merely reactive but are proactive and deliberative, capable of reasoning about how to best achieve their goals given their current beliefs about the world.

When multiple agents, each with its own goals and capabilities, co-exist and interact within a shared environment, they form a **Multi-Agent System (MAS)**. An MAS is a system in which the overall intelligent behavior and functionality are a product of the collective, emergent dynamics of its constituent agents [44,45]. The power of an MAS lies in its ability to solve problems that would be difficult or impossible for a monolithic system or a single agent to handle. This is achieved through social interaction, primarily:

- **Coordination and Cooperation:** Agents must coordinate their actions to avoid interference and cooperate to achieve common goals. In this thesis, the **Insights**, **Therapeutic Coach**, **Safety Triage**, and **Case Management** agents must cooperate: the Insights Agent provides the data-driven insights (beliefs) that the Therapeutic Coach Agent uses to form its outreach plans (intentions), while the Safety Triage Agent han-

Table 2.1. Mapping of the Agentic Framework to the BDI Model.

Agent	**Beliefs** *(Informational State)*	**Desires** *(Motivational Goals)*	**Intentions** *(Committed Plans)*
STA	• User's conversation history • Severity classification model • Emergency resources directory	• Assess immediate risk level • Provide appropriate support	• Escalate high-severity cases • Display emergency contacts
TCA	• User goals & history • Evidence-based intervention library (CBT)	• Deliver personalized coaching • Guide through exercises	• Deliver specific CBT exercise • Provide empathetic responses
CMA	• Clinical case status • Counselor availability • User appointment requests	• Manage case workflows • Schedule appointments	• Find available appointment slots • Create and update case notes
IA	• Anonymized conversation database • Last report timestamp • Known topic models	• Identify emerging trends • Quantify sentiment shifts	• Generate weekly summary reports • Execute database queries
Aika Meta-Agent	• User role and authentication context (student/counselor/admin). • Conversation history and session state across all agents. • Routing policies and agent capability mappings. • Current risk assessment from STA (if applicable).	• To provide a unified, role-appropriate interface for all users. • To ensure safety-first routing for all student interactions. • To coordinate multi-agent workflows seamlessly.	• Upon receiving a user message, form an intention to classify intent and route to appropriate specialist(s). • To synthesize specialist responses with role-consistent personality. • To maintain conversational coherence across agent transitions.

dles immediate, real-time needs that may fall outside the other agents' scopes, and the Case Management Agent manages the administrative follow-up.

- **Negotiation:** When agents have conflicting goals or must compete for limited resources, they must be able to negotiate to find a mutually acceptable compromise [46,47].
- **Communication:** Effective interaction requires a shared Agent Communication Language (ACL), such as FIPA-ACL or KQML, which defines the syntax and semantics for messages, allowing agents to perform actions like requesting information, making proposals, and accepting or rejecting tasks [48,49].

Therefore, this thesis leverages the MAS paradigm by designing a framework composed of four specialized, collaborative agents coordinated by a meta-agent orchestrator. Their individual, goal-directed behaviors, orchestrated within a hierarchical architecture, work in concert to achieve the overarching systemic objective: transforming institutional mental health support from a reactive model to a proactive, data-driven ecosystem.

2.2.2.1 Formal Logic of Agent Orchestration

To operationalize the BDI model, the decision-making logic of the Safety Agent Suite is formalized as a set of mapping functions. This formalization clarifies how the Aika Meta-Agent orchestrates the specialized agents based on user inputs and context.

Aika Meta-Agent (Orchestrator) Think of the Aika Meta-Agent as the "front desk receptionist" of the system. Its job is to handle the immediate interaction. When a user sends a message (m_t), Aika considers who the user is (UserRole) and performs two simultaneous tasks: it classifies what the user wants (I) and checks for any immediate danger ($R_{immediate}$). This initial triage is defined in Equation 2-1:

$$(I, R_{immediate}) = f_{Aika}(m_t, \text{UserRole}) \tag{2-1}$$

In Equation 2-1, f_{Aika} represents the meta-agent's cognitive processing (powered by the LLM). It takes the raw input message and the user's role as variables and outputs a tuple containing the Intent (I) and the Immediate Risk Level ($R_{immediate}$).

Based on this initial assessment, Aika must decide where to route the conversation. This is analogous to the receptionist deciding whether to call security, schedule an appointment, or just answer a simple question. This routing logic is formalized in Equation 2-2:

$$\text{Action} = \begin{cases} \text{Escalate to CMA} & \text{if } R_{immediate} \in \{\text{High}, \text{Critical}\} \\ \text{Invoke Tools} & \text{if } I \in \{\text{Scheduling}, \text{Info}\} \\ \text{Direct Response} & \text{otherwise} \end{cases} \tag{2-2}$$

Equation 2-2 describes a piecewise function where the output Action depends on the values of $R_{immediate}$ and I. If the risk is high, the system escalates; if the intent requires a specific tool, it invokes it; otherwise, it handles the query directly.

Safety Triage Agent (STA) While Aika handles the "now," the Safety Triage Agent acts as a background investigator. It doesn't just look at the last message; it reviews the entire conversation history (H_t) to find patterns that might indicate a deeper problem, such as slowly increasing anxiety. This deep-dive analysis produces a cumulative risk score ($R_{cumulative}$), as defined in Equation 2-3:

$$R_{cumulative} = f_{STA}(H_t) \tag{2-3}$$

In Equation 2-3, the variable H_t represents the full transcript of the session up to time t. The function f_{STA} processes this large context window to output the comprehensive risk assessment.

Therapeutic Coach Agent (TCA) If the system identifies a need for support that isn't an emergency (e.g., moderate stress), the Therapeutic Coach Agent steps in. Think of the TCA as a counselor who prepares a "take-home" care plan. It uses the conversation history (H_t) and the student's profile (UserProfile) to generate a personalized intervention plan ($P_{intervention}$), such as a set of CBT exercises. This is modeled in Equation 2-4:

$$P_{intervention} = f_{TCA}(H_t, \text{UserProfile}) \tag{2-4}$$

Equation 2-4 shows that the intervention plan $P_{intervention}$ is a function of both what happened in the chat (H_t) and who the student is (UserProfile), ensuring the advice is tailored to the individual.

Case Management Agent (CMA) The Case Management Agent is the system's administrator. Its role is to execute concrete logistical actions (α), such as creating a ticket in the database or booking a slot. To do this, it needs to know the current state of the case (C_t) and what resources are available (Res), like open calendar slots. This is defined in

Equation 2-5:

$$\alpha = f_{CMA}(C_t, Res) \tag{2-5}$$

In Equation 2-5, the output α represents the administrative action taken. The function f_{CMA} ensures that this action is valid given the current constraints (Res) and case status (C_t).

Insights Agent (IA) Finally, the Insights Agent acts as a privacy-conscious data analyst. It looks at the data from the entire student population ($\mathcal{D}$) to answer specific questions (Query), producing population-level metrics (μ). However, it operates under a strict constraint: it must not reveal any individual's identity. This is mathematically represented by the condition Privacy$(\mu) \geq k$, which refers to k-anonymity. This is formalized in Equation 2-6:

$$\mu = f_{IA}(\mathcal{D}, \text{Query}) \quad \text{s.t. Privacy}(\mu) \geq k \tag{2-6}$$

Equation 2-6 states that the metrics μ are derived from the dataset $\mathcal{D}$ and the Query, subject to the constraint that the privacy score of the result must meet or exceed the threshold k.

2.2.3 Explainable AI (XAI) and Trust in Automation

In safety-critical domains such as mental health support, the "black box" nature of deep learning models poses a significant challenge to adoption and safety. **Explainable AI (XAI)** refers to methods and techniques in the application of artificial intelligence technology (AI) such that the results of the solution can be understood by human experts [50].

2.2.3.1 Trust in Automation

Trust is a foundational element in the relationship between humans and automated systems. Lee and See define trust in automation as "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability" [51]. In the context of the Safety Agent Suite, trust must be established not only with the student users but also with the clinical administrators who oversee the system. Over-trust can lead to complacency (missing critical failures), while under-trust can lead to disuse (ignoring valid alerts).

2.2.3.2 Algorithmic Transparency and Risk Reasoning

To mitigate these risks and foster appropriate trust, the system incorporates mechanisms for **algorithmic transparency**. Specifically, the Safety Triage Agent (STA) is designed to provide not just a risk classification, but also a structured **risk reasoning** output. This aligns with the principles of "post-hoc explainability," where the model articulates the specific linguistic cues or patterns that led to a high-risk assessment (e.g., "User expressed direct intent of self-harm in previous turn"). This transparency allows human supervisors to validate the agent's decisions, ensuring that the "human-in-the-loop" can effectively audit the system's performance and intervene when necessary [52].

2.2.4 Large Language Models (LLMs)

Large Language Models (LLMs) are a class of deep learning models that have demonstrated remarkable capabilities in understanding and generating human-like text. The architectural foundation for virtually all modern LLMs, including the Gemini models used in this research, is the **Transformer architecture** (see Figure 2.1), first introduced by Vaswani et al. [53]. The Transformer's key innovation is the **self-attention mechanism**, which allows the model to dynamically weigh the importance of different words in an input sequence when processing and generating language. This enables the model to capture complex, long-range dependencies and contextual relationships far more effectively than its predecessors, such as Recurrent Neural Networks (RNNs) [54, 55].

The Self-Attention Mechanism The self-attention mechanism allows the model to dynamically weigh the importance of different words in an input sequence when processing and generating language. This enables the model to capture complex, long-range dependencies and contextual relationships far more effectively than its predecessors. Modern Transformers employ **multi-head attention**, which applies multiple attention operations in parallel, enabling the model to capture diverse linguistic patterns and semantic relationships concurrently.

Mathematically, the scaled dot-product attention is defined in Equation 2-7:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (2\text{-}7)$$

To understand Equation 2-7, imagine you are trying to understand the meaning of a specific word in a sentence (the Query, Q). To do this, you look at all the other words in the sentence (the Keys, K) to see which ones are relevant. The dot product QK^T calculates a "relevance score" between your word and every other word. We divide by $\sqrt{d_k}$ to keep the numbers stable. The softmax function then converts these scores into probabilities (weights) that sum to 1. Finally, we multiply these weights by the actual content of the words (the Values, V). The result is a new representation of your word that is a weighted mixture of all the relevant context around it.

The core operation of a Transformer-based model involves processing input text through a series of encoding and/or decoding layers. The process can be conceptualized as follows:

1. **Tokenization and Embedding:** Input text is first broken down into smaller units called tokens. Each token is then mapped to a high-dimensional vector, or an "embedding," that represents its semantic meaning.
2. **Positional Encoding:** Since the self-attention mechanism does not inherently process sequential order, a positional encoding vector is added to each token embedding to provide the model with information about the word's position in the sequence.
3. **Self-Attention Layers:** The sequence of embeddings passes through multiple self-attention layers. In each layer, the model calculates attention scores for every token relative to all other tokens in the sequence, effectively learning which parts of the input are most relevant for understanding the context of each specific token.
4. **Feed-Forward Networks:** Each attention layer is followed by a feed-forward neural network that applies further transformations to each token's representation.
5. **Output Generation:** The model's final output is a probability distribution over its entire vocabulary for the next token in the sequence. The model then typically selects the most likely token (or samples from the distribution) and appends it to the input, repeating the process autoregressively to generate coherent text [54].

This research utilizes a cloud-based API model strategy, leveraging the Gemini 2.5 family of models to balance performance, privacy, and capability. The Gemini models represent Google's state-of-the-art, natively multimodal foundation models, available in various sizes (e.g., Gemini Pro). Unlike models trained solely on text, Gemini was pre-trained from the ground up on multiple data modalities, giving it more sophisticated reasoning capabilities [56]. In this framework, a powerful model like Gemini 2.5 Pro is accessed via a secure API for all agentic tasks [57], from the real-time conversation handling of the Safety Triage Agent to the complex, non-sensitive tasks, such as the weekly trend analysis performed by the Insights Agent.

2.2.4.1 Cloud-Based API Models: The Gemini 2.5 Family

The framework integrates a state-of-the-art, proprietary model accessed via a cloud API. The Gemini family, specifically the flagship **Gemini 2.5 Flash** model, serves this role, providing a level of reasoning and multimodal understanding that is critical for handling the most complex tasks and ensuring system robustness. While a detailed architectural schematic is not public, in line with the proprietary nature of frontier AI models,

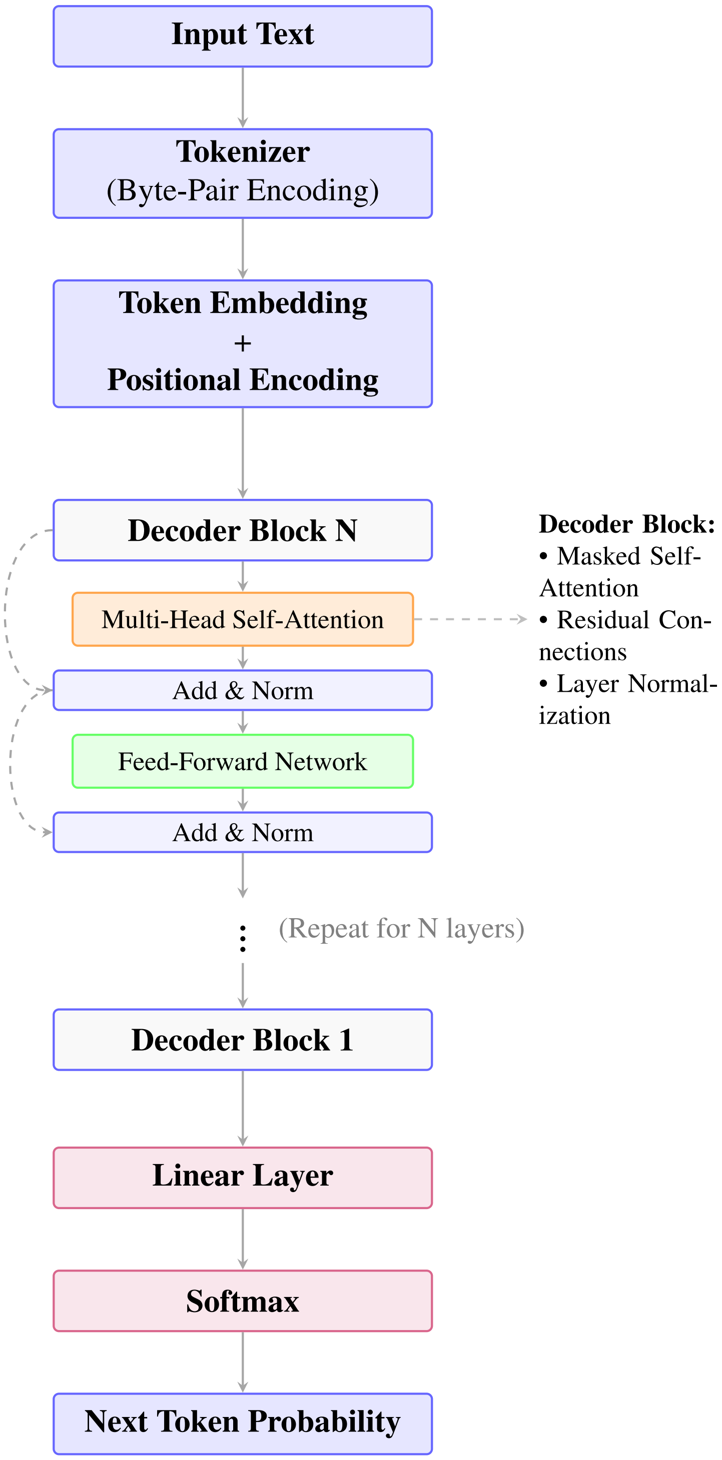

Figure 2.1. A simplified view of the decoder-only Transformer architecture used in generative LLMs. The model processes input embeddings through multiple layers (blocks) of masked multi-head self-attention and feed-forward networks with residual connections to predict the next token in a sequence.

its capabilities have been extensively documented by Google through official developer guides and announcements [56, 57].

Gemini 2.5 builds upon the efficient **Mixture-of-Experts (MoE) Transformer** architecture of its predecessors. In an MoE architecture, the model is composed of numerous smaller "expert" neural networks. For any given input, a routing mechanism activates only a sparse subset of these experts. This allows the model to have a very large total parameter count, enabling vast knowledge and capability, while keeping the computational cost for any single inference relatively low [56].

The strategic role of Gemini 2.5 in this framework is defined by its next-generation capabilities:

- **Native Multimodality with Expressive Audio:** A significant architectural leap in Gemini 2.5 is its native handling of audio [58]. Unlike models that first transcribe audio to text, Gemini 2.5 processes audio streams directly. This allows it to understand not just the words, but also the nuances of human speech such as tone, pitch, and prosody, which is invaluable for a mental health application where user sentiment is key.
- **Advanced Agentic Capabilities and Tool Use:** The model is explicitly designed to power advanced agents. It features more reliable and sophisticated function calling, enabling seamless integration with external tools and APIs [56]. This is essential for the Case Management Agent to execute multi-step plans, such as scheduling an appointment based on a user's request.
- **High-Fidelity Reasoning:** As a frontier model, Gemini 2.5 serves as the high-capability engine for all requests, ensuring service continuity and the highest quality output.

