Diploma Project

Design and Development of a Smartphone-Based Mental Health Companion App for Early Detection of Depression and Anxiety

Lydia Paschalidou

University of Cyprus
Department of Computer Science

Department of Computer Science

May 2026

UNIVERSITY OF CYPRUS

DEPARTMENT OF COMPUTER SCIENCE

Design and Development of a Smartphone-Based Mental Health Companion App for Early Detection of Depression and Anxiety

Lydia Paschalidou

Supervisor: Dr. Giorgos Papadopoulos

Co-Supervisor: Christos Mettouris

The Individual Diploma Thesis was submitted towards partially meeting the requirements for obtaining the degree of Computer Science of the Department of Computer Science of the University of Cyprus

May 2026

Declaration

I declare that this dissertation and any accompanying software are my own original work, conducted in full compliance with the University of Cyprus Code of Conduct for Research (https://www.ucy.ac.cy/legislation/kodikas-deontologias-kai-politikes/), and with full accountability on my part for the accuracy, integrity, and validity of all content.

Name: Lydia Paschalidou

Date: May 2026

Acknowledgements

Completing this thesis would not have been possible without the generous support, guidance, and encouragement provided to me along the way.

First, my heartfelt thanks go to my supervisor, Dr. Giorgos Papadopoulos, for trusting me with this project and creating the opportunity to explore such a meaningful topic. I also extend sincere gratitude to my co-supervisor Christos Mettouris, whose ongoing support and excellent collaboration throughout the thesis preparation were truly invaluable.

Lastly, I am deeply thankful to my family, friends, and colleagues for their unwavering emotional support and encouragement that sustained me through this challenging journey.

Abstract

MoodSense is a smartphone-based self-monitoring application for Android that passively collects behavioural data throughout the day and combines it with brief daily self-reports to help users understand patterns in their own mood and well-being. The application tracks physical activity, screen usage, sleep-related inactivity, and, where the user consents, location mobility, communication frequency, and physiological signals from a connected wearable via Android Health Connect. Over time, MoodSense constructs a personal behavioural baseline for each individual and surfaces simple, readable insight cards when a metric deviates meaningfully from that baseline. It does not produce diagnoses, clinical assessments, or population-relative scores. It offers context, reflection, and pattern visibility to the user in plain, supportive language.

The system is designed around four guiding principles: simplicity, trust, personalisation, and low burden. A local-first architecture ensures that raw sensor data never leaves the device. Only structured daily summaries and mood entries are optionally synchronised to a Firebase cloud account under the user's own identifier. All optional sensors are disabled by default, and the user can delete all local data or sign out at any time from the Settings screen.

The application was evaluated by a group of participants who used MoodSense for a minimum of three days. Their experience was assessed using a structured questionnaire based on the User Experience Questionnaire (UEQ), supplemented by questions on technical problems, permission concerns, and open-ended feedback. The results indicate that participants found the application easy to use, clear, and supportive, and that the daily check-in and behavioural insights were the most valued features. The rationale for sensitive permissions, particularly calls, SMS logs, and location access, was identified as the primary area for improvement in future iterations.

TABLE OF CONTENTS

Acknowledgements **ii**

Abstract **iii**

TABLE OF CONTENTS **iv**

LIST OF FIGURES **vii**

LIST OF TABLES **viii**

1 Introduction **1**

1.1 Overview 1
1.2 Motivation 2
1.3 About the Application 2
1.4 Aims and Objectives 3
1.5 Structure of the Thesis 4

2 Background **5**

2.1 Introduction 5
2.2 Depression and Anxiety 5
2.3 Wearables, IoT, and Mobile Devices 6
2.4 Smartphone Sensors 7
2.5 Digital Phenotyping 8
2.6 Privacy and Ethical Considerations 9

3 Related Work **10**

3.1 Introduction 10
3.2 Smartphone-Based Mental Health Monitoring 10
3.3 Wearables and Passive Physiological Sensing 12
3.4 Mobile Mental Health Applications 13
3.5 Research Gaps and Positioning of MoodSense 15
3.6 Summary 16

4 **System Design and Application Design** **17**
4.1 Overview 17
4.2 Use Case Modelling 18
4.3 System Requirements 20
4.3.1 Functional Requirements 20
4.3.2 Non-Functional Requirements 22
4.4 System Architecture 24
4.5 Data Model 26
4.6 User Interface Design 27
4.7 Privacy by Design 31
4.8 Data and Interaction Flow 33

5 **Implementation** **35**
5.1 Overview 35
5.2 Technologies and Tools 35
5.3 Application Architecture in Practice 37
5.4 Key Modules and Features 37
5.4.1 Passive Data Collection 37
5.4.2 Daily Check-in 40
5.4.3 Baseline Computation and the Weekly Summary 41
5.4.4 Insight Engine 41
5.4.5 Settings and Privacy Controls 42
5.4.6 Daily Reminder Notifications 43
5.4.7 Data Export 44
5.4.8 Health Connect Integration 44
5.5 Challenges and Solutions 45

6 **Testing and Evaluation Methodology** **48**
6.1 Introduction 48
6.2 Evaluation Objectives 48
6.3 Participants 48
6.4 Procedure 50
6.5 Questionnaire Design 51
6.6 Ethical Considerations 52

7 **Evaluation Results and Discussion** **53**
7.1 Introduction 53
7.2 User Experience Ratings 53

7.2.1 Item-Level Results . . . 53
7.2.2 Subscale Results . . . 54
7.3 Technical Problems . . . 55
7.4 Permission Concerns . . . 56
7.5 Qualitative Feedback . . . 57
7.5.1 Most Useful Features . . . 57
7.5.2 Positive Themes . . . 58
7.5.3 Improvement Suggestions . . . 58
7.6 Discussion . . . 59
7.6.1 Usability and Clarity . . . 60
7.6.2 Privacy and Trust . . . 60
7.6.3 Perceived Usefulness and Self-Awareness . . . 61
7.6.4 Limitations . . . 61

8 Conclusion **62**
8.1 Final Remarks . . . 62
8.2 Future Work . . . 63

BIBLIOGRAPHY **65**

APPENDICES **68**

Appendix A: Evaluation Questionnaire **69**

LIST OF FIGURES

4.1 Use Case Diagram of MoodSense 18
4.2 High-level System Architecture of MoodSense 24
4.3 Data Model of MoodSense 27
4.4 MoodSense Application Screens: Welcome, Login, Dashboard, and Daily Check-in 28
4.5 MoodSense Application Screens: Trends, Check-in History, and Insights 29

LIST OF TABLES

4.1 MoodSense Use Cases 19
4.2 Functional Requirements 21
4.3 Non-Functional Requirements 23
4.4 Data Sources in MoodSense 25
4.5 Privacy by Design in MoodSense 32

6.1 Participant profile (n = 21) 49
6.2 Frequency of mood reflection or tracking before MoodSense (n = 21) 50
6.3 Prior use of mood, sleep, or wellbeing apps (n = 21) 50
6.4 Semantic differential adjective pairs used in the evaluation 51

7.1 UEQ item scores (corrected, 1–7; n = 21) 53
7.2 UEQ subscale scores (3 to +3; n = 21) 54
7.3 Technical problems reported by participants 55
7.4 Permission concerns reported by participants (n = 21) 56
7.5 Features identified as most useful (multiple selections permitted) 57

1 Introduction

1.1 Overview

Mental health difficulties rarely announce themselves clearly. Depression and anxiety tend to develop gradually, shaped by small but persistent shifts in everyday behavior: changes in sleep, in movement, in how often a person reaches for their phone or connects with others. These patterns are easy to overlook, particularly when they unfold slowly over weeks rather than appearing as a single, identifiable moment [1, 2].

Smartphones and wearable devices occupy an unusual intersection of everyday life and behavioral measurement. Because people carry them continuously, these devices can passively capture a range of behavioral signals without placing any deliberate burden on the user. Movement patterns, screen usage, sleep duration, communication frequency, and location variety are all observable through the sensors and logs already built into the hardware. Research in digital phenotyping has demonstrated that patterns drawn from this kind of data can reflect meaningful changes in psychological state, sometimes before the person themselves has fully recognised that something has shifted [3–5].

This thesis presents the design and development of MoodSense, a smartphone-based mental health companion application. MoodSense is not a diagnostic instrument. It is an early self-awareness tool that helps users notice changes in their own behavioral patterns before those changes become harder to ignore. The application combines passive sensing from phone usage and movement data with lightweight user input in the form of brief daily mood check-ins. Where a compatible wearable device is connected through Android's Health Connect platform, additional metrics such as heart rate and sleep duration can be incorporated to enrich the behavioral picture [2, 6].

A core design principle of MoodSense is personalisation over fixed thresholds. Behavioral signals do not carry the same meaning for every person. A drop in physical activity that is entirely unremarkable for one individual may represent a significant change for another, and the same applies to shifts in sleep, screen time, or communication patterns. Rather than comparing users against population-level norms, MoodSense tracks deviations from each individual's own established behavioral baseline, making the insights it surfaces more personally meaningful and less prone to misinterpretation [2, 4].

1.2 Motivation

Despite being among the most widespread health conditions worldwide, depression and anxiety are frequently identified late. In many cases, meaningful behavioral changes have already taken hold, quietly affecting sleep, physical activity, social engagement, and daily routine, well before a person seeks support or receives any formal assessment [1, 2]. This gap between early behavioral change and eventual recognition represents one of the more pressing challenges in mental health care, and it is a challenge that technology may be genuinely well placed to address.

The devices most people already carry offer a practical foundation for continuous behavioral observation. Smartphones log movement, screen interactions, communication patterns, and location throughout the day, often without any deliberate input from the user. Where users have a compatible wearable and choose to connect it, additional physiological signals such as heart rate and sleep duration become available through Health Connect, building a more complete picture of behavioral state [3–5]. When observed carefully over time, these sources can reveal behavioral trends associated with periods of psychological distress, not as a replacement for clinical judgment, but as a way of bringing gradual changes into view before they become more difficult to address.

Existing research, however, reveals persistent limitations. Many studies rely on small samples monitored over short periods, which makes it difficult to draw reliable conclusions about longer-term behavioral patterns [1, 4]. Practical systems frequently raise concerns about privacy, transparency, and intrusiveness, issues that carry particular weight in mental health contexts, where users are understandably cautious about what data they share and how it is handled [5, 7]. And while research prototypes have demonstrated the technical feasibility of passive sensing, translating those prototypes into applications that people genuinely want to use in their everyday lives remains a significant challenge [5, 6].

MoodSense is motivated by these gaps, grounded in the belief that a well-designed everyday tool can make gradual behavioral change visible before it becomes difficult to address [4, 5].

1.3 About the Application

MoodSense is a smartphone-based mental health companion designed to support the early recognition of behavioral changes that may be associated with depression, anxiety, or general psychological distress. Its role is that of a personal self-monitoring aid: one that helps users build awareness of their own patterns over time and notice when those patterns begin to shift [1, 5].

Behavioral data enters the application through two complementary channels. Passive sensing draws on signals already available through the smartphone, including step count, screen usage, sleep duration, and communication frequency, all collected without requiring any active effort from the user. Brief daily check-ins covering mood and stress provide a second, self-reported layer that gives context to the patterns the sensors detect, the check-in also records the user's location, which the application uses to assess location variety across days as a behavioural indicator. Users who have a compatible wearable device can optionally connect it through Android's Health Connect platform, which makes heart rate and sleep duration data available to supplement the phone-based signals [2–4].

From the user's perspective, MoodSense presents information through a simple and accessible interface. Rather than displaying raw sensor measurements, the application offers clear summaries, trend visualizations, and contextual feedback that help users interpret what their own data means over time. Research on self-tracking technologies consistently shows that users engage more meaningfully with systems that offer genuine insight rather than uninterpreted numbers [5–7].

Privacy and user control are central to the design rather than peripheral additions. The application is built on the principle that users should understand what is being collected, why it is collected, and how it is used. Data is processed locally where possible, features can be enabled or disabled according to individual preference, and no raw sensor data is shared externally without explicit consent. These choices reflect the understanding that trust is a prerequisite for any tool intended to support mental well-being [5, 7].

1.4 Aims and Objectives

The aim of this thesis is to design, develop, and evaluate a smartphone-based application that supports early self-awareness of behavioral changes associated with depression, anxiety, and related forms of psychological distress. Rather than attempting to diagnose mental health conditions, the system is intended to help users observe meaningful shifts in their own daily patterns and reflect on them in a timely and understandable way.

To support this aim, the project pursues the following objectives:

- To review the literature on digital phenotyping, passive smartphone sensing, and wearable-assisted mental health monitoring.
- To identify behavioral indicators that can be feasibly and ethically monitored through a smartphone, with optional enrichment from connected wearable devices.
- To design a privacy-aware and user-friendly mobile application architecture for continuous behavioral self-monitoring.

- To implement the MoodSense prototype on Android, integrating passive sensing, daily self-report check-ins, and optional Health Connect data where a compatible wearable is available.
- To evaluate the application in terms of usability, clarity, perceived usefulness, and user trust.

The outcome of these objectives is MoodSense: a fully functional Android application evaluated with real users, grounded in an individual-baseline model, and designed to operate without dedicated hardware. The thesis also documents the design decisions and evaluation findings in a way that can inform future work in the same space.

1.5 Structure of the Thesis

This thesis is organized into eight chapters. Following this introduction, the second chapter covers the theoretical and technical background that underpins the work: depression and anxiety in the context of digital health, smartphone sensing and wearable technologies, digital phenotyping as the conceptual framework connecting sensing capabilities to mental health monitoring, and the privacy and ethical considerations relevant to systems of this kind. The third chapter reviews existing research on smartphone and wearable-based mental health monitoring, examining key studies on passive sensing and behavioral indicators of psychological distress, comparing their approaches and findings, and identifying the gaps this thesis aims to address.

The design of MoodSense is covered in the fourth chapter, which presents the main use cases, functional and non-functional requirements, system architecture, user interface prototypes, and the privacy by design principles that guided key decisions throughout the planning phase. This is followed by the implementation chapter, which explains the selected technologies, software structure, and main system modules, while also discussing practical challenges encountered during development and the solutions adopted to address them.

The sixth chapter outlines the testing and evaluation methodology, including the study objectives, participant profile, evaluation procedure, selected metrics, and the ethical considerations relevant to a study conducted in a mental health context. The seventh chapter presents the evaluation results and discusses their implications in relation to the project goals and the broader literature, acknowledging the limitations of the current work. The eighth and final chapter concludes the thesis with a summary of contributions and directions for future development.

2 Background

2.1 Introduction

Building a smartphone-based mental health application requires grounding in several interconnected areas. The system draws on knowledge from clinical psychology, mobile computing, sensor technology, and privacy-aware design, and understanding how these areas relate to one another is essential before examining the design and development decisions that follow [3, 4].

This chapter introduces the main concepts and technical foundations that support this work. It begins with depression and anxiety as the central mental health conditions of interest, discussing how they manifest behaviorally and why that makes them particularly relevant to digital monitoring. It then examines the role of smartphones, wearable devices, and IoT technologies in capturing behavioral and physiological signals from everyday life, before introducing digital phenotyping as the guiding framework that ties these sensing capabilities to mental health monitoring [1, 2, 5].

The chapter closes with a discussion of the privacy and ethical considerations that any responsible mental health monitoring system must address. Because applications of this kind collect sensitive and continuous personal data, questions of transparency, informed consent, user control, and responsible data handling are not peripheral concerns but foundational to the design of a system that people can genuinely trust [5, 7].

2.2 Depression and Anxiety

Depression and anxiety are not simply states of low mood or occasional worry. They are recognized clinical conditions that can significantly disrupt daily functioning, affecting sleep quality, concentration, energy levels, social engagement, and a person's overall sense of routine and stability. Although they are distinct in their presentation, they frequently co-occur, and their behavioral manifestations often overlap in ways that make early recognition both important and difficult [1, 2, 5].

What makes these conditions particularly relevant to digital health research is that they tend to leave traces in everyday behavior long before they are formally identified. Depression is commonly associated with reduced physical activity, increased time spent at home, lower communication frequency, and a withdrawal from previously normal routines. Anxiety, by contrast, is more often linked to sleep disturbance, irregular phone use, heightened nighttime activity, and difficulty maintaining stable patterns of attention and engagement. Neither set of signals is conclusive on its own, but when observed together and over time, they can serve as meaningful early indicators of a change in psychological state [2–4].

This behavioral dimension is what makes smartphones a relevant monitoring tool. Because these devices are present throughout the day, they can passively capture exactly the kinds of signals that depression and anxiety tend to affect, including movement, location variety, communication patterns, sleep duration, and screen usage. This creates an opportunity to observe gradual changes in a more continuous and naturalistic way than periodic clinical questionnaires alone would allow [1, 3, 5].

