

Design and Usability Evaluation of a Weather-Aware Mobile Movie Recommendation System Using a Human-Centered AI Approach Entertainment

Vincentius Bryan Kwandou¹

¹*Informatics Engineering Department, Universitas Atma Jaya Makassar, Makassar, Indonesia*

*Corresponding author: Vincentius.kwandou@gmail.com

ABSTRACT

The global digital entertainment industry demands personalized, context-aware recommendation systems that respond to users' immediate environmental circumstances. NNG Cinema is a weather-aware mobile movie recommendation system developed using Flutter 3.16 and Dart 3.2, integrating real-time meteorological data from the OpenWeatherMap API with movie discovery capabilities from The Movie Database (TMDB) API. Grounded in Context-Aware Recommender System (CARS) theory, mood management theory, and human-centered AI design principles, the system automatically maps seven meteorological condition categories to curated genre clusters derived from environmental psychology literature, requiring no additional user input. This study employed a six-phase design-and-build methodology. Twenty-four black-box functional test cases were executed across six categories in three cycles, and 30 purposively sampled participants completed the System Usability Scale (SUS) following a single ten-minute session of free system interaction. All 24 test cases achieved Pass status across all three cycles (100% pass rate). SUS evaluation produced a mean score of 80.33 (SD = 7.84), placing NNG Cinema in the Good-to-Excellent usability range and 12.33 points above the established population median. Findings indicate that fully automated environmental context integration is technically feasible and elicits a positive usability perception as a basis for mobile movie recommendation. These results should be read as evidence of design feasibility and perceived usability only: recommendation accuracy, perceived relevance, user satisfaction, and technology acceptance in the sense of TAM or UTAUT were not measured, and no comparative baseline was evaluated.

Keywords: *context-aware recommender systems, Flutter, human-centered AI, mobile applications, weather-based recommendation*

1. INTRODUCTION

The global film industry generated revenues exceeding one hundred billion US dollars annually before streaming platforms transformed content distribution (Escandon, 2020). As digital entertainment libraries expanded to millions of titles, recommender systems became the dominant mechanism for content discovery. Netflix attributes over eighty percent of consumed content to recommendation-driven selection (Gomez-Urbe & Hunt, 2015). Conventional systems operate on historical behavioral data, collaborative filtering signals, and content metadata, without accounting for the user's immediate environmental circumstances at the moment of consumption.

Context-Aware Recommender Systems (CARS) extend traditional recommendation architectures by incorporating situational variables as first-class inputs (Adomavicius & Tuzhilin, 2005). Weather represents a compelling contextual variable: it is objective, automatically measurable via public APIs, and repeatedly reported to be associated with human

affective states, although the association is probabilistic, not deterministic. Environmental psychology reports that precipitation and reduced sunlight tend to be associated with introspective, subdued mood states, while clear conditions tend to correlate with positive affect and social orientation (Schwarz & Clore, 1983; Zhang & Li, 2025). These mood states may in turn influence media selection, consistent with Zillmann's (1988) mood management theory, which holds that individuals select content that counteracts or complements their current emotional state. This study treats weather as one contextual cue among several, moderated by individual differences, habit, culture, and viewing situation, and not as a deterministic predictor of film preference.

Despite these theoretical connections, weather remains underutilized as a recommendation signal. The MAWI Rec system (Duncan et al., 2024) demonstrated at ACM RecSys 2024 that severe weather data improves recommendation quality, yet consumer-facing mobile implementations remain sparse. Mateos and Bellogín (2025) confirm through systematic literature review that automatic context sensing is a recognized open problem in the CARS research agenda. Prior work falls into three groups, and each one leaves the present problem unsolved. Duncan et al. (2024) use weather as a predictive feature but do so server-side and offline, in the setting of severe weather events, not everyday conditions. Casillo et al. (2026) deploy automatic context sensing in a mobile setting, but for cultural-heritage tourism, where context is location, not weather, and the item catalogue is small and geographically bounded. Piçarra et al. (2022) survey emotion-driven movie retrieval systems, but in almost all of them the affective state must be declared by the user, which reintroduces exactly the input burden that context sensing is meant to remove. What remains absent is a consumer-facing mobile client in which weather sensing, affect-informed genre mapping, and catalogue retrieval are chained end to end with no user input at all. This research addresses that specific gap by developing and evaluating NNG Cinema: a Flutter-based mobile application delivering fully automated, weather-contextualized film recommendations. Consistent with this scope, the study evaluates functional correctness (24 black-box test cases) and perceived usability (SUS instrument, 30 participants); recommendation accuracy and user satisfaction were not measured and lie outside the present scope.

1.1. Background and Problem Statement

The proliferation of streaming platforms has created a content abundance paradox: users face choice paralysis when selecting from libraries numbering in the hundreds of thousands. A user deciding what to watch on a rainy evening has fundamentally different contextual needs than the same user browsing on a sunny afternoon. Today's systems present

identical recommendations regardless of environmental context, representing a gap between theoretical CARS capabilities and deployed system behavior.