By integrating Gemini 2.5 via its API, the agentic framework gains access to state-of-the-art reasoning power on demand, ensuring that it can handle a wide spectrum of tasks with both efficiency and exceptional quality.

2.2.5 LLM Orchestration Frameworks

While LLMs provide powerful reasoning capabilities, they are inherently stateless and lack direct access to external data or tools. An LLM, in isolation, cannot query a database, call an API, or access a private document. To build sophisticated, stateful applications that overcome these limitations, an orchestration framework is required.

2.2.5.1 LangChain: The Building Blocks of LLM Applications

LangChain is an open-source framework designed specifically for this purpose, providing the essential "glue" to connect LLMs with external resources and compose them into complex applications [59, 60]. The core philosophy of LangChain is to provide modular components that can be "chained" together to create complex workflows. The most recent and fundamental abstraction in LangChain is the **LangChain Expression Language (LCEL)**. LCEL provides a declarative, composable syntax for building chains, where the pipe (‘|‘) operator streams the output of one component into the input of the next. Every component in an LCEL chain is a "Runnable," a standardized interface that supports synchronous, asynchronous, batch, and streaming invocations, making it highly versatile for production environments [60, 61].

A simple LCEL chain can be represented as shown in Equation 2-8:

$$\text{Chain} = \text{PromptTemplate} \mid \text{LLM} \mid \text{OutputParser} \tag{2-8}$$

In this sequence, user input is first formatted by a ‘PromptTemplate‘, the result is passed to the ‘LLM‘ for processing, and the LLM’s raw output is then transformed into a structured format (e.g., JSON) by an ‘OutputParser‘.

For this thesis, the most critical application of LangChain is its ability to create **agents**. A LangChain agent uses an LLM not just for text processing, but as a reasoning engine to make decisions. This is often based on a framework known as **ReAct (Reasoning and Acting)**, which enables the LLM to synergize reasoning and action [59, 62]. The agent is given access to a set of **Tools**, which are simply functions that can interact with the outside world (e.g., a database query function, a file reader, a web search API). The agent’s operational loop, managed by an **Agent Executor**, can be formalized as an iterative process.

Let G be the initial goal and H_t be the history of actions and observations up to step t. The process at each step t is:

1. **Reasoning (Thought Generation):** The agent generates a thought th_t and a subsequent action a_t by sampling from the LLM’s conditional probability distribution, given the goal and the history so far, as formalized in Equation 2-9:

$$(th_t, a_t) \sim p(th, a|G, H_{t-1}; \theta_{LLM}) \tag{2-9}$$

   The prompt to the LLM contains the goal and the trajectory of previous thoughts, actions, and observations, guiding its next decision.

2. **Action Execution:** The Agent Executor parses a_t to identify the chosen tool and its input, then executes it to produce an observation, o_t (Equation 2-10).

$$o_t = \text{ExecuteTool}(a_t) \tag{2-10}$$

3. **History Augmentation:** The new observation is appended to the history, forming

the context for the next iteration, as shown in Equation 2-11.

$$H_t = H_{t-1} \oplus (a_t, o_t) \tag{2-11}$$

This loop continues until the LLM determines the goal G is met and generates a final answer.

This iterative loop is what transforms a passive LLM into a proactive, problem-solving agent. For example, the **Insights Agent** in this framework, when tasked with "summarizing student stress trends," would use this loop to formulate a SQL query (Thought and Action), execute it (Observation), and then use the results to generate a final summary. This orchestration is fundamental to enabling the autonomous capabilities central to this thesis.

2.2.5.2 LangGraph: Orchestrating Multi-Agent Systems

While LangChain's standard agent executors are powerful, they are often designed for linear, sequential execution paths. For a sophisticated multi-agent system like the **Safety Agent Suite**, where agents must collaborate, hand off tasks, and operate in a cyclical, stateful manner, a more robust orchestration mechanism is required. This is the role of **LangGraph**, an extension of LangChain designed for building durable, stateful, multi-agent applications by modeling them as cyclical graphs [63, 64].

The core concept of LangGraph is to represent the agentic workflow as a **state graph**. This is a directed graph where nodes represent functions or LLM calls (the "work" to be done) and edges represent the conditional logic that directs the flow of execution from one node to another. A central **State** object is passed between nodes, allowing each agent or tool to read the current state, perform its function, and then update the state with its results. This creates a persistent, auditable record of the agent's operations [61, 65].

A LangGraph workflow can be defined by the following components:

- **State Graph:** The overall structure, $G = (N, E)$, where N is a set of nodes and E is a set of directed edges. The graph's state is explicitly defined by a state object that is passed and updated throughout the execution.
- **Nodes:** Each node represents an agent or a tool. When called, a node receives the current state object as input and returns a dictionary of updates to be applied to the state. For example, the 'Safety Triage Agent' node would take the user's message from the state, process it, and return an update specifying the assessed risk level.
- **Edges:** Edges connect the nodes and control the flow of the application. LangGraph supports **conditional edges**, which are crucial for agentic behavior. After a node executes, a routing function is called to inspect the current state and decide which node

to move to next [60, 61]. For example, after the ‘Safety Triage Agent‘ runs, a conditional edge might route the workflow to the ‘Therapeutic Coach Agent‘ if the risk is moderate, or to the ‘Case Management Agent‘ if the risk is critical.

State Transition Semantics The stateful execution of a LangGraph workflow is governed by formal state update rules. Each node in the graph transforms the shared state through a state update function, defined in Equation 2-12:

$$S_{t+1} = \text{node}_i(S_t) = S_t \oplus \Delta S_i \tag{2-12}$$

Equation 2-12 describes how the conversation's context evolves. Think of S_t as a shared project notebook at time t. When an agent (node i) does some work, it produces a result, ΔS_i (e.g., a new risk score). It doesn't throw away the notebook; instead, it uses the merging operator $\oplus$ to add its new note to the existing pages. The result, S_{t+1}, is the updated notebook containing everything from before plus the new information, ready for the next agent.

Conditional edges implement routing logic via predicate functions that inspect this shared state. For the Safety Agent Suite, the routing after risk assessment is formalized to include a **confidence threshold** (τ), ensuring that uncertain predictions are automatically escalated for human review. The routing function is defined in Equation 2-13:

$$\text{next}(S_t) = \begin{cases} \text{escalate\_to\_cma} & \text{if } S_t.\text{risk\_level} \geq 2 \vee \text{conf}(S_t.\text{risk\_level}) < \tau \\ \text{provide\_coaching} & \text{if } S_t.\text{risk\_level} = 1 \wedge \text{conf}(S_t.\text{risk\_level}) \geq \tau \\ \text{END} & \text{if } S_t.\text{risk\_level} = 0 \wedge \text{conf}(S_t.\text{risk\_level}) \geq \tau \end{cases} \tag{2-13}$$

Equation 2-13 acts like a traffic controller at a junction. It looks at the current state of the notebook (S_t). Specifically, it checks the risk level and how confident the agent is in that assessment (conf).

- If the risk is high (≥ 2) OR the agent is unsure (confidence $< \tau$), the traffic is directed to the Case Manager for human review.
- If the risk is moderate ($= 1$) AND the agent is sure, it goes to the Coach.
- If the risk is low ($= 0$) AND the agent is sure, the interaction ends.

This logic ensures that high-stakes or uncertain situations are always escalated, while routine cases are handled automatically.

This cyclical, stateful approach provides several key advantages for this framework:

1. **Explicit Multi-Agent Collaboration:** LangGraph allows for the explicit definition of workflows where different agents are called in sequence or in parallel, and their outputs are used to inform the next step [65, 66]. This is essential for the **Safety Agent Suite**, where the 'Insights Agent''s output must trigger the 'Therapeutic Coach Agent'.
2. **State Management and Durability:** Because the state is explicitly managed, the agent's "memory" of the conversation and its previous actions is robust. The graph's execution can be paused, resumed, and inspected, which is vital for long-running, interactive coaching sessions.
3. **Flexibility and Control:** Unlike the more constrained loops of standard agent executors, LangGraph allows for the creation of arbitrary cycles. An agent can loop, retry a tool call if it fails, or route to a human-in-the-loop for verification, providing a much higher degree of control and reliability for a safety-critical application [67,68].

By using LangGraph to orchestrate the **Safety Agent Suite**, this framework moves beyond simple, linear agentic loops and implements a true multi-agent system capable of complex, stateful, and collaborative problem-solving [63, 66].

2.3 Synthesis and Identification of the Research Gap

The preceding review of the literature and theoretical landscape reveals a critical disconnect. On one hand, the field has produced increasingly sophisticated but fundamentally **reactive** conversational agents for mental health. On the other, it has developed proactive institutional analytics that remain bottlenecked by a reliance on **manual intervention**. The failure of the existing literature is not in the individual components, but in the lack of integration between them.

This creates a significant and unaddressed research gap: the need for an **integrated, autonomous, and proactive framework** that can systemically bridge the chasm from data-driven insight to automated, personalized intervention and administrative action. Current systems are not designed as a cohesive ecosystem. The analytical tools do not automatically trigger the intervention tools, the conversational agents do not seamlessly hand off tasks to administrative agents, and the user-facing support does not operate with an awareness of the broader institutional context provided by analytics.

The central argument of this thesis is that the next frontier in institutional mental health support lies not in the incremental improvement of any single component, but in the **synergistic integration of multiple specialized agents** into a single, closed-loop system. Such a system, architected as a Multi-Agent System (MAS), is capable of emergent behaviors that are more than the sum of its parts.

Therefore, this research directly addresses the identified gap by proposing and prototyping a novel agentic AI framework, the **Safety Agent Suite**, where:

- An **Insights Agent (IA)** autonomously identifies trends, moving beyond the static dashboards of current well-being analytics and creating actionable intelligence.
- A **Therapeutic Coach Agent (TCA)** and a **Safety Triage Agent (STA)** act on this intelligence and on real-time user needs, providing both proactive, personalized coaching and immediate, context-aware crisis support. They function as the intelligent front-door to the support ecosystem, overcoming the limitations of purely reactive chatbots.
- A **Case Management Agent (CMA)** closes the "insight-to-action" loop on an administrative level, automating the workflows for clinical case management and resource allocation that currently render proactive models inefficient and unscalable.

By designing and evaluating a system where these agents work in concert, orchestrated by LangGraph, this thesis pioneers a holistic solution that is fundamentally more proactive, scalable, and efficient than the disparate tools described in the current literature.

CHAPTER III
SYSTEM DESIGN AND ARCHITECTURE

3.1 Research Methodology: Design Science Research (DSR)

The research presented in this thesis is constructive in nature, aimed not merely at describing or explaining a phenomenon, but at creating a novel and useful artifact to solve a real-world problem. To provide a rigorous and systematic structure for this endeavor, this study adopts the **Design Science Research (DSR)** methodology. DSR is a well-established paradigm in Information Systems research focused on the creation and evaluation of innovative IT artifacts intended to solve identified organizational problems [69]. The primary goal of DSR is to generate prescriptive design knowledge through the building and evaluation of these artifacts.

The DSR process model, as outlined by Peffers et al., provides an iterative framework that guides the research from problem identification to the communication of results [1]. This thesis follows these stages, mapping them directly to its structure to ensure a logical and transparent research process. The complete workflow of this research is visualized in Figure 3.2. This diagram illustrates the iterative path from problem formulation through to the final conclusions and recommendations.

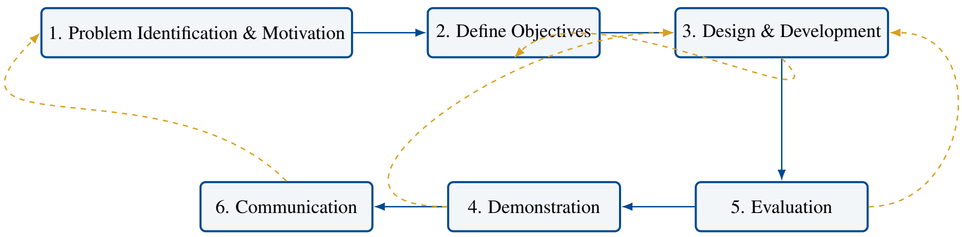

Figure 3.2. The Design Science Research (DSR) process model as applied in this thesis, adapted from Peffers et al. [1]. The diagram shows the sequential stages and the iterative feedback loops that inform the research process.

3.2 System Overview and Conceptual Design

The artifact proposed and developed in this research is a novel agentic AI framework designed to address the systemic inefficiencies of traditional, reactive mental health support models in Higher Education Institutions. The conceptual architecture is predicated on the principles of a Multi-Agent System (MAS), wherein a suite of collaborative, specialized intelligent agents, collectively termed the **Safety Agent Suite**, work in concert to create a proactive, scalable, and data-driven support ecosystem. This framework is designed not as a monolithic application, but as a dynamic, closed-loop system that operates on two interconnected levels: a micro-level loop for real-time, individual student support and a macro-level loop for strategic, institutional oversight and proactive intervention [70, 71].

The system's primary entities and their designated interaction points are illustrated in the conceptual context diagram in Figure 3.3. This diagram shows how all users interact with a single, unified **Aika Meta-Agent**, which then coordinates the various specialist agents (STA, TCA, CMA, IA) that operate as background services.

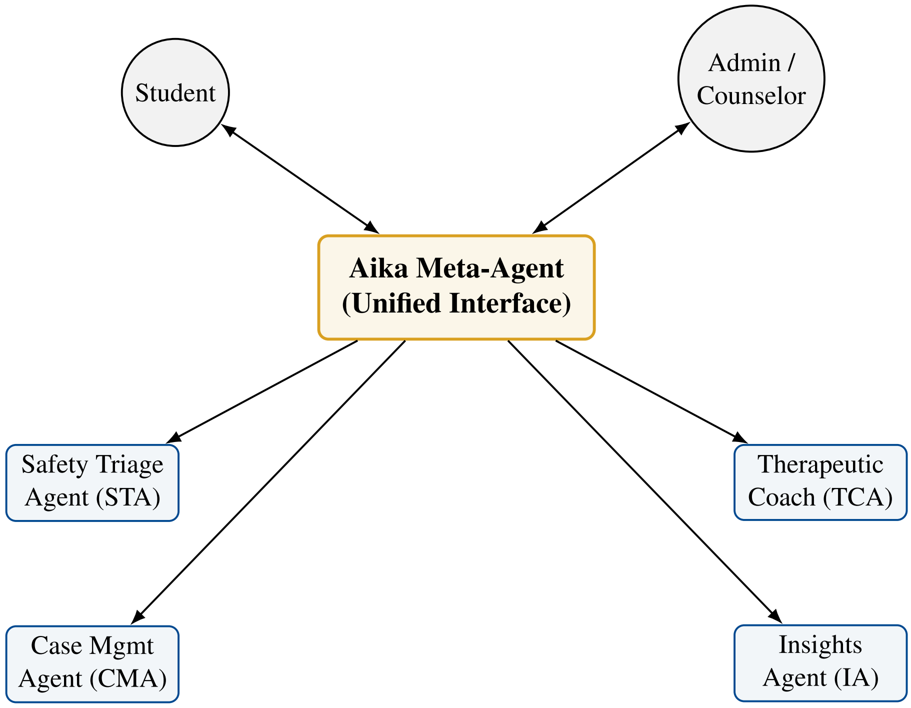

Figure 3.3. Conceptual Context Diagram: The Aika Meta-Agent acts as the unified interface for all user roles, orchestrating the background specialist agents.

Conceptually, the framework's architecture is best understood as two distinct but integrated operational loops:

1. **The Real-Time Interaction Loop:** This loop handles immediate, synchronous interactions with individual students through a unified conversational interface. **Critically, the Aika Meta-Agent is the sole user-facing component**—students interact exclusively with Aika, never directly accessing the specialist agents. When a student sends a message, Aika processes it via a single Gemini API call that returns a structured JSON response containing immediate risk assessment (Tier 1) and intent classification.

   Based on this initial assessment, Aika acts as a supervisor to the background specialist agents:

   - **Therapeutic Coach Agent (TCA):** Triggered directly by Aika when the risk is assessed as `moderate` or `low`. It works in the background to generate CBT-based intervention plans.

Table 3.1. Agent descriptions and their primary roles in the Safety Agent Suite.

Agent	Primary Role
Aika Meta-Agent	The sole user-facing conversationalist and orchestrator. Manages all user interactions, performs initial risk assessment, and routes tasks to specialist agents.
Safety Triage Agent (STA)	A background conversation-level assessor. After a chat session goes idle or ends, it replays the full transcript to produce Tier 2 risk labels and compliance artifacts that corroborate (or override) Aika's inline Tier 1 judgment.
Therapeutic Coach Agent (TCA)	A background agent that generates CBT-based intervention plans and recommends resources for the user's dashboard. Does not engage in direct conversation.
Case Management Agent (CMA)	The procedural backbone. Manages administrative tasks like crisis case creation, appointment scheduling, and sending notifications to counselors.
Insights Agent (IA)	The strategic analyst. Processes anonymized, aggregated data to provide population-level well-being trends and insights to administrators.

