Taxonomy of cooperative adaptation level for cooperative adaptive mobile applications

Berhanyikun Amanuel Gebreselassie[1], Nuno M. Garcia[2], Dida Midekso[1]
[1]IT Doctoral Program, School of Information Science, College of Natural and Computational Sciences, Addis Ababa University, Addis Ababa, Ethiopia
[2]Institute of Biophysics and Biomedical Engineering (IBEB), Faculty of Sciences, University of Lisbon, Lisbon, Portugal

Article Info

Article history:

Received Sep 9, 2025
Revised Jan 17, 2026
Accepted Jan 30, 2026

Keywords:

Adaptive
Adaptive mobile application
Cooperation level
Mobile application
Taxonomical framework

ABSTRACT

Adaptive mobile applications (AMAs) are software systems designed to dynamically adjust their behavior in response to contextual changes. When multiple AMAs coexist on the same device, they create an ecosystem of heterogeneous applications with distinct functionalities, interaction models, and sensor requirements. This diversity enables opportunities for cooperative adaptation, where applications synchronize their behavior for collective benefit. Building on prior work that identified cooperation as a key dimension of adaptive mobile systems, this study proposes a refined taxonomy of cooperation levels for AMAs. The taxonomy is validated through case studies and formal specification methods, demonstrating its theoretical soundness and practical applicability. The findings advance the understanding of cooperative adaptation mechanisms and provide structured guidance for designing and classifying cooperative AMAs.

Corresponding Author:

Berhanyikun Amanuel Gebreselassie
IT Doctoral Program, School of Information Science
College of Natural and Computational Sciences, Addis Ababa University
Addis Ababa, Ethiopia
Email: berhanyihun@gmail.com

1. INTRODUCTION

Mobile applications are software programs designed to run on mobile devices such as smartphones and tablets, providing users with various services ranging from entertainment and social networking to productivity and health monitoring [1], [2]. Mobile computing is distinctively affected by continuously changing resource availability and user preferences. This property introduces a requirement for mobile applications to be more adaptive to changes, which were not normally present in the relatively static desktop world [3]. Self-adaptive software modifies its own behavior in response to changes in its operating environment. The operating environment can be an end-user input, external hardware devices and sensors or program instrumentation [4]. The ability to adapt autonomously and context awareness has informed many mobile applications.

An adaptive mobile application (AMA) exhibits self-modifying behavioral characteristics. It does so in response to dynamic changes in its operational environment. The operational contexts include user profile parameters, end-user input modalities, hardware configurations, and sensory data streams. The context information originate from the mobile device and external systems [5], [6].

There are numerous works, e.g., [7]–[11] dealing with adaptive systems in the context of mobile applications. A number of systematic reviews [5], [6] also characterize dimensions related to AMA. Moreover, there are studies [12]–[15] that highlight the importance of coordination among mulitiple AMA. However, there is a scarcity of published literature that incorporates the cooperativeness dimension to characterize adaptive mobile apps. Our previous work [16] on cooperative AMA identified: structure of monitor–analyze–plan–execute over a shared knowledge base (MAPE-K), domain, goals, context, triggers, aspects, coordination and level of cooperation as the major dimensions that characterize the phenomena. Level of cooperation defines the depth and nature of interaction between participating applications and has cascading effects on various aspects of their collaborative behavior.

Classification schemes like taxonomies provide four key benefits for researchers and practitioners: they facilitate knowledge sharing, help identify research gaps, and clarify interrelationships between factors within a knowledge domain [17]. Additionally, these structured frameworks support informed decision-making processes by providing systematic ways to organize and compare relevant information [18]. This study aims to characterize the level of cooperative dimension in more detail to provide a validated framework to guide the development of cooperative AMA.

The framework put forward in this study, presents a more detailed description of the discovered coarse range of cooperative adaptation levels in [16]. It will refine the concepts, relationships and constraints embedded in the levels of cooperation in formal manner, providing a guiding principles to follow for designers and developers of cooperative AMA based on the constraints of the levels. The proposed framework will be assessed through the application of a hypothetical use case scenario that demonstrates cooperative adaptation mechanisms.

This paper is organized as follows: section 2 establishes the theoretical foundations and conceptual framework underlying this research. Section 3 provides a comprehensive analysis of cooperation levels and examines their implications for cooperative adaptation mechanisms. Section 4 presents a validation scenario that demonstrates the applicability of the proposed framework through detailed case study analysis. The empirical findings and evaluation results are systematically analyzed and discussed in section 5. Finally, section 6 concludes the paper with a synthesis of key findings and identifies directions for future research endeavors.