At the same time, it is important to maintain a clear sense of what digital monitoring can and cannot do. Depression and anxiety are complex conditions shaped by biology, environment, personal history, and circumstance. Behavioral changes captured by a smartphone may have many explanations entirely unrelated to mental health, including illness, academic pressure, or simple changes in routine. Digital systems should therefore be understood as tools for surfacing patterns and encouraging reflection, not as instruments capable of diagnosis. This distinction is foundational to the ethical design of any mental health monitoring application [1, 5, 7].

2.3 Wearables, IoT, and Mobile Devices

The growing availability of smartphones, wearable devices, and connected IoT technologies has opened new possibilities for health monitoring outside of clinical settings. Unlike traditional assessments that depend on periodic appointments or self-administered questionnaires, these devices can observe behavior continuously and in the context of everyday life, making them particularly well suited to detecting the kind of gradual changes that characterize conditions like depression and anxiety [3, 4].

Smartphones and wearables each contribute different but complementary types of data. Smartphones capture behavioral signals such as screen usage, app activity, communication patterns, movement, and location. Wearables add physiological measurements including heart rate, heart rate variability, sleep duration, and physical activity intensity. When used together, these sources can offer a more complete picture of a person's daily routine and emotional state than either could provide independently [1, 2, 4].

A key advantage of these technologies is that they are already embedded in most people's daily lives, which makes passive sensing practical at scale. Data can be collected over extended periods without deliberate action from the user, which matters particularly for mental health monitoring, where behavioral changes often unfold over weeks rather than appearing as discrete events [3, 5].

These technologies also introduce real challenges that cannot be ignored. Continuous sensing can feel intrusive, data quality varies across device types and manufacturers, and battery consumption remains a practical concern in long-term deployments. In a mental health context, users tend to be particularly sensitive about how their data is handled, making trust, transparency, and user control essential design considerations from the outset [5, 7, 8].

Their value ultimately depends not only on the signals they can collect, but on how responsibly that data is processed and communicated back to the user [4, 5].

2.4 Smartphone Sensors

Modern smartphones contain a range of built-in sensors and usage logs that can provide a detailed picture of everyday behavioral patterns. For mental health monitoring purposes, the most relevant of these are the accelerometer, GPS, communication logs, and screen interaction data, each of which captures a different dimension of the user's daily routine [3, 4].

The accelerometer measures physical movement and can be used to estimate activity levels, restlessness, and periods of inactivity throughout the day. In the context of depression, reduced movement and prolonged inactivity are among the more consistently reported behavioral indicators. In anxiety, irregular or heightened movement patterns may also be relevant. Because the accelerometer operates continuously in the background, it can track these patterns over time without requiring any input from the user [2, 3].

GPS data adds a spatial and routine-based dimension to this picture. Spending more time at home, visiting fewer locations, or showing a narrower daily movement range may all reflect withdrawal or decreased engagement with everyday life. Research suggests that these mobility patterns are most informative when interpreted as deviations from a person's own established baseline, since individual routines vary considerably [2, 4].

Communication behavior, captured through call logs, message frequency, and social app usage, provides another useful signal. A reduction in outgoing communication or decreased interaction with others can reflect social withdrawal, a recognized behavioral marker of depression. Changes in the timing of social interactions may also correlate with shifts in mood or stress levels [1, 3, 6].

Screen usage patterns offer a further layer of behavioral information. Metrics such as total screen time, unlock frequency, and late-night phone activity can reveal disruptions to sleep routine or signs of restlessness. Increased nighttime phone use in particular has been associated with poor sleep quality and anxiety, and these patterns are notable because they often emerge before the user themselves has registered that something has changed [2, 4, 9].

Despite their value, smartphone sensors have real limitations. Data can be noisy or inconsistent across different devices and operating system versions, and behavioral changes captured by sensors always have multiple possible explanations beyond mental health. Smartphone sensor data is therefore most useful when treated as one layer within a multi-source system, combined with self-reports and wearable physiological data where available [1, 5, 7].

2.5 Digital Phenotyping

Digital phenotyping is the practice of using data from personal digital devices to build a continuous picture of an individual's behavior, cognitive functioning, and social patterns over time. The term was introduced to describe a shift away from one-off clinical appointments and questionnaires toward a more ongoing and naturalistic form of observation, one that captures how a person actually lives rather than how they report feeling at a single point in time [3, 5].

In mental health research, the appeal of this approach lies in its sensitivity to change that happens slowly. Symptoms of depression and anxiety rarely appear overnight and tend to build across days and weeks, often showing first in subtle behavioral shifts that the person themselves may not yet have consciously registered. By tracking multiple signals simultaneously and comparing them against each individual's own established patterns, digital phenotyping can surface these early deviations in a way that one-off assessments cannot [1, 2, 5].

The practical value of this approach increases significantly when multiple data sources are combined. A single dropped signal, such as fewer steps on a given day, carries little meaning on its own. But when reduced activity occurs alongside increased time at home, decreased communication, and disrupted sleep patterns, the combination begins to tell a more coherent story. Research consistently shows that multi-signal models outperform single-signal approaches in identifying behavioral changes associated with psychological distress [4, 6].

Crucially, digital phenotyping systems are designed to support awareness and reflection rather than to render clinical verdicts. Rather than telling a user they are depressed, a well-designed system might instead observe that their step count has dropped notably over the past two weeks and that they have been spending more time at home than usual, and invite them to reflect on why. Users have reported finding these kinds of pattern-based observations genuinely helpful, particularly when they are presented clearly and without judgment [5, 7].

Because the data involved is continuous, personal, and often collected passively without moment-to-moment awareness, digital phenotyping raises serious questions about privacy, consent, and data responsibility that are central to any system intended for mental health monitoring [1, 5, 7].

2.6 Privacy and Ethical Considerations

Mental health monitoring applications occupy a particularly sensitive position in the privacy landscape. The data they collect, including movement, location, communication patterns, screen behavior, and sleep, may appear unremarkable in isolation. But when combined and analyzed over time, these signals can reveal deeply personal information about a person's psychological state, social relationships, and daily struggles. Under frameworks such as the General Data Protection Regulation (GDPR), health-related and behavioral data of this kind is classified as sensitive personal data, carrying stricter obligations around collection, processing, and storage than ordinary personal information [5, 7].

A foundational challenge is that passive data collection often happens without the user's active awareness. Unlike filling out a questionnaire, where responding makes the data collection visible, passive sensing runs continuously in the background. This makes informed consent more complex than a simple one-time acceptance of terms. Users need to understand not only what data is being collected, but how it is processed, what insights it generates, and what the limitations of those insights are. Because these systems support self-awareness rather than diagnosis, this distinction must be communicated clearly to avoid misleading expectations [1, 7].

Privacy by Design offers a principled approach to embedding these protections into a system's architecture from the outset rather than adding them as an afterthought [5]. Applied to a mental health application, this means building in data minimization so that only the signals genuinely necessary for the application's purpose are collected. It means using local processing so that sensitive behavioral data is handled on-device where possible rather than transmitted to external servers. And it means providing meaningful user control so that individuals can review, adjust, or delete their data at any time. These are not merely compliance measures but expressions of respect for the user's autonomy.

Fairness and potential bias are also important considerations. Behavioral patterns vary widely across cultures, age groups, socioeconomic backgrounds, and device types. A system calibrated on one population may misinterpret entirely normal behavioral diversity in another as a warning sign. Mental health applications should therefore focus on individual baseline deviations rather than population-level comparisons to reduce this risk [1, 7].

Finally, developers behind mental health applications carry a responsibility to document their data practices clearly, establish retention and deletion policies, and take seriously the potential for misuse in consumer contexts. The risk of sensitive behavioral data being repurposed or accessed without consent is real and demands deliberate safeguards. By treating privacy and ethics as design requirements rather than compliance checkboxes, these systems can become tools people genuinely trust [5, 7].

3 Related Work

3.1 Introduction

The use of smartphones and wearable devices to monitor mental health has become an active area of research over the past decade, driven by the dual recognition that conditions like depression and anxiety are both widespread and chronically underdetected. Researchers have explored a range of approaches, from passive sensing platforms that collect behavioural data continuously in the background, to CBT-based mobile applications that engage users through structured daily interactions, to large-scale longitudinal studies that attempt to establish reliable relationships between sensor signals and psychological state. Together, this body of work has demonstrated both the feasibility and the complexity of mobile mental health monitoring, and it reveals a recurring set of limitations that motivate the design of MoodSense.

This chapter reviews the most relevant research across three areas. The section on smartphone-based mental health monitoring examines empirical studies that trace the evolution of mobile sensing from early feasibility work through to more recent large-scale longitudinal research. The section on wearables and passive physiological sensing reviews how the addition of heart rate, sleep staging, and activity data strengthens the behavioural picture that smartphones alone can provide. The section on mobile mental health applications discusses both their contributions and the significant gap that separates most commercial tools from the passive sensing capabilities explored in research. The final sections draw these threads together to identify the specific limitations and open questions that this thesis addresses and offer a brief synthesis.

3.2 Smartphone-Based Mental Health Monitoring

The case for using smartphones as mental health monitoring platforms rests on a straightforward observation: people carry these devices throughout their day, and the behavioural signals they generate, movement, location, screen use, communication, and sleep-related inactivity, tend to be precisely the signals that depression and anxiety affect. The challenge has been to determine whether these signals are reliable enough, and interpretable enough, to support meaningful monitoring in practice.

One of the earliest and most influential demonstrations of this possibility was the StudentLife study conducted at Dartmouth College by Wang and colleagues [10]. Over a ten-week university term, the study used a continuous sensing app on the Android phones of 48 students to track sleep, physical activity, mood, sociability, and conversation levels alongside academic performance and validated mental health measures. The results showed a clear and sustained pattern across the cohort: as the term progressed and academic pressure intensified, stress increased while sleep, physical activity, conversation frequency, and positive affect all declined. The study was significant not only for its findings but for demonstrating the feasibility of longitudinal, naturalistic behavioural monitoring through a smartphone without requiring active effort from participants. It established the methodological foundation on which much subsequent work has been built.

Building on this foundation, Wang et al. later examined depression dynamics more directly in a college population, combining mobile phone data with wearable sensing to track depressive symptom changes over time [11]. This work highlighted that passive sensing could detect not only the presence of depressive states but their longitudinal trajectory, an important distinction for any system concerned with early recognition rather than diagnosis at a single point in time.

A particularly informative feasibility study was conducted by Boonstra and colleagues, who deployed the Socialise app with 32 participants who had lived experience of mental health challenges [3]. The app collected Bluetooth proximity, GPS, and battery data over four weeks. Despite the modest technical scope, the study yielded important practical insights: sensor data was successfully collected for 55 % of scheduled scans on Android devices and 45 % on iOS, while continuous scanning reduced battery life by approximately 12 %, from an average of 21.3 to 18.8 hours. Although participants described the battery impact as noticeable, the overall acceptability of the system was strongly shaped by factors that had nothing to do with the sensors themselves. Trust in the organisation collecting the data, transparency about why the data was being gathered, and perceived control over privacy were reported as the primary determinants of whether participants were comfortable using the app at all. These findings were an early and clear signal that passive sensing acceptability is as much a social and design problem as a technical one.

More recently, Meyerhoff and colleagues conducted a large-scale longitudinal study with 282 Android users followed over 16 weeks, examining how smartphone sensor data related to changes in depression and anxiety symptoms over time [2]. By clustering participants according to their symptom profiles, the study found that sensor changes tended to precede symptom changes rather than merely coincide with them, a finding with direct implications for early recognition systems. Notably, GPS-derived mobility features, including the number of locations visited and time spent at home, were among the most reliable predictors of depression severity changes, with correlation coefficients in the range of $r = -0.20$ to $r = -0.36$ across different participant groups. Increased home time relative to a participant's own average emerged as an early and sustained signal of worsening depression across observation windows of two weeks, one week, and concurrent assessment [12]. These findings strengthen the case for personal baselines: the predictive value of mobility data came not from comparing people against one another, but from tracking deviations from each individual's own established patterns.

A study by Terhorst and colleagues reinforced this picture by showing that reduced communication frequency, irregular usage patterns, heavier engagement with entertainment and social media applications, and lower levels of physical movement were all independently associated with higher depression severity [6]. Crucially, models that combined multiple features substantially outperformed those that relied on any single signal. The study also highlighted the practical limitations of this approach: data was collected only on Android devices, sensor availability varied considerably across manufacturers, and model accuracy, while encouraging, remained moderate, reflecting the inherent challenge of capturing a complex psychological state through behavioural proxies alone.

Taken together, these studies trace a clear trajectory. Early work demonstrated that smartphone sensing was feasible and that behavioural signals correlated with mental health indicators. More recent work has refined this picture considerably, identifying the sensors most predictive of specific conditions, establishing the longitudinal time scales over which changes are detectable, and demonstrating that individual baselines are more informative than population-level thresholds. What remains comparatively rare is the translation of these insights into systems designed for general use rather than controlled research settings.

3.3 Wearables and Passive Physiological Sensing

While smartphone sensors capture behavioural patterns at the level of daily activity and routine, wearable devices add a physiological dimension that behavioural data alone cannot easily provide. Heart rate, heart rate variability, sleep duration and staging, and fine-grained activity intensity all contribute information that is difficult to infer from GPS or screen logs, and their combination with smartphone behavioural data has repeatedly produced stronger associations with mental health indicators than either source on its own.

A comprehensive review by Shen and colleagues examined 42 peer-reviewed studies that used passive sensing from smartphones and wearables with machine learning to monitor clinically relevant mental health conditions [4]. Across this literature, a reliable core feature package emerged: step count, accelerometer data, heart rate, and sleep metrics appeared most frequently and were among the strongest predictors of depression, anxiety, and psychological stress. Multi-feature models outperformed single-sensor approaches in the large majority of studies, and simpler machine learning methods such as logistic regression and random forests often achieved results comparable to more complex architectures, suggesting that the quality and breadth of input features matters more than the sophistication of the model. The review also identified serious methodological concerns: only 14 % of studies addressed data anonymisation explicitly, fewer than half reported sufficient information to assess the risk of overfitting, and the majority relied on small samples monitored over short periods.

A related review focusing on the *Frontiers in Digital Health* literature examined wearable, environmental, and smartphone-based sensors together, finding that physiological signals such as heart rate variability and skin conductance added meaningful sensitivity for detecting stress and anxiety states that behavioural mobility data alone would miss [13]. The study concluded that combining sensing modalities was not simply additive but synergistic, with physiological and behavioural signals capturing different temporal scales of change: physiological signals tended to reflect acute states while behavioural signals tracked more gradual trends.

Longitudinal engagement with wearable-based research systems has also been studied directly. The RADAR-MDD study, a large multinational observational study of depression using smartphones and Fitbit devices, followed 614 participants across sites in the United Kingdom, Spain, and the Netherlands for up to two years [14]. Of particular note was the relationship between engagement and symptom severity: participants with the highest depression severity were disproportionately represented among the least engaged group, responding to survey notifications more slowly and dropping out at higher rates. This creates a systematic bias in passive sensing research that is rarely acknowledged directly. If the people most affected by the condition being studied are also the least likely to maintain engagement with the monitoring system, the resulting data may underrepresent the very population the system is designed to help.

From a platform perspective, research-grade tools such as the Beiwe system developed at the Onnela Lab at Harvard have demonstrated the technical feasibility of large-scale, HIPAA-compliant digital phenotyping using smartphones and wearables [15]. The HOPES platform extended this foundation by integrating a wider sensor suite and adding clinical dashboards, demonstrating the architecture required for a complete digital phenotyping system [16]. While these platforms are valuable for academic research, they are designed for controlled clinical studies rather than everyday consumer use, and they do not prioritise the kind of accessible, self-explanatory interface that a general-population application requires.

3.4 Mobile Mental Health Applications

Alongside the research literature on passive sensing, a substantial market of consumer-facing mental health applications has developed. Understanding how these applications work, and where they fall short, is essential for positioning MoodSense within the broader landscape.

The most widely studied and deployed applications in this space, including Woebot and Wysa, are built primarily around conversational AI and cognitive-behavioural therapy techniques. Fitzpatrick and colleagues conducted a randomised controlled trial of Woebot with 70 participants, finding that two weeks of engagement with the chatbot produced significant reductions in depression symptoms as measured by the PHQ-9, compared to an information-only control group [17]. The conversational format proved engaging and accessible, and participants reported finding the tool useful as a supplement to or substitute for formal support. Wysa and comparable tools have shown similarly positive short-term outcomes in several small trials [18].

These applications have genuine strengths. They are accessible around the clock, low in cost, and evidence-based in their therapeutic frameworks. However, they share a fundamental limitation: they rely entirely on what users actively report. There is no passive sensing, no collection of behavioural data from the phone's sensors, and no trend analysis over time. A user who is withdrawing, sleeping poorly, and moving less will not see any of these patterns reflected in an application that responds only to what they choose to type. For users experiencing the low motivation and reduced engagement that often accompany depression, this dependency on active participation is a structural vulnerability.