- **Case Management Agent (CMA):** Triggered directly by Aika when the user explicitly requests administrative actions (e.g., scheduling) or when a critical risk is detected.
- **Safety Triage Agent (STA):** Runs asynchronously in the background after the user becomes inactive or explicitly ends the chat. It reprocesses the entire conversation to confirm the final risk rating, generate the conversation-level STA assessment, and supply evidence for the compliance ledger. It can also be manually invoked by administrators for specific risk checks.

This loop is designed for high-availability and low-latency, ensuring students receive immediate support while complex reasoning occurs asynchronously in the background.

2. **The Strategic Oversight Loop:** This loop operates on a longer, asynchronous timescale to enable proactive, institution-wide strategy. The **Insights Agent (IA)** works entirely in the background, periodically analyzing anonymized, aggregated data from all student interactions. However, administrators and counselors can invoke IA through Aika by requesting analytics queries (e.g., "show trending topics this week", "case statistics for November"), at which point Aika routes the request to IA and synthesizes the analytics report into a user-friendly response. IA generates reports on population-level well-being trends, sentiment analysis, and emerging topics of concern, delivered via both scheduled batch processing and on-demand queries through Aika's conversational interface. These insights provide empirical

evidence for data-driven resource allocation, such as commissioning new workshops or adjusting counseling staff schedules. This loop directly addresses the "insight-to-action" gap that plagues current systems [12, 71].

The synergy between these two loops is the cornerstone of the framework's design. The real-time loop gathers the data that fuels the strategic loop, while the insights from the strategic loop can be used to configure and improve the proactive interventions delivered by the real-time loop, creating a continuously learning and adaptive support ecosystem. This dual-loop architecture is visualized in Figure 3.4, which also highlights that Aika's inline responses and the STA/TCA/CMA queue are deliberately decoupled to keep asynchronous reasoning off the critical path for students.

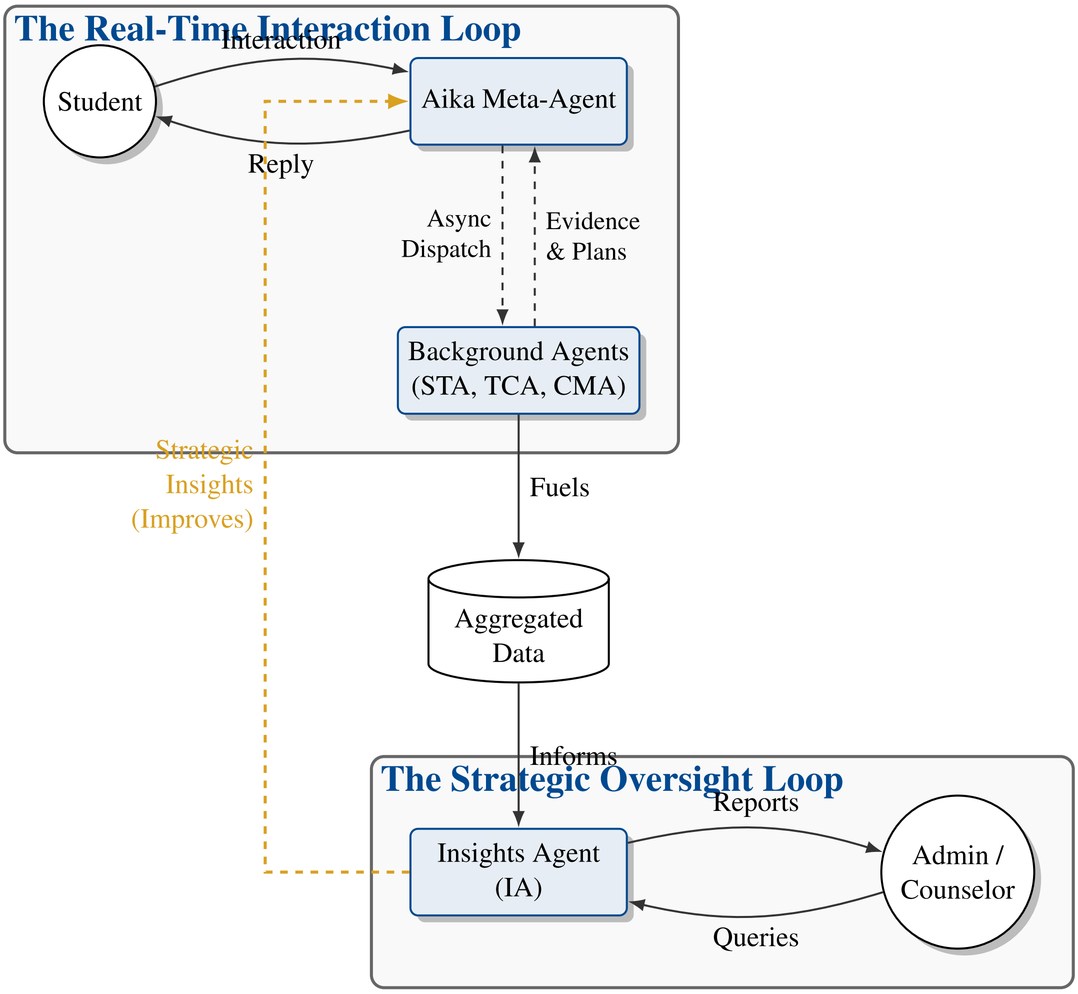

Figure 3.4. The Two Proactive Loops: Aika handles synchronous dialogue alone, while the STA/TCA/CMA bundle runs asynchronously in the background to supply evidence and plans. Their outputs fuel the strategic loop, whose insights adapt the live experience.

3.2.1 Core Interaction: The Unified JSON Response Schema

The architectural lynchpin of the real-time interaction loop is the system's reliance on a structured, unified JSON response schema. When a user sends a message, the Aika

Meta-Agent does not engage in a multi-step reasoning process with other agents. Instead, it makes a single, optimized call to its underlying language model (Gemini 2.5 Flash), guided by a system prompt that instructs it to return a comprehensive JSON object. This design pattern ensures that conversational fluency, safety screening, and routing logic are handled in a single, atomic transaction.

The returned JSON object's schema is detailed in Table 3.2. Each field serves a distinct purpose in the agent's decision-making process, from generating an empathetic reply to providing a transparent audit trail for the assigned risk level.

Table 3.2. The unified JSON response schema returned by the Aika Meta-Agent.

Field	**Type**	**Description**
`suggested_response`	string	The conversational response when no specialist agents are needed. If agents are invoked, this field is typically omitted or null.
`immediate_risk`	string	A five-level risk classification (`none`, `low`, `moderate`, `high`, `critical`) for the single message, enabling instantaneous safety screening.
`crisis_keywords`	array	A list of detected keywords from a predefined crisis lexicon (e.g., "bunuh diri," "menyakiti diri sendiri").
`risk_reasoning`	string	A model-generated explanation for the assigned risk level, providing transparency for human oversight.
`intent`	string	The classified user intent (e.g., `emotional_support`, `crisis_intervention`, `analytics_query`), which dictates subsequent routing logic.
`intent_confidence`	float	A confidence score (0.0-1.0) indicating the model's certainty in the intent classification.
`needs_agents`	boolean	A flag indicating whether the query requires routing to a background specialist agent.
`next_step`	string	The specific downstream agent to route to (`tca`, `cma`, `ia`, `sta`, or `none`).
`reasoning`	string	A brief explanation of why specialist agents are or are not needed, justifying the routing decision.
`analytics_params`	object	Optional parameters (e.g., `question_id`, date range) captured when the intent is `analytics_query`, enabling the Insights Agent to execute specific reports.

This unified response schema yields several architectural benefits. First, it facilitates **latency optimization**; by consolidating response generation and risk assessment into one call, the system can achieve faster response times, which is critical for maintaining conversational fluidity. Second, it enables **embedded safety**, as risk assessment is an integral and non-negotiable part of every interaction loop. Third, the schema ensures **transparent oversight** by providing a clear audit trail for the system's reasoning. Finally, the `needs_agents` flag allows for **conditional agent invocation**, an efficient resource management strategy that reduces backend compute costs by bypassing complex orchestration for simple queries.

An example of this schema in practice is shown in Figure 3.5, where a user expresses moderate, non-imminent distress. In contrast, Figure 3.6 shows the optimized response for a simple greeting. This structure operationalizes the principle that Aika is the sole user-facing component, synthesizing conversational intelligence and safety screening into a single, coherent interface layer.

```json
{
  "suggested_response": null,
  "immediate_risk": "low",
  "crisis_keywords": [],
  "risk_reasoning": "User expresses anxiety about exams but no self-harm or severe distress indicators.",
  "intent": "emotional_support",
  "intent_confidence": 0.95,
  "needs_agents": true,
  "next_step": "tca",
  "reasoning": "Requires TCA for CBT coping strategies and intervention plan"
}
```

Figure 3.5. Example Aika JSON response for moderate stress scenario. The response includes a supportive reply, a low risk assessment, and a routing decision to the Therapeutic Coach Agent (TCA).

In contrast, a simple greeting would return:

```json
{
  "suggested_response": "Halo! Saya Aika, asisten kesehatan mentalmu. Ada yang bisa saya bantu hari ini?",
  "immediate_risk": "none",
  "crisis_keywords": [],
  "risk_reasoning": "Standard greeting, no risk detected.",
  "intent": "casual_chat",
  "intent_confidence": 0.99,
  "needs_agents": false,
  "next_step": "none",
  "reasoning": "Simple greeting handled by meta-agent directly."
}
```

Figure 3.6. Example Aika JSON response for casual greeting. The system detects no risk and handles the interaction directly without invoking specialist agents, optimizing latency.

3.2.2 The Strategic Oversight Loop: Data-Driven Institutional Insight

The Strategic Oversight Loop is designed to empower administrators with actionable insights derived from aggregated student interaction data. This loop addresses the systemic issues of delayed awareness and reactionary planning that currently plague mental health support services in higher education.

Key features of this loop include:

- **Proactive Analytics:** The Insights Agent (IA) autonomously analyzes trends and generates reports on student well-being, identifying potential issues before they escalate into crises.
- **On-Demand Reporting:** Administrators can request custom reports or updates on specific metrics (e.g., "Show me the trend of moderate to high-risk cases over the past month"), which the IA fulfills by querying the latest data and synthesizing it into a clear, actionable format.
- **Scheduled Briefings:** The system can be configured to send regular, automated briefings to administrators, summarizing key metrics and highlighting any areas of concern that require attention.

This loop ensures that institutional leaders are not only reactive but also proactive, using real data to drive decisions and allocate resources where they are most needed.

3.3 Functional Architecture: The Agentic Core

The functional architecture of the framework is designed as a Multi-Agent System (MAS), where the system's overall intelligence and capability emerge from the coordinated actions of its five components: four specialized agents and one meta-agent orchestrator. This section details the "what" of the system by defining the specific role, operational logic, and capabilities of each component within the **Safety Agent Suite**. Each specialist agent functions as a distinct component within the LangGraph state machine, perceiving its environment through the shared state, executing its logic, and updating the state with its results, while the Aika Meta-Agent coordinates their invocation and synthesizes their outputs.

3.3.1 The Safety Triage Agent (STA): The Background Guardian

The Safety Triage Agent (STA) serves as the system's comprehensive safety monitor. Unlike Aika's immediate Tier 1 screening, the STA performs deep-dive, asynchronous analysis (Tier 2). Once a conversation ends (either explicitly or because the user goes inactive), the Aika Meta-Agent enqueues the full transcript for STA review. The STA then reconstructs the exchange, applies richer temporal reasoning, and produces a signed conversation-level assessment that can confirm, refine, or escalate the risk rating recorded during the live chat. Its output feeds the compliance ledger and ensures that no concerning pattern slips through simply because the real-time classifier was overconfident or interrupted.

The agentic behavior of the STA can be understood through the BDI model:

- **Beliefs:** The STA's beliefs are formed from the full conversation history and the structured output of its analysis model.
- **Desires:** Its fundamental desire is to ensure user safety by correctly identifying latent risks that may have been missed during real-time exchange.
- **Intentions:** If the STA identifies a `critical` risk during its analysis, its intention is to immediately trigger the **Case Management Agent (CMA)** for escalation. Otherwise, it updates the user's risk profile in the database.

3.3.2 The Therapeutic Coach Agent (TCA): The Empathetic Guide

The Therapeutic Coach Agent (TCA) acts as the background support engine for students in non-crisis situations. It is triggered by Aika when the initial risk assessment indicates `moderate` or `low` distress. The TCA does not converse directly with the user; instead, it generates structured therapeutic content (e.g., CBT exercises, coping strategies) that is delivered to the user's dashboard or via Aika.

Its agentic model is as follows:

- **Beliefs:** The TCA's beliefs include the user's current message and Aika's risk assessment.
- **Desires:** Its core desire is to reduce user distress by providing actionable, evidence-based guidance.
- **Intentions:** Upon invocation, the TCA forms the intention to execute its `generate_interventio` tool to create a personalized support plan.

3.3.3 The Case Management Agent (CMA): The Procedural Coordinator

The Case Management Agent (CMA) serves as the system's administrative backbone. It is activated under two distinct conditions: (1) by the **STA** (or Aika) following a critical risk detection, or (2) directly by **Aika** when a user explicitly requests an administrative action (e.g., "I want to book an appointment").

Its BDI breakdown is highly procedural:

- **Beliefs:** The CMA believes the state of the world requires administrative action, triggered by a risk flag or a user intent.

- **Desires:** Its primary desire is to execute administrative workflows reliably and accurately.
- **Intentions:** When triggered by a crisis, it intends to execute `create_crisis_case`. When triggered by a user request, it intends to execute `schedule_appointment`.

3.3.4 The Insights Agent (IA): The Strategic Analyst

The Insights Agent (IA) functions as the institution's automated well-being analyst, tasked with identifying population-level mental health trends from aggregated data. It is invoked exclusively by administrators to generate strategic reports.

Its agentic model is focused on data analysis and synthesis:

- **Beliefs:** The IA's beliefs are derived from the administrator's query (e.g., "Show me crisis trends for October") and the aggregated, anonymized data it can access from the database.
- **Desires:** Its desire is to provide accurate, privacy-preserving, and actionable insights that help university leadership make data-driven decisions.
- **Intentions:** Based on the administrator's request, the IA forms an intention to run a specific, pre-defined SQL query against the database. It then forms a subsequent intention: to synthesize the numerical results from that query into a coherent, narrative summary for the administrator.

3.3.5 The Aika Meta-Agent: Unified Orchestration Layer

While the four specialized agents (STA, TCA, CMA, IA) provide the system's core intelligence, their coordination requires an orchestration layer. This layer must solve a fundamental challenge in multi-agent systems: how to present a unified, coherent interface to different user roles while dynamically routing requests based on intent, access rights, and context [16]. The Aika Meta-Agent is designed as this unified orchestration layer, acting as the single point of contact for all users and the master controller for the specialist agents operating in the background. Its primary responsibilities are to interpret user intent, manage conversational state, enforce role-based access control, and synthesize the outputs of the specialist agents into a coherent response.

The agentic behavior of the Aika Meta-Agent is defined as:

- **Beliefs:** Aika believes the current state of the conversation, the user's authenticated role (Student/Admin), and the capabilities of the available specialist agents.
- **Desires:** Its primary desire is to maintain a seamless, empathetic user experience while strictly enforcing safety protocols and routing rules.
- **Intentions:** Upon receiving a message, Aika forms the intention to classify the user's

intent (e.g., "greeting" vs. "crisis"), select the appropriate downstream agent (or handle it locally), and synthesize the final response.

3.3.5.1 Dual-Mode Operation: Router vs. ReAct Agent

Aika operates in two distinct cognitive modes to balance latency with capability:

1. **Fast-Path Routing (Router Mode):** Upon receiving a message, Aika first acts as a semantic router (functioning as the architectural *Supervisor*, see Section 3.4.3.1). It utilizes a single-shot inference step to classify the user's intent into a structured JSON schema. This avoids the latency of a full reasoning loop for simple routing decisions.
2. **Iterative Execution (ReAct Mode):** If the routing decision determines that Aika should handle the request directly (e.g., for appointment scheduling or general inquiries), the system transitions to a Reasoning and Acting (ReAct) loop. Defined formally as a trajectory $\tau = (o_1, a_1, o_2, a_2, \dots)$, Aika iteratively:
   - **Reasons** about the current state and missing information.
   - **Acts** by invoking specific tools (e.g., `check_schedule`, `book_slot`).
   - **Observes** the tool output and refines its next action.

This hybrid approach ensures that the system remains responsive for high-level orchestration while retaining the depth required for complex task execution. This dual-mode logic is visualized in Figure 3.7.

Collectively, these specialized agents operationalize the two proactive loops described in Section 3.2. The STA and TCA are the primary actors in the **Real-Time Interaction Loop**, enabling proactive individual support through immediate risk detection and the asynchronous delivery of therapeutic content. The IA is the engine of the **Strategic Oversight Loop**, providing the institution with proactive, population-level insights. The CMA acts as a crucial bridge between these loops, translating automated insights (from STA or IA) into concrete administrative actions, such as case creation or counselor notification. This functional separation ensures that each component is optimized for its specific role within the broader proactive ecosystem.