2. THEORETICAL BACKGROUND

Self-adaptability enables a system to adapt itself to changes in its execution conditions and user requirements in order to achieve particular quality goals [6]. An AMA is a mobile application that has the capacity to modify its structure and behavior at run time in response to changes in operational context without major involvement from a user or human agent [6]. The operational context should be a well-defined set of internal and external conditions that affect the defined performance of the mobile application. The context considered can be user activity, device resources, environment and the internal state of the mobile application [5], [16] An AMA uses the context and employs its adaptive mechanisms to implement an adaptive output aligned with adaptation goals. Adaptive mechanisms are the computational processes, algorithms, or decision-making frameworks within an AMA that enable it to dynamically modify its behavior in response to changes in context [6]. While, adaptive daptive output refers to the specific, observable behaviors, actions, or modifications produced by an AMA as a result of its adaptive mechanisms, tailored to the current user state and context to enhance functionality or user experience [5]. These outputs are the tangible results of the adaptation process, affecting the app's interaction with the user or device resources.

An AMA ecosystem contains one or more AMAs that share a computing platform comprising a common hardware, operating system, sensors and external computing elements serving a single user. They are usually deployed in the same mobile phone device. The ecosystem comprises of AMAs that are aware of the presence of at least one other AMA. This predicates that the AMAs have the possibility to share context and operational context.

Cooperative self-AMA within an AMA ecosystem are defined as a collection of two or more AMAs that demonstrate mutual awareness of co-existing AMA and exhibit coordinated collaborative behaviors to optimize collective system performance. These applications achieve enhanced effectiveness through the consideration and integration of adaptive actions performed by other applications within the ecosystem, rather than relying solely on individual, isolated adaptive responses.

Two theories primarily inform our proposed framework of cooperation level for AMA. The first is coordination theory that provides a lens to examine how different systems manage interdependencies. Malone defines coordination theory as "a body of principles about how activities can be coordinated, that is, about how actors can work together" [19]. Coordination theory identifies goals, activities, actors and interdependencies as the major components of coordination in various disciplines. The theory further relates the identified components of coordination processes, namely; managing shared resources, managing task decomposability, producer-consumer relationships that are integral to the coordination phenomenon.

The second theory is cooperation theory, which a broad theory is based on the tenant that individuals and groups working together with shared goals achieve better results than those that are based on competition.

There are numerous works discussing cooperative theory in various fields including management, economics, education and biology. The work in [20] denoted that cooperation resulted in more productivity. The core principles behind this cooperation theory in the field of management are interdependence, mutual benefit, shared goals, trust and communication and positive relationships [21]. The concepts of cooperation theory manifest in this study through:

– Actors: these are individual AMAs.
– Interactions: there are different possibilities in which individual AMA can interact with each other. It can be a direct peer to peer interaction, an interaction facilitated by an intermediary or a hybrid mechanism that allows for direct and managed interactions.

This study draws upon concepts from multi-agent systems (MAS) within the field of computing. MAS are based on automomous agents that are capable of perceiving their environment, making independent decisions, and acting to achieve goals. An MAS comprises a collection of agents that interact within a defined environment [22]. These agents facilitate information exchange via a designated communication channel, and they often cooperate to proficiently address a shared problem. The field of MAS addresses fundamental questions about how autonomous entities can coordinate, cooperate, compete, and negotiate in complex environments [23].

In management science, cooperative behavior has been extensively examined through rigorous theoretical and empirical research frameworks [24], [25]. Similarly, the field of MAS has developed substantial literature addressing how multiple autonomous agents interact and collaborate to solve complex computational problems [23], [26]. However, within the domain of computing, and particularly in the context of self-adaptive systems and AMAs, cooperative behavior remains significantly underexplored in the existing literature.

Mobile applications operate in dynamic environments. They must constantly adapt to changing networks and resources. Current research, however, focuses on individual adaptation. It overlooks cooperative strategies among competing applications. This is a critical gap on devices with shared resources. This paper introduces a validated taxonomy for cooperative adaptation. The taxonomy defines application interaction levels, from isolation to full collaboration. It provides practical guidance for developers. We validate the taxonomy through mathematical formalisms and an application of a hypothetical use case scenario that demonstrates cooperative adaptation mechanisms.

3. METHOD

Level of cooperation between AMA can be considered as fundamental determinant that shapes cooperative adaptation in mobile applications. It qualifies the intensity, scope, and sophistication of collaborative interactions among AMAs. It is predicated in the assumption that cooperation is not a mere binary concept limited to presence and absence of the act. A higher level of cooperation is related to greater information exchange and more integrated adaptive actions. Trust, user preference, security requirements and task complexity affect level of cooperation.

Our previous work in [16] introduced a cooperative adaptation continuum that delineated a range of cooperative adaptation approaches, organized by their level of cooperation as illustrated in Figure 1. This spectrum initiated with a “None” category, signifying the absence of cooperative adaptation. Progressing along this continuum, the study identified “Decentralized Adaptation” and “Abstraction of Adaptation Concepts” as subsequent stages. Further levels of complexity included “Multi-Device and Distributed Adaptation,” “Recognition of Other Adaptive Applications,” “Context Sharing and Management,” and “Conflict Resolution.” The spectrum culminated in “Global Goal,” representing the most integrated form of cooperative adaptation observed.