Other applications, such as Moodpath and Bearable, have moved in a different direction, emphasising mood and symptom tracking through daily self-reports and offering trend visualisations over time. Analysis of user reviews of depression self-management applications found that app-based tracking genuinely increased users' sense of self-awareness and supported self-management when the features matched individual needs and goals [19]. Users repeatedly requested clearer visualisations, more personalisation, and greater control over what data was tracked and how it was displayed. At the same time, most of these applications still do not collect passive sensor data, meaning that the behavioural picture they construct is limited to what users remember and choose to report at the moment they open the app.

The AWARE framework represents a research-facing platform that does implement passive sensing across a wide range of sensors, including GPS, accelerometer, screen events, communication, and application usage [20]. Studies using AWARE have produced valuable research findings, but the platform is not designed as a consumer product. It requires technical configuration, produces outputs oriented toward researchers rather than users, and provides no self-facing insight layer that would help someone make sense of their own behavioural patterns.

The landscape that emerges from reviewing these applications is one of a persistent gap. On one side are research platforms and academic studies that implement sophisticated passive sensing but require technical expertise, are not designed for everyday use, and focus on researchers rather than users. On the other side are consumer applications that are well-designed, accessible, and engaging but limit themselves almost entirely to self-report. The combination of passive behavioural monitoring, personal baseline analysis, and accessible self-facing feedback that MoodSense proposes occupies a space that the existing landscape has not adequately addressed.

3.5 Research Gaps and Positioning of MoodSense

Four recurrent limitations emerge from this body of work, each of which directly informs a design decision in MoodSense.

The most prominent methodological concern across passive sensing studies is the reliance on small samples observed over short periods. Many studies in this domain involve fewer than one hundred participants monitored for days or weeks, which is insufficient to establish stable individual behavioural baselines, particularly for conditions that manifest differently across people and circumstances [1, 4]. The work of Meyerhoff and colleagues stands as an exception in its scale and duration, but even sixteen weeks falls short of what would be needed to establish truly stable long-term baselines for a general population.

A second gap concerns the near-universal reliance on population-level comparisons in both research systems and consumer applications. The predictive value of behavioural signals has been shown repeatedly to depend not on absolute levels but on deviations from individual norms, yet most systems continue to apply fixed thresholds or train classifiers on aggregated population data. This means that a naturally sedentary person may appear distressed to a population-calibrated system even during a period of psychological stability, while an ordinarily very active person whose activity has halved might pass undetected. The personal baseline approach proposed in MoodSense addresses this directly, reflecting findings from Meyerhoff and others that intra-individual change, rather than between-person comparison, is the more informative signal [2, 12].

Privacy and transparency have been chronically underemphasised in both the research and commercial literature. Boonstra and colleagues found that perceived privacy impact and trust in the collecting organisation were the dominant factors in acceptability, yet most systems treat these as secondary concerns addressed through terms-of-service rather than through the design of the application itself [3, 7]. The systematic review by Shen and colleagues found that fewer than one in six studies explicitly addressed data anonymisation, reflecting how marginal privacy has been as a design priority in this field [4].

User engagement and long-term retention remain underexplored problems. Traditional mental health apps are known to lose the majority of their users within a matter of weeks, and passive sensing research studies face the same challenge in a more acute form: the RADAR-MDD study demonstrated that the most severely affected participants disengaged at disproportionately high rates, creating a structural sampling bias that calls the generalisability of findings into question [14]. A system that loses its most vulnerable users first cannot claim to serve them effectively, and this consideration shapes the design priorities of MoodSense toward simplicity, low burden, and non-judgmental feedback.

Finally, the gap between research prototypes and consumer-ready applications remains wide. Sophisticated systems like Beiwe, AWARE, and HOPES have demonstrated what is technically possible, but they are designed for researchers and clinical study teams, not for individuals managing their own mental health in everyday life. Consumer apps like Woebot and Wysa have achieved broad adoption but sacrifice behavioural sensing entirely in favour of conversational interaction. MoodSense occupies the space between these poles: a system that implements passive behavioural monitoring and personal baseline analysis in an interface designed for ordinary users, with privacy and user control treated as foundational rather than incidental.

3.6 Summary

The literature reviewed in this chapter establishes a clear picture of what the field has achieved and where it has not yet arrived. Smartphone and wearable sensing can reliably capture behavioural patterns relevant to depression and anxiety, and multi-signal models that track intra-individual change over time outperform simpler approaches.The practical challenges of doing this well, including short study durations, small samples, privacy concerns, and the persistent gap between research systems and consumer-ready tools, remain genuinely open. MoodSense is designed in direct response to these challenges: a privacy-aware, baseline-driven, user-facing application that brings passive behavioural monitoring out of the research environment and into everyday use.

4 System Design and Application Design

4.1 Overview

The design of MoodSense is grounded in four guiding principles: simplicity, trust, personalisation, and low burden. Each of these reflects a deliberate response to the limitations identified in the related work. Systems that place too much demand on users tend to lose them. Systems that present data without explanation tend to confuse them. Systems that collect sensitive behavioural data without transparency tend to lose their trust. MoodSense is designed to avoid each of these pitfalls by being clear about what it does, sparing in what it demands from the user, and honest about what its insights mean and what they do not.

At its core, MoodSense is a self-awareness tool. It collects behavioural data passively from the smartphone throughout the day and invites the user to supplement that data with brief daily self-reports. Over time, it constructs a picture of each individual's behavioural patterns and surfaces meaningful deviations as simple, readable insights. It does not produce diagnoses, scores, or clinical assessments. It offers context, reflection, and pattern visibility to the user, framed in language that is supportive rather than clinical [5, 7].

MoodSense is designed for Android smartphone users who want to better understand their own behavioural patterns in the context of their mental well-being. The intended user is any adult who experiences or is curious about the relationship between their daily habits and their mood, particularly those who would benefit from a low-effort, non-judgmental tool that works quietly in the background. The application does not assume any prior familiarity with mental health monitoring or self-tracking technology, and it is designed to be equally accessible to younger adults who are comfortable with smartphone technology and to older users who may be less so. No wearable device is required, although the experience is enriched for users who have a compatible smartwatch connected through Android Health Connect.

This chapter describes how that design was realised. It begins with the use case modelling that defines what the system does from the user's perspective, then moves through the functional and non-functional requirements that constrain and shape the design, the system architecture that connects the components, the data model that structures what is stored, the user interface design that makes the system accessible, and the privacy-by-design principles embedded throughout. The chapter closes with an overview of the main data and interaction flows that link the sensing, logging, and feedback components together.

4.2 Use Case Modelling

MoodSense is designed around a single primary actor: the smartphone user who wishes to monitor their own behavioural patterns over time. There is no clinical operator, no therapist portal, and no external data consumer. The user is both the subject of the monitoring and the intended beneficiary of its output, and this shapes every design decision in the system.

The use case diagram in Figure 4.1 gives a structural overview, while Table 4.1 details the trigger and outcome for each case.

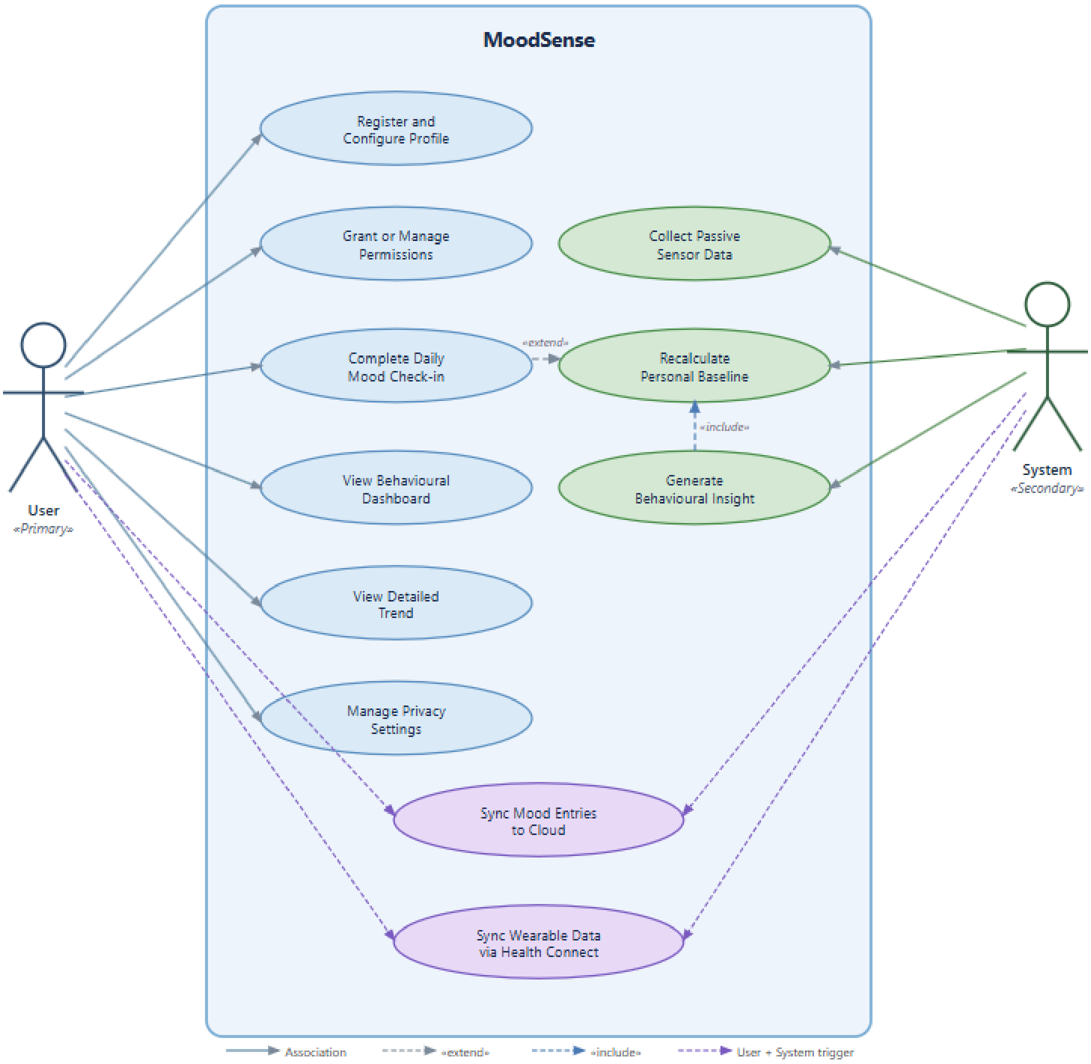

Figure 4.1: Use Case Diagram of MoodSense

Table 4.1: MoodSense Use Cases

Use Case	Actor	Trigger	Expected Outcome
Register and configure profile	User	First app launch	Onboarding completed. User preferences and permission choices saved
Grant or manage permissions	User	Onboarding or Settings screen	Selected sensors enabled. Optional sensors remain off by default
Complete daily mood check-in	User	Daily notification or manual entry	Mood, stress, sleep quality, and energy recorded. Self-report and daily context stored
View behavioural dashboard	User	Opens app home screen	Daily summary data and weekly overview displayed
View detailed trend	User	Opens Trends screen	Mood and stress graph, sleep quality and energy graph, steps and screen time card, and check-in history displayed
Manage privacy settings	User	Opens Settings screen	Sensors toggled. Data deletion or sign-out initiated if requested
Collect passive sensor data	System	Background sensing and periodic tasks	Sensor readings captured and summarised for the current day
Recalculate personal baseline	System	Dashboard refresh or periodic maintenance task	Rolling baseline updated. Deviations available for insight generation
Generate behavioural insight	System	Baseline comparison detects meaningful deviation	Insight card created and surfaced on the Insights screen
Sync mood entries to cloud account	System / User	Successful check-in while signed in	Self-report and attached daily summary synced to Firebase under the user's account
Sync wearable data via Health Connect	System / User	Background task or user request	Heart-rate and sleep duration data retrieved from Health Connect and incorporated into summaries

The most important distinction in this use case structure is the separation between user-initiated and system-initiated actions. The user's role is minimal and voluntary: they configure the app once, complete brief daily check-ins, and interact with the dashboard when they choose. Everything else, including most data collection, baseline computation, and insight generation, happens automatically in the background or when the app is opened. This separation reflects a deliberate low-burden design approach. Users are only asked to provide brief daily input, everything else runs automatically. This matters because prior work has shown that apps requiring consistent active effort see substantially higher abandonment rates among the very populations they are intended to support [5, 14].

4.3 System Requirements

4.3.1 Functional Requirements

The functional requirements describe what MoodSense must do. They are organised by area and assigned a priority using the MoSCoW method, where **Must** indicates a core requirement without which the system does not meet its purpose, **Should** indicates a high-priority requirement that meaningfully strengthens the system, and **Could** indicates a desirable enhancement that adds value but is not essential to the core experience.

Table 4.2: Functional Requirements

ID	Requirement	Priority
FR-01	The system shall collect step count and physical activity data passively from Android sensors or Health Connect	Must
FR-02	The system shall collect screen usage time passively, subject to the user's permission	Must
FR-03	The system shall determine sleep duration using sleep session records from Health Connect where available, falling back to phone inactivity periods recorded locally when Health Connect data is unavailable	Must
FR-04	The system shall store all collected self-reports and behavioural summaries locally and associate them with the current user account	Must
FR-05	The system shall allow the user to complete a daily mood, stress, sleep quality, and energy check-in	Must
FR-06	The system shall compute a rolling personal behavioural baseline for each tracked metric	Must
FR-07	The system shall generate simple insight cards when a metric deviates from its personal baseline	Must
FR-08	The system shall display trend visualisations for mood and stress, sleep quality and energy, and steps and screen time over selectable time periods	Must
FR-09	The system shall allow the user to enable or disable each optional data source independently (e.g. location, call and SMS logs, wearable data)	Must
FR-10	The system shall allow the user to delete their locally stored data at any time and to sign out of their cloud account	Must
FR-11	The system shall collect GPS-based mobility data when the user has enabled location tracking	Should
FR-12	The system shall collect communication frequency data (call and message counts) when the user has granted the relevant permissions	Should
FR-13	The system shall retrieve heart-rate data from Android Health Connect when a compatible wearable is connected and the user has opted in	Should
FR-14	The system shall send a daily reminder notification to prompt mood check-in completion	Should
FR-15	The system shall provide an onboarding flow explaining what is collected and why before any data collection begins	Must
FR-16	The system shall present a weekly behavioural summary to the user on the dashboard	Should
FR-17	The system shall allow the user to export their own data in a readable format for personal use or research participation	Should

4.3.2 Non-Functional Requirements

Non-functional requirements define the quality constraints within which the system must operate. These are particularly significant for MoodSense because the application runs continuously in the background and handles sensitive personal data, making performance, privacy, and reliability essential rather than secondary concerns.

Table 4.3: Non-Functional Requirements

ID	Requirement	Category	Rationale
NR-01	Raw high-frequency behavioural data (e.g. screen events, GPS traces, detailed heart-rate samples) must remain on the device. Only summarised records and self-reports may be synced to the user's cloud account	Privacy	Reduces exposure of sensitive continuous data while preserving the ability to back up and export meaningful summaries
NR-02	The application must not reduce device battery life by more than approximately 5 % under normal usage conditions	Performance	Battery impact was a key acceptability concern in passive sensing research [3]
NR-03	The user interface must be navigable by a first-time user without instruction	Usability	Self-monitoring tools must be accessible to non-technical users [5]
NR-04	All insight text and trend labels must be written in plain, non-clinical language	Clarity	Avoids creating distress or misinterpretation. Supports the self-awareness rather than diagnostic purpose
NR-05	Core features (check-ins, dashboard, summaries) must function fully offline, with no dependency on network connectivity	Reliability	Ensures consistent access regardless of network conditions. Cloud sync is treated as an enhancement rather than a requirement
NR-06	Background sensing and reminders must use Android-appropriate mechanisms (foreground services and WorkManager) and comply with system restrictions	Reliability	Ensures consistent data availability while respecting platform power-management constraints
NR-07	The onboarding flow must clearly explain each data type collected and each permission requested before any collection begins	Transparency	Informed consent requirement. Foundational to ethical design in mental health applications
NR-08	The system must handle missing data gracefully, skipping gaps rather than inferring or fabricating values	Data integrity	Sensor gaps are common across Android devices. Graceful degradation protects baseline accuracy
NR-09	The application must request only permissions that are actively used and must link each permission to a clear user-facing benefit	Privacy	Principle of data minimisation. Avoids unnecessary access
NR-10	The user interface must follow Material Design guidelines and support standard Android accessibility features	Accessibility	Ensures the application is usable across different devices and user needs

4.4 System Architecture

MoodSense follows a hybrid, local-first architecture in which all sensing, processing, and insight generation occur on the user's device, while summarised mood entries are optionally synchronised to a cloud backend under the user's account. Raw, high-frequency behavioural streams such as detailed screen usage, GPS traces, and individual heart-rate samples are never sent to the server. Only daily summaries and self-reports are transmitted, and only when the user is signed in. This design seeks to balance the privacy requirements of a mental health application with the practical advantages of secure cloud backup and export.