1.2. Literature Review

1.2.1. Context-Aware Recommender Systems

Burke (2002) provides the foundational taxonomy of hybrid recommender systems. Adomavicius and Tuzhilin (2005) formally define CARS and establish the contextual pre-filtering, post-filtering, and contextual modeling paradigms. Adomavicius et al. (2011) extend this framework to include automatic context sensing. Mateos and Bellogín (2025) survey 147+ CARS papers from 2015–2024, confirming weather as an emerging but underexplored signal. Casillo et al. (2026) demonstrate CARS application in cultural heritage recommendation, establishing a template for domain-specific deployment. These studies converge on the usefulness of contextual pre-filtering but diverge sharply on how context is obtained, and that divergence matters here. Burke (2002) and Adomavicius and Tuzhilin (2005) treat context as given, which keeps the modelling tractable but silently assumes an acquisition mechanism that is never specified. Adomavicius et al. (2011) name automatic sensing explicitly, yet their treatment stays architectural and never becomes empirical. Casillo et al. (2026) do sense context automatically, but their signal is location within a bounded heritage site, so their result does not transfer directly to an unbounded film catalogue. Mateos and Bellogín (2025) sharpen the criticism: most of the 147 systems they survey are evaluated on static datasets with context replayed from logs, so reported gains are not evidence that context can be captured usefully in a live client. The open question is not whether context helps. It is whether context can be acquired at the moment of consumption without cost to the user. That is the question this study takes up.

1.2.2. Weather, Mood, and Media Preference

Schwarz and Clore (1983) established the foundational mood-as-information framework. Zhang and Li (2025) synthesize over a century of weather-wellbeing research confirming robust links between meteorological conditions and affective states. Piçarra et al. (2022) review emotion-based movie recommendation, identifying weather-driven affective states as a valid input to genre preference modeling. Cohen-Kalaf et al. (2022) demonstrate that films possess stable, measurable emotional signatures that align with genre-based affective predictions. Duncan et al. (2024) confirm that weather context carries information predictive of user satisfaction with recommended content. These strands are complementary but unequal in evidential weight, and conflating them would overstate the case for weather-driven recommendation. The weather–affect link is the strongest of the three: it is replicated across large populations and long time series (Zhang & Li, 2025), though the reported effects are population-level averages and modest at the level of an individual on a given day. The affect–genre link is weaker, resting largely on correlational, self-report, and content-annotation

evidence (Piçarra et al., 2022; Cohen-Kalaf et al., 2022), and mood management theory itself allows two opposite behaviours (mood repair and mood congruence) without specifying in advance which will occur. Duncan et al. (2024) supply the only direct computational evidence for the full chain, but under severe-weather conditions and offline evaluation, which is a narrower and more extreme case than ordinary daily weather. The literature supports weather as a defensible contextual prior for an initial recommendation set. It does not support weather as a substitute for personalization. The mapping used in this study should be read in that light.

1.2.3. Human-Centered AI and Mobile Development

Shneiderman (2020) defines human-centered AI as systems enhancing human capability, providing transparency, and minimizing user burden. Lu et al. (2024) find that transparency without complexity best supports user trust in AI systems. Flutter, used by 46% of professional mobile developers (Stack Overflow, 2023), provides an accessible cross-platform implementation environment. Aung et al. (2024) demonstrate Flutter's practical deployment viability in structured development contexts. One limitation runs through this body of work. Shneiderman (2020) and Lu et al. (2024) are conceptual and review-based respectively, and although both are widely cited as design guidance, neither specifies how principles such as transparency or reduced user burden are to be instantiated in a recommender interface, nor how their presence should be verified. Aung et al. (2024), correspondingly, evaluate the development environment and not any user-facing outcome. The consequence is that “human-centered” is frequently asserted of deployed systems without a stated referent. To avoid that weakness, this study does not claim the label on the strength of its usability score alone; Section 2.6 sets out the specific design features through which each principle is realized, and Section 3.3 states which of them the present evaluation can and cannot substantiate.

1.2.4. Research Gap

Three observations follow from the literature reviewed above. First, CARS research remains dominated by desktop and web settings in which context is either supplied manually by the user or attached to a static offline dataset, and systematic review continues to list automatic context acquisition as an open problem (Mateos & Bellogín, 2025). Second, where weather has been used computationally it has been used as an offline feature under severe-weather conditions (Duncan et al., 2024), not as a live signal inside a shipped consumer client. Third, work that is genuinely mobile-native and automatically context-sensing has been demonstrated in narrow domains such as cultural heritage (Casillo et al., 2026), not in large-catalogue entertainment. The gap addressed here is narrow and specific: a mobile-native entertainment recommender in which environmental context is acquired automatically at the moment of consumption. NNG Cinema is offered as one instantiation of that design, and this study reports its technical feasibility and its perceived usability against established usability benchmarks, not its recommendation quality relative to existing approaches.

2. METHODS

This study employed a design-and-build research methodology, appropriate for applied software engineering research where the primary contribution is a functioning artifact evaluated against technical and usability criteria. Two distinct methods are reported below and should be kept separate. The system-development method is the six-phase design-and-build cycle (requirements analysis, architectural design, implementation, functional testing, usability evaluation, and reporting) that produced the artifact and is described in Sections 2.2 to 2.6. The evaluation method consists of two instruments applied to the finished artifact: black-box functional testing, which assesses whether the implementation conforms to its specification (Section 2.7.1), and the System Usability Scale, which assesses perceived usability (Section 2.7.2). The study is descriptive and single-group: no experimental manipulation, control condition, or comparative baseline system was involved, and no claim of relative effectiveness can be derived from this design. Six sequential phases were executed as illustrated in Figure 1.



Figure 1. Research Methodology Flowchart (6-Phase Design-and-Build)

2.1. Research Design

The research adopted a descriptive evaluative design. The independent variable is the weather-context signal derived from the OpenWeatherMap API; dependent variables are functional test outcomes (pass/fail per test case across three cycles) and user experience quality (SUS composite score). Table 1 summarizes the research design parameters.