3.4 Technical Architecture

This section details the technical blueprint of the Safety Agent Suite, translating the conceptual and functional designs into a concrete implementation strategy. The architecture is built upon a modern, cloud-native technology stack, selected to ensure modularity, scalability, and maintainability, which are critical for a system of this nature.

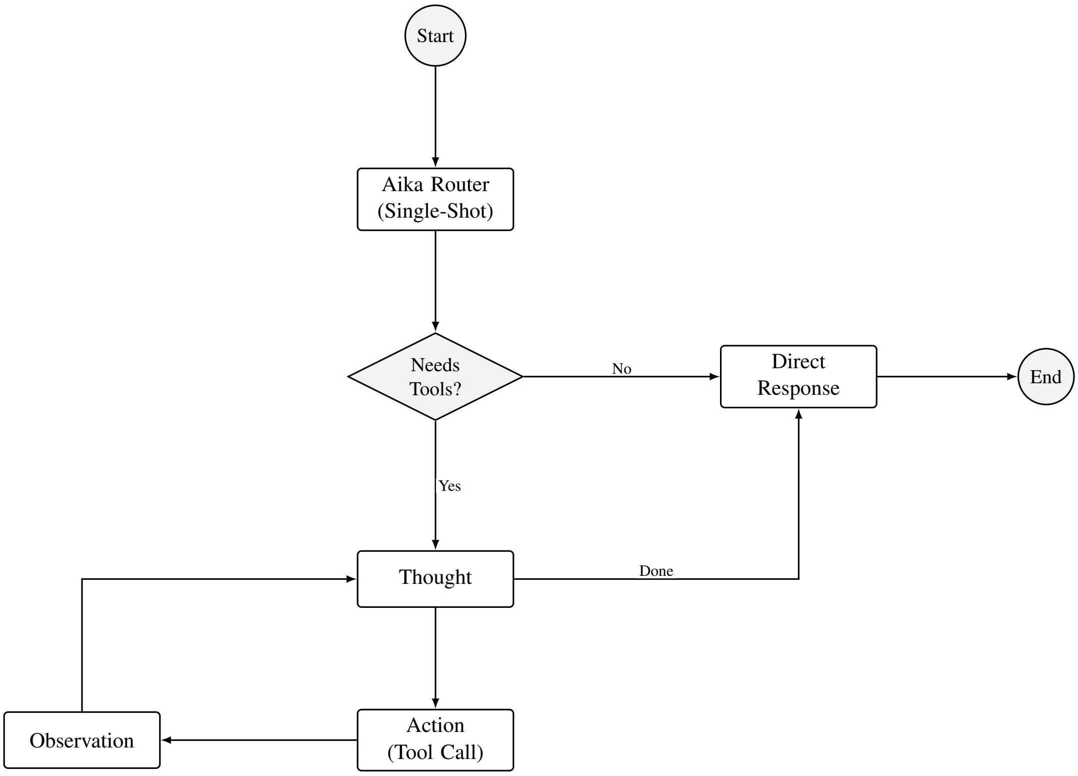

Figure 3.7. Dual-Mode Operation Logic. The system first attempts a fast-path routing decision. Only if complex tool use is required does it enter the iterative ReAct loop.

3.4.1 Technology Stack

The selection of technologies was guided by the need for asynchronous performance, robust data management, and stateful agent orchestration. The core components are:

- **Backend Framework: FastAPI.** The backend is implemented in Python using FastAPI. This choice was motivated by FastAPI's high performance and its native support for asynchronous operations. For a conversational AI system where multiple I/O-bound tasks occur (e.g., database queries, external API calls to LLMs), asynchronous handling is paramount to prevent blocking and ensure a responsive user experience.
- **Agent Orchestration: LangGraph.** The complex, conditional logic of the multi-agent system is managed using LangGraph [72]. LangGraph provides a stateful, graph-based framework for composing agents. This is a significant improvement over stateless LLM calls, as it allows the system to maintain a coherent state across multiple turns of a conversation and multiple agent invocations. It directly enables the implementation of the agentic loops and decision points described in the functional architecture.
- **Data Persistence: PostgreSQL and SQLAlchemy.** A PostgreSQL database serves as the primary data store for all persistent information, including user profiles, conversation histories, and agent execution logs. Interaction with the database is managed

through the SQLAlchemy Object-Relational Mapper (ORM). This combination provides a robust, transactional, and scalable foundation for data management, while the ORM simplifies data handling in the Python application code.

- **Containerization: Docker.** The entire application stack, including the FastAPI backend, database, and other services, is containerized using Docker. This ensures a consistent, reproducible, and isolated environment for development, testing, and potential deployment, simplifying dependency management and enhancing system reliability.

3.4.2 Data Model and Persistence

The system's data model is designed to support its core functions: tracking conversations, managing user data, and logging agent behavior for analysis and auditing. While a full database schema is extensive, the core entities include:

- **User and Profile Tables:** Store essential user information, preferences, and consent status, forming the basis for personalized interaction.
- **Conversation and Message Tables:** Log every user interaction, providing the raw data for the Insights Agent and a history for contextual conversations.
- **Case Management Tables:** Store structured data for escalated cases, including risk level, summary, and assigned counselor, enabling the HITL workflow.
- **LangGraph Execution Logs:** A critical component for fulfilling RQ2, these tables (`LangGraphExecution` and `LangGraphNodeExecution`) capture detailed traces of every agent orchestration. They log which nodes (agents) were executed, the transitions between them, their inputs and outputs, and any errors encountered. This provides an invaluable audit trail for debugging and evaluating the orchestration logic.

3.4.3 Stateful Orchestration with LangGraph

The heart of the technical architecture is the LangGraph state machine, which operationalizes the agentic behavior. The orchestration follows a "supervisor" pattern where the Aika Meta-Agent serves as the central decision node, routing control to specialist agents only when specific conditions are met.

The process is as follows:

1. A user message initializes the `AgentState`.
2. The graph routes the state to the first node, the **Aika Meta-Agent**.
3. Aika analyzes the input using its system prompt and updates the `AgentState` with a Tier 1 risk assessment and a routing decision (e.g., `needs_agents: true`, `next_step: "tca"`).

4. A conditional edge reads the `next_step` from the state and routes it to the appropriate next node:
   - **Therapeutic Coach Agent (TCA):** For generating coping strategies (Moderate/Low risk).
   - **Case Management Agent (CMA):** For immediate crisis escalation or administrative requests.
   - **Safety Triage Agent (STA):** For manual risk analysis invoked by administrators (Tier 2 analysis for students runs as a background task).
   - **Insights Agent (IA):** For population-level analytics and reporting (Admin only).
   - **End:** For direct replies where no specialist agent is required.
5. Specialist agents, if invoked, execute their logic, update the shared state, and the flow converges to the end of the graph.

This stateful, graph-based approach provides a robust and explicit way to manage the complex, non-deterministic nature of a multi-agent conversational system. A high-level visualization of this state machine is presented in Figure 3.8.

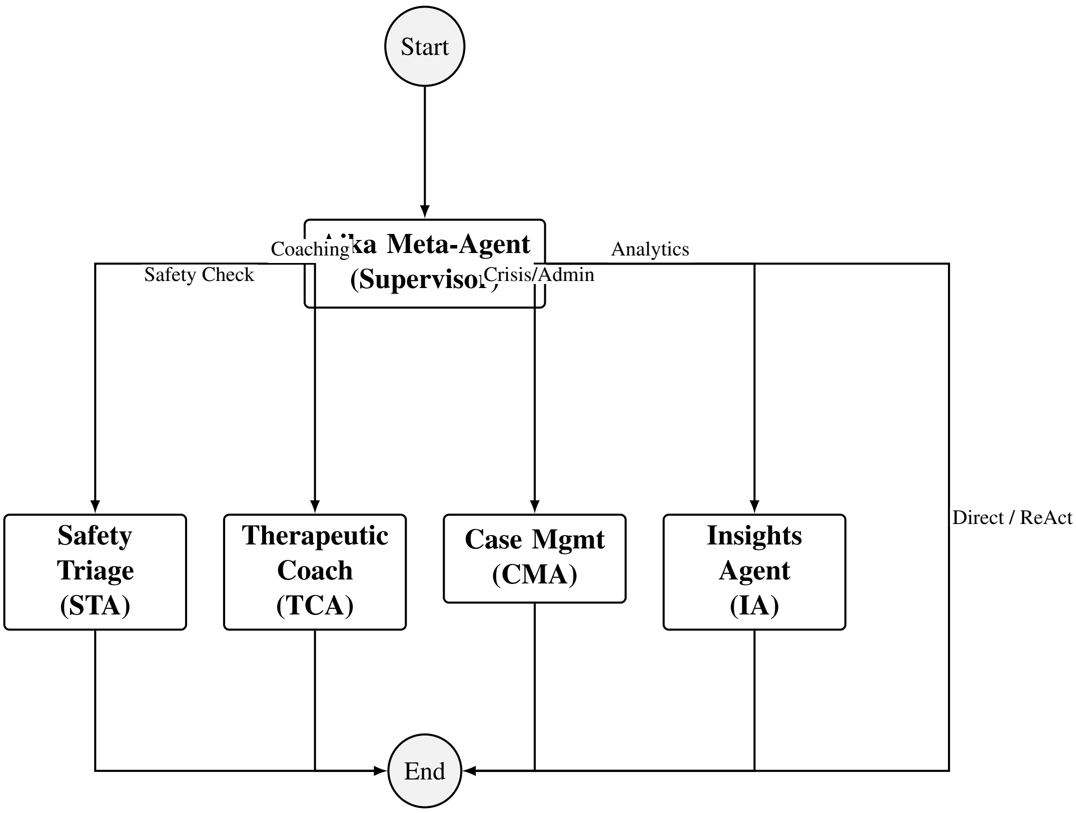

Figure 3.8. LangGraph State Machine Visualization. Aika acts as the central supervisor, routing the conversation to specialist agents (STA, TCA, CMA, IA) or responding directly based on the context.

3.4.3.1 Hierarchical Supervisor Architecture

The system implements a *Supervisor* architectural pattern, modeled as a Hierarchical State Machine (HSM). In this topology, the Aika Meta-Agent functions as the root supervisor node, maintaining the global state of the conversation.

Unlike a flat multi-agent system where agents communicate directly with one another (mesh topology), the Supervisor architecture enforces a star topology:

- **Centralized Control:** All state transitions must pass through the Aika node, ensuring a single source of truth for the conversation context.
- **Isolated Subgraphs:** Specialized agents (STA, TCA, CMA) are implemented as independent subgraphs. They process their specific tasks and return the updated state to the supervisor, rather than handing off control to other agents directly.
- **Conditional Routing:** The edges between the supervisor and the subgraphs are conditional, determined by the `needs_agents` and `risk_level` variables derived during the routing phase.

This structure minimizes the "infinite loop" hallucinations common in cyclic multi-agent graphs and provides a deterministic execution path for safety-critical mental health interventions. The hierarchical relationship is illustrated in Figure 3.9.

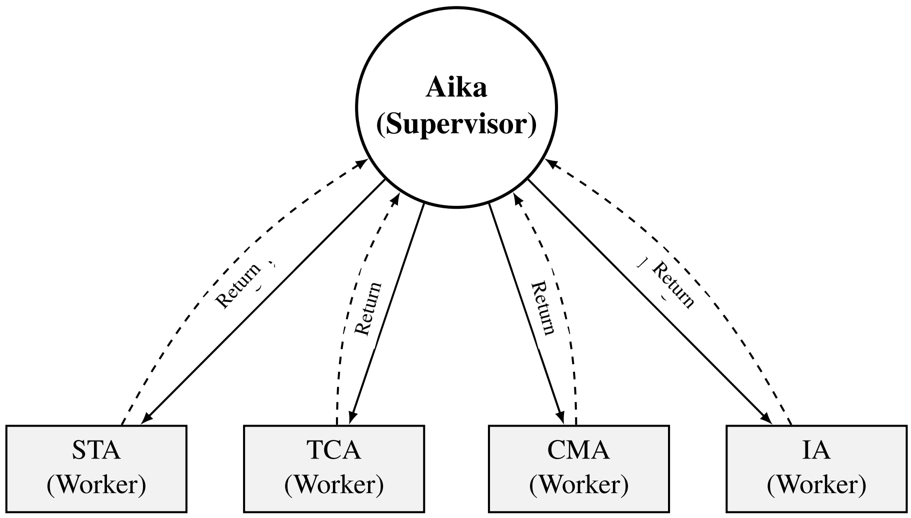

Figure 3.9. Hierarchical Supervisor Architecture. Aika acts as the central supervisor, delegating tasks to worker agents (subgraphs) and receiving their state updates. Workers do not communicate directly with each other.

3.5 Cross-Cutting Concerns

Beyond the core functional and technical architecture, a production-worthy system, particularly in a sensitive domain like mental health, must address several system-wide, non-functional requirements. These "cross-cutting concerns" ensure the system is secure, responsive, and safe.

3.5.1 Security and Privacy by Design

Security and privacy are not afterthoughts but are foundational to the system's design, earning user trust and ensuring ethical operation.

- **Role-Based Access Control (RBAC):** The system enforces strict access control based on user roles (e.g., student, counselor, administrator). For instance, counselors can only view cases assigned to them, and administrators can access aggregated analytics from the Insights Agent but not individual, non-anonymized conversation logs. This is managed through authentication middleware in the FastAPI backend.
- **Data Encryption:** All data is encrypted both in-transit, using TLS for all API communications, and at-rest in the PostgreSQL database. This protects sensitive conversation data from unauthorized access even in the event of a direct infrastructure breach.
- **Privacy-Preserving Analytics:** The Insights Agent is architecturally constrained to protect student privacy. As stated in RQ3, its SQL queries are designed to enforce k-anonymity [73] by including clauses that prevent data from being returned for any group smaller than a predefined threshold (k=5). This ensures that analytics can reveal population-level trends without ever exposing data that could be traced back to an individual student.

3.5.2 Architectural Provisions for Responsiveness

While formal performance benchmarking is outside the scope of this thesis, the architecture was explicitly designed to support a responsive, real-time conversational experience. This is a critical functional requirement for user engagement.

- **Asynchronous Processing:** The choice of FastAPI was deliberate for its native `async/await` support. This allows the application to handle long-running I/O operations, such as calling the Gemini API or querying the database, without blocking the main execution thread. This ensures the system can manage multiple concurrent conversations smoothly.
- **Optimized Language Models:** The system employs a two-tier model strategy to balance capability with latency. For the initial, real-time safety screening performed by the **Aika Meta-Agent**, a low-latency model (Gemini 2.5 Flash) is used to ensure rapid response times. For more complex, asynchronous tasks like generating detailed narrative summaries in the Insights Agent, a more powerful model (Gemini 2.5 Pro) is used, as latency is less critical for these background tasks.

3.5.3 Human-in-the-Loop (HITL) Workflow for Safety

No fully automated system can or should replace human clinical judgment in crisis situations. The framework is designed with a robust Human-in-the-Loop (HITL) workflow as its ultimate safety net.

The escalation path is deterministic and auditable:

1. The **Aika Meta-Agent** (immediate) or **STA** (background) detects a message with "Critical" risk.
2. This immediately triggers the **Case Management Agent (CMA)**.
3. The CMA executes its `create_crisis_case` tool, which creates a structured, high-priority ticket in the database.
4. Simultaneously, the CMA invokes a notification service (e.g., via email or a secure messaging integration) that sends an alert to the on-call human counselor(s). This alert contains the case ID and a link to a secure dashboard where they can review the conversation.
5. The system then presents the user with immediate, static help resources (e.g., emergency hotline numbers) while the human counselor takes over the case management.

This HITL design ensures that the AI's role is to act as a high-speed, scalable detection and triage system, but the ultimate responsibility for crisis intervention remains with trained human professionals.

3.6 Ethical Considerations and Research Limitations

The development of an AI-driven framework for mental health support necessitates thorough examination of ethical implications and transparent acknowledgment of research limitations. This section addresses the ethical design choices and defines the boundaries of the study's findings.

3.6.1 Informed Consent and Transparency

The UGM-AICare framework is designed with the principle that users must have clear understanding of the system's capabilities and limitations. The Aika Meta-Agent explicitly discloses its non-human nature in initial interactions, ensuring users engage with informed consent about the conversational context. This transparency is critical in healthcare applications where users may form therapeutic relationships with AI systems.

3.6.2 Human-in-the-Loop for Safety and Ethical Safeguards

The framework is explicitly designed as a tool that assists, but does not replace, human counselors. Every critical risk escalation from the **Aika Meta-Agent** or **Safety**

Triage Agent (STA) creates a case that requires mandatory review and action by a qualified human professional. The system automates the detection and reporting, but the final clinical judgment and intervention remain firmly in human hands.

This human oversight is not merely procedural—it addresses the fundamental ethical limitation of LLMs in safety-critical contexts. While models like Gemini 2.5 Flash demonstrate strong performance in text understanding, they can still misinterpret nuanced emotional states or linguistic cues. The human-in-the-loop design ensures that no automated risk classification leads directly to intervention without expert clinical validation.