The architecture is organised into four principal layers, illustrated in Figure 4.2. The **presentation layer** renders the dashboard, trend screens, check-in interface, and settings using Jetpack Compose, observing data changes through ViewModels that follow the MVVM architectural pattern [21]. The **ViewModel layer** handles baseline computation, deviation detection, and insight generation through domain components such as `InsightEngine`. The **repository layer** is responsible for querying device sensors and external data sources, including the step counter, screen usage, optional GPS-based mobility tracking, Health Connect integration, and repositories for communication data. The **local storage layer** manages persistence of settings and mood entries through Android DataStore and a local database, with optional synchronisation to Firebase Realtime Database under the authenticated user's account.

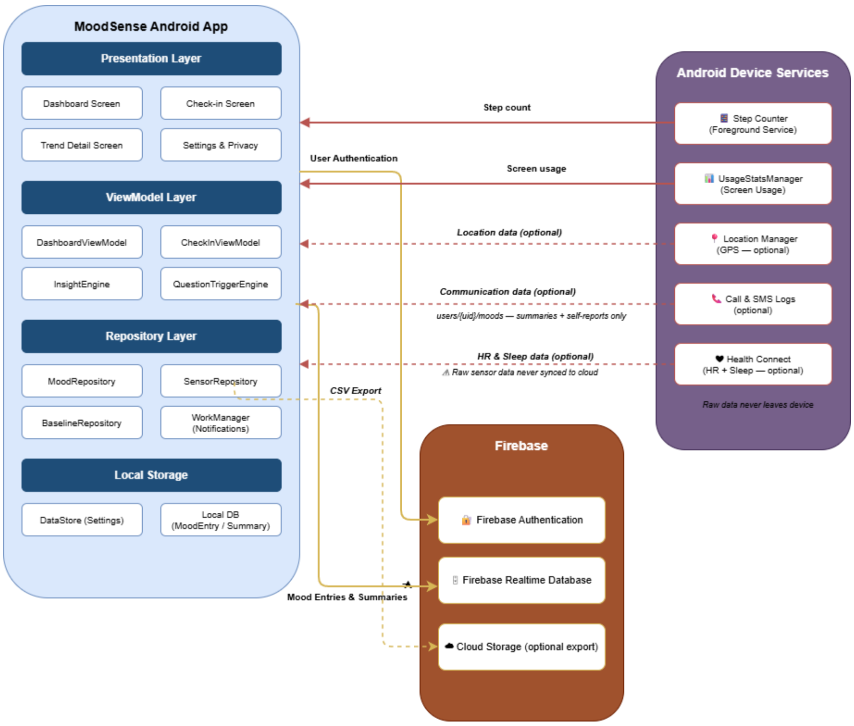

Figure 4.2: High-level System Architecture of MoodSense

Table 4.4 describes each data source used by MoodSense, the signals it provides, how those signals are used within the application, and any limitations that apply.

Table 4.4: Data Sources in MoodSense

Data Source	**Signals Collected**	**Use in MoodSense**	**Limitations / Notes**
Android step counter and activity sensors	Daily step count, activity episodes	Physical activity tracking and mobility baseline	Availability and accuracy vary by device. Some devices batch step counts
Usage statistics and screen events	Screen-on duration, app session lengths	Screen behaviour tracking	Requires usage access permission. Some devices restrict background access
Phone inactivity (derived)	Periods of extended screen-off and no interaction	Sleep duration fallback when Health Connect data is unavailable	Indirect measure. Confounded by the device being left idle while the user is active
GPS / location services	Location coordinates, time spent stationary	Low-mobility detection. Extended stationary periods linked to low mood	Optional. Permission state controls both data collection and UI display
Call and message logs	Call counts, SMS counts	Communication behaviour tracking and social withdrawal signals	Optional and sensitive. Requires clear explanation and explicit permissions
Daily self-report (user input)	Mood (1–5), stress (1–5), sleep quality (1–5), energy (1–5), free-text notes	Self-reported context. Supplements passive data and anchors interpretations	Dependent on user compliance. Missed entries create gaps in the record
Android Health Connect	Heart-rate samples, sleep duration, resting heart rate	Physiological data from compatible wearables. Enriches activity and sleep insights	Works with devices that integrate with Health Connect. Huawei Health currently does not, which limits coverage on Huawei wearables
Firebase Realtime Database	Daily mood entries and attached behavioural summaries	Secure cloud storage and export of user-level records	Stores only summarised data under the user's authenticated account. No raw sensor streams

Background sensing is implemented using a combination of Android components. A foreground service tracks steps continuously with minimal battery impact, while WorkManager handles reliable periodic tasks including daily reminder notifications, scheduled through a `PeriodicWorkRequest` with a calculated initial delay that aligns delivery to the user's configured notification time. If the scheduled time has already passed for the current day, the first notification is automatically deferred to the following day. Baseline computations and insight generation run on demand when the dashboard or trends screens are loaded, ensuring that the user always sees the most recent interpretation of their data without requiring heavy scheduled jobs.

4.5 Data Model

MoodSense organises its persistent data around two core structures: the `MoodEntry` and the `TodaySummary`. A `MoodEntry` represents a single completed check-in and contains the user's self-reported mood, stress, sleep quality, and energy scores, any additional free-text notes, a timestamp, and a reference to the `TodaySummary` captured at the moment of submission. A `TodaySummary` encapsulates the passive sensor readings for that calendar day: total steps, screen-on duration in minutes, estimated sleep duration, call and SMS counts where permitted, and heart-rate statistics where available from Health Connect.

Both structures are stored locally on the device and, when the user is signed in, synchronised to Firebase Realtime Database under a path keyed to the user's unique identifier (`users/{uid}/moods`). Raw sensor readings are never written to Firebase. Only the structured daily records produced by the check-in flow are transmitted. The `WeeklySummary` is a derived structure computed on demand from the last seven `MoodEntry` records and is never persisted independently.

The `InsightCard` is a transient structure generated by the `InsightEngine` when a metric deviates meaningfully from its personal baseline. It carries a type, title, detail text, severity level, and priority score, and is surfaced on the Insights screen without being persisted to local storage or Firebase. Figure 4.3 illustrates the relationships between these structures.

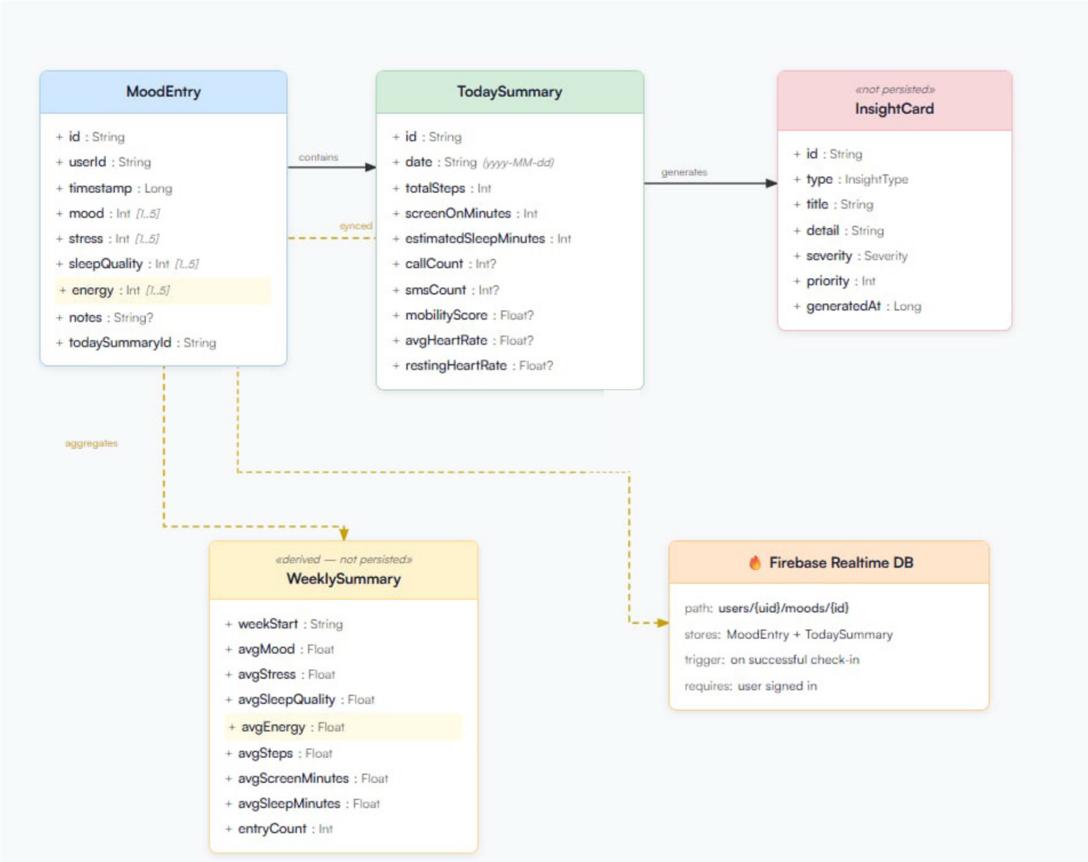

Figure 4.3: Data Model of MoodSense

4.6 User Interface Design

The user interface of MoodSense is organised around five primary screens accessible through the bottom navigation bar: Dashboard, Mood Check-in, Trends, Insights, and Resources. The design prioritises clarity and legibility over density, keeping each screen focused on a single purpose and avoiding the presentation of raw sensor numbers that users would have no natural framework to interpret.

Figures 4.4 and 4.5 show screenshots of the running application across the primary screens.

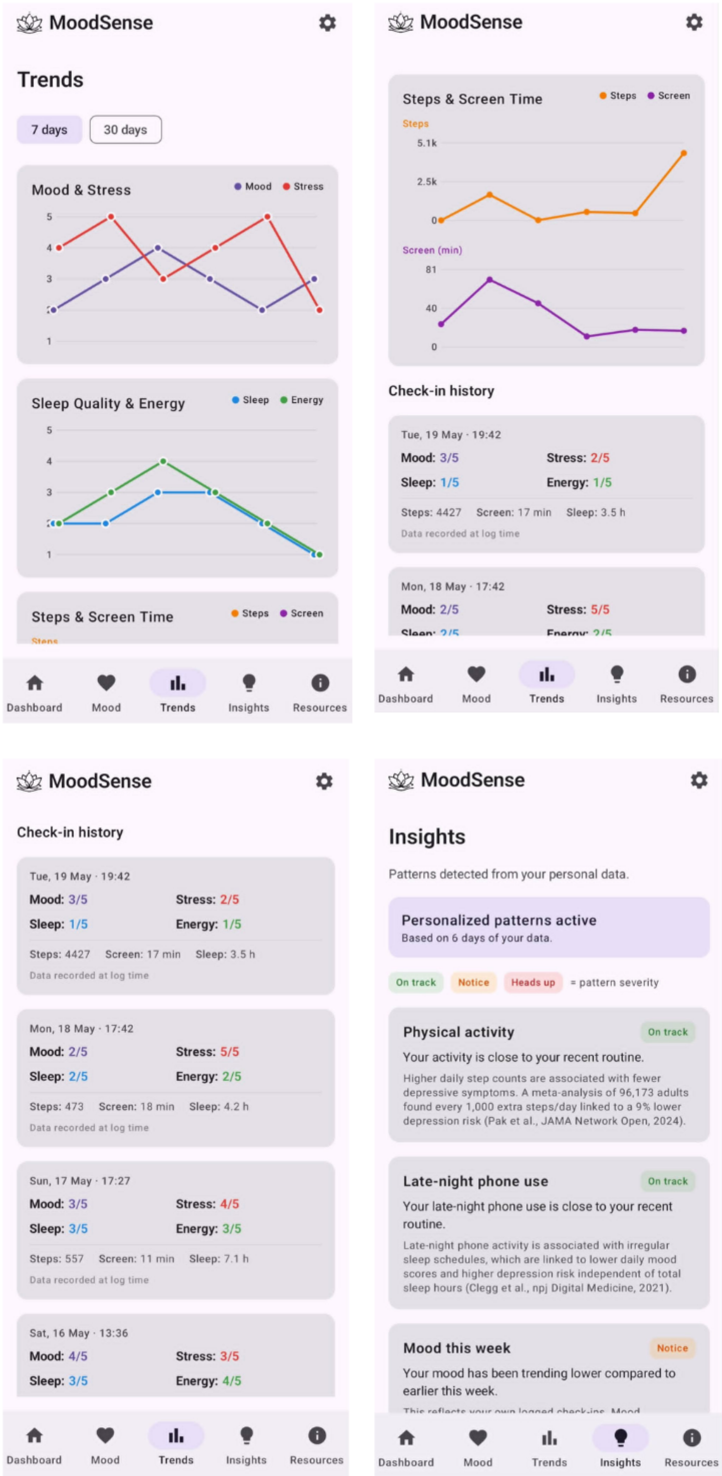

Figure 4.4: MoodSense Application Screens: Welcome, Login, Dashboard, and Daily Check-in

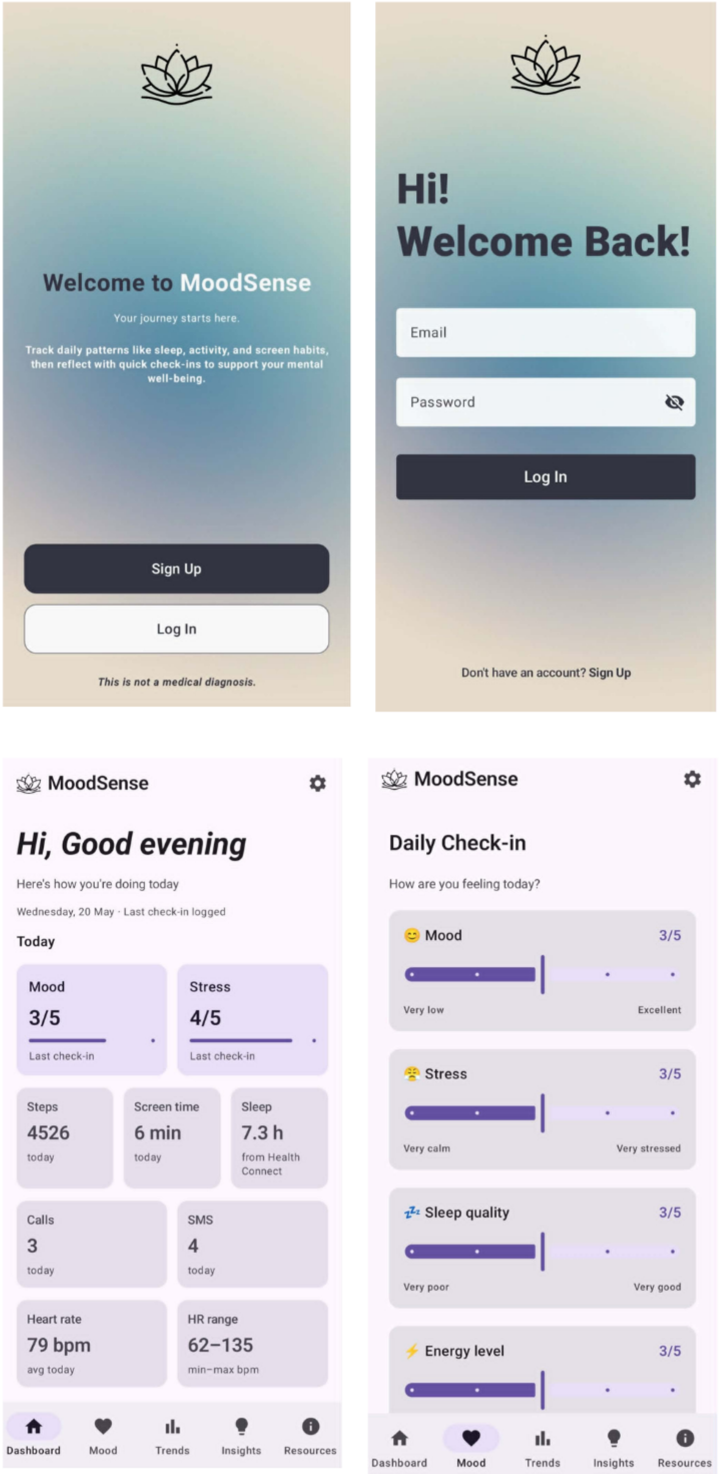

Figure 4.5: MoodSense Application Screens: Trends, Check-in History, and Insights

Welcome and Login

The onboarding flow consists of two screens. The Welcome screen presents a brief statement of the application's purpose (tracking daily patterns such as sleep, activity, and screen habits) and provides entry points to sign up or log in. The Login screen follows standard credential-based authentication with email and password, and includes a link to account registration for new users. Both screens use the application's ambient gradient background to establish a calm, non-clinical tone before the user reaches any data.