Table 1. Research Design Summary

Parameter	Description
Research type	Design-and-build (applied software engineering)
Research design	Descriptive evaluative
Independent variable	Weather context (OWM API condition code + temperature)
Dependent variables	Functional correctness (pass rate); Perceived usability (SUS score)
Functional evaluation	Black-box test cases (24 cases, 6 categories, 3 cycles)
Usability instrument	System Usability Scale — SUS (Brooke, 1996)
Participant sampling	Purposive: age 18-40, smartphone user, streaming app experience
Sample size	N = 30 participants
Ethics	Voluntary, anonymized, right to withdraw

2.2. System Architecture

NNG Cinema implements a three-layer Clean Architecture separating presentation, domain, and data concerns. Figure 2 illustrates the complete architecture. The BLoC state management pattern employed across the three primary BLoCs.

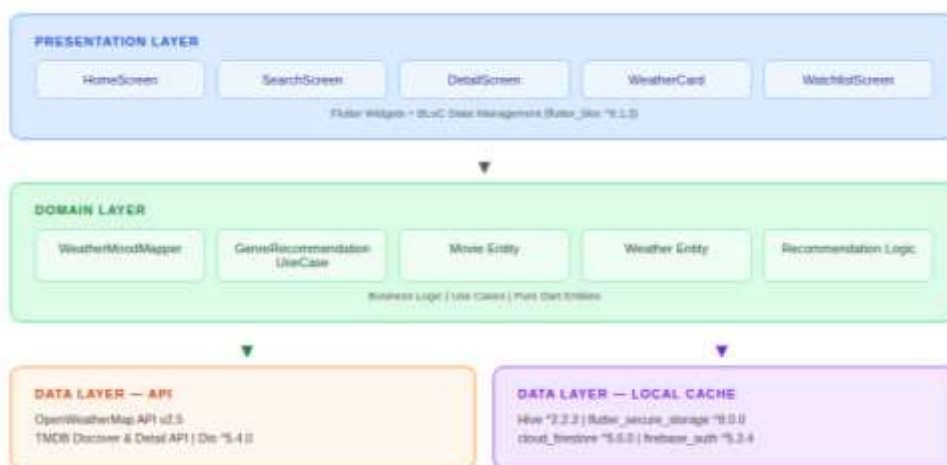


Figure 2. NNG Cinema System Architecture (Clean Architecture — 3 Layers)

The Presentation Layer renders all UI components and delegates user events to the appropriate BLoC (Business Logic Component). The Domain Layer encapsulates the WeatherMoodMapper algorithm and GenreRecommendationUseCase, operating on pure Dart entities without external dependencies. The Data Layer manages all external API communication through repository implementations. Dependency injection is managed through GetIt (^7.6.4); navigation through go_router (^13.0.1).

Each BLoC class follows the unidirectional data flow: Event dispatched by UI Widget → BLoC processes event via repository → emits new State → UI Widget rebuilds. The three BLoCs are fully decoupled: WeatherBloc handles location and weather fetch events; MovieBloc handles TMDb discovery requests keyed to genre_ids[]; SearchBloc manages user-initiated keyword searches. All BLoCs register repository dependencies through GetIt at application startup.

2.3. Application Data Flow

On application launch, the system initiates a four-step weather context acquisition pipeline: (1) device location request via Geolocator; (2) HTTP GET to OpenWeatherMap API; (3) JSON response parsing; (4) WeatherMoodMapper invocation to derive genre_ids[]. Figure 3 illustrates the complete data flow.



Figure 3. NNG Cinema Application Data Flow Diagram

API responses are cached locally using Hive. Cache invalidation occurs after 30 minutes for weather data and 6 hours for movie listings. Error states — network unavailability, API quota exceeded, location permission denial — trigger dedicated error screens with retry affordances.

2.4. Weather-to-Genre Mapping Algorithm

The WeatherMoodMapper translates OpenWeatherMap condition code groups into TMDB genre identifier clusters. Seven meteorological states are defined, each mapped to two to five genre IDs synthesized from mood management theory (Zillmann, 1988), affective genre research (Piçarra et al., 2022; Cohen-Kalaf et al., 2022), and environmental psychology (Schwarz & Clore, 1983; Zhang & Li, 2025). Table 2 presents the complete algorithmic specification.

Table 2. Complete Weather Condition to Genre Mapping

Weather State	OWM Codes	TMDB Genres	Psychological Rationale	Key References
Clear Sky	800	Action, Adventure, Comedy, Animation	Positive affect; elevated arousal favors stimulating, social content	Schwarz & Clore (1983); Zillmann (1988)
Cloudy	801-804	Drama, Documentary, Biography	Neutral-subdued affect; reflective mood suits informational content	Zhang & Li (2025)
Rainy	500-531	Drama, Romance, Mystery, Crime	Subdued arousal; emotionally resonant narratives align with mood	Piçarra et al. (2022); Cohen-Kalaf et al. (2022)
Stormy	200-232	Thriller, Horror, Sci-Fi	High arousal from external stimulus; tension genres channel it	Zillmann (1988)
Snowy	600-622	Family, Animation, Fantasy	Snow associates with warmth-seeking; comforting genres reinforce coping	Cohen-Kalaf et al. (2022)
Cold <15°C	temp<15	Adventure, Fantasy, History, War	Physical discomfort drives escapism; immersive genres counteract it	Zillmann (1988); Duncan et al. (2024)
Drizzle	300-321	Romance, Drama, Music, Animation	Mildly subdued state; warm, aesthetically rich narratives suit mood	Cohen-Kalaf et al. (2022)

2.5. Application Implementation

NNG Cinema was implemented across a 14-week development cycle using Flutter 3.16 and Dart 3.2, targeting Android (API 21+) and iOS (12.0+) simultaneously. The home screen

presents a dynamic weather card and a scrollable 2-column film grid. Table 3 presents the full dependency manifest.