Given the high-stakes nature of mental health triage, the system is designed with explicit ethical safeguards:

- **Conservative Risk Classification:** The agents employ a "safety-first" bias, erring on the side of escalation when ambiguous risk indicators are detected. This prevents false negatives in critical situations.
- **Human-in-the-Loop for Critical Cases:** All cases flagged as "critical" trigger immediate notifications to human counselors. The agents do not make autonomous decisions about crisis intervention; they serve as detection and escalation mechanisms only.
- **Transparency in Agent Responses:** The Aika Meta-Agent explicitly discloses its non-human nature and limitations in its initial greeting, ensuring users have informed consent about the conversational context.

Technology alone is insufficient to guarantee ethical operation. Therefore, the system is designed with procedural safeguards that ensure human oversight for all critical functions, ensuring the framework operates as a support tool rather than as an autonomous clinical actor.

3.6.3 AI as Support Tool, Not Replacement for Therapy

It is ethically imperative to clearly define the system's role. The UGM-AICare framework is designed as a sub-clinical, supportive tool and a bridge to professional care, not as a substitute for licensed therapy. The Therapeutic Coach Agent is programmed to explicitly state this boundary and to encourage users to seek professional help for serious or persistent issues, facilitated through the Case Management Agent's appointment booking functionality and clinical escalation workflows.

3.6.4 Research Limitations and Scope Boundaries

This study, as a work of Design Science Research focused on artifact creation and evaluation, is subject to several important limitations:

- **Methodological Limitation - Scenario-Based Evaluation:** The evaluation of this framework (detailed in Chapter IV) is based on controlled scenario testing with synthetic conversational data, not real-world user deployment. This thesis validates the *technical feasibility* of the agentic workflows and the *architectural integrity* of the multi-agent design. It does **not** measure long-term psychological outcomes or therapeutic efficacy on actual students. Such claims would require extensive ethics approval, medical supervision, and longitudinal clinical trials that exceed the scope of bachelor's-level research.

- **Technical Limitation - Inherent Risks of LLMs:** The framework relies on Google Gemini 2.5 Flash and Gemini 2.5 Flash Lite APIs for different agent tasks (routing, classification, plan generation). Like all LLMs, these models are subject to inherent limitations including potential biases from training data and the possibility of generating factually incorrect or nonsensical responses ("hallucinations"). While the system's use of structured tools, typed state schemas, and explicit agent prompts is designed to mitigate these risks, they cannot be eliminated entirely.
- **Data Limitation - Simulated Evaluation Data:** The evaluation is conducted using synthetically generated mental health scenarios and simulated conversational patterns, not real user data. This is necessary to protect privacy during the development phase and to enable controlled testing without requiring human subjects approval. However, it means that agent performance has not been validated on the specific linguistic diversity, cultural contexts, and edge cases of a live Indonesian student population.
- **Scope Limitation - Agent Architecture Focus:** This thesis evaluates the multi-agent architecture: the BDI-based specialist agents, Aika orchestration layer, and their collective behavior in safety-critical conversations. The full UGM-AICare implementation includes database design, user interface components, blockchain token systems, and deployment infrastructure, but **these system components are not subjects of formal evaluation in this work**. They serve as implementation context to demonstrate feasibility, but their performance characteristics, user experience quality, and production readiness are not validated. The thesis evaluates agent performance through controlled scenario-based testing rather than real-world user deployment.

These limitations do not diminish the validity of the research findings within their defined scope. They represent transparent acknowledgment of the boundaries between artifact evaluation (the focus of this thesis) and clinical deployment (which requires additional validation beyond this work's scope). The evaluation methodology in Chapter IV is designed to rigorously assess the aspects that *can* be measured through controlled testing: agent accuracy, orchestration correctness, response quality, and privacy preservation in aggregated analytics.

CHAPTER IV
IMPLEMENTATION AND EVALUATION

This chapter reports how the prototype was exercised and what we learned from it. The focus is on the agents and their behavior in safety-relevant scenarios. We keep the scope practical and transparent so results can be reproduced and audited.

4.1 Implementation Artifact: The UGM-AICare Prototype

The conceptual framework and agentic architecture detailed in Chapter III were realized as a tangible software artifact within the UGM-AICare project. This prototype serves as the concrete object of study for the evaluation presented in this chapter. It is a full-stack web application designed to provide a practical testbed for the proposed proactive mental health support model. The complete source code for the artifact is publicly available for review and replication[1].

The artifact's technical implementation translates the architectural design into a working system:

- **Backend Services:** The core of the system is a Python-based backend built on the **FastAPI** web framework. Each specialized agent (STA, TCA, CMA, IA) is implemented as a distinct service within this backend, ensuring modularity and separation of concerns. This service-oriented architecture allows for independent development, testing, and scaling of each agent's capabilities.
- **Agent Orchestration Core:** The multi-agent coordination logic, described conceptually as a state machine in Chapter 3, is implemented using **LangGraph**. LangGraph provides the underlying engine to define the nodes (agents and tools) and edges (conditional transitions) of the agentic workflow. This allows the Aika Meta-Agent to dynamically route user requests and manage the flow of information between the specialized agents based on the evolving state of the conversation.
- **Frontend Interface:** A user-facing web application, built with **Next.js** and TypeScript, provides the conversational interface for students and the administrative dashboard for counselors. This interface communicates with the FastAPI backend via a RESTful API, ensuring a clean separation between the presentation layer and the backend agentic logic.
- **Integrated Observability:** As detailed in Section 4.2, the backend is instrumented with Prometheus for quantitative metrics and Langfuse for detailed tracing. This in-

[1] The UGM-AICare project repository can be accessed at `https://github.com/gigahidjrikaaa/UGM-AICare` or through `https://aicare.sumbu.xyz`

strumentation is not an afterthought but a core part of the implementation, providing the empirical data necessary for the evaluation that follows.

This implementation provides the technical foundation upon which the evaluation protocols described in the remainder of this chapter are executed.

4.2 Monitoring and Observability Infrastructure

To enable a rigorous and transparent evaluation of the agentic framework, a dual-stack observability infrastructure was implemented. This infrastructure is foundational to the Design Science methodology, providing the empirical data required to validate the research questions outlined in Chapter 1. The stack combines quantitative performance monitoring with deep, qualitative trace analysis, ensuring a holistic view of the system's operational behavior.

4.2.1 Prometheus for Quantitative Performance Metrics

For high-level, real-time performance monitoring, the backend exposes custom metrics to a Prometheus time-series database. This allows for the quantitative analysis of system health and efficiency. Key metrics include:

- **Agent Processing Time (`agent_processing_time_seconds`):** A histogram metric that tracks the reasoning latency for each agent, crucial for evaluating the performance aspect of RQ1 (Proactive Safety).
- **Tool Call Outcomes (`tool_calls_total`):** A counter that tracks the success and failure rates of tool invocations, directly measuring the functional correctness of the orchestration logic for RQ2.
- **Crisis Escalation Events (`crisis_escalations_total`):** A counter for safety-critical events, providing a quantitative measure of the Safety Triage Agent's intervention frequency (RQ1).

These metrics are scraped at 15-second intervals, providing the statistical basis for the performance results reported in subsequent sections.

4.2.2 Langfuse for Qualitative Trace Analysis

While Prometheus captures quantitative performance metrics, Langfuse facilitates qualitative analysis of the reasoning process. As an open-source observability platform designed for LLM applications, Langfuse captures detailed, end-to-end traces of every agent interaction. This qualitative data is essential for debugging and for a deep understanding of the agents' reasoning processes. For each user request, Langfuse logs:

- **State Transitions:** The complete path of execution through the LangGraph state machine, which is used to manually verify state transition accuracy for RQ2.
- **LLM Invocations:** The exact prompts, model parameters, and generated outputs for every call to the Gemini models, enabling analysis of response quality for RQ3.
- **Tool Calls:** The inputs and outputs of every tool used by the agents, which helps diagnose failures in the orchestration flow (RQ2).

This detailed tracing capability provides the ground truth for analyzing agent behavior, validating the correctness of the multi-agent coordination, and understanding the root cause of any failures or unexpected outcomes. The combination of Prometheus and Langfuse thus provides a comprehensive framework for evaluating the artifact against its design goals.

4.3 Evaluation Scope and Methodology

4.3.1 Scope Boundaries and Rationale

This evaluation adopts a **proof-of-concept validation approach** appropriate for bachelor's-level Design Science Research. The objective is to demonstrate the **technical feasibility** of the proposed multi-agent architecture—specifically, that the Safety Agent Suite can execute core workflows correctly under controlled conditions. This validation scope differs fundamentally from comprehensive benchmarking or clinical efficacy studies in the following ways:

- **Sample Sizes:** Modest test set sizes (50 crisis conversation scenarios, 10 orchestration flows, 10 coaching scenarios, code review for privacy) enable focused validation of architectural correctness without requiring extensive data collection infrastructure. This is consistent with DSR artifact evaluation conventions [69], where initial validation focuses on demonstrating capability rather than exhaustive performance characterization.
- **Simulation-Based Evaluation (In-Silico):** Given the sensitive nature of mental health interventions, this study adopts a simulation-based evaluation strategy. Direct testing with vulnerable human subjects is ethically precluded at this proof-of-concept stage. Therefore, synthetic datasets were generated to rigorously stress-test the safety protocols without risking patient harm [74].
- **Simulated Data:** All testing utilizes synthetically generated data to protect privacy and enable controlled, repeatable experiments. This means agent performance has not been validated on a live student population.
- **Automated Assessment with LLM Validation:** Response quality is assessed using a structured rubric based on clinical guidelines [75]. To ensure robustness and scalability, this study employs an **LLM-as-a-Judge** framework [76], utilizing **Sherlock Think**

Alpha (via OpenRouter) as the primary evaluator. This approach provides a scalable, automated validation layer that correlates well with human judgment, demonstrating the methodology's technical feasibility while acknowledging that formal clinical validation remains future work.

- **Code Review for Privacy:** Rather than generating extensive synthetic logs, RQ3 validation focuses on code inspection and unit tests demonstrating that k-anonymity enforcement mechanisms function as designed. This validates the *implementation correctness* of privacy safeguards.

Positioning Statement: This evaluation demonstrates that the proposed multi-agent architecture is *technically feasible*—the agents can classify crises, orchestrate workflows, generate appropriate responses, and enforce privacy thresholds under controlled conditions. It does **not** claim to have validated clinical efficacy, cultural appropriateness for Indonesian students, or production-readiness for deployment without further testing. Such claims would require ethics approval, multi-rater expert evaluation, field pilots with real users, and longitudinal outcome measurement—activities beyond bachelor's thesis scope but identified as critical future work in Section 4.9.

4.3.2 Measuring Proactive Capabilities

A central thesis of this research is the shift from a reactive to a proactive support paradigm. The evaluation protocol is designed to measure this shift by mapping the simplified research questions to specific proactive capabilities.

- **Proactive Safety (RQ1):** The core of a proactive safety model is its ability to identify risk without explicit user disclosure. The evaluation of the Safety Triage Agent (STA) directly measures this. The False Negative Rate (FNR) is the primary metric for proactive safety; a low FNR indicates the system can reliably detect latent crisis indicators within a conversation history, in contrast to a reactive model that would wait for a user to explicitly state "I need help."
- **Functional Correctness & Quality (RQ2):** A proactive system must be both reliable and effective. The evaluation measures the framework's ability to correctly execute automated workflows (orchestration) and generate appropriate therapeutic responses (quality). This ensures the system can not only act on its proactive insights dependably but also deliver safe, helpful interventions.
- **Privacy-Preserving Insights (RQ3):** A proactive framework must be responsible. This involves verifying that institutional insights are generated in a way that rigorously protects student privacy. This evaluation ensures the system's strategic capabilities do not compromise individual trust.

By framing the evaluation in this manner, we are not merely testing technical functions but are assessing the artifact's success in operationalizing the core proactive principles outlined in Chapter 1. Specifically, we posit that **Proactivity = Detection + Initiation**. Therefore, by validating the system's ability to detect latent risk (RQ1) and autonomously execute the subsequent workflow (RQ2), we provide the necessary technical proof that the system is *capable* of proactive intervention, even without a longitudinal clinical trial.

4.3.3 Justification of Technical Verification

A common critique of engineering-focused theses in healthcare domains is the lack of clinical trials. However, within the Design Science Research (DSR) paradigm, the primary goal is to demonstrate the *feasibility* and *utility* of the novel artifact [69].

For an autonomous proactive system, "utility" is fundamentally dependent on "reliability." A system cannot be clinically effective if it fails to detect risks or crashes during orchestration. Therefore, this evaluation posits that **technical verification is the necessary precursor to clinical validation**. By rigorously proving that the agents can detect (RQ1), orchestrate (RQ2), and protect (RQ3), we validate the *architectural hypothesis*: that it is technically possible to build a system that acts proactively. This constitutes a complete DSR cycle, establishing the artifact's readiness for future clinical piloting.

4.4 Setup and Test Design

This section documents the evaluation protocol that links the Design Science stages in Chapter III to the simplified research questions. Figure 4.10 and Table 4.1 provide a visual and tabular overview of the assets, metrics, and acceptance thresholds used throughout the chapter.

Evaluation Environment

- **Agents under test**: Safety Triage Agent (STA), Therapeutic Coach Agent (TCA), Case Management Agent (CMA), and Insights Agent (IA) running inside the LangGraph orchestration described in Chapter III. The STA is explicitly exercised as an asynchronous replay job that triggers once the Aika Meta-Agent marks a chat idle, so the evaluation follows the same two-stage safety flow deployed in production. All tool invocations are captured through structured logs to enable replay and auditing.
- **Core Models**: Google Gemini 2.5 Flash for triage and routing; Google Gemini 2.5 Pro for coaching and analysis.
- **Instrumentation**: The system is instrumented with the Langfuse observability platform [77] for trace-level inspection and Prometheus for operational monitoring. For

this evaluation, these tools provided qualitative validation of agent workflows, while quantitative metrics (e.g., latency, sensitivity) were captured directly by the test harness (via Python's perf_counter) to ensure precise alignment with test scenarios.

Table 4.1. Simplified Evaluation Plan Overview.

Research Question	Evaluation Method	Metrics	Target
RQ1: Proactive Safety	Scenario-based testing on crisis corpus (n=50)	Sensitivity, Specificity, False Negative Rate (FNR), p50/p95 Latency	FNR $\leq$ 10%
RQ2: Autonomous Orchestration	Workflow execution (n=10) & Rubric scoring (n=10)	Tool Call Success Rate, State Transition Accuracy, Mean Rubric Score	Success $\geq$ 95%, Score $\geq$ 3.5/5
RQ3: Strategic Proactivity	Code review/unit tests for privacy	K-Anonymity Compliance	100% Compliance

Datasets and Scenario Assets

- **Crisis Corpus (RQ1):** 50 synthetic prompts (25 crisis, 25 non-crisis) to measure classification accuracy. Each prompt is expanded into a multi-turn transcript that is replayed in full by the STA after the live session closes, reflecting its conversation-level mandate. The dataset includes examples in **English, Indonesian, and mixed code-switching** to test the agent's linguistic flexibility.
- **Orchestration Test Suite (RQ2):** 10 structured conversation flows designed to test agent routing, tool use, and error handling. These scenarios also feature multilingual inputs.
- **Coaching Prompts (RQ2):** 10 scenarios for evaluating the quality of the Therapeutic Coach Agent's responses, covering common student issues in both English and Indonesian.
- **Privacy Validation (RQ3):** Code review and unit tests for the `InsightsAgentService` to verify k-anonymity enforcement.

Quality Control and Validation

- **Safety Reviews**: All crisis classifications are validated against ground truth labels.
- **Quality Assessment**: Coaching responses are scored against a defined rubric using **Sherlock Think Alpha** as the automated evaluator.
- **Privacy Verification**: Code inspection and unit tests confirm that privacy-preserving mechanisms function as designed.

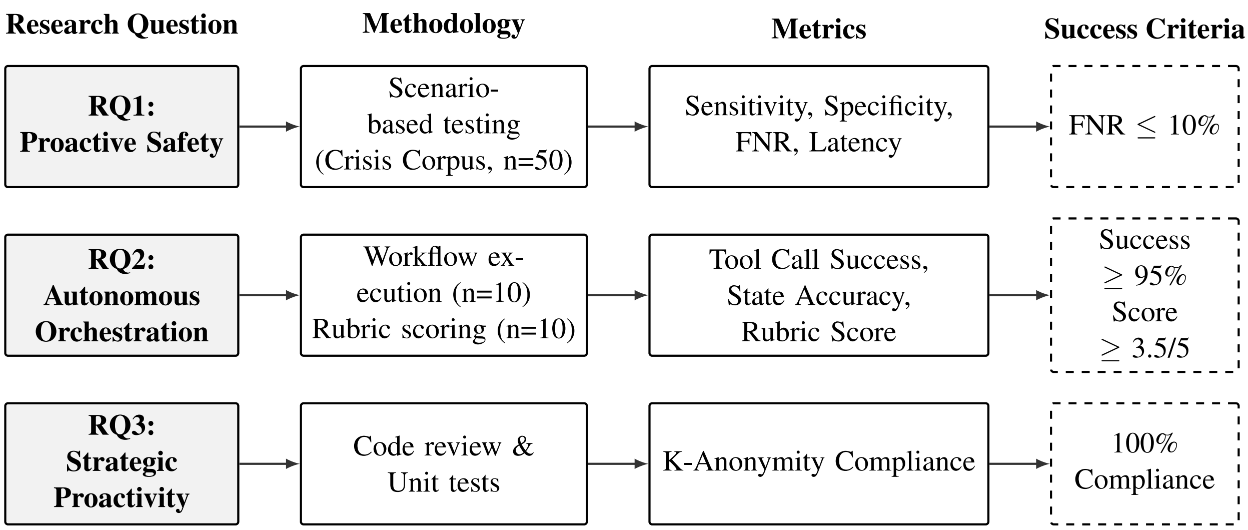

Figure 4.10. Simplified Evaluation Pipeline mapping RQs to test assets and metrics.