Dashboard

The Dashboard is the home screen and the primary point of interaction for most users on most days. It greets the user by time of day and displays a weekly summary card showing this week's behavioural averages, including mood, stress, sleep, steps, screen time, and heart rate where available. The design deliberately avoids scores, percentages relative to population norms, or any phrasing that could be interpreted as a clinical assessment.

Daily Check-in

The Daily Check-in screen presents four questions covering mood, stress level, sleep quality, and energy level, each answered on a labelled five-point scale ranging from very negative to very positive. Users may also add free-text notes, which are entirely optional. The check-in is designed to be completable in under thirty seconds. A daily notification prompts the user at a time they configure during onboarding. Missing a check-in has no consequence. The system records what is available and treats missing entries as gaps rather than defaults.

Trends and Check-in History

The Trends screen displays line graphs for each tracked metric over a selectable period (7 days or 30 days). The metrics shown include Mood and Stress, Sleep Quality and Energy, and Steps and Screen Time, each pair sharing a single chart to allow visual comparison. The same screen also includes a scrollable log of past check-in entries below the charts. Each entry shows the date and time of submission alongside the values recorded for mood, stress, energy, and sleep, as well as the passive metrics captured that day. This allows users to review their own history in full without relying solely on aggregated trend charts.

Insights

The Insights screen surfaces the personalised pattern cards generated by the system. Each card is tagged with a category label (Activity, Sleep, Screen, or Mood) and a severity indicator that classifies the pattern as On Track, Notice, or Heads Up, giving the user an at-a-glance sense of how the pattern compares to their personal baseline. Each card presents a plain-language interpretation of the detected pattern alongside a supporting research reference. This keeps the application grounded in evidence while remaining accessible to users without a clinical background.

Resources

The Resources screen serves as a safety and signposting layer rather than a psychoeducational content library. It prominently states that MoodSense is not a substitute for professional mental health care, and directs users in crisis to contact a helpline. The screen lists a range of international and local emergency contacts, helpline numbers, and mental health websites, ensuring that users who may be experiencing distress have immediate access to appropriate support regardless of their location.

Settings and Privacy

Although a dedicated Settings screen is not shown in the current screenshots, the application exposes privacy controls through the settings icon accessible from every screen. Each optional sensor is represented by a clearly labelled toggle with a one-sentence explanation of what it collects. A prominent option to delete all locally stored data and to sign out of the cloud account is accessible from this menu, and a link to the application's full privacy policy and data transparency statement is also included. The design reflects the understanding that users who feel in control of their data are more likely to engage with the system over time [7, 15].

4.7 Privacy by Design

Privacy by Design, as articulated by Ann Cavoukian, holds that privacy protections should be embedded into a system's architecture from the outset rather than added as compliance measures afterward [22]. For an application that collects continuous behavioural data in a mental health context, this is not merely good practice but a foundational obligation. The sensitive nature of the data MoodSense handles demands that every architectural decision be examined through the lens of what it implies for the user's privacy and autonomy.

Table 4.5 maps the seven foundational principles of Privacy by Design to the specific design decisions made in MoodSense and the benefit each decision delivers to the user.

Table 4.5: Privacy by Design in MoodSense

Privacy Principle	Design Decision in MoodSense	User Benefit
Proactive, not reactive	Privacy implications considered at every design stage. No unnecessary data flows or analytics SDKs included	Privacy risks reduced at the source rather than managed after the fact
Privacy as the default setting	All optional sensors (location, communication, wearable data) are disabled by default. The user must actively opt in	Users who do not engage with settings are protected by the most private configuration
Privacy embedded into design	Hybrid local-first architecture. Raw sensor data remains on device. Only summarised records and self-reports are synced to the user's cloud account	Minimises the amount and sensitivity of data that ever leaves the device
Full functionality, positive-sum	Core functionality is available without enabling any optional sensors. Privacy protections do not reduce access to self-monitoring features	Users can benefit from MoodSense without sharing location, communication, or wearable data
End-to-end security	Data stored in system-protected storage on the device and transmitted over secure channels when synced. No third-party data sharing	Behavioural data protected throughout its lifecycle, from collection to optional backup
Visibility and transparency	Onboarding explains each data type and permission before collection begins. Settings shows active sensors in plain language	Users understand exactly what is collected and can verify or change it at any time
Respect for user privacy	The user can disable any sensor, adjust notification timing, sign out, and delete all local data from Settings	Full user control. No data persists without ongoing consent

Two design decisions in particular deserve additional comment. The choice to disable all optional sensors by default reflects the principle that privacy should be the starting position from which users choose to move, rather than a setting they must seek out to enable. This is directly applicable to MoodSense because the most sensitive data the application could collect (location history and communication patterns) are exactly the data that the related literature identifies as a primary source of user discomfort [1, 3]. Requiring an active opt-in means that a user who installs the app and never visits the settings page will never have their location or call logs collected.

The second decision concerns data deletion and sign-out. Providing permanent, immediate local data deletion and simple account sign-out is not a purely regulatory concession but a trust mechanism. Users who know they can delete everything are more willing to allow collection in the first place, and the research on long-term engagement consistently suggests that perceived control is one of the strongest predictors of sustained use [5, 7].

4.8 Data and Interaction Flow

Three interaction sequences are central to the operation of MoodSense and are described below.

4.8.0.0.1 Passive data collection and summarisation. Throughout the day, a lightweight foreground service tracks step counts in the background. At suitable intervals, and when the device is idle, the application queries usage statistics, call and SMS counts, and, if enabled, location and Health Connect data for the current day. These readings are summarised into daily metrics such as total steps, estimated sleep duration, communication counts, and basic heart-rate statistics. The daily summary is stored locally and can later be attached to the day's mood entry.

4.8.0.0.2 Mood check-in flow. At the time configured by the user during onboarding, a `PeriodicWorkRequest` managed by WorkManager fires a local notification inviting the user to complete their daily check-in. If the user taps the notification, the Daily Check-in screen opens. The core questions presented cover mood, stress, sleep quality, and energy level. On confirmation, the ViewModel requests a fresh daily summary from the sensing layer and passes both the self-report and the summary to the repository. The entry is stored locally and, when the user is signed in, synchronised to Firebase under their account. The dashboard updates reactively so that new entries are immediately reflected in trends and insights. If the user dismisses the notification, no entry is recorded for that day and no additional reminder is sent.

4.8.0.0.3 Baseline comparison and insight generation. When the dashboard or trends screens are opened, the ViewModels request the most recent seven to fourteen days of summaries and self-reports. The logic layer computes rolling baselines and compares the most recent period to the preceding one. If a deviation exceeds a configured threshold, the insight engine creates an insight object with a title, detail text, severity, and priority. The system also distinguishes between an early mode, used when fewer than five days of data are available, and a baseline mode, used once enough history has accumulated to make personal comparisons meaningful. Examples of generated insights include physical activity, sleep pattern, and communication pattern cards, each tied to a specific metric. The Insights screen observes these insight objects and surfaces the most relevant cards, keeping the interface focused rather than overwhelming.

The implementation of these design decisions, including the Kotlin code, libraries, and Android APIs used to realise each component, is described in Chapter 5.

5 Implementation

5.1 Overview

This chapter describes how the design presented in Chapter 4 was translated into a working Android application. MoodSense is implemented in Kotlin using the MVVM architectural pattern, Jetpack Compose for the user interface, Firebase Realtime Database and Firebase Authentication for optional cloud synchronisation, Android DataStore for local settings persistence, and Android Health Connect for wearable integration. All sensing, baseline computation, and insight generation occur on the device. Only the structured daily records produced by the check-in flow, containing summarised behavioural counts and self-reported scores, are optionally synchronised to the user's Firebase account when signed in. No raw sensor streams are ever transmitted.

The chapter is organised as follows. The Technologies and Tools section describes the libraries selected and the reasoning behind each choice. The Application Architecture section explains how the package structure reflects the layered architecture described in Chapter 4. The Key Modules section covers the core components of MoodSense, each illustrated with a selected code excerpt where the implementation itself is the clearest explanation. The chapter closes with a discussion of the practical challenges encountered during development and the solutions adopted to address them.

5.2 Technologies and Tools

MoodSense targets API level 26 (Android 8.0) and above, with a target SDK of 35. The following technologies form the foundation of the implementation.

5.2.0.0.1 Kotlin. The application is written entirely in Kotlin. Kotlin's coroutine support, null safety, and concise syntax are well suited to an application that must coordinate asynchronous sensor queries, reactive data observation across multiple sources, and strict separation between sensing, logic, and presentation layers.

5.2.0.0.2 Jetpack Compose. The user interface is built with Jetpack Compose, Android's declarative UI toolkit. Composable functions describe the UI as a function of state, and Compose recomposes only the parts of the screen affected by a state change. ViewModels expose state as `StateFlow` streams. Screens observe these through `collectAsState()` and update automatically when values change. This reactive model is well suited to a dashboard where multiple independent data sources update at different times.

5.2.0.0.3 Firebase Realtime Database and Authentication. Cloud synchronisation and user account management are handled through Firebase. Firebase Authentication provides email-based sign-in and associates each user with a unique identifier. When a user is signed in, mood entries and their attached daily summaries are written to Firebase Realtime Database under a path keyed to that identifier (`users/{uid}/moods`). No raw sensor data is written to Firebase. Only the structured daily records produced by the check-in flow are synchronised. Firebase was selected for its low-latency read and write operations, per-user data isolation through security rules, and clean integration with Kotlin coroutines [23].

5.2.0.0.4 Android DataStore. User preferences, including notification time, enabled sensors, and onboarding completion state, are persisted using Jetpack DataStore (Preferences). DataStore provides a type-safe, coroutine-friendly replacement for `SharedPreferences` and exposes preferences as `Flow` streams, which integrate naturally with the reactive patterns used throughout the application. Among the preferences stored is a boolean that records whether onboarding has been completed, observed in the main activity to determine the initial navigation destination.

5.2.0.0.5 WorkManager. Daily reminder notifications are scheduled using WorkManager through a `PeriodicWorkRequest` configured to repeat every twenty-four hours. The `ReminderScheduler` object computes the delay to the user's next configured notification time and sets this as the initial delay on the request, using `enqueueUniquePeriodicWork` with a `CANCEL_AND_REENQUEUE` policy to ensure only one active reminder exists at any time. WorkManager was chosen over `AlarmManager` because it handles device-restart survival and Doze-mode compatibility automatically across the range of Android versions and manufacturers targeted by the application [21].

5.2.0.0.6 Android Health Connect. The Health Connect API provides a standardised, permission-gated interface for reading health data contributed by compatible wearable devices. MoodSense uses Health Connect to read `HeartRateRecord` and `SleepSessionRecord` data when a compatible device is connected and the user has opted in. Permissions are managed through a system-level consent dialog that cannot be bypassed, which aligns with the privacy-by-design principle of transparent, granular consent [21].

5.2.0.0.7 Android Communication and Location APIs. Communication frequency data is collected through the `CallLog.Calls` and `Telephony.Sms` content providers, both requiring runtime permissions. Location data for mobility is obtained through `LocationManager` using GPS, network, and passive providers in sequence. Both sources are treated as fully optional and queried only when the user has enabled them in Settings.

5.3 Application Architecture in Practice

The application's package structure directly reflects the four-layer architecture described in Chapter 4.

The `data/` package contains the DataStore preference keys and the `UserPrefs` helper used to persist onboarding state. The `model/` package contains the core data classes (`MoodEntry`, `TodaySummary`, `WeeklySummary`, `HeartRateSummary`) and the repository classes that retrieve, aggregate, and write them (`MoodRepository`, `SensorsRepository`, `DashboardRepository`, `HeartRateRepository`, `CommunicationRepository`, `MobilityRepository`). The `domain/` package contains the pure Kotlin data models shared across layers (`DailySignals`, `Question`).

The `insights/` package contains the insight engine, which computes Pearson correlations and rolling baseline deviations and produces typed `InsightData` objects. The `logic/` package contains `QuestionTriggerEngine`, an adaptive question selection component designed to extend the check-in with contextually triggered questions. Its integration with the live check-in screen is identified as a direction for future development. The `notifications/` package contains `ReminderWorker` and `ReminderScheduler`. The `export/` package contains `ExportHelper`. The `screens/` package contains all Composable screen functions and their associated ViewModels, organised by screen.

Data flows in one direction throughout the application. Repositories read from device APIs, Firebase, and Health Connect and expose results as suspend functions or `Flow` streams. ViewModels call repositories, transform results into UI state, and expose them as `StateFlow`. Composable screens observe these flows and redraw reactively. No screen reads from or writes to any data source directly.

5.4 Key Modules and Features

5.4.1 Passive Data Collection

Passive data collection is distributed across the repository layer rather than consolidated into a single background job. Step data, screen usage, communication frequency, and location are each handled by a dedicated repository that queries its respective Android API and returns a typed result to the sensing layer.

5.4.1.0.1 Step counting. Step data is read through a two-source strategy in `SensorsRepository`. The repository first queries Android Health Connect using an `AggregateRequest` for `StepsRecord.COUNT_TOTAL` over a time window from midnight to the current moment. If Health Connect returns a positive count, that value is used directly. If Health Connect is unavailable or returns zero, the repository falls back to the hardware `TYPE_STEP_COUNTER` sensor, which reports a cumulative total since the last device boot. Because the cumulative counter does not reset at midnight, the fallback logic stores the sensor reading at the start of each calendar day as a baseline and computes the day's steps as the difference between the current total and that baseline. Listing 5.1 shows this fallback calculation.

```kotlin
val today = Calendar.getInstance().get(Calendar.DAY_OF_YEAR)
val savedDay = prefs.getInt("steps_day", -1)
val lastTotal = prefs.getInt("steps_last_total", 0)
var baseline = prefs.getInt("steps_baseline", -1)

if (savedDay != today || baseline == -1) {
    prefs.edit()
        .putInt("steps_day", today)
        .putInt("steps_baseline", lastTotal)
        .apply()
    baseline = lastTotal
}

val sensorSteps = kotlin.math.max(0, lastTotal - baseline)
```

Listing 5.1: Step baseline subtraction in the sensor fallback path of SensorsRepository

5.4.1.0.2 Screen time. Screen usage is queried through `UsageStatsManager.queryUsageStats()` with an `INTERVAL_BEST` granularity over a window spanning from midnight to the present moment. The API returns per-package foreground duration records. Because multiple overlapping records can be returned for the same package within a single day, the repository deduplicates by package name, retaining only the maximum `totalTimeInForeground` value for each app. MoodSense's own package is excluded from the total. The summed foreground duration across all remaining packages is converted from milliseconds to minutes and returned as a single daily total. Listing 5.2 shows this aggregation.

```kotlin
val stats = usm.queryUsageStats(
    UsageStatsManager.INTERVAL_BEST, startTime, endTime
```

```kotlin
)

val deduped = stats
    .filter { it.packageName != ctx.packageName }
    .filter { it.totalTimeInForeground > 0 }
    .groupBy { it.packageName }
    .mapValues { (_, entries) ->
        entries.maxOf { it.totalTimeInForeground }
    }

val totalMs = deduped.values.sum()
return (totalMs / 1000 / 60).toInt()
```

Listing 5.2: Screen time aggregation via UsageStatsManager in SensorsRepository

5.4.1.0.3 Communication frequency. Communication data is collected by `CommunicationRepository`, which queries the `CallLog.Calls` and `Telephony.Sms` content providers for records created since midnight. Rather than returning zero when a permission has not been granted, both methods return `-1` explicitly. This allows the baseline engine and insight layer to distinguish a genuine zero reading (no calls or messages that day) from a permission gap and exclude the latter from averages. Listing 5.3 shows this pattern for call log access.

```kotlin
fun getCallsToday(): Int {
    val hasPermission = ContextCompat.checkSelfPermission(
        ctx, Manifest.permission.READ_CALL_LOG
    ) == PackageManager.PERMISSION_GRANTED

    if (!hasPermission) return -1   // -1 signals missing permission,
                                    // not zero calls

    return try {
        val startOfDay = getStartOfDayMillis()
        val cursor = ctx.contentResolver.query(
            CallLog.Calls.CONTENT_URI,
            arrayOf(CallLog.Calls._ID),
            "${CallLog.Calls.DATE} >= ?",
            arrayOf(startOfDay.toString()),
            null
        )
        val count = cursor?.count ?: 0
```

```
        cursor?.close()
        count
    } catch (e: Exception) { 0 }
}
```

Listing 5.3: Permission-aware call log query in CommunicationRepository

5.4.1.0.4 Sleep data. Sleep duration is read from `SleepSessionRecord` via the Health Connect API, which aggregates records contributed by compatible wearable devices or sleep-tracking applications. When Health Connect is unavailable or no sleep records exist for the previous night, the sleep field in that day's summary is left null and excluded from baseline calculations. A phone-inferred sleep fallback based on device inactivity and charging timestamps was considered during design but is not implemented in the current version.