Table 3. NNG Cinema Dependency Manifest (pubspec.yaml)

Package	Version	Purpose
flutter_bloc	^8.1.3	BLoC state management pattern
get_it	^7.6.4	Dependency injection and service locator
go_router	^13.0.1	Declarative navigation with type-safe routes
dio	^5.4.0	HTTP client for API communication
geolocator	^11.0.0	Device GPS location acquisition
cached_network_image	^3.3.0	Efficient poster image loading with caching
hive / hive_flutter	^2.2.3 / ^1.1.0	Local key-value caching store
firebase_auth	^5.3.4	User authentication (Google Sign-In)
cloud_firestore	^5.6.0	Watchlist persistence (NoSQL cloud)
firebase_messaging	^15.2.10	Push notification support
firebase_crashlytics	^4.3.10	Crash reporting and analytics
equatable	^2.0.5	Value equality for BLoC state comparison
dartz	^0.10.1	Functional programming (Either monad for errors)
shimmer	^3.0.0	Skeleton loading placeholder animations
google_fonts	^6.1.0	Typography customization

2.6. Operationalization of Human-Centered AI Principles

Because “human-centered AI” is otherwise an unfalsifiable label, the four principles drawn from Shneiderman (2020) and Lu et al. (2024) are stated here together with the concrete design decisions that implement them in NNG Cinema, so that each claim can be traced to an artifact feature and not to a usability score.

Minimized user burden. The system requires no mood declaration, no onboarding questionnaire, no genre pre-selection, and no cold-start preference elicitation. The weather signal is acquired from device location and the OpenWeatherMap API at launch, so the user reaches a contextualized recommendation set with zero deliberate input. This is the principle most directly embodied in the artifact and the one the SUS results speak to most closely, since items Q3, Q7, and Q10 concern ease of use, learnability, and prior learning required.

Transparency of the contextual basis. The home screen renders a persistent weather card showing the detected condition and temperature above the film grid, so the input driving the current recommendation set stays visible at all times and is never buried inside an opaque model. The mapping is rule-based and fully enumerated in Table 2, which makes the system’s

reasoning inspectable and auditable in a way that a learned latent model would not be. The application displays the weather state, however, but does not currently render an explicit natural-language explanation of the form “rain detected, therefore Drama and Romance”; Transparency here is partial, and Section 3.3.4 returns to the point.

Preservation of user control. Automatic contextualization sets the default view only and is never exclusive. A keyword search function (SearchScreen) lets the user bypass the weather-derived set entirely and retrieve any title in the catalogue, and a persistent watchlist lets the user retain films independently of the prevailing condition. Swipe-to-refresh re-triggers acquisition on demand. The user can override the contextual inference at any point. This sets the design apart from a system that silently constrains what can be reached.

Informative failure and non-punitive error handling. Location denial, network loss, and API quota exhaustion each resolve to a dedicated screen that names the condition and offers a retry affordance or a route to system settings, and not to a crash, a blank grid, or a silent fallback that would misrepresent the contextual basis of the results. Test cases TC06, TC22, and TC23 verify these paths specifically.

Two principles frequently associated with human-centered AI are deliberately not claimed here. Fairness across user subgroups was not assessed, since the mapping is uniform by construction and no subgroup analysis was performed; and contestability, meaning the ability of a user to correct the system’s inference about their state, is not implemented, as the mapping cannot currently be adjusted or overridden at the rule level. These omissions are treated as limitations, not as achieved properties.

2.7. Evaluation Methods

2.7.1. Black-Box Functional Testing

System functional evaluation employed 24 black-box test cases across six categories: Weather API Integration (6), Recommendation Algorithm (6), UI Navigation (5), Movie Detail Display (4), Error Handling (2), and Search Functionality (1). Each test case specified scenario, preconditions, expected output, observed output, and pass/fail status. All cases were executed three times. Table 4 presents the complete 24-case inventory.

Table 4. Complete Black-Box Functional Test Case Inventory (N = 24)

No.	Category	Test Scenario	Expected Output	Status	Cycles
TC01	Weather API Integration	App fetches weather on launch via GPS	Condition & temp displayed within 2 s	PASS	3/3
TC02	Weather API Integration	Parses OWM code 800 (Clear Sky)	WeatherLoadedState with sunny data	PASS	3/3
TC03	Weather API Integration	Parses codes 500-531 (Rainy group)	Rainy state triggers Drama/Romance mapping	PASS	3/3