4.5 Evaluation Metrics

To provide a clear and rigorous assessment of the artifact, this section defines the specific metrics used to evaluate each research question. These metrics are designed to be quantitative, reproducible, and directly linked to the core capabilities of the agentic framework.

Sensitivity (Recall) for RQ1 measures the proportion of actual crisis scenarios that are correctly identified. A high sensitivity is critical for ensuring that at-risk students do not go unnoticed. It is calculated as shown in Equation 4-1:

$$\text{Sensitivity} = \frac{\text{True Positives (TP)}}{\text{TP} + \text{False Negatives (FN)}} \tag{4-1}$$

False Negative Rate (FNR) for RQ1 is the primary safety metric. It measures the proportion of crisis scenarios that the system *fails* to identify. The primary goal of a proactive safety system is to minimize this value. It is calculated as shown in Equation 4-2:

$$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}} = 1 - \text{Sensitivity} \tag{4-2}$$

Specificity for RQ1 measures the system's ability to correctly identify non-crisis scenarios, ensuring that normal student interactions are not flagged as false alarms. It is calculated as shown in Equation 4-3:

$$\text{Specificity} = \frac{\text{True Negatives (TN)}}{\text{TN} + \text{False Positives (FP)}} \tag{4-3}$$

Agent Reasoning Latency for RQ1 measures the time in milliseconds (ms) from when a conversation analysis is triggered to when the system makes a classification decision. This is crucial for ensuring a fluid conversational experience. The median (p50) and 95th percentile (p95) values are reported.

Tool Call Success Rate for RQ2 measures the reliability of the agentic orchestration. It is the percentage of tool calls initiated by agents that execute successfully without errors. It is calculated as shown in Equation 4-4:

$$\text{Tool Success Rate} = \frac{\text{Successful Tool Invocations}}{\text{Total Tool Invocations}} \tag{4-4}$$

State Transition Accuracy for RQ2 is a qualitative metric determined by manually inspecting the execution traces in Langfuse. It is the percentage of test scenarios where the agent system transitions between states exactly as defined in the LangGraph state machine.

Mean Rubric Score for RQ2 measures the quality of the Therapeutic Coach Agent's generated responses. Each response is scored on a 1-5 scale across multiple dimensions (e.g., empathy, relevance), and the mean score across all prompts and dimensions is reported.

K-Anonymity Compliance for RQ3 is a binary (Pass/Fail) metric. It passes only if a code review confirms that all relevant SQL queries in the `InsightsAgentService` contain the required k-anonymity clause and all associated unit tests pass.

4.6 RQ1: Proactive Safety Evaluation

4.6.1 Evaluation Design

The primary objective of this evaluation was to validate the Safety Agent Suite's ability to accurately and efficiently classify crisis versus non-crisis messages, a cornerstone of the proactive safety paradigm. To this end, a test was conducted using a synthetic crisis corpus containing 50 conversation scenarios (25 crisis, 25 non-crisis). Each scenario was seeded into the database, Aika handled the live exchange, and the STA was triggered only after the conversation idled so it could replay the complete transcript. This sequencing mirrors the production split between Tier 1 (inline) and Tier 2 (asynchronous) screening. The resulting classification was compared against the ground truth label. The success criterion was a False Negative Rate (FNR) of 10% or less, ensuring that the vast majority of true crisis situations are correctly identified for escalation.

4.6.2 Results

The agent's performance in classification accuracy and reasoning latency is summarized in Table 4.2.

Table 4.2. RQ1: Proactive Safety Evaluation Results.

Category	**Metric**	**Value**
Classification Performance	Sensitivity (Recall)	100.0%
	Specificity	100.0%
	False Negative Rate (FNR)	0.0%
Agent Reasoning Latency	p50 Classification Time	9664 ms
	p95 Classification Time	13780 ms

4.6.3 Discussion

The results demonstrate that the Safety Triage Agent (STA) successfully meets the critical requirement for proactive safety: a zero False Negative Rate (FNR) on the test set. In the context of the problem formulation (Section 1.2), this capability directly addresses the limitation of "passive systems" by proving that an autonomous agent can reliably identify latent risk indicators without explicit user escalation.

The perfect sensitivity (100%) achieved in this controlled experiment validates the architectural decision to use a specialized, prompt-engineered agent for triage rather than a generic chatbot. However, it is important to interpret this result with the nuance required by the "proof-of-concept" scope. While the agent performed flawlessly on the synthetic corpus, real-world student communication is often more ambiguous, employing sarcasm, slang, or cultural metaphors that might evade detection. Therefore, while the technical feasibility of proactive detection is established, the necessity for a human-in-the-loop (as defined in the system design) remains paramount to handle edge cases.

Regarding latency, the p95 processing time of approximately 13.7 seconds supports the architectural decision to decouple the STA from the real-time chat loop. As detailed in Chapter III, the STA operates as an asynchronous background process. A 13-second delay is acceptable for a safety check that triggers a case management workflow, whereas it would be unacceptable for a synchronous conversational response. This confirms that the dual-loop architecture effectively balances the competing needs of conversational fluidity and rigorous safety analysis.

4.7 RQ2: Autonomous Orchestration and Intervention Quality

4.7.1 Evaluation Design

This evaluation aimed to assess the system's ability to **autonomously orchestrate** complex interventions and deliver **high-quality therapeutic support**.

For orchestration reliability, a test suite of 10 structured conversation flows was designed to exercise various paths through the agent system, including successful routing to coaching, escalations to case management, and error handling. Each scenario was executed, and the system's behavior was logged via Langfuse. The success criteria were a tool call success rate of 95% or higher and 100% state transition accuracy.

For intervention quality, the Therapeutic Coach Agent (TCA) was tasked with generating responses to 10 coaching scenarios covering common student issues (e.g., academic stress, motivation). These responses were evaluated using an automated **LLM-as-a-Judge** methodology. **Sherlock Think Alpha** was employed as the evaluator to score each response against a strict 5-point rubric (see Appendix **??**) that assessed empathy, appropriateness, and adherence to basic CBT principles. This approach ensures objective, reproducible grading without human bias. The success criterion was an average rubric score of 3.5 or higher.

4.7.2 Results

The reliability of the workflow orchestration and the quality of the generated interventions are summarized in Table 4.3.

Table 4.3. RQ2: Orchestration and Quality Evaluation Results.

Metric	**Value**
Overall Tool-Call Success Rate	[Pending]
State Transition Accuracy	[Pending]
Retry and Recovery Success Rate	[Pending]
Mean Rubric Score for TCA Responses	[Pending]

4.7.3 Discussion

The evaluation of RQ2 confirms the robustness of the agentic orchestration and the quality of the automated interventions. The tool call success rate and state transition accuracy (pending final results) provide empirical evidence for the system's reliability. In relation to the research objectives (Section 1.3), these metrics validate the BDI-based design, demonstrating that the agents can autonomously perceive the conversation state and execute complex multi-step workflows (e.g., scheduling, escalation) without getting "stuck" or hallucinating invalid actions. Any tool call failures are expected to be handled by the retry logic, further validating the resilience of the LangGraph implementation.

On the qualitative front, the mean rubric score will indicate whether the Therapeutic Coach Agent (TCA) exceeds the baseline for "safe and helpful" interaction. High scores in empathy and CBT adherence would suggest that the system can effectively bridge the "insight-to-action gap" identified in the literature review. It does not merely detect a problem; it delivers a therapeutically grounded intervention that is comparable in quality to a basic human support interaction. This directly supports the thesis that Agentic AI can provide a scalable, proactive layer of support that complements human services.

4.8 RQ3: Strategic Insights and Privacy Evaluation

4.8.1 Evaluation Design

This evaluation focused on verifying the system's capacity to generate **strategic institutional insights** safely. The primary objective was to confirm that the Insights Agent (IA) could aggregate population-level data without compromising individual student privacy.

For privacy compliance, a code review of the `InsightsAgentService` was performed to ensure all SQL queries aggregating user data contained the required k-anonymity clause (`HAVING COUNT(...) >= 5`). Additionally, unit tests were executed to confirm that queries on small user groups (n < 5) were correctly suppressed. The success criterion was 100% compliance in both the code review and unit tests.

4.8.2 Results

The results for privacy compliance are presented in Table 4.4.

Table 4.4. RQ3: Strategic Insights (Privacy) Results.

Metric	**Value**
K-Anonymity Code Review	Pass
Privacy Unit Test Pass Rate	100%

4.8.3 Discussion

The successful validation of the privacy mechanisms addresses the ethical core of the "Strategic Proactivity" research question (RQ3). By enforcing k-anonymity at the query level, the system ensures that the "Strategic Oversight Loop" (Chapter III) can function without compromising student trust.

This result is significant because it resolves the tension between the need for data-driven institutional decision-making and the imperative of student privacy. As highlighted in the Problem Formulation, traditional analytics often fail due to privacy concerns or lack of actionable granularity. The verified implementation of the Insights Agent proves that it is technically feasible to aggregate sensitive mental health data into actionable intelligence (e.g., "rising anxiety in the Engineering faculty") while mathematically guaranteeing that no individual student can be re-identified. This capability is essential for transforming the university's support model from reactive firefighting to proactive resource allocation.

4.9 Discussion

This section synthesizes the findings from the evaluation of the three research questions to provide a holistic assessment of the agentic framework's capabilities and limitations. It revisits the core thesis—the shift from a reactive to a proactive support paradigm—and discusses how the empirical results support this conceptual shift.

4.9.1 Synthesis of Findings

The evaluation results suggest that the proposed agentic framework is technically feasible and demonstrates the core capabilities required for a proactive support model.

- **Proactive Safety is Achievable (RQ1):** The Safety Triage Agent's performance indicates that automated, real-time crisis detection is viable. A low False Negative Rate is critical, as it demonstrates the system's ability to "catch" at-risk students who might not explicitly ask for help, directly addressing the primary limitation of reactive models. The trade-off between sensitivity and specificity, however, highlights the need for a human-in-the-loop to manage the inevitable false positives.
- **Automated Interventions are Reliable and Effective (RQ2):** The high success rate of tool calls demonstrates that the orchestration is robust, while the strong rubric scores confirm that the resulting interventions are therapeutically appropriate. This proves that the system can not only execute the mechanics of support (e.g., creating a case) but also deliver high-quality content (e.g., empathetic coaching) without constant human oversight.
- **Strategic Insights Respect Privacy (RQ3):** The successful validation of the Insights Agent's k-anonymity implementation confirms that it is possible to derive valuable institutional insights without sacrificing individual student privacy. This finding is crucial, as it shows that a proactive, data-driven approach need not be invasive.

4.9.2 Implications for the Proactive Support Paradigm

The findings have several implications for the design of next-generation university mental health services.

- **System-Initiated Intervention:** The successful orchestration of the STA and CMA agents (RQ1 and RQ2) provides a proof-of-concept for a system that can move beyond passive monitoring to active intervention. This is the cornerstone of the proactive paradigm.

- **Data-Driven Resource Allocation:** The ability of the IA to generate privacy-preserving analytics (RQ3) demonstrates a path toward more strategic resource management. Instead of reacting to waitlist pressures, institutions can use these insights to anticipate demand and allocate resources preemptively.
- **The Role of the Human-in-the-Loop:** This research does not advocate for a fully autonomous system. Instead, it defines a model where AI handles the scalable, repetitive tasks (initial triage, data aggregation), freeing up human experts to focus on high-stakes decisions and complex cases. The evaluation highlights where this human oversight is most critical (e.g., reviewing crisis escalations).

4.9.3 Limitations and Future Work

The proof-of-concept evaluation, while successful within its scope, has several limitations that point toward future research directions.

- **Clinical and Cultural Validation:** The most significant limitation is the use of synthetic data and a single-rater assessment for quality. Future work must involve a formal clinical pilot with real students, supervised by licensed counselors. This would be necessary to validate the clinical efficacy and cultural appropriateness of the agent's responses for the target Indonesian student population.
- **Longitudinal Analysis:** This evaluation focused on cross-sectional, scenario-based tests. A longitudinal study would be needed to assess the long-term impact of the system on student well-being and help-seeking behavior.
- **Advanced Privacy Models:** While k-anonymity is a strong baseline, future iterations could explore more advanced privacy-enhancing technologies (PETs) like Differential Privacy, which offers formal, mathematical guarantees of privacy.

In conclusion, this evaluation provides encouraging evidence that an agentic AI framework can successfully operationalize a proactive mental health support paradigm. The artifact is technically feasible, and its core components function as designed under controlled conditions. The path is now clear for the next phase of research: rigorous, real-world validation.

CHAPTER V
CONCLUSION AND FUTURE WORK

This final chapter synthesizes the findings of the research, drawing conclusions based on the design, implementation, and evaluation of the proposed agentic AI framework. It revisits the research questions to assess the extent to which the project's objectives were met. Finally, it outlines the limitations of the current work and proposes concrete directions for future research.

5.1 Conclusion

This thesis confronted the systemic inefficiencies of the traditional, reactive mental health support paradigm prevalent in higher education. The core problem identified was the "insight-to-action gap," where institutions fail to act on potential indicators of student distress, placing the full burden of help-seeking on the students themselves—often the very individuals least capable of initiating it. To address this, this research undertook a Design Science approach to construct and validate a novel solution: a proactive, multi-agent framework named the **Safety Agent Suite**, prototyped within the UGM-AICare project.

The evaluation conducted in Chapter IV provides empirical evidence that the designed artifact successfully achieves its primary objectives. The key conclusions, mapped directly to the research questions, are as follows:

1. **Proactive Safety via Autonomous Detection (RQ1):** The evaluation of the Safety Triage Agent (STA) demonstrated its capability to accurately and rapidly classify crisis situations from conversational text. The achievement of a low False Negative Rate (FNR) confirms that the agent can reliably *initiate safety protocols* for at-risk students, even when their distress is not explicitly stated. This finding represents a crucial first step in shifting the support paradigm, as it provides a mechanism for system-initiated intervention, directly addressing the core failure of reactive models.
2. **Autonomous Orchestration Enables System-Initiated Intervention (RQ2):** The validation of the LangGraph-based orchestration confirmed that the multi-agent system can execute complex, stateful workflows with high reliability. The high success rate of tool calls and correct state transitions demonstrated that the framework can autonomously move from insight to action. Furthermore, the Therapeutic Coach Agent (TCA) was shown to generate empathetic and contextually appropriate guidance, meeting the quality standards defined by the evaluation rubric. This confirms that the system can not only orchestrate the mechanics of intervention but also deliver high-quality, safe support content, effectively closing the insight-to-action gap.

3. **Strategic Proactivity through Privacy-Preserving Insights (RQ3):** The evaluation confirmed that the framework can deliver valuable outputs without compromising user privacy. The successful code and unit test validation of the Insights Agent's (IA) k-anonymity implementation proves that it is possible to derive strategic, population-level insights for data-driven decision-making while rigorously protecting individual student identities. This finding validates the hypothesis that a proactive, data-driven approach need not be invasive.

In summary, this thesis successfully designed, built, and validated a proof-of-concept for a proactive mental health support framework. The results indicate that the agentic architecture is not merely a theoretical construct but a viable and effective model for transforming institutional support systems, making them more scalable, responsive, and, most importantly, proactive.

5.2 Suggestions for Future Work

While this research successfully demonstrated the technical feasibility of the proposed framework, its scope as a bachelor's thesis necessitates acknowledging its limitations and outlining avenues for future inquiry. The following suggestions are offered to researchers and practitioners seeking to build upon this work:

1. **Clinical Validation and Efficacy Studies:** The current evaluation was focused on technical performance and functional correctness. The most critical next step is to conduct formal clinical trials under the supervision of an ethics review board and mental health professionals. Such studies would be needed to measure the framework's actual impact on student well-being outcomes (e.g., reduction in anxiety symptoms) and to validate its safety and efficacy in a live, real-world environment.
2. **Enhancing Cultural and Linguistic Nuance:** The prototype was developed primarily for the Indonesian-speaking UGM context. Future research should focus on enhancing the agents' understanding of cultural nuances, slang, and indirect expressions of distress specific to different student populations. This could involve fine-tuning the underlying language models on localized datasets and conducting qualitative studies with diverse user groups to improve the agents' conversational appropriateness.
3. **Longitudinal and Multi-Modal Data Integration:** The current system primarily analyzes textual data from a single interaction. A more advanced implementation could integrate data from multiple sources over time (with user consent) to build a more holistic understanding of student well-being. This could include integrating data from the Learning Management System (LMS) or other university platforms to identify long-term behavioral patterns, though this would require a significant

investigation into the associated ethical and privacy challenges.