5.4.1.0.5 Mobility. Location data is obtained through `MobilityRepository`, which calls `LocationManager.getLastKnownLocation()` in provider priority order: GPS first, network second, passive third. The repository returns the most recent location fix available rather than requesting continuous updates, minimising battery impact. In the current version, the repository records whether a location fix was obtained at all during the day. The insight engine uses this as a binary indicator of whether the user left a fixed location, rather than computing a displacement or distance score. Extending this into a richer mobility metric is identified as a direction for future development.

5.4.2 Daily Check-in

The daily check-in is the core active feature of MoodSense and the primary source of labelled data for the insight engine. The check-in screen presents four slider questions covering mood, stress, sleep quality, and energy level, each rated on a 1–5 scale, together with an optional free-text notes field. The `MoodEntry` data class also contains fields for mental exhaustion and social connectedness. In the current version these are not collected through the check-in screen. They are reserved for the adaptive `QuestionTriggerEngine`, which is designed to surface them conditionally based on passive sensor signals but whose integration with the live screen is not completed. Both fields remain null in all entries produced by the current check-in flow.

On submission, `MoodViewModel.submitMood()` collects the slider answers into a typed map and passes them to `MoodRepository` alongside a fresh `TodaySummary` requested from the sensing layer. The repository constructs a `MoodEntry` object combining the self-reported scores with the day's passive sensor readings, including steps, screen time, sleep duration, call count, SMS count, and heart rate summary, and writes it to Firebase Realtime Database under the authenticated user's path (`users/{uid}/moods/{entryId}`). The ViewModel exposes a `submitSuccess StateFlow` that the screen observes. When it transitions to `true`, the screen navigates back to the dashboard, which reactively reloads to reflect the new entry.

If the user has not yet granted coarse location permission at the point of submission, the check-in screen requests it before calling `submitMood()`. This request is placed at submission time rather than during onboarding because location is an optional enrichment signal and the check-in is the only moment the app actively needs a location fix. Submission completes regardless of whether the permission is granted.

An `answeredToday` flag read from Firebase prevents duplicate submissions by replacing the check-in form with a confirmation message if an entry for the current calendar day already exists.

5.4.3 Baseline Computation and the Weekly Summary

The weekly summary and baseline computation are implemented in `DashboardRepository.getWeeklySummary()`. The method retrieves the last seven days of `MoodEntry` records from Firebase and computes averages for mood, stress, steps, screen time, sleep duration, and communication counts. To detect directional change, entries are split into a recent group (the last three days) and an older group (the four days before that), and the percentage difference between the two averages is stored as a change field for each metric. Sensor fields that carry a value of `-1`, indicating a missing permission rather than a genuine reading, are excluded from these averages so that permission gaps do not distort the baseline.

The number of distinct calendar days with at least one entry is tracked as `daysWithData`. This value is passed to the insight engine, which uses it to determine whether enough history has accumulated to make personalised comparisons meaningful, as described in the following section.

5.4.4 Insight Engine

The insight engine is implemented in `InsightEngine.kt` as a collection of pure functions. The entry point, `buildInsights()`, receives a `WeeklySummary` and the full entry list, invokes each individual insight builder, filters out invisible results, and returns the four highest-priority insights for display.

The first decision the engine makes is which operating mode to use. Listing 5.4 shows this logic.

```
fun getInsightMode(daysWithData: Int): InsightMode =
```

```
    if (daysWithData >= 5) InsightMode.BASELINE else InsightMode.
       EARLY
```

Listing 5.4: Insight mode selection based on available data history

When fewer than five days of data are available, the engine operates in `InsightMode.EARLY` and returns generic, non-comparative observations that acknowledge the limited baseline. Once five or more days have been recorded, it switches to `InsightMode.BASELINE` and generates personalised comparisons against the user's own rolling history. This cold-start handling ensures that newly onboarded users are never shown misleading deviation alerts before a meaningful baseline has been established.

Two of the nine insight types are computed using the Pearson correlation coefficient over the available mood and sensor entry pairs. Listing 5.5 shows the on-device implementation.

```
private fun pearsonCorrelation(xs: List<Float>, ys: List<Float>):
   Float? {
    if (xs.size != ys.size || xs.size < 2) return null
    val meanX = xs.average().toFloat()
    val meanY = ys.average().toFloat()
    val num = xs.zip(ys)
        .sumOf { (x, y) -> ((x - meanX) * (y - meanY)).toDouble() }
        .toFloat()
    val denX = xs.sumOf { ((it - meanX) * (it - meanX)).toDouble() }.
       toFloat()
    val denY = ys.sumOf { ((it - meanY) * (it - meanY)).toDouble() }.
       toFloat()
    val den = kotlin.math.sqrt(denX * denY)
    return if (den == 0f) null else num / den
}
```

Listing 5.5: Pearson correlation computed on-device for mood-activity and mood-sleep insights

A minimum of five valid paired observations is required before a correlation insight is generated, preventing spurious results from small samples. A correlation of 0.35 or above triggers a positive insight. A correlation of -0.35 or below triggers a contrasting observation. All correlation computation occurs entirely on the device. No entry data is transmitted for analysis.

5.4.5 Settings and Privacy Controls

The Settings screen is the primary interface through which users control what MoodSense collects. It is implemented as a scrollable list of grouped toggle rows and navigation rows, each reading its state from `SettingsViewModel`, which in turn observes a `settingsFlow` exposed by `SettingsRepository` from Android DataStore.

Each data source toggle, covering steps, screen time, sleep, GPS, communication logs, and wearable metrics, is backed by a boolean preference in DataStore. When the user enables a toggle, the ViewModel writes the updated preference and, where a runtime permission is also required, immediately launches the system permission dialog via the appropriate `ActivityResultLauncher`. When a toggle is disabled, the ViewModel writes `false` to DataStore and the relevant repository checks this flag at the start of every query, returning `null` for that field rather than querying the API. This ensures that disabling a source in Settings takes effect immediately on the next sensing cycle without requiring an app restart.

The Settings screen also provides direct navigation to Android's system permission management pages, including the Usage Access settings page for screen time, the app's permission detail page for runtime permissions, and the Health Connect app for wearable permissions. Users can review and revoke any permission at the system level at any time. The repositories re-check grant state on every data access, so revocations made outside the app are respected silently.

5.4.6 Daily Reminder Notifications

Daily reminders are delivered through a WorkManager `PeriodicWorkRequest` targeting `ReminderWorker`. `ReminderScheduler.schedule()` calculates the delay from the current moment to the user's next configured notification time. If the target time has already passed for the current day, the delay is extended to the following day. Listing 5.6 shows the scheduling logic.

```kotlin
fun schedule(context: Context, hour: Int, minute: Int) {
    val now = Calendar.getInstance()
    val target = Calendar.getInstance().apply {
        set(Calendar.HOUR_OF_DAY, hour)
        set(Calendar.MINUTE, minute)
        set(Calendar.SECOND, 0)
    }
    if (target.before(now)) target.add(Calendar.DAY_OF_YEAR, 1)

    val delay = target.timeInMillis - now.timeInMillis

    val request = PeriodicWorkRequestBuilder<ReminderWorker>(24,
       TimeUnit.HOURS)
        .setInitialDelay(delay, TimeUnit.MILLISECONDS)
```

```
        .addTag(WORK_TAG)
        .build()

    WorkManager.getInstance(context).enqueueUniquePeriodicWork(
        WORK_TAG,
        ExistingPeriodicWorkPolicy.CANCEL_AND_REENQUEUE,
        request
    )
}
```

Listing 5.6: WorkManager periodic reminder scheduling with computed initial delay

The `CANCEL_AND_REENQUEUE` policy ensures that when the user changes their notification time in Settings, the previously scheduled reminder is replaced immediately rather than running at the old time until its next natural expiry. `ReminderWorker.doWork()` constructs a high-priority notification through `NotificationCompat.Builder`, creates the required notification channel on Android 8.0 and above, and attaches a `PendingIntent` that navigates the user to `MainActivity` on tap.

5.4.7 Data Export

The `ExportHelper` object queries Firebase Realtime Database for all mood entries belonging to the authenticated user, constructs a two-section CSV file containing the full entry log followed by a seven-day summary table, writes the file to the application's cache directory, and shares it through the Android share sheet using a `FileProvider` URI. The export covers mood, energy level, stress, sleep quality, sleep duration, screen time, steps, call count, SMS count, and timestamps. Columns for mental exhaustion and social connectedness are included in the CSV header but carry empty values in the current version, as these fields are not yet collected through the check-in screen.

5.4.8 Health Connect Integration

Health Connect integration is implemented through `HealthConnectManager` and `HeartRateRepository`. `HealthConnectManager` exposes a `hasAllPermissions()` suspend function that checks the current grant state before each data retrieval, accounting for the possibility that the user has revoked permissions through the Health Connect settings panel between sessions. `HeartRateRepository.getHeartRateToday()` reads `HeartRateRecord` objects for the current day, extracts beat-per-minute samples, and returns a `HeartRateSummary` containing average, minimum, and maximum values. All fields are nullable. If no records are found or any exception occurs, the method returns an empty summary rather than propagating an error to the calling ViewModel.

A known limitation is that Huawei Health does not write data to Android Health Connect, excluding users of Huawei wearables from the physiological data integration. This limitation is discussed further in the evaluation chapter.

5.5 Challenges and Solutions

5.5.0.0.1 Permission prompts not appearing on first load. During development, the location permission prompt was found to never appear on the dashboard at all, and the communication logs permission prompt only appeared after the user had visited the Settings screen and navigated back. It did not display on the first load of the dashboard. Both issues stemmed from the same root cause. The Composable functions responsible for showing the prompts were being composed before the ViewModel had finished reading the permission state from DataStore, so the conditional rendering evaluated to false on the first frame and was never re-triggered. The fix involved ensuring that permission prompt visibility was driven by a `StateFlow` that emitted only after the initial DataStore read had completed, so that the first composition reflected the true state rather than a default.

5.5.0.0.2 Settings toggles not reflecting actual permission state. Several toggles in the Settings screen, including screen time, GPS, communication logs, and wearable metrics, remained visually switched on even after the user pressed “Not Now” on the corresponding permission card during onboarding. Steps and sleep correctly defaulted to off, but the other sources did not, producing inconsistent behaviour across the same screen. The underlying problem was that the ViewModel wrote `true` to DataStore at the moment the user enabled the toggle, before the system permission dialog had been resolved. If the user then declined the permission, the DataStore value was never corrected. The fix was to move the DataStore write to the permission callback result handler, writing `true` only when the permission was actually granted and leaving the preference unchanged if the user declined, so the toggle state always reflected the real grant state.

5.5.0.0.3 Permission prompts reappearing after navigation. Pressing “Not Now” on the screen time, Health Connect, and step counting permission prompts dismissed them correctly, but they reappeared each time the user navigated to a different screen and returned. The prompts were being controlled by a local `remember{mutableStateOf}` variable scoped to the Composable, which was destroyed and recreated on every recomposition triggered by navigation. The solution was to move the dismissed flag into the ViewModel as a `StateFlow`-backed field, persisted across navigation events, so that a prompt dismissed once would not reappear during the same session.

5.5.0.0.4 Health Connect dialog dismissing unrelated prompts. Granting the Health Connect permission was found to simultaneously dismiss an unrelated permission dialog that happened to be visible at the same time, causing that second permission to be silently skipped. This occurred because both dialogs were being managed by the same Activity result launcher. The Health Connect system dialog completing its flow caused the launcher to fire its result callback unconditionally, which the application interpreted as a resolution of both pending requests. The fix was to assign each permission request its own dedicated `ActivityResultLauncher` instance and to ensure that no two permission dialogs could be pending simultaneously by serialising their presentation.

5.5.0.0.5 Health Connect dialog appearing at unexpected moments. Separately from the above, the Health Connect permission dialog was found to reappear at random moments even after it had already been handled, including on screens where it should not have been triggered. The cause was a `LaunchedEffect` block that checked Health Connect permission state and launched the consent flow whenever its key recomposed, which happened more frequently than intended due to upstream state changes. The fix was to add an explicit already-requested guard in the ViewModel so that the consent flow was launched at most once per app session, and to scope the `LaunchedEffect` key to a stable value rather than a recomposing state.

5.5.0.0.6 Firebase Authentication session persisting after reinstall. After a fresh install or reinstall, the app was found to skip the welcome screen and open the dashboard directly, as if the user were already signed in, even though no account was active. Sign-out also failed to clear the authentication state correctly, allowing brief access to the dashboard without a valid session. The root cause was that Firebase Authentication caches the user token on device and this cache was not being invalidated on reinstall or during sign-out. The fix involved explicitly calling `FirebaseAuth.getInstance().signOut()` on sign-out and auditing the main activity's navigation decision logic to re-read the current `FirebaseAuth.currentUser` at launch rather than relying on a previously cached preference. The onboarding completion flag in DataStore was also cleared on sign-out, ensuring that a reinstalled or signed-out user always enters through the welcome screen.

5.5.0.0.7 Cold-start baseline accuracy. Early testing showed that users who had logged only one or two entries were receiving deviation alerts that compared their latest entry against an average of nearly identical data, producing misleading observations such as "your mood is higher than your baseline" based on a sample of two days. The `InsightMode` mechanism was introduced to address this by switching between early mode and baseline mode based on the `daysWithData` count, using a threshold of five days as a pragmatic balance between prompt feedback and baseline reliability. Below that threshold the engine returns only generic, non-comparative observations.

5.5.0.0.8 WorkManager notification timing. The initial implementation scheduled the reminder using a fixed twenty-four-hour period without an initial delay, which caused the first notification to fire immediately after scheduling rather than at the time the user had configured. On some devices this meant the reminder arrived within seconds of completing onboarding. The revised `ReminderScheduler` computes the exact delay to the user's next target time and passes it as `setInitialDelay()`, ensuring that the first notification fires at the correct moment and that subsequent ones maintain the twenty-four-hour cadence from that anchor point.

6 Testing and Evaluation Methodology

6.1 Introduction

This chapter describes the methodology used to evaluate MoodSense. The evaluation was designed to assess the application's usability, perceived usefulness, and user perceptions of privacy following a period of naturalistic use. Rather than conducting a single demonstration session, participants were asked to install MoodSense on their personal Android smartphones and use it across a minimum of three consecutive days. This approach was chosen so that feedback would reflect genuine interaction with the system rather than first impressions alone.

The chapter covers the objectives that guided the evaluation, the profile of the participants who took part, the procedure followed during the study period, the design of the questionnaire, and the ethical framework within which the study was conducted.

6.2 Evaluation Objectives

The evaluation was guided by three primary objectives.

The first was to assess whether users found MoodSense useful for becoming more aware of their own behavioural patterns over time. Because MoodSense is designed as a self-awareness tool rather than a diagnostic instrument, perceived usefulness in the context of personal reflection was considered the most meaningful indicator of whether the application achieves its central goal.

The second objective was to evaluate usability: whether the interface was clear, navigable, and understandable without prior training or guidance.

The third objective was to examine user perceptions of privacy and trust. MoodSense requests a range of sensitive permissions relating to screen usage, communication patterns, movement, sleep, and optionally location. Understanding which permissions caused discomfort, and the reasoning behind those concerns, was considered important both for assessing the current design and for informing future iterations of the system.

6.3 Participants

The evaluation involved **21** adult Android users who installed MoodSense on their own smartphones and used it for at least three consecutive days before completing the questionnaire. This ensured that responses reflected real-world use of the application in everyday contexts rather than first impressions from a single demonstration session.

Table 6.1 summarises the participant profile. Gender is not reported as an outcome variable, as the sample size does not permit meaningful gender-based comparisons and it was not directly relevant to the usability or privacy objectives of the study.

Table 6.1: Participant profile (n = 21)

Category	**Group**	**Count**	**%**
Age	18–24	10	47.6%
Age	25–34	6	28.6%
Age	35–44	2	9.5%
Age	45–54	2	9.5%
Age	55–64	2	9.5%
Device	Google Pixel	9	42.9%
Device	Samsung	8	38.1%
Device	Xiaomi	4	19.0%
Wearable used	Yes	15	71.4%
Wearable used	No	6	28.6%

As Table 6.1 shows, just under half of the sample were aged 18–24, with the remainder distributed across the 25–34, 35–44, 45–54, and 55–64 age groups. The device distribution was balanced across the three main Android manufacturers represented in the study (Google, Samsung, and Xiaomi), and around two-thirds of participants used a smartwatch or fitness band during the evaluation. This mix provides feedback from both younger and older adults, and from users with and without wearables, which is relevant to understanding how well MoodSense works across different usage contexts.

Table 6.2 presents participants' self-reported frequency of mood reflection or tracking prior to the study, and Table 6.3 shows whether they had previously used any app for mood, sleep, or wellbeing monitoring. These background questions were included to contextualise the evaluation results. Participants with little prior experience of self-monitoring tools provide a useful indication of whether MoodSense is accessible to new users without prior exposure to the genre.