No.	Category	Test Scenario	Expected Output	Status	Cycles
TC04	Weather API Integration	Parses codes 200-232 (Thunderstorm)	Stormy state triggers Thriller/Horror mapping	PASS	3/3
TC05	Weather API Integration	Temperature < 15°C pathway activated	Cold mapping: Adventure/Fantasy/History/War	PASS	3/3
TC06	Weather API Integration	Invalid GPS coordinates supplied	Error message shown; app does not crash	PASS	3/3
TC07	Recommendation Algorithm	Clear Sky produces Action/Comedy grid	Correct genre_ids[] dispatched to TMDB	PASS	3/3
TC08	Recommendation Algorithm	Rainy weather produces Drama/Mystery grid	TMDB returns movies of mapped genres	PASS	3/3
TC09	Recommendation Algorithm	Snowy weather produces Family/Fantasy grid	Genre set changes correctly with weather	PASS	3/3
TC10	Recommendation Algorithm	Drizzle produces Romance/Music/Animation grid	Correct genre mapping and TMDB response	PASS	3/3
TC11	Recommendation Algorithm	Movie grid updates on weather change	BLoC emits new state; UI re-renders grid	PASS	3/3
TC12	Recommendation Algorithm	Minimum 20 results returned per state	TMDB Discover returns >=20 results	PASS	3/3
TC13	UI Navigation	Tap poster card goes to detail screen	DetailScreen shows full movie metadata	PASS	3/3
TC14	UI Navigation	Back navigation returns to home screen	Home grid state preserved on back	PASS	3/3
TC15	UI Navigation	Search tab navigates to SearchScreen	SearchScreen renders with empty state	PASS	3/3
TC16	UI Navigation	Watchlist tab shows saved movies	WatchlistScreen renders saved movie list	PASS	3/3
TC17	UI Navigation	Swipe-to-refresh triggers weather fetch	WeatherBloc emits loading then loaded	PASS	3/3
TC18	Movie Detail Display	Detail shows title and full synopsis	Title and synopsis visible and readable	PASS	3/3
TC19	Movie Detail Display	Detail shows runtime and TMDB rating	Runtime (min) and rating (0-10) visible	PASS	3/3
TC20	Movie Detail Display	Detail shows cast list	Cast names visible in scrollable row	PASS	3/3
TC21	Movie Detail Display	Poster image loads in detail view	High-res poster renders without placeholder	PASS	3/3
TC22	Error Handling	Simulated network loss during API call	Error message shown; retry available; no crash	PASS	3/3
TC23	Error Handling	Location permission denied by user	Fallback message shown; guides to settings	PASS	3/3
TC24	Search Functionality	Keyword search returns relevant results	Matching films displayed within acceptable time	PASS	3/3

2.7.2. System Usability Scale (SUS) Evaluation

The SUS instrument (Brooke, 1996) comprises ten Likert-format statements alternating between positive (items 1,3,5,7,9) and negative (items 2,4,6,8,10) phrasing, each rated 1–5. Scoring: contribution = (response-1) x 2.5 for positive items and (5-response) x 2.5 for negative items; composite score = sum x 2.5 (Sauro & Lewis, 2011). Benchmarks (Bangor et al., 2008): above 80.3 = Excellent, 68–80.3 = Good, 51–68 = OK, below 51 = Poor. Population median = 68.0 (Sauro & Lewis, 2011). Thirty participants were recruited purposively against three criteria: age 18–40, active smartphone ownership, and prior experience with at least one video streaming application. The composition of the sample (gender, age group, occupation, streaming-service usage frequency, film-viewing frequency, location, and testing device) is reported in Table 5. Sessions were conducted individually and in person in a quiet room on the Universitas Atma Jaya Makassar campus, with the researcher present but not intervening. The application was distributed as an Android build through Google Play Store; each participant used either the researcher’s own Android handset or their own device, connected to a 4G mobile network. Each participant received a five-minute orientation covering navigation only. The weather-to-genre mapping was not explained, so as not to bias perception of the recommendations. Participants then interacted freely with the application for ten minutes and completed the SUS questionnaire in printed form. A single ten-minute session exposes each participant to only the one meteorological condition prevailing at the time of their session. That window is adequate for judging interface usability, which is what the SUS measures, but it is too short to observe the behaviour that most distinguishes this system, namely how the recommendation set shifts as weather changes across hours and days. It also means that no participant experienced the weather-adaptivity of the system as a longitudinal user would. Weather-state coverage in testing came from the functional test cases (TC02–TC05, TC07–TC11), not from participant experience. This constraint is revisited in Section 3.3.4.

Table 5. Participant Demographic Profile (N = 30)

Variable	Category	n	%
Gender	Male	15	50.0
	Female	15	50.0
Age group	19–24 years	25	83.3
	31–38 years	5	16.7
Occupation	Undergraduate student	25	83.3
	Campus staff	5	16.7

Streaming-service usage	Daily	15	50.0
	Three to five times per week	7	23.3
	Weekly	8	26.7
Film-viewing frequency	Daily	15	50.0
	Three to five times per week	5	16.7
	Weekly	10	33.3
Testing location	Makassar, Indonesia	30	100.0
Testing device	Android handset	30	100.0

Participant age ranged from 19 to 38 years ($M = 23.6$, $SD = 5.16$). The sample falls into two clusters and does not spread evenly across the eligible range: 25 undergraduate students aged 19–24 and five campus staff aged 31–38, with no participants in the 25–30 band. Gender was balanced at 15 men and 15 women. All participants were resident in Makassar, Indonesia, so the weather conditions encountered during testing were drawn from a single tropical climate zone. All thirty sessions were conducted on an Android handset, so the usability data reflect the Android build exclusively; the iOS target, although supported by the codebase and exercised in functional testing, was not evaluated with users.

3. RESULTS AND DISCUSSIONS

3.1. Functional Testing Results

All 24 black-box test cases passed across all three execution cycles, yielding a 100 percent functional correctness rate. Figure 4 presents results by category. Weather API Integration tests confirmed correct parsing across all seven condition code groups, with API response times averaging 1.3 seconds (range: 0.9–1.9 s) under 4G mobile network conditions.