We have adapted and refined the cooperative adaptive continuum in Figure 1 to propose a qualitative level of cooperation to guide cooperative adaptation in mobile applications. The classification system is based on a taxonomical framework grounded in seven principles as illustrated in Figure 2.

a. Level of awareness: this indicates the degree of awareness of other AMA in an ecosystem. AMA can be completely unaware of each other, aware about functional capabilities of other AMA, aware of adaptive capabilities,
b. Autonomy: the degree of independent decision-making authority retained by individual AMA. It is considered as an important dimension in MAS that is related in many aspects to AMA [27]. A complete autonomy signifies AMA making adaptive decisions independently without consideration to the self-adaptive capabilities of other AMA. A moderate level of autonomy enables an AMA to participate in and be influenced by other AMA in the decision-making process concerning its own adaptive mechanisms. In low autonomy AMA operate under a significant external control from other AMA.

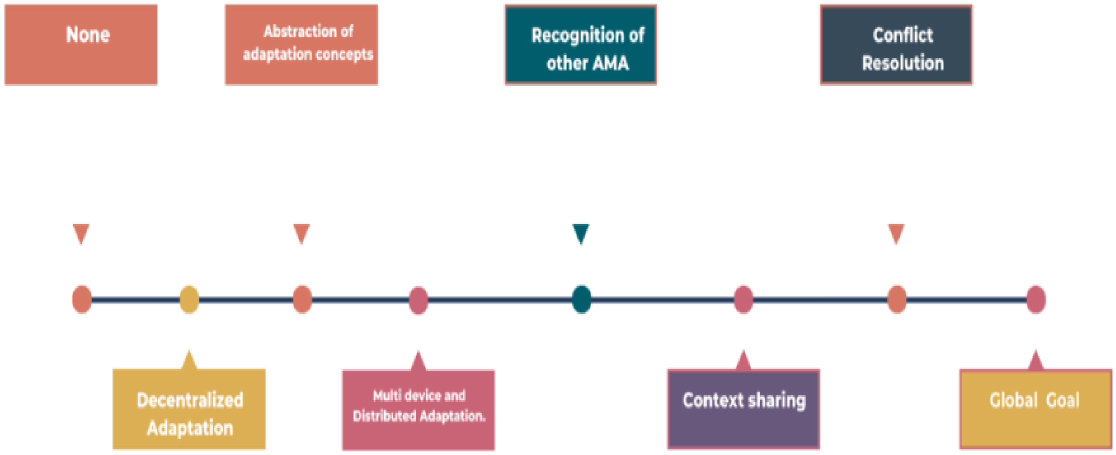


Figure 1. Cooperative adaptation continuum

Figure 2. Taxonomical framework for levels of cooperation in AMA

c. Information sharing: the extent, frequency, and nature of information exchange between AMA in the AMA ecosystem [28]. It can refer to the following instances:
   - None: no information is exchanged among AMA
   - Minimal: basic information sharing about the functional capabilities.
   - Selective: sharing of non critical contextual data.
   - Considerable: sharing of contextual data.
   - Comprehensive: extensive data sharing including the context, adaptive capabilities and mechanisms.
   - Complete: full transparency with complete knowledge followed by real time synchronization of adaptive knowledge.

d. Duration of cooperation refers to the temporal scope of an AMA cooperative behavior. This duration can be categorized into three distinct levels:
   - Short-term interactions are brief, lasting only a few seconds. These are typically for sharing basic contextual information and do not require the preservation of historical data.
   - Medium-term interactions involve cooperation specific to an adaptive action. This level necessitates the logging of interaction data for subsequent analysis and reference.

- Long-term cooperation represents a persistent relationship between AMAs. This enables continuous collaboration and facilitates the learning and evolution of shared adaptive mechanisms over an extended period.
- The temporal dimension also can specify for adhoc, event driven, periodic or continuous interactions.

e. Engagement mode: it captures how AMA initiate, respond to, and manage cooperation. Cooperations can be reactive that respond to cooperation request and environmental triggers or they can be proactive cooperation that are anticipatory and goal driven.

f. Goal alignment: this describes how the adaptive objectives of AMA relate to each other. Cooperating AMA can have:
   - Orthogonal goal alignment: AMA persue independent adaptive goals.
   - Conflicting goal alignment: one AMA's adaptive goal impedes another's goal achievement.
   - Partial goal alignment: have some overlapping adaptive goals.
   - Shared goal alignment: AMA have common aligned goals and success is measured collectively.

g. Scope: it captures the breadth, scale, and boundary characteristics of AMA interactions and coordinations. It is assumed that AMA share a common computing platform in the mobile device of a particular user. The user's profile and behaviour are also shared by the AMA. The scope of cooperation can be minimal, intermediate or extensive. In minimal cooperation AMA don't have any common adaptive goals and the majority of cooperation revolves around sharing context information. The adaptive mecahnisms of an AMA have little impact on the adaptive mechanisms of another AMA. In intermediate cooperation scope AMA have the opportunity to impact each other in a limited capacity in a selected common adaptive goal. Information about context and adaptive strategies are shared among AMA. In an extensive scope of cooperation in AMA. In an extensive scope cooperation two or