Table 6.2: Frequency of mood reflection or tracking before MoodSense (n = 21)

Frequency	**Count**	**%**
Every day	1	4.8%
A few times a week	3	14.3%
A few times a month	7	33.3%
Rarely	9	42.9%
Never	1	4.%

Table 6.3: Prior use of mood, sleep, or wellbeing apps (n = 21)

Response	**Count**	**%**
Yes	12	57.1%
No	9	42.9%

Overall, most participants reported reflecting on or tracking their mood only rarely or a few times a month, and just over half had previously used any app for mood, sleep, or wellbeing tracking. This suggests that MoodSense was evaluated both by participants familiar with digital self-monitoring tools and by those approaching such an application for the first time.

6.4 Procedure

Participants were asked to download and install MoodSense on their personal Android smartphone and use it over a minimum of three consecutive days. During this period they were instructed to complete at least one daily check-in per day and to explore the trends and insights features at their own pace. No specific tasks were assigned beyond the daily check-in, and participants were free to interact with the application in whatever way felt natural to them. This approach was chosen to reflect a realistic adoption scenario, in which a person begins using a self-monitoring application as part of their everyday routine rather than following a researcher-directed task sequence.

At the end of the usage period, participants completed a structured questionnaire covering background information, user experience evaluation using semantic differential scales, technical problem reporting, permission concerns, and open-ended qualitative feedback. The questionnaire was administered digitally using Google Forms and required approximately ten to fifteen minutes to complete. The full questionnaire is reproduced in Appendix 8.2.

6.5 Questionnaire Design

The questionnaire was structured into six parts.

The first part confirmed informed consent and the minimum three-day usage requirement, and the second part collected the background information described in Section 6.3. These were followed by the substantive evaluation parts described below.

The third part formed the core evaluation instrument. It used semantic differential scales drawn from the User Experience Questionnaire (UEQ) [24], a validated instrument designed to assess both pragmatic and hedonic aspects of interactive systems. Each scale presents a pair of opposing adjectives at either end of a seven-point continuum. The UEQ covers six quality dimensions: *attractiveness*, *perspicuity*, *efficiency*, *dependability*, *stimulation*, and *novelty*. A subset of fifteen adjective pairs was selected to cover all six dimensions while keeping the questionnaire concise enough to complete within the overall ten-to-fifteen minute target. The adjective pairs used in this study are listed in Table 6.4.

Table 6.4: Semantic differential adjective pairs used in the evaluation

Q	Adjective Pair	Left Pole	Right Pole
1	Annoying / Enjoyable	Negative	Positive
2	Not understandable / Understandable	Negative	Positive
3	Easy to learn / Difficult to learn	Positive	Negative
4	Valuable / Inferior	Positive	Negative
5	Boring / Exciting	Negative	Positive
6	Not interesting / Interesting	Negative	Positive
7	Obstructive / Supportive	Negative	Positive
8	Complicated / Easy	Negative	Positive
9	Secure / Not secure	Positive	Negative
10	Motivating / Demotivating	Positive	Negative
11	Clear / Confusing	Positive	Negative
12	Impractical / Practical	Negative	Positive
13	Organized / Cluttered	Positive	Negative
14	Attractive / Unattractive	Positive	Negative
15	Friendly / Unfriendly	Positive	Negative

Because some adjective pairs place the positive pole on the left and others on the right, raw scores require directional correction before analysis. For pairs where the positive pole is on the right, the raw score is used directly. For pairs where the positive pole is on the left, the score is inverted using the formula $8 - \textit{raw score}$, such that a corrected score of 7 always represents the most positive evaluation and 1 the most negative, regardless of pair orientation. Subscale scores are subsequently mean-centred to the range -3 to $+3$, where 0 represents the scale midpoint. Scores above $+0.8$ are considered positive, scores between -0.8 and $+0.8$ neutral, and scores below -0.8 negative [24].

The fourth part asked participants to report any technical problems encountered during the study period. Those who had experienced a problem were asked to describe it, identify which screen or feature was affected, rate its severity, and confirm whether they were able to continue using the application. This section was included to identify device-specific or platform-level issues that might affect the reliability of the evaluation results.

The fifth part collected permission concerns. Participants who reported discomfort with a requested permission were asked to identify which one, briefly explain their concern, and indicate whether the concern had affected their use of the application.

The sixth part collected open-ended qualitative feedback. Participants were asked to identify the feature they found most useful, describe what they liked most about MoodSense, and suggest any improvements. These questions were optional and designed to complement the quantitative results with personally grounded observations.

6.6 Ethical Considerations

The study was conducted in accordance with the ethical principles governing research involving human participants. Participation was entirely voluntary, and participants were informed at the outset that they could withdraw at any time without consequence. The questionnaire included a clear informed consent statement that each participant was required to confirm before proceeding.

No personally identifiable information was collected. Participants were not asked for their names, email addresses, or any other identifying details. All responses were accessible only to the researcher for the purposes of this thesis.

Because MoodSense operates in a mental health adjacent domain, additional care was taken to ensure that participation did not require or imply the disclosure of personal mental health information. The questionnaire did not ask about participants' mental health status, history, diagnoses, or symptoms. Demographic questions were limited to age group and device type, included solely to contextualise usability ratings. Questionnaire data was stored within Google Forms and will be deleted following the submission of this thesis.

7 Evaluation Results and Discussion

7.1 Introduction

This chapter presents and interprets the results of the evaluation described in Chapter 6. The analysis is organised around the three evaluation objectives: assessing perceived usefulness for self-awareness, evaluating usability, and understanding user perceptions of privacy and trust. Quantitative results from the UEQ are presented first, followed by technical problem reports, permission concerns, and the qualitative feedback gathered through open-ended questions. The chapter closes with a discussion that relates these findings to the design decisions described in Chapter 4 and to the relevant literature.

7.2 User Experience Ratings

7.2.1 Item-Level Results

Table 7.1 reports the mean and median corrected score for each of the fifteen semantic differential items on a scale of 1 to 7, where 7 represents the most positive evaluation.

Table 7.1: UEQ item scores (corrected, 1–7; n = 21)

Q	Adjective Pair	Mean	Median
1	Annoying / Enjoyable	6.19	6.0
2	Not understandable / Understandable	6.76	7.0
3	Easy to learn / Difficult to learn	6.90	7.0
4	Valuable / Inferior	6.14	6.0
5	Boring / Exciting	5.76	6.0
6	Not interesting / Interesting	6.00	6.0
7	Obstructive / Supportive	6.14	6.0
8	Complicated / Easy	6.62	7.0
9	Secure / Not secure	6.14	6.0
10	Motivating / Demotivating	5.81	6.0
11	Clear / Confusing	6.57	7.0
12	Impractical / Practical	6.14	6.0
13	Organized / Cluttered	6.62	7.0
14	Attractive / Unattractive	6.52	7.0
15	Friendly / Unfriendly	6.81	7.0

All items were corrected for pole direction prior to analysis, as described in Section 6.5. The adjective pairs are listed in their original orientation. Corrected scores reflect the direction of the positive pole for each pair, ensuring that higher values consistently indicate a more favourable evaluation regardless of which end of the scale the positive adjective appeared on. Scores were consistently high across all items, with means ranging from 5.76 to 6.90. The highest-rated items were *Easy to learn* (mean 6.90), *Friendly* (mean 6.81), and *Understandable* (mean 6.76), reflecting strong performance on the clarity and learnability dimensions that were central design priorities. The lowest-rated items were *Exciting* (5.76) and *Motivating* (5.81), which, while still clearly positive, suggest that the hedonic dimension of stimulation was perceived as somewhat less strong than the pragmatic qualities of clarity and ease of use.

7.2.2 Subscale Results

Table 7.2 reports the mean score for each UEQ subscale, converted to the standard -3 to $+3$ range. Following the UEQ benchmarking criteria [24], scores above $+0.8$ are considered positive, scores between -0.8 and $+0.8$ are neutral, and scores below -0.8 are negative.

Table 7.2: UEQ subscale scores (3 to +3; n = 21)

Subscale	**Score**	**Interpretation**
Perspicuity	+2.71	Positive
Attractiveness	+2.51	Positive
Efficiency	+2.30	Positive
Dependability	+2.14	Positive
Novelty	+2.00	Positive
Stimulation	+1.79	Positive

All six subscales received positive ratings, all substantially above the $+0.8$ threshold. Perspicuity was the highest-rated subscale ($+2.71$), indicating that users found the application easy to understand and learn without external guidance. This is consistent with the non-functional requirement NR-03 (Section 4.3.2), which specified that the interface must be navigable by a first-time user without instruction. Attractiveness ($+2.51$) and Efficiency ($+2.30$) were rated closely together and also strongly, suggesting that users found the interface both visually appealing and easy to use effectively. Stimulation was the lowest subscale ($+1.79$), though it remained clearly positive, consistent with the item-level finding that *Exciting* and *Motivating* were the least highly rated individual items.

7.3 Technical Problems

Five of twenty-one participants (24 %) reported a technical problem during the study period. All five continued using the application. Table 7.3 summarises the reported issues.

Table 7.3: Technical problems reported by participants

Issue	Severity	Notes
Sleep data from third-party app not reaching Health Connect	Minor	Participant used a dedicated sleep-tracking app that did not reliably sync data to Health Connect, resulting in missing sleep records. Not a MoodSense issue but reflects a dependency on third-party integration.
Permissions setup confusion at onboarding	Minor	One participant experienced difficulty understanding the permissions flow at initial setup. The issue was resolved with guidance and did not recur.
Huawei watch incompatible with Health Connect	Moderate	Participant's Huawei wearable does not integrate with Android Health Connect, preventing heart-rate and sleep stage data from being retrieved. This is a known platform-level limitation.
Screen time inflated on Samsung device	Minor	Samsung's usage statistics API counts background activity, resulting in higher-than-actual screen time readings. Baseline comparison was not meaningfully affected.
Sleep and step data incomplete without a wearable	Minor	Participant initially without a watch experienced gaps in sleep and step data. The issue was resolved after being provided with a compatible wearable.

The pattern across these reports is notable. None of the five issues represent defects in MoodSense's core logic. Three are attributable to device-specific or platform-level behaviour (Huawei Health Connect incompatibility, Samsung screen time API, missing wearable) rather than to the application itself. The fourth reflects a dependency on Health Connect being populated by a compatible third-party sleep app, a limitation acknowledged in the system design. The fifth, the permissions setup confusion, is the only issue that points to an improvable aspect of the onboarding flow, and is taken up in the qualitative feedback discussion below. It is also worth noting that three of the five issues involved participants who were not using a compatible wearable at the time. Among the fifteen participants who used a smartwatch or fitness band throughout the study, no data completeness issues were reported, suggesting that wearable integration meaningfully improves the reliability and richness of the data MoodSense is able to collect.

7.4 Permission Concerns

Sixteen of twenty-one participants (76 %) reported no discomfort with any of the requested permissions. Five participants (24 %) indicated that a particular permission had given them pause. The distribution of concerns across permission types is shown in Table 7.4.

Table 7.4: Permission concerns reported by participants (n = 21)

Permission	**Participants**	**%**
Calls and SMS logs only	3	14 %
Location only	1	5 %
Location and Calls/SMS logs	1	5 %
No concerns	16	76 %

Importantly, none of the five participants who reported a concern stated that the concern caused them to stop using the application or to refuse the permission entirely. In several cases the concern resolved once the in-app explanation was read. One participant described this directly:

> *"At first I was a little sceptical about having an application have permission with my calls and SMS logs, but then the app explained that it wouldn't look into what my messages said but only the number of what I received, and I felt more comfortable."*

Another participant who raised concerns about location and call logs framed the issue as a general privacy disposition rather than a specific objection to MoodSense:

> *"I do not usually grant such permissions to any apps, because they feel intrusive to my privacy."*

These responses suggest that the permission rationale screen is functioning as intended for most users, but that the explanation could be made more visible or prominent for users who skip or skim it. This is discussed further in Section 7.6.

7.5 Qualitative Feedback

7.5.1 Most Useful Features

Participants were asked to identify the feature or screen they found most useful. Table 7.5 summarises the responses.

Table 7.5: Features identified as most useful (multiple selections permitted)

Feature	**Count**	**%**
Insights screen	16	76 %
Sensor tracking	13	62 %
Trends screen	12	57 %
Daily check-in	8	38 %
Reminders	1	5 %

The Insights screen was the most frequently cited useful feature (76 %), followed by Sensor tracking (62 %) and Trends screen (57 %). The prominence of the Insights screen is significant because it represents the core output of the system's baseline comparison logic, the feature that transforms raw behavioural data into personally relevant observations. Several participants described the Insights screen as particularly meaningful after a few days of use:

> *"The insights screen at first it wasn't my most liked feature but after a few days of using the application it was really insightful and I got many useful information."*

> *"In general I liked it, but what I liked the most was the insights screen. I think it was good that my data were being compared together and not based on what the norm should be. It was really interesting to compare the different versions of myself."*

The second quote is particularly relevant to the design objective of personalisation. The participant explicitly recognises and values the fact that the application compares against the individual's own baseline rather than against population norms, which is precisely the approach described in Chapter 5.

7.5.2 Positive Themes

Three themes emerged from the open-ended responses about what participants liked most.

7.5.2.0.1 Clarity and ease of understanding. Multiple participants commented on the accessibility of the information presented. Descriptions such as "easy to understand," "clear," "tidy," and "easy to navigate" recurred across responses, consistent with the high Perspicuity subscale score. One participant wrote:

> *"I really liked how easy it was to understand the data presented. They were shown in a really tidy way and it was easy to immediately find your information and whatever you wanted to see."*

7.5.2.0.2 Personal baseline rather than external comparison. Several participants explicitly noted that they appreciated being compared to themselves rather than to a standard or norm. The non-judgmental framing of insights was highlighted by multiple participants across different age groups:

> *"It told me what was happening, for example my sleep varied or that my steps were down, and it didn't make me feel that I was doing something wrong or bad about myself."*
>
> *"Seeing sleep pattern changes during the days I was stressed was insightful, and in general I didn't get a feeling of being judged or being told what is right and wrong."*

7.5.2.0.3 Low effort and low intrusiveness. Several participants noted that MoodSense did not demand much of their time. The brief daily check-in and passive background sensing were mentioned positively:

> *"I think that it didn't require very much of my time. The only thing I needed to do was the daily check-in that didn't even need a minute of my time, and since I had the notifications on I remembered to do it as well."*
>
> *"It was something different and definitely helpful. Didn't need too much of my time for it to work correctly and give me helpful information."*

7.5.3 Improvement Suggestions

Four categories of improvement were raised across the open-ended responses.

7.5.3.0.1 Enhanced wearable and data integration. The most frequently raised improvement theme was the desire for deeper wearable integration. Participants who used a compatible smartwatch generally reported a richer experience, and several noted that the application would benefit from broader wearable support and from being able to collect more physiological data:

> *"I think the fact that I wasn't using a watch, from what I understood, was not as ideal as it would be with a watch, since a watch can track more data easier than phones, like sleep and steps."*

7.5.3.0.2 Additional check-in flexibility. Two participants suggested that offering multiple check-ins per day could add value, for example one in the morning and one in the evening. One participant suggested this while also acknowledging that the non-mandatory nature of the existing check-in felt appropriate:

> *"I think checking the mood on a standard time each time or multiple times (when you wake up and before you sleep) would be more practical. But it was nice that it didn't feel mandatory. It allowed me to be more honest."*

7.5.3.0.3 Onboarding and permission explanations. Two participants suggested that the explanation of each permission could be clearer or more prominent before acceptance:

> *"Maybe a clearer explanation of why each permission is needed and what each permission is used for before accepting it would be an improvement, so that the user feels more secure and comfortable."*

This feedback directly echoes the non-functional requirement NR-07 (Section 4.3.2), which specified that the onboarding flow must explain each permission before collection begins. The responses suggest that while this requirement was implemented, its visibility could be improved in a future iteration.

7.5.3.0.4 Content enrichment. One participant suggested adding contextually relevant content to complement the behavioural insights, such as articles, guides, or resources tailored to the user's current patterns. While outside the scope of the current implementation, this represents a meaningful direction for future development and is discussed in Chapter 8.

7.6 Discussion

The evaluation results present a coherent and largely positive picture of MoodSense's performance against its stated design goals.

7.6.1 Usability and Clarity

The UEQ results confirm that the application meets its usability objectives. A Perspicuity score of $+2.71$ is the highest of all subscales, reflecting the finding that participants found the interface easy to understand and learn. This is consistent with the qualitative responses, in which clarity, ease of navigation, and legible presentation of data were cited repeatedly and unprompted. The design decision to present behavioural metrics in plain language rather than raw sensor values, and to build the interface around a small number of focused screens with a single purpose each, appears to have been effective.

The relatively lower Stimulation score ($+1.79$), while still clearly positive, points to a trade-off inherent in the design. An application that prioritises clarity and low burden over richness and novelty will naturally score lower on hedonic dimensions than a more feature-dense product. Given that the design explicitly prioritised simplicity and low burden to serve users who would benefit most from a sustainable, undemanding tool [5, 14], this trade-off was deliberate and the results confirm it was understood and accepted by participants.