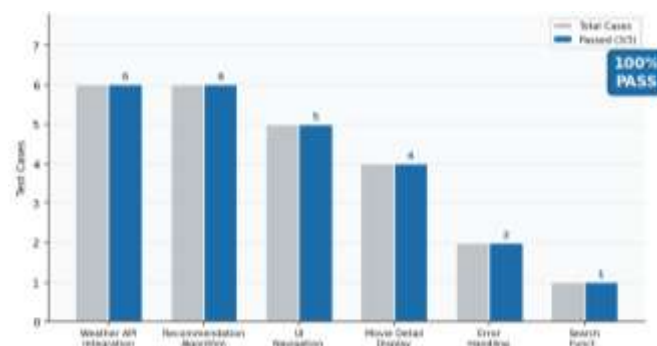


Figure 4. Functional Testing Results — 24/24 Test Cases Passed (100% Pass Rate)

Recommendation Algorithm tests verified all seven weather states trigger distinct and correct genre sets, and TMDB Discover queries return a minimum of 20 results per request. UI Navigation tests confirmed correct screen transitions and back-stack behavior on both Android

and iOS targets. Error Handling tests (TC22–TC23) demonstrated clean recovery from simulated network unavailability and location permission denial, with informative messaging and functional retry mechanisms.

3.2. SUS Evaluation Results

The SUS evaluation produced a mean score of 80.33 (SD = 7.84) across 30 participants — at the Good–Excellent boundary, 12.33 points above the population median. Figure 5 presents the score distribution and per-item contribution analysis. Figure 6 presents a radar chart of per-item normalized scores for a holistic usability profile view.

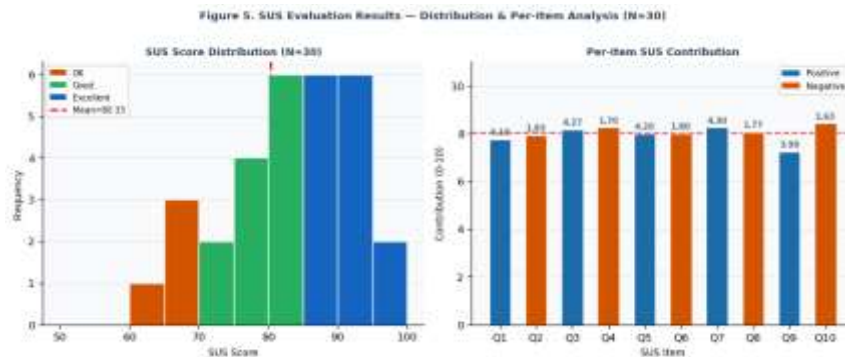


Figure 5. SUS Evaluation Results — Score Distribution and Per-Item Analysis (N=30)

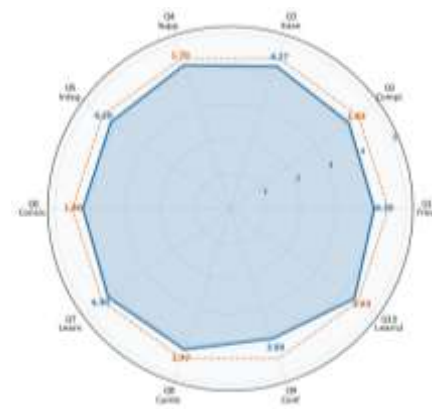


Figure 6. SUS Per-Item Radar Chart — Normalized Score Profile (N=30)

Table 6. SUS Evaluation Results Summary (N = 30)

Metric	Value	Interpretation
Number of respondents	30	—
Mean SUS score	80.33	Good / Excellent (Bangor et al., 2008)
Standard deviation	7.84	—
Minimum score	62.50	OK category
Maximum score	95.00	Best Imaginable category

Respondents scoring ≥ 80.3 (Excellent)	15 (50.0%)	Above Good-Excellent boundary
Respondents scoring 68-80.3 (Good)	11 (36.7%)	Above population median
Respondents scoring 51-68 (OK)	4 (13.3%)	Below population median
Population median benchmark	68.00	Reference: Sauro & Lewis (2011)
Score above population median	+12.33 pts	Above-average usability confirmed

Table 7. Per-Item SUS Score Analysis (N = 30)

Item	SUS Statement (abbreviated)	Type	Mean (SD)	Contribution	Category
Q1	I would like to use this system frequently	Positive	4.10 (0.71)	7.75	Good
Q2	I found the system unnecessarily complex	Negative	1.83 (0.65)	8.43	Excellent
Q3	I thought the system was easy to use	Positive	4.27 (0.64)	8.18	Excellent
Q4	I would need technical support to use this	Negative	1.70 (0.60)	8.25	Excellent
Q5	The functions were well integrated	Positive	4.20 (0.61)	8.00	Good
Q6	There was too much inconsistency	Negative	1.80 (0.71)	8.00	Good
Q7	Most people would learn this quickly	Positive	4.30 (0.60)	8.25	Excellent
Q8	I found the system very cumbersome	Negative	1.77 (0.63)	8.08	Excellent
Q9	I felt very confident using the system	Positive	3.90 (0.88)	7.25	Good
Q10	I needed to learn many things first	Negative	1.63 (0.56)	8.43	Excellent

3.3. Discussion

3.3.1. Functional Correctness Analysis

The 100 percent functional pass rate across 72 total test executions (24 cases x 3 cycles) establishes that NNG Cinema's technical pipeline operates deterministically across all seven meteorological states. The scope of that claim should be stated precisely: black-box testing verifies that the implementation conforms to its own specification, that is, that a given weather code reliably produces the genre set the mapping prescribes. It says nothing about whether that prescribed genre set is the right one for the user. A pass rate of 100 percent is a statement about implementation correctness, not about recommendation quality. For a rule-based mapping run

against a stable API, such a result is expected. The average API response time of 1.3 seconds is consistent with acceptable mobile UX thresholds (under 2 seconds for primary content operations). Both error handling scenarios produced clean user experiences without crashes, satisfying the human-centered AI principle of providing informative feedback without placing burden on the user (Shneiderman, 2020).