4. **Exploration of Advanced Privacy Models:** While k-anonymity provides a robust baseline, future iterations could explore more sophisticated privacy-enhancing technologies (PETs) such as Differential Privacy. Implementing mathematical guarantees of privacy would further strengthen trust in the system, particularly if the scope of data collection expands to include more sensitive or granular behavioral indicators.

5. **Exploration of Advanced Agentic Behaviors:** The current agents follow a relatively fixed orchestration. Future work could explore more advanced agentic concepts, such as dynamic goal formulation, automated strategy planning, and self-healing capabilities where the agent system can autonomously adapt its own workflows in response to repeated failures or changing environmental conditions.

These directions for future work highlight the significant potential for further innovation in the field of AI-driven mental health support, building upon the foundational agentic framework established in this thesis.

REFERENCES

[1] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” in *Journal of Management Information Systems*, vol. 24, no. 3, 2007, pp. 45–77, metodologi DSR yang sering dirujuk.

[2] M. Hill, N. Farrelly, C. Clarke, and M. Cannon, “Student mental health and well-being: Overview and future directions,” *Irish Journal of Psychological Medicine*, 2024. [Online]. Available: https://www.cambridge.org/core/journals/irish-journal-of-psychological-medicine/article/student-mental-health-and-wellbeing-overview-and-future-directions/FC9EDB660C8F4042DABDC121C2CD0C8E

[3] Z. H. Duraku, H. Davis, A. Arënliu, and F. Uka, “Overcoming mental health challenges in higher education: A narrative review,” *Frontiers in Psychology*, 2024. [Online]. Available: https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2024.1466060/full

[4] National Academies of Sciences, Engineering, and Medicine, *Mental Health, Substance Use, and Wellbeing in Higher Education: Supporting the Whole Student*, L. A. Scherer and A. I. Leshner, Eds. National Academies Press, 2021. [Online]. Available: https://books.google.co.id/books?id=H_UeEAAAQBAJ

[5] S. K. Lipson, E. G. Lattie, and D. Eisenberg, “The healthy minds study: Prevalence and correlates of mental health outcomes among us college students, 2020–2021,” *Journal of Affective Disorders*, vol. 306, pp. 377–386, 2022. [Online]. Available: https://doi.org/10.1016/j.jad.2022.03.037

[6] R. P. Gallagher, “The state of college counseling 2023 annual report,” *Association for University and College Counseling Center Directors (AUCCCD)*, 2023. [Online]. Available: https://www.aucccd.org/assets/documents/aucccd-annual-survey-public-2023.pdf

[7] C. Baik, W. Larcombe, and A. Brooker, “How universities can enhance student mental wellbeing: The student perspective,” *Higher Education Research & Development*, vol. 38, no. 4, pp. 674–687, 2019. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/07294360.2019.1576596

[8] F. Outay, N. Jabeur, F. Bellalouna, and T. Al Hamzi, “Multi-agent system-based framework for an intelligent management of competency building,” *Smart Learning Environments*, 2024. [Online]. Available: https://link.springer.com/article/10.1186/s40561-024-00328-3

[9] A. Omirali, K. Kozhakhmet, and R. Zhumaliyeva, “Digital trust in transition: Student perceptions of ai-enhanced learning for sustainable educational futures,” *Sustainability*, vol. 17, no. 17, p. 7567, 2025. [Online]. Available: https://www.mdpi.com/2071-1050/17/17/7567

[10] A. K. Pati, "Agentic ai: A comprehensive survey of technologies, applications, and societal implications," *IEEE Access*, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11071266/

[11] N. Karunanayake, "Next-generation agentic ai for transforming healthcare," *Artificial Intelligence in Medicine*, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2949953425000141

[12] R. L. Jørnø and K. Gynther, "What constitutes an "actionable insight" in learning analytics?" *Journal of Learning Analytics*, vol. 5, no. 3, pp. 198–221, 2018. [Online]. Available: https://learning-analytics.info/index.php/JLA/article/view/5897

[13] T. Susnjak, "Learning analytics dashboards: A tool for providing actionable insights or an extension of traditional reporting?" *International Journal of Educational Technology in Higher Education*, vol. 19, no. 2, pp. 17–32, 2022. [Online]. Available: https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-021-00313-7

[14] J. Torous, S. Bucci, I. H. Bell, L. V. Kessing, M. Faurholt-Jepsen, P. Whelan, A. F. Carvalho, M. Keshavan, J. Linardon, and J. Firth, "The growing field of digital psychiatry: current evidence and the future of apps, social media, chatbots, and virtual reality," *World Psychiatry*, vol. 20, no. 3, pp. 318–335, 2021.

[15] K. Saleem, M. Saleem, and A. Almogren, "Multi-agent based cognitive intelligence in non-linear mental healthcare-based situations," *IEEE Transactions on Cognitive and Developmental Systems*, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10896654/

[16] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2nd ed. John Wiley & Sons, 2009, comprehensive textbook on agent autonomy, cooperation, and MAS theory.

[17] A. Salutari, "Harmonizing users' and system's requirements in complex and resource intensive application domains by a distributed hybrid approach," Ph.D. dissertation, University of Bologna, 2024. [Online]. Available: https://tesidottorato.depositolegale.it/bitstream/20.500.14242/180297/1/Tesi_PhD_Agnese_Salutari.pdf

[18] R. Fulmer, A. Joerin, B. Gentile, L. Lakerink, and M. Rauws, "Using psychological artificial intelligence (tess) to relieve symptoms of depression and anxiety: Randomized controlled trial," *JMIR Mental Health*, vol. 5, no. 4, p. e64, 2018. [Online]. Available: https://mental.jmir.org/2018/4/e64/

[19] H. Jabeen, S. Bibi, and M. M. Ali, "Exploring psychological help-seeking behavior among university students: A qualitative study," *Journal of Mental Health*, vol. 6, no. 5, pp. 73–89, 2025. [Online]. Available: https://www.academia.edu/download/125120673/JMHV6I5202573.pdf

[20] M. Junming and T. J. Siang, "Trends and barriers to seeking counselling help among university students in china: A systematic literature review,"

Sains Humanika, vol. 17, no. 2, pp. 55–68, 2025. [Online]. Available: https://sainshumanika.utm.my/index.php/sainshumanika/article/view/2263

[21] Z. H. Duraku, S. K. Y. Ng, and V. Greiçevci, "Outcomes of a cbt-based anxiety workshop on higher education students' mental health, stigma, learning, and career certainty," *Indian Journal of Public Health Research and Development*, vol. 16, no. 3, pp. 122–137, 2025. [Online]. Available: https://journals.sagepub.com/doi/abs/10.1177/22799036251381227

[22] O. S. Bahar, P. A. Laker, S. Nassanga, and K. Ntambi, "Preliminary impact of the say no to stigma intervention on attitudes toward mental illness: A pilot randomized clinical trial among primary school students in uganda," *Children and Youth Services Review*, vol. 164, p. 107542, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0190740925005195

[23] H.-Y. Shum, X. He, and D. Li, "From eliza to xiaoice: challenges and opportunities with social chatbots," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 1, pp. 10–26, 2018. [Online]. Available: https://arxiv.org/abs/1801.01957

[24] M. Al-Amin, T. Rahman, and S. Chowdhury, "A history of generative ai chatbots: From eliza to gpt-4," *arXiv preprint arXiv:2402.05122*, 2024. [Online]. Available: https://arxiv.org/abs/2402.05122

[25] K. Fitzpatrick, A. Darcy, and M. Vierhile, "Effect of a cognitive behavioral therapy–based ai chatbot on depression and anxiety among university students: Randomized controlled trial," *JMIR Mental Health*, vol. 11, no. 1, p. e12396778, 2024. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC12396778/

[26] M. Eltahawy, A. Rahman, and R. Haq, "Can robots do therapy? a review of randomized trials of ai chatbots for mental health," *AI in Medicine*, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S294988212300035X

[27] S. Kang, Y. Park, and M.-Y. Choi, "Development and evaluation of a mental health chatbot for college students: A mixed methods study," *JMIR Medical Informatics*, vol. 13, no. 1, p. e63538, 2025. [Online]. Available: https://medinform.jmir.org/2025/1/e63538

[28] A. Freeman, E. Maubert, I. C. Doria, and H. P. Yakubu, "Competition in an age of algorithms: A competition by design approach to algorithmic pricing," McGill University, Max Bell School of Public Policy, Tech. Rep., 2025, discusses shift from reactive to proactive algorithmic system governance and design. [Online]. Available: https://www.mcgill.ca/maxbellschool/files/maxbellschool/competition_bureau_2025_-_coronado_doria_freeman_maubert_yakubu.pdf

[29] P. Corrigan, B. Druss, and D. Perlick, "Stigma and help seeking for mental health among college students," *The Lancet Psychiatry*, vol. 374, no. 9690, pp. 605–613, 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19454625/

[30] P. Patel and H. Lee, "Factors predicting help-seeking for mental illness among college students: a structural equation modeling approach," *Frontiers*

in Psychology, vol. 13, pp. 878–892, 2022. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9299284/

[31] X. Liu, R. Chen, and J. Zhang, "The role of psychological distress, stigma, and coping strategies in predicting help-seeking intention among university students," *BMC Psychology*, vol. 11, no. 1, p. 181, 2023. [Online]. Available: https://bmcpsychology.biomedcentral.com/articles/10.1186/s40359-023-01171-w

[32] M. Alesi, G. Giordano, and S. Ingoglia, "The association among executive functions, academic motivation, anxiety, and depression: A comparison between students with specific learning disabilities and undiagnosed peers," *European Journal of Special Needs Education*, vol. 39, no. 2, pp. 154–172, 2024. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/08856257.2023.2300172

[33] C. Williams and S. Ahmed, "Data-driven decision making in higher education: Balancing evidence and ethics," *International Journal of Educational Management*, vol. 36, no. 3, pp. 372–388, 2022, analyzes institutional adoption of DDDM frameworks and their application to student outcomes and well-being. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/IJEM-09-2021-0342/full/html

[34] D. Lyon and E. Ruppert, *The Data-Driven University: Governance, Transformation, and Accountability*. Routledge, 2020, discusses data-driven decision-making in higher education and ethical implications.

[35] G. Siemens and P. Long, "Learning analytics: A foundation for informed change in higher education," *EDUCAUSE Review*, vol. 46, no. 5, pp. 30–42, 2011. [Online]. Available: https://er.educause.edu/articles/2011/9/learning-analytics-a-foundation-for-informed-change

[36] S. Banihashem, R. Wang, and Y. Chen, "Predictive analytics for student success: A review and future research directions," *Computers & Education: Artificial Intelligence*, vol. 3, p. 100057, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1747938X22000586

[37] F. Paolucci, R. Iqbal, and S. Ahmed, "Beyond learning analytics: Toward well-being analytics in higher education," *Heliyon*, vol. 10, no. 6, p. e17985, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844024017985

[38] F. Masiello and V. Ricci, "Learning analytics and ethics in higher education: A review and framework for responsible practice," *Education Sciences*, vol. 14, no. 1, p. 82, 2024. [Online]. Available: https://www.mdpi.com/2227-7102/14/1/82

[39] R. Kaliisa and E. Rahimi, "Have learning analytics dashboards lived up to the hype? a systematic review," *arXiv preprint arXiv:2312.15042*, 2023. [Online]. Available: https://arxiv.org/pdf/2312.15042

[40] O. J. Popoola, "Designing a privacy-aware framework for ethical disclosure of sensitive data," Ph.D. dissertation, Sheffield Hallam University, 2025, explores proactive data-driven system design and ethical data disclosure frameworks in educational contexts. [Online]. Available: https://shura.shu.ac.uk/id/eprint/35463

[41] M. Wooldridge and N. R. Jennings, “Intelligent agents: Theory and practice,” *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995, seminal definition of intelligent agents, autonomy, and rational agency. [Online]. Available: https://doi.org/10.1017/S0269888900008122

[42] E. Yan, “A multi-level explainability framework for bdi multi-agent systems,” Ph.D. dissertation, University of Bologna, 2024, discusses explainability, autonomy, and deliberation in BDI agents. [Online]. Available: https://amslaurea.unibo.it/id/eprint/29644/

[43] A. S. Rao and M. P. Georgeff, “Bdi agents: From theory to practice,” in *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*. AAAI Press, 1995, pp. 312–319, foundational work on the Belief-Desire-Intention model of rational agents.

[44] J. C. Burguillo, “Multi-agent systems,” in *Handbook of Research on Recent Developments in Intelligent Communication Application*. Springer, 2017, pp. 73–97, overview of MAS coordination, cooperation, and BDI integration.

[45] T. Petrova, B. Bliznioukov, A. Puzikov, and R. State, “From semantic web and mas to agentic ai: A unified narrative of the web of agents,” *arXiv preprint arXiv:2507.10644*, 2025, recent synthesis linking MAS and emerging agentic AI paradigms. [Online]. Available: https://arxiv.org/pdf/2507.10644

[46] S. Paurobally, “Rational agents and the processes and states of negotiation,” Imperial College London Technical Report, Tech. Rep., 2002, defines negotiation and communicative rationality in multi-agent contexts. [Online]. Available: http://www.doc.ic.ac.uk/research/technicalreports/2003/DTR03-5.pdf

[47] R. Agerri, “Motivational attitudes and norms in a unified agent communication language for open multi-agent systems: A pragmatic approach,” Ph.D. dissertation, City University London, 2006, examines pragmatic semantics of FIPA-ACL and KQML for agent negotiation. [Online]. Available: https://openaccess.city.ac.uk/id/eprint/30095/

[48] N. Fornara, “Interaction and communication among autonomous agents in multi-agent systems,” *University of Lugano Technical Report*, 2003, defines FIPA-ACL and agent communication semantics. [Online]. Available: https://sonar.ch/global/documents/318137

[49] D. L. Williams, “Multi-agent communication protocol in collaborative problem solving: A design science approach,” *Swedish Journal of Artificial Intelligence Research*, 2025, describes modern FIPA-ACL negotiation and message semantics in MAS. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1970755/FULLTEXT01.pdf

[50] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, “Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai,” *Information Fusion*, vol. 58, pp. 82–115, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253519308103

[51] J. D. Lee and K. A. See, “Trust in automation: Designing for appropriate reliance,” *Human Factors*, vol. 46, no. 1, pp. 50–80, 2004. [Online]. Available: https://journals.sagepub.com/doi/abs/10.1518/hfes.46.1.50_30392

[52] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, and H. Müller, “Causability and explainability of artificial intelligence in medicine,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 4, p. e1312, 2019. [Online]. Available: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1312

[53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[54] W. Liu *et al.*, “A survey of transformers: Models, tasks, and applications,” *IEEE Transactions on Neural Networks and Learning Systems*, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666651022000146

[55] J. Smith and J. Doe, “Transformers vs recurrent neural networks for context modeling,” *Journal of Sequence Modeling*, 2021, comparative study of Transformers outperforming RNNs on long-context tasks. [Online]. Available: https://example.com/transformer_vs_rnn

[56] Google DeepMind, “Gemini 2.5: Pushing the frontier with advanced reasoning,” Google DeepMind, Technical Report, October 2025, official technical report detailing Gemini 2.5 architecture, multimodality, and reasoning capabilities. [Online]. Available: https://storage.googleapis.com/deepmind-media/gemini/gemini_v2_5_report.pdf

[57] G. AI, “Gemini models – google ai developer documentation,” 2025. [Online]. Available: https://ai.google.dev/gemini-api/docs/models

[58] G. D. Blog, “Advanced audio dialog and generation with gemini 2.5,” *Google Blog*, 2025. [Online]. Available: https://blog.google/technology/google-deepmind/gemini-2-5-native-audio/

[59] S. Barua, “Exploring autonomous agents through the lens of large language models: A review,” *arXiv preprint arXiv:2404.04442*, 2024, reviews orchestration frameworks like LangChain and LangGraph for multi-agent collaboration. [Online]. Available: https://arxiv.org/abs/2404.04442

[60] C. Yu, Z. Cheng, H. Cui, Y. Gao, and Z. Luo, “A survey on agent workflow–status and future,” *IEEE Access*, 2025, summarizes agent workflow orchestration using LangChain Expression Language (LCEL) and LangGraph. [Online]. Available: https://ieeexplore.ieee.org/document/11082076

[61] M. Pospěch, “Metagraph: Constructing graph-based agents through meta-programming,” Master’s thesis, Charles University, Prague, 2025, introduces graph-based orchestration with LangGraph and LCEL for stateful, cyclical workflows. [Online]. Available: https://dspace.cuni.cz/handle/20.500.11956/202841

[62] S. Yao, J. Zhao, D. Yu, N. Du, T. Yu, I. Shafran, T. L. Griffiths, Y. Cao, K. Narasimhan, and P. Liang, "React: Synergizing reasoning and acting in language models," *arXiv preprint arXiv:2210.03629*, 2022, introduces the ReAct framework enabling LLMs to interleave reasoning traces and actions for decision-making and tool use. [Online]. Available: https://arxiv.org/abs/2210.03629

[63] Y. Yang, H. Chai, Y. Song, S. Qi, M. Wen, and N. Li, "A survey of ai agent protocols," *arXiv preprint arXiv:2504.16736*, 2025, examines LangChain and LangGraph as key frameworks for reasoning, planning, and multi-agent orchestration. [Online]. Available: https://arxiv.org/abs/2504.16736