7.6.2 Privacy and Trust

The permission concern results are encouraging. Three-quarters of participants reported no concerns at all, and among the five who did, none described the concern as preventing them from using the application. The Dependability subscale ($+2.14$), which reflects perceived security and trustworthiness, was rated positively by all participants. Several qualitative responses specifically mentioned the experience of understanding and then accepting permissions they had initially viewed with caution, suggesting that the in-app permission rationale was functioning as intended.

The focus of concern on Calls and SMS logs (four participants, across both the sole and combined responses) and Location (two participants) is consistent with findings in the literature that identify communication and location data as the categories users find most sensitive [1, 3]. The design decision to disable these sensors by default and require explicit opt-in is directly validated by this finding: users who chose not to enable these permissions were protected from a concern they would have had, while those who did opt in were given a clear explanation first.

The onboarding permission explanation was identified by two participants as an area for improvement. This is a constructive finding: the mechanism exists and was described positively, but its visibility at the moment of the first permission request could be made more prominent. A revised onboarding flow that surfaces the rationale in closer proximity to the permission prompt itself, rather than in a separate screen, would address this directly.

7.6.3 Perceived Usefulness and Self-Awareness

The qualitative data provides the strongest evidence that MoodSense achieves its primary purpose. Unprompted, multiple participants described moments of genuine self-awareness that arose from using the application. One participant observed a connection between stress and reduced physical activity and social contact:

> *"I realised that when I had many things to do and I was stressed, my steps were down and also my calls were nearly zero."*

Another described a broader change in self-perception:

> *"It kind of changed the way I look at my data and my mood."*

A third made a similar observation:

> *"I think it made me connect more with myself. How I looked at my different data and how they are connected to my mental health. I think it opened my mind into a different way of looking at my life in general."*

These responses align closely with the goals articulated in the related work: that passive sensing tools can be meaningful not because they produce clinical outputs but because they make visible patterns that individuals were not previously aware of [5, 7]. The emphasis on personal baselines rather than external norms was explicitly recognised and valued by participants, confirming that the decision to avoid population-level comparisons and clinical language was the right approach for this user group and context.

7.6.4 Limitations

Several limitations of the evaluation should be acknowledged. The sample of 21 participants is relatively small and was drawn from the researcher's personal and academic network, which introduces a degree of familiarity bias. Participants who agreed to help with the study may have been more positively disposed toward the application than an independent sample would be. The sample is also skewed toward younger adults, with 47.6 % aged 18–24, limiting the generalisability of the findings to older populations. The minimum usage period of three days, while sufficient for the Insights screen to begin generating comparisons, is short relative to the multi-week periods typical in passive sensing research [3], and longer use might reveal usability issues or engagement patterns that did not emerge within the study window. Finally, the absence of Huawei device users in the sample, despite Huawei Health's known incompatibility with Health Connect being a documented limitation, means that this limitation could not be directly evaluated.

8 Conclusion

8.1 Final Remarks

This thesis presented the design, implementation, and evaluation of MoodSense, a passive behavioural sensing application for Android that supports personal mental health self-awareness. The central motivation was a gap identified in the related literature: existing mood and well-being applications tend either to demand too much active effort from users, leading to high dropout rates, or to present behavioural data without sufficient context for users to interpret it meaningfully. MoodSense was designed to address both problems simultaneously by collecting behavioural signals passively in the background and surfacing them as plain-language, personally contextualised insights rather than raw numbers or clinical assessments.

The system was built around four guiding principles: simplicity, trust, personalisation, and low burden. These principles were not merely aspirational. They shaped concrete architectural and interface decisions throughout the project. The local-first hybrid architecture ensures that raw sensor data never leaves the device, directly addressing the privacy concerns that research consistently identifies as a barrier to adoption in mental health applications. The rolling personal baseline approach means that insights are always relative to the individual's own history rather than to population norms, avoiding the misinterpretation and distress that clinical or comparative framing can cause. The minimal daily check-in, completable in under thirty seconds, reflects the finding that sustainable engagement requires low rather than high daily effort. The opt-in design for all sensitive data sources, with all optional sensors disabled by default, embeds privacy as a starting position rather than an afterthought.

The implementation brought together a range of Android platform capabilities, including foreground services and WorkManager for reliable background sensing, Android Health Connect for wearable integration, Firebase Realtime Database for optional cloud backup, and Jetpack Compose with the MVVM architectural pattern for the user interface. The result is a fully functional Android application that collects step count, screen usage, sleep duration, mobility, communication patterns, and heart-rate data from compatible wearables, computes rolling personal baselines for each metric, and generates insight cards when meaningful deviations are detected. The daily check-in presents four fixed questions covering mood, stress, sleep quality, and energy level.

The evaluation involved twenty-one participants who used MoodSense for a minimum of three consecutive days on their personal Android smartphones. Results across all six UEQ subscales were positive, with Perspicuity (+2.71) and Attractiveness (+2.51) rated highest, confirming that the application was perceived as clear, easy to understand, and visually well-designed. Technical problems were reported by five participants, all of which were attributable to device-specific platform behaviour or third-party wearable compatibility rather than defects in the application itself. Permission concerns were raised by five participants, primarily regarding calls and SMS logs and location access, consistent with the sensitivity of these data categories identified in the literature. In every case, the concern did not prevent continued use, and several participants described how the in-app permission rationale resolved their initial hesitation.

Most significantly, the qualitative responses provided direct evidence that MoodSense achieves its primary purpose. Participants described noticing connections between their mood, their physical activity, their sleep, and their social behaviour that they had not previously been aware of. Several described the experience as changing the way they relate to their own data. These responses confirm that a well-designed passive sensing application can serve as a meaningful self-awareness tool for everyday users, without requiring clinical framing, population comparisons, or significant daily effort from the people it is intended to help.

8.2 Future Work

MoodSense provides a functional foundation, but several directions for future development would meaningfully extend its value.

The most frequently requested improvement across the evaluation responses was deeper wearable integration. Currently, MoodSense retrieves heart-rate and sleep duration data from Android Health Connect, which limits compatibility to devices that expose data through that API. Huawei wearables, for example, do not currently integrate with Health Connect, leaving a significant portion of the Android wearable market unsupported. Future versions could explore direct integration with manufacturer SDKs or broader Health Connect adoption as the platform matures, enabling richer physiological data for a wider range of users.

A second direction is the enrichment of the check-in experience. The current implementation presents four fixed questions covering mood, stress, sleep quality, and energy level. A `QuestionTriggerEngine` component was designed and implemented to select additional questions dynamically based on the day's passive signals, for example surfacing a mental exhaustion question when HRV drops, or a social connectedness question when both communication frequency and mobility are low. This component was not wired to the live check-in screen in the current version. Completing that integration, and extending the check-in to support configurable schedules such as a morning and evening entry, represents a well-scoped and meaningful direction for future development.

Third, participants suggested that the Insights screen could be made more actionable by providing contextually relevant content alongside each insight. For example, when a participant's sleep pattern is disrupted, a link to a relevant article, breathing exercise, or evidence-based resource could complement the observational insight with something practical. This direction would need to be approached carefully to preserve the non-clinical framing that participants responded to positively, but done well it could meaningfully increase the utility of the application.

Fourth, the onboarding permission explanation could be made more prominent. Two participants suggested that the rationale for each permission would be more effective if surfaced at the moment of the permission request itself, rather than in a separate screen. A revised onboarding flow that presents a brief, plain-language explanation immediately before each system permission dialog would address this directly and is a straightforward improvement to implement.

Finally, the scope of MoodSense is currently limited to Android. A cross-platform extension, whether through a shared backend and separate native clients or through a cross-platform framework, would substantially broaden the population for whom the application is accessible. An iOS implementation would also open access to HealthKit, Apple's equivalent of Health Connect, and to the Apple Watch ecosystem, which represents a large share of the wearable market.

With continued development in wearable integration, onboarding clarity, and cross-platform reach, MoodSense could reach a substantially broader audience, supporting more people in understanding themselves while remaining a tool that is gentle, private, and undemanding by design.

BIBLIOGRAPHY

[1] J. R. Beames *et al.*, "Use of smartphone sensor data in detecting and predicting depression and anxiety in young people," 2024. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/39166029/

[2] J. Meyerhoff *et al.*, "Evaluation of changes in depression, anxiety, and social anxiety using smartphone sensor features: Longitudinal cohort study," 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34477562/

[3] T. W. Boonstra *et al.*, "Using mobile phone sensor technology for mental health research: Integrated analysis to identify hidden challenges and potential solutions," 2018. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC6090171/

[4] N. Shen *et al.*, "Passive sensing for mental health monitoring using machine learning with wearables and smartphones: Scoping review," 2025. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/40811794/

[5] Y. Zakai *et al.*, "Harnessing digital phenotyping for early self-detection of psychological distress," 2025. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/40868624/

[6] Y. Terhorst *et al.*, "Investigating smartphone-based sensing features for depression severity prediction: Observation study," 2025. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/39883512/

[7] K. P. Kruzan *et al.*, "The perceived utility of smartphone and wearable sensor data in digital self-tracking technologies for mental health," 2023. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/38873656/

[8] Unknown, "Internet of things for mental health: Open issues in data acquisition, self-organization, service-level agreement and identity management," 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33535714/

[9] S. Thomée, "Mobile phone use and mental health: A review of the research that takes a psychological perspective on exposure," 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30501032/

[10] R. Wang, F. Chen, Z. Chen, T. Li, G. Harari, S. Tignor, X. Zhou, D. Ben-Zeev, and A. T. Campbell, "StudentLife: Assessing mental health, academic performance and behavioral trends of college students using smartphones," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '14. ACM, 2014, pp. 3–14.

[11] R. Wang, W. Wang, A. daSilva, J. F. Huckins, W. M. Kelley, T. F. Heatherton, and A. T. Campbell, "Tracking depression dynamics in college students using mobile phone and wearable sensing," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 43:1–43:26, 2018.

[12] J. Zulueta, A. Piscitello, M. Rasic, R. Easter, P. Babu, S. A. Langenecker, M. McInnis, O. Ajilore, L. Zhan, and A. Leow, "Predicting mood disturbance severity with mobile phone keystroke metadata: A biaffect digital phenotyping study," *JAMA Psychiatry*, 2025, published online ahead of print.

[13] M. Abujelala, C. Abellanoza, S. Sharma, and F. Makedon, "Brain-ee: Brain enjoyment-engagement model using commercial EEG headset," *Frontiers in Digital Health*, vol. 3, 2021.

[14] F. Matcham *et al.*, "Remote assessment of disease and relapse in major depressive disorder (RADAR-MDD): A multinational, multicentre prospective cohort study," *PLOS Medicine*, vol. 20, no. 1, p. e1004102, 2023.

[15] J. Torous, M. Keshavan, E. J. Lenze, and J.-P. Onnela, "Digital mental health research: From promises to progress," *World Psychiatry*, vol. 24, no. 1, pp. 43–54, 2025.

[16] M. J. Sliwinski, J. A. Mogle, J. Hyun, E. Munoz, J. M. Smyth, and R. B. Lipton, "Reliability and validity of ambulatory cognitive assessments," *Assessment*, vol. 25, no. 1, pp. 14–30, 2021.

[17] K. K. Fitzpatrick, A. Darcy, and M. Vierhile, "Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): A randomized controlled trial," *JMIR Mental Health*, vol. 4, no. 2, p. e19, 2017.

[18] B. Inkster, S. Sarda, and V. Subramanian, "An empathy-driven, conversational artificial intelligence agent (Wysa) for digital mental well-being: Real-world data evaluation mixed-methods study," *JMIR mHealth and uHealth*, vol. 6, no. 11, p. e12106, 2018.

[19] S. Lagan, P. Aquino, M. R. Emerson, K. Fortuna, R. Walker, and J. Torous, "Actionable health app evaluation: Translating evidence-based medicine framework for mhealth apps," *JMIR mHealth and uHealth*, vol. 8, no. 7, p. e15847, 2022.

[20] D. Ferreira, V. Kostakos, and A. K. Dey, "AWARE: Mobile context instrumentation framework," in *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, framework actively maintained; see https://awareframework.com/.

[21] Google LLC, "Guide to app architecture," Android Developers Documentation, 2024.

[Online]. Available: https://developer.android.com/topic/architecture

[22] A. Cavoukian, "Privacy by design: The 7 foundational principles," Information and Privacy Commissioner of Ontario, Tech. Rep., 2011. [Online]. Available: https://www.ipc.on.ca/wp-content/uploads/resources/7foundationalprinciples.pdf

[23] Google LLC, "Firebase documentation," Firebase by Google, 2024. [Online]. Available: https://firebase.google.com/docs

[24] B. Laugwitz, T. Held, and M. Schrepp, "Construction and evaluation of a user experience questionnaire," in *HCI and Usability for Education and Work*, ser. Lecture Notes in Computer Science, vol. 5298. Springer, 2008, pp. 63–76.

APPENDICES

Appendix A: Evaluation Questionnaire

This appendix presents the questionnaire used to evaluate the MoodSense application. The questionnaire was distributed to participants after they had used the application for at least three consecutive days. It consisted of informed consent and eligibility confirmation, participant background questions, a user experience scale based on the User Experience Questionnaire (UEQ), technical problem reporting, permission concern questions, and an open-ended final comments section.

A.1 Informed Consent and Eligibility

Participants were asked to confirm the following before proceeding:

1. I confirm that I have used MoodSense for at least 3 days.
    - Yes
    - No
2. I have read the study information above and agree to participate voluntarily.
    - Yes
    - No

Note: Participation in this study was voluntary. Responses were anonymous and used only for academic research. No passwords or sensitive personal information were requested.

A.2 Participant Background

3. What is your age group?
    - 18–24
    - 25–34
    - 35–44
    - 45–54
    - 55–64
    - 65+
4. What is your gender?
    - Female

- Male
- Non-binary
- Prefer not to say

5. What type of phone did you use with MoodSense?
    - Samsung
    - Xiaomi
    - Huawei
    - Google Pixel
    - Other Android
6. Did you use a smartwatch or fitness band during the study?
    - Yes
    - No
7. Before using MoodSense, how often did you reflect on or track your mood?
    - Every day
    - A few times a week
    - A few times a month
    - Rarely
    - Never
8. Before using MoodSense, had you used any app for mood, sleep, or wellbeing tracking?
    - Yes
    - No

A.3 User Experience Questionnaire

This section consisted of pairs of contrasting attributes drawn from the User Experience Questionnaire (UEQ). Participants were asked to select a value from 1 to 7 based on their immediate impression of the application, where 1 indicated the left-hand attribute and 7 indicated the right-hand attribute.

Table A.1: User Experience Questionnaire items

Attribute	1	2	3	4	5	6	7	Opposite Attribute
annoying	○	○	○	○	○	○	○	enjoyable
not understandable	○	○	○	○	○	○	○	understandable
easy to learn	○	○	○	○	○	○	○	difficult to learn
valuable	○	○	○	○	○	○	○	inferior
boring	○	○	○	○	○	○	○	exciting
not interesting	○	○	○	○	○	○	○	interesting
obstructive	○	○	○	○	○	○	○	supportive
complicated	○	○	○	○	○	○	○	easy
secure	○	○	○	○	○	○	○	not secure
motivating	○	○	○	○	○	○	○	demotivating
clear	○	○	○	○	○	○	○	confusing
impractical	○	○	○	○	○	○	○	practical
organized	○	○	○	○	○	○	○	cluttered
attractive	○	○	○	○	○	○	○	unattractive
friendly	○	○	○	○	○	○	○	unfriendly

A.4 Technical Problems

16. Did you experience any technical problems while using MoodSense?
    - Yes *(proceed to questions 17–19)*
    - No *(skip to question 20)*
17. Please briefly describe the technical problem you experienced.

    ____

18. Which screen or feature had the problem? *(select all that apply)*
    - Daily check-in
    - Trends
    - Insights
    - Permissions / setup
    - Notifications
    - Other

19. How serious was the problem for you?
    - Minor
    - Moderate
    - Serious
20. Were you still able to continue using the app after the problem?
    - Yes
    - No

A.5 Permission Concerns

21. Was there any permission you felt uncomfortable granting?
    - Yes *(proceed to questions 22–24)*
    - No *(skip to question 25)*
22. Which permission concerned you most? *(select all that apply)*
    - Location
    - Health Connect
    - Screen time
    - Calls / SMS logs
    - Notifications
    - Other
23. Please briefly explain why this permission concerned you.

    ____

24. Did this concern affect how you used the app?
    - Yes
    - No
    - Not sure

A.6 Final Comments

25. Which feature did you find most useful? *(select all that apply)*

- Daily check-in
- Trends screen
- Insights screen
- Sensor tracking
- Reminders
- Other

26. What did you like most about MoodSense?

    ____

27. What would you improve in MoodSense?

    ____

28. Any other comments?

    ____