3.3.2. Usability Analysis

The SUS score of 80.33 places NNG Cinema at the Good–Excellent boundary. This indicates a positive perception of usability and nothing further. The SUS is a unidimensional measure of perceived ease of use and does not assess usefulness, perceived relevance of the recommendations, behavioural intention, or acceptance in the sense of TAM or UTAUT, and it is not intended to be read as an endorsement of the recommendation logic. Q7 (learnability: $M = 4.30$) and Q3 (ease of use: $M = 4.27$) received the strongest positive scores, confirming the interface achieves immediate comprehensibility without prior training. Q4 and Q10 — measuring perceived need for technical support and prior learning — scored lowest on negative items ($M = 1.70$ and 1.63 respectively), indicating participants perceived the system as independently usable. The lowest positive item was Q9 (confidence: $M = 3.90$, $SD = 0.88$), plausibly attributable to the novelty of automatic weather-driven recommendation for first-time users, although with a single ten-minute session and no follow-up interview this interpretation remains a conjecture, not a finding. The radar chart (Figure 6) illustrates a strong and consistent usability profile, with Q9 as the sole dimension notably below the 4.0 threshold.

3.3.3. Theoretical Implications

The research extends CARS literature by demonstrating that contextual pre-filtering (Adomavicius & Tuzhilin, 2005) can be implemented as a fully automatic, zero-friction consumer mobile experience. The weather-mood-genre chain operationalized in NNG Cinema confirms the theoretical linkages proposed by Schwarz and Clore (1983) and Zillmann (1988) in a deployed application context. From a Human-Centered AI perspective (Shneiderman, 2020; Lu et al., 2024), NNG Cinema demonstrates that recommendation systems can achieve situational transparency and environmental responsiveness in a production mobile application.

Two qualifications apply to these theoretical claims. The artifact shows that the weather–mood–genre chain can be implemented and run end to end. It does not show that the chain holds for an individual user. Establishing that would mean measuring whether recommendations drawn from the mapping are perceived as more relevant than recommendations that ignore weather, and this study did not measure it. The transparency claim is bounded in a similar way. What the interface exposes is the detected weather state and an enumerable rule set, not a full explanation of the inference. The human-centered properties evidenced here are the ones set out in Section 2.6, chiefly reduced input burden and informative failure handling.

3.3.4. Limitations

Six limitations constrain the interpretation and generalizability of these results. Several bear directly on what the study can be said to have shown, so all six are stated in full.

The weather–preference relationship is weaker at the individual level than the aggregate literature implies. Population-level associations between meteorological conditions and affect (Zhang & Li, 2025) do not license inference about one user on one evening. Mood management theory itself permits both mood-congruent and mood-repairing selection (Zillmann, 1988). The same rainy evening may justify Drama for one user and Comedy for another. The mapping settles this ambiguity in one direction only, and the study offers no evidence that it settles it correctly.

The mapping is rule-based, static, and uniform. The seven weather-to-genre rules encode the authors’ reading of the literature, not empirically learned associations. They do not adapt to individual history, they carry no confidence weighting, and every user in a given condition receives the same genre cluster. The rule-based approach was chosen deliberately, for inspectability and for its immunity to cold-start failure. The cost is that the system cannot improve with use, and cannot represent a user whose preferences run counter to the rule.

The design carries a risk of recommendation bias. In a locality with a persistent climate, such as an extended monsoon, the mapping would narrow the catalogue to the same four genres day after day. That produces a weather-induced filter bubble and under-exposes films outside the mapped clusters. Genres absent from all seven rules cannot be reached through the contextual path at all, and surface only by explicit search. The result is a loss of catalogue diversity that this study did not measure. Diversity, coverage, and novelty metrics should be reported alongside relevance in any later evaluation.

The evaluation measured functional correctness and perceived usability only. Recommendation accuracy, perceived relevance, satisfaction, and behavioural outcomes such as selection rate or completion rate were not measured. No comparative baseline against collaborative-filtering or content-based recommendation was tested. Nothing in these results supports a claim that weather-aware recommendation performs better than a conventional approach.

The participant sample is small ($N = 30$), purposive, and drawn entirely from one institution in one city. Its age distribution is bimodal, not continuous: 25 undergraduates aged 19–24 and five campus staff aged 31–38, with the 25–30 band empty. The sample is better described as two narrow groups than as coverage of the 18–40 bracket, and 83.3% of it consists of students. All participants lived in Makassar, a single tropical climate zone. The snow and cold-temperature pathways of the mapping (Table 2) were never exercised by a real user, and were verified only in functional testing. Generalization to older users, to users unfamiliar with streaming interfaces, and to temperate or seasonal climates is not warranted.

The evaluation was a single ten-minute session. Each participant experienced essentially one weather state, and none saw a transition between states. The study reports first-impression usability, not sustained use. The property most specific to this system, its responsiveness to changing conditions over time, remains untested at the level of user experience.

These constraints mean that the contribution should be read as a demonstration of design feasibility and positive initial usability for automatic weather-contextualized mobile recommendation, not as evidence of recommendation effectiveness.

4. CONCLUSION

This study developed and evaluated NNG Cinema, a weather-aware mobile movie recommendation system integrating real-time meteorological data with TMDB movie discovery to deliver fully automated, context-sensitive recommendations grounded in mood management theory, environmental psychology, and human-centered AI design principles. Functional evaluation confirmed a 100% pass rate across 24 test cases and 3 cycles, with average API response times of 1.3 seconds, indicating that the implementation conforms to its specification. SUS evaluation (N=30) produced a mean score of 80.33 (SD=7.84), placing NNG Cinema in the Good-to-Excellent usability range and 12.33 points above the established population median of 68.0.