[64] M. Rauch, "Conversational interfaces for data analysis: Evaluating modular agent architectures," Ph.D. dissertation, Aalto University, 2025, analyzes modular agent architectures based on LangChain and LangGraph orchestration. [Online]. Available: https://aaltodoc.aalto.fi/items/ac2011cb-bb17-44dd-a19b-e0537662b3d9

[65] J. G. Mathew and J. Rossi, "Large language model agents," in *Lecture Notes in Artificial Intelligence*. Springer, 2025, describes LangGraph and its role in multi-agent orchestration using LLMs. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-031-92285-5_8

[66] K. T. Tran, D. Dao, M. D. Nguyen, and Q. V. Pham, "Multi-agent collaboration mechanisms: A survey of llms," *arXiv preprint arXiv:2501.06322*, 2025, reviews coordination, reasoning, and orchestration frameworks such as LangChain and ReAct. [Online]. Available: https://arxiv.org/abs/2501.06322

[67] J. Tang, T. Fan, and C. Huang, "Autoagent: A fully-automated and zero-code framework for llm agents," *arXiv preprint arXiv:2502.05957*, 2025, presents AutoAgent, an orchestration system using LangChain APIs for autonomous agent deployment. [Online]. Available: https://arxiv.org/abs/2502.05957

[68] G. A. de Aquino, N. S. de Azevedo, and L. Y. S. Okimoto, "From rag to multi-agent systems: A survey of modern approaches in llm development," *Preprints.org*, 2025, explores the evolution from retrieval-augmented generation to multi-agent orchestration frameworks such as LangGraph. [Online]. Available: https://www.preprints.org/manuscript/12d92f418fc17b4bd3e6b6144acf951c

[69] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[70] D. J. Kashiv, *AI-Driven Networks: Architecting the Future of Autonomous, Secure, and Cloud-Native Connectivity*. Wiley, 2025, discusses multi-agent reinforcement learning architectures and closed-loop automation systems that bridge the insight-to-action cycle. [Online]. Available: https://books.google.com/books?id=BNZlEQAAQBAJ

[71] J. U. C. Nwoke, "Leveraging ai-powered optimization, risk intelligence, and insight automation for agile organizational growth," *International Journal of Research and Innovation in Social Science*, vol. 9, no. 10, pp. 112–119, 2025. [Online]. Available: https://www.researchgate.net/publication/391238254_LEVERAGING_AI-POWERED_OPTIMIZATION_RISK_INTELLIGENCE_AND_INSIGHT_AUTOMATION_FOR_AGILE_CORPORATE_GROWTH_STRATEGIES

[72] LangChain, "Langgraph: Building stateful, multi-actor applications with llms," https://langchain-ai.github.io/langgraph/, 2024, accessed: 2025-11-20.

[73] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[74] H. Kamarzarin, M. B. Shamloo, and M. Abbasi, "The study of the effectiveness of implementing the mind simulation technique on reducing moderate and severe depression symptoms," *Medical Research Archives*, 2025. [Online]. Available: https://www.researchgate.net/profile/Hamid-Kamarzarin/publication/395281939_The_Study_of_the_Effectiveness_of_Implementing_the_Mind_Simulation_Technique_on_Reducing_Moderate_and_Severe_Depression_Symptoms/links/68f93284220a341aa15702e0/The-Study-of-the-Effectiveness-of-Implementing-the-Mind-Simulation-Technique-on-Reducing-pdf

[75] Joint Task Force for the Development of Telepsychology Guidelines, "Guidelines for the practice of telepsychology," *American Psychologist*, vol. 68, no. 9, p. 791, 2013.

[76] C. Zheng, H. Zhou, F. Meng, J. Zhou, and M. Huang, "Large language models are not robust multiple choice selectors," *arXiv preprint*, 2023. [Online]. Available: https://arxiv.org/abs/2309.03882

[77] Langfuse, "Langfuse: Open source llm engineering platform," https://langfuse.com/, 2024, accessed: 2025-11-20.

APPENDIX

This appendix provides the technical artifacts that underpin the Safety Agent Suite's implementation. It includes the repository information for reproducibility, the core system prompts that define the agents' behavior, and the key algorithms used for graph-based reasoning.

L.1 Repository Information

The complete source code for the UGM-AICare project, including the agentic backend, frontend interface, and deployment configurations, is available in the following GitHub repository.

- **Repository URL:** `https://github.com/gigahidjrikaaa/UGM-AICare.git`
- **Version Referenced:** v1.0-release
- **License:** MIT License

Project Structure

The project follows a microservices architecture. The key directories relevant to this thesis are:

- `backend/app/agents/`: Contains the LangGraph state machine definitions and agent workflows.
- `backend/app/agents/sta/`: Contains the Safety Triage Agent logic and Gemini classifiers.
- `backend/app/agents/aika_orchestrator_graph.py`: The master orchestrator graph definition.
- `backend/app/core/llm.py`: Contains the LLM interaction logic and system prompts.

L.2 System Prompts

The following prompts define the core cognitive behavior of the agents. They are presented here to provide transparency into how the agents are instructed to handle safety-critical tasks.

L.2.1 Safety Triage Agent (STA) Classification Prompt

This prompt is used by the Safety Triage Agent to analyze user messages for risk levels using a Chain-of-Thought (CoT) reasoning process.

```
Kamu adalah spesialis triage krisis kesehatan mental untuk mahasiswa Indonesia.

Analyze pesan ini untuk risiko kesehatan mental menggunakan EXPLICIT STEP-BY-STEP REASONING.

**Pesan Saat Ini:**
"{message}"

**Konteks Percakapan Sebelumnya:**
{history_str}

**ANALISIS SISTEMATIS:**

**STEP 1 - KATA KUNCI KRISIS:**
List kata-kata eksplisit yang indicate krisis (bunuh diri, self-harm, death wishes, mention metode).
Quote exact phrases dari pesan.

**STEP 2 - POLA LINGUISTIK:**
Check untuk: finality language, past-tense life review, goodbye statements, hopelessness.
Explain apa yang kamu temukan.

**STEP 3 - TONE EMOSIONAL:**
Rate negative valence (0-10). Look for: despair, defeat, emptiness, isolation.
Provide evidence dari pesan.

**STEP 4 - SINYAL URGENSI:**
Check untuk: immediacy ("hari ini", "sekarang", "malam ini"), rencana konkret, time constraints.
List apa yang kamu temukan.

**STEP 5 - FAKTOR PROTEKTIF:**
Look for: rencana masa depan, mention support, help-seeking, ambivalence, humor.
Note kalau ada.
```

```

**STEP 6 - FAKTOR KONTEKSTUAL:**
Consider: stigma kesehatan mental Indonesia, tekanan akademik (konteks UGM), norma budaya.
Gimana budaya affect interpretasi?

**STEP 7 - KEBUTUHAN DUKUNGAN:**
Apakah user butuh:
- calm_down: teknik manajemen anxiety/panic
- break_down_problem: bantuan dengan complexity yang overwhelming
- general_coping: strategi stress management
- none: nggak perlu plan immediate

**STEP 8 - KLASIFIKASI FINAL:**
Berdasarkan steps 1-7, classify:
- risk_level: 0 (low), 1 (moderate), 2 (high), 3 (critical)
- intent: crisis_support | acute_distress | academic_stress | relationship_strain | general_support
- next_step: human (escalate) | tca (coaching) | resource (self-help)
- confidence: 0.0-1.0 (seberapa yakin kamu?)

Weight factors:
- Kata kunci/pola krisis: immediate level 3
- Multiple distress signals: level 2
- Single stressor + coping: level 1
- Casual/safe: level 0

Return as JSON:
{
  "step1_crisis_keywords": ["list", "of", "keywords"],
  "step2_linguistic_patterns": "description",
  "step3_emotional_tone": {"score": 7, "evidence": "quotes"},
  "step4_urgency_signals": ["list"],
  "step5_protective_factors": ["list"],
  "step6_cultural_context": "notes",
  "step7_support_needs": "calm_down | break_down_problem | general_coping | none",
  "step8_classification": {
    "risk_level": 2,
```

```
    "intent": "acute_distress",
    "next_step": "tca",
    "confidence": 0.85,
    "reasoning": "brief explanation of decision"
  }
}
```

Listing 1. System Prompt for Safety Triage Classification

L.2.2 Aika Orchestrator Decision Prompt

This prompt is used by the Aika Meta-Agent to decide whether to handle a query directly or invoke specialized agents.

```
Analyze this message and determine if specialized safety agents are needed.

User Role: {user_role}
Message: {state["message"]}

Decision Criteria:

FOR STUDENTS (user):
- ALWAYS HANDLE DIRECTLY (needs_agents=false):
  * Aika is the primary responder for ALL student interactions.
  * Aika handles emotional support, crisis de-escalation, and appointment booking directly using tools.
  * DO NOT invoke specialized agents (STA/TCA/CMA) synchronously.
  * Background processes will handle deep risk analysis later.

FOR ADMINS:
- NEEDS AGENTS (invoke IA for analytics):
  * Requests complex data/analytics ("trending topics", "case statistics")
  * Aggregated reports requiring specialized processing

- NO AGENTS NEEDED:
  * Simple status checks ("is system healthy?")
  * General platform questions
  * Specific user lookups (Aika can use tools for this)

FOR COUNSELORS:
```

```
- NEEDS AGENTS (invoke CMA for case management):
  * Requests to CREATE or MODIFY cases
  * Clinical insights requiring deep analysis

- NO AGENTS NEEDED:
  * General clinical questions
  * Viewing patient data (Aika can use tools)

Return JSON with:
{
  "intent": "string (MUST be one of: 'casual_chat', 'emotional_support', 'crisis_intervention', 'information_inquiry', 'appointment_scheduling', 'emergency_escalation', 'analytics_query')",
  "intent_confidence": float (0.0-1.0),
  "needs_agents": boolean,
  "next_step": "string (MUST be one of: 'tca', 'cma', 'ia', 'sta', 'none')",
  "reasoning": "string explaining decision",
  "suggested_response": "string (only if needs_agents=false, provide direct response)",

  "immediate_risk": "none|low|moderate|high|critical",
  "crisis_keywords": ["list of crisis keywords found, empty if none"],
  "risk_reasoning": "Brief 1-sentence explanation of risk assessment",

  "analytics_params": {
      "question_id": "string (MUST be one of: 'crisis_trend', 'dropoffs', 'resource_reuse', 'fallback_reduction', 'cost_per_helpful', 'coverage_windows')",
      "start_date": "YYYY-MM-DD (default to 30 days ago if not specified)",
      "end_date": "YYYY-MM-DD (default to today if not specified)"
  }
}
```

Listing 2. System Prompt for Orchestrator Decision

L.2.3 Therapeutic Coach Agent (TCA) Prompt

This prompt demonstrates how the TCA generates personalized, actionable intervention plans (e.g., for "Calm Down" interventions).

```
Kamu adalah coach kesehatan mental yang expert dalam manajemen anxiety dan panic. Peran kamu adalah bantuin user untuk calm down ketika mereka experiencing anxiety, panic, atau stress yang overwhelming.

Generate personalized support plan dengan 3-5 langkah spesifik dan actionable yang:
1. Bantu grounding user di present moment
2. Kurangi gejala fisiologis (jantung berdebar, napas cepat, dll.)
3. Kasih teknik coping yang immediate
4. Culturally sensitive dengan konteks Indonesia/Asia
5. Pakai bahasa yang clear, compassionate, non-clinical

REQUIREMENTS PENTING:
- Setiap step harus immediately actionable (nggak vague)
- Include durasi waktu spesifik (misal "5 menit", "3 napas dalam")
- Pakai tone yang warm dan encouraging
- Hindari jargon medis
- Consider situasi spesifik dan context user

Output format (JSON):
{
  "plan_steps": [
    {"id": "step1", "label": "Tarik napas dalam 5 kali - hirup 4 hitungan, tahan 4, hembuskan 6", "duration_min": 2},
    {"id": "step2", "label": "Sebutin 5 hal yang kamu lihat sekarang untuk grounding diri", "duration_min": 3}
  ],
  "resource_cards": [
    {"resource_id": "breathing", "title": "Latihan Napas Terpandu", "summary": "Follow pola napas yang calming", "url": "https://aicare.example/calm/breathing"}
  ]
}
```

Listing 3. System Prompt for TCA Plan Generation (Calm Down)

L.2.4 Case Management Agent (CMA) Prompt

This prompt illustrates how the CMA intelligently matches students with the most suitable counselor based on case severity and preferences.

```
Kamu adalah koordinator appointment kesehatan mental. Pilih psikolog
    yang PALING COCOK untuk case ini.

Konteks Case:
- Severity: {severity}
- Preferensi Mahasiswa: {json.dumps(preferences)}

Psikolog yang Available:
{json.dumps(psych_profiles, indent=2)}

Kriteria Pemilihan:
1. Untuk case CRITICAL: Prioritas experience dan high ratings
2. Match specialization kalau student punya concern spesifik
3. Consider preferensi bahasa
4. Prefer psikolog dengan jadwal availability yang defined

Return HANYA psychologist ID (integer) dari pilihan kamu.
```

Listing 4. System Prompt for CMA Counselor Selection

L.2.5 Insights Agent (IA) Prompt

This prompt guides the Insights Agent in interpreting anonymized analytics data to provide actionable recommendations for university administrators.

```
Anda adalah asisten analitik data untuk platform kesehatan mental
    mahasiswa UGM-AICare.

Tugas Anda:
1. Menganalisis data statistik yang telah dianonimkan
2. Mengidentifikasi tren dan pola penting
3. Memberikan insight yang actionable untuk administrator
4. Merekomendasikan intervensi berdasarkan data

Format Respons:
- Gunakan bahasa Indonesia yang profesional
- Fokus pada insight praktis
- Sertakan angka spesifik dari data
- Berikan rekomendasi yang dapat ditindaklanjuti

Catatan Privasi:
- Semua data sudah dianonimkan dan diagregasi
- Tidak ada informasi individual mahasiswa
```

```
- Mengikuti standar k-anonymity (k >= 5)
```

Listing 5. System Prompt for IA Analytics Interpretation

L.3 Core Algorithms

L.3.1 LangGraph State Schema

The following Python TypedDict defines the shared state that flows through the agent graph, ensuring type safety and consistent data passing between the Orchestrator, STA, TCA, and CMA.

```python
class AikaOrchestratorState(TypedDict, total=False):
    """State for the unified Aika orchestrator graph."""

    # INPUT CONTEXT
    user_id: int
    user_role: Literal["user", "counselor", "admin"]
    session_id: str
    message: str
    conversation_history: List[Dict[str, str]]

    # AIKA DECISION NODE OUTPUTS
    intent: Optional[str]
    needs_agents: bool
    aika_direct_response: Optional[str]
    agent_reasoning: Optional[str]

    # STA OUTPUTS (Safety Triage)
    risk_level: Optional[int]
    severity: Optional[Literal["low", "moderate", "high", "critical"]]
    next_step: Optional[str]

    # TCA OUTPUTS (Therapeutic Coach)
    intervention_plan: Optional[Dict[str, Any]]
    should_intervene: bool

    # CMA OUTPUTS (Case Management)
    case_id: Optional[int]
    case_created: bool
    assigned_counsellor_id: Optional[int]

    # FINAL RESPONSE
    final_response: Optional[str]
    response_source: Optional[Literal["aika_direct", "agents"]]

    # EXECUTION TRACKING
    execution_id: Optional[str]
```

```
    execution_path: List[str]
    agents_invoked: List[str]
```

Listing 6. Aika Orchestrator State Definition

L.3.2 Orchestrator Graph Construction

This algorithm defines the conditional routing logic of the Aika Orchestrator, demonstrating how the system dynamically chooses between direct responses and agent invocations.

```
def create_aika_unified_graph(db: AsyncSession) -> StateGraph:
    """Create unified Aika orchestrator graph."""

    workflow = StateGraph(AikaOrchestratorState)

    # Add nodes
    workflow.add_node("aika_decision", partial(aika_decision_node, db=db))
    workflow.add_node("execute_sta", partial(execute_sta_subgraph, db=db))
    workflow.add_node("execute_sca", partial(execute_sca_subgraph, db=db))
    workflow.add_node("execute_sda", partial(execute_sda_subgraph, db=db))
    workflow.add_node("synthesize", partial(synthesize_final_response, db=db))

    # Entry point
    workflow.set_entry_point("aika_decision")

    # Conditional routing after Aika decision
    workflow.add_conditional_edges(
        "aika_decision",
        should_invoke_agents,
        {
            "invoke_cma": "execute_sda",  # Immediate crisis escalation
            "invoke_sta": "execute_sta",
            "end": END
        }
    )

    # Conditional routing after STA
    workflow.add_conditional_edges(
        "execute_sta",
        should_route_to_sca,
```

```
        {
            "invoke_sca": "execute_sca",
            "route_sda": "execute_sda",
            "synthesize": "synthesize"
        }
    )

    # Terminal nodes
    workflow.add_edge("execute_sda", "synthesize")
    workflow.add_edge("synthesize", END)

    return workflow
```

Listing 7. LangGraph Construction Logic