These outcomes are best characterized as evidence of technical feasibility and of positive initial usability. The study shows that weather sensing, genre mapping, and catalogue retrieval can be chained inside a single mobile client without user input, and that users find the resulting interface easy to learn and to operate. It does not show that the resulting recommendations are relevant, satisfying, or better than those a conventional recommender would produce, because neither recommendation quality nor a comparative baseline was measured, and because the SUS assesses perceived usability, not acceptance or usefulness. Further validation is required before the approach can be advanced as an improvement over existing practice: specifically, relevance and satisfaction measures collected alongside usability, a controlled comparison against collaborative-filtering and content-based baselines, longitudinal deployment spanning multiple weather states per user, and diversity and coverage metrics capable of detecting the filter-bubble risk identified in Section 3.3.4.

Future research should: (1) incorporate machine learning adaptation to personalize genre probabilities per weather state; (2) expand contextual signals to include time of day, social context, and device usage patterns; (3) conduct comparative evaluation against collaborative filtering baselines; (4) extend the participant demographic beyond the 18–40 age bracket, beyond a single institution, and beyond a single tropical climate zone so that the snow and cold-temperature pathways can be exercised by real users; and (5) complement the SUS

with an acceptance model such as TAM or UTAUT, together with relevance, satisfaction, and diversity measures, so that usefulness and behavioural intention are measured directly.

5. REFERENCES

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), 734–749. <https://doi.org/10.1109/TKDE.2005.99>
- Adomavicius, G., Mobasher, B., Ricci, F., & Tuzhilin, A. (2011). Context-aware recommender systems. *AI Magazine*, 32(3), 67–80. <https://doi.org/10.1609/aimag.v32i3.2364>
- Aung, S. T., Funabiki, N., Aung, L. H., Kinari, S. A., Mentari, M., & Wai, K. H. (2024). A study of learning environment for initiating Flutter app development using Docker. *Information*, 15(4), 191. <https://doi.org/10.3390/info15040191>
- Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, 24(6), 574–594. <https://doi.org/10.1080/10447310802205776>
- Brooke, J. (1996). SUS: A quick and dirty usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland (Eds.), *Usability Evaluation in Industry* (pp. 189–194). Taylor & Francis.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331–370. <https://doi.org/10.1023/A:1021240730564>
- Casillo, M., Colace, F., Conte, D., Lombardi, M., Santaniello, D., & Valentino, C. (2026). A context-aware recommender system-based framework for improving cultural experiences. *User Modeling and User-Adapted Interaction*, 36, Article 1. <https://doi.org/10.1007/s11257-025-09437-1>
- Cohen-Kalaf, M., Lanir, J., Bak, P., & Mokryn, O. (2022). Movie emotion map: An interactive tool for exploring movies according to their emotional signature. *Multimedia Tools and Applications*, 81(11), 14663–14684. <https://doi.org/10.1007/s11042-021-10803-5>
- Duncan, B. A., Kallumadi, S., Berg-Kirkpatrick, T., & McAuley, J. (2024). MAWI Rec: Leveraging severe weather data in recommendation. *Proceedings of the 18th ACM Conference on Recommender Systems (RecSys 2024)*, 850–854. <https://doi.org/10.1145/3640457.3688157>
- Escandon, R. (2020). The film industry made a record-breaking \$100 billion last year. *Forbes*. <https://www.forbes.com/sites/rosaescandon/2020/03/12/the-film-industry-made-a-record-breaking-100-billion-last-year/>

- Gomez-Uribe, C. A., & Hunt, N. (2015). The Netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems*, 6(4), 1–19. <https://doi.org/10.1145/2843948>
- Lu, Y., Yang, Y., Zhao, Q., Zhang, C., & Li, T. J.-J. (2024). AI assistance for UX: A literature review through human-centered AI. *arXiv:2402.06089*. <https://arxiv.org/abs/2402.06089>
- Mateos, P., & Bellogín, A. (2025). A systematic literature review of recent advances on context-aware recommender systems. *Artificial Intelligence Review*, 58(1), Article 20. <https://doi.org/10.1007/s10462-024-10939-4>
- Piçarra, N., Reis, E., Chambel, T., & Arriaga, P. (2022). Searching, navigating, and recommending movies through emotions: A scoping review. *Human Behavior and Emerging Technologies*, 2022, Article 7831013. <https://doi.org/10.1155/2022/7831013>
- Sauro, J., & Lewis, J. R. (2011). When designing usability questionnaires, does it hurt to be positive? *Proceedings of CHI 2011*, 2215–2224. <https://doi.org/10.1145/1978942.1979266>
- Schwarz, N., & Clore, G. L. (1983). Mood, misattribution, and judgments of well-being. *Journal of Personality and Social Psychology*, 45(3), 513–523. <https://doi.org/10.1037/0022-3514.45.3.513>
- Shneiderman, B. (2020). Human-centered AI. *Issues in Science and Technology*, 37(1), 56–61. <https://issues.org/human-centered-ai/>
- Stack Overflow. (2023). Developer Survey 2023. <https://survey.stackoverflow.co/2023/>
- Zhang, W., & Li, W. (2025). How weather conditions affect well-being: An explanation from the perspective of environmental psychology. *Frontiers in Public Health*, 13, 1553315. <https://doi.org/10.3389/fpubh.2025.1553315>
- Zillmann, D. (1988). Mood management through communication choices. *American Behavioral Scientist*, 31(3), 327–340. <https://doi.org/10.1177/000276488031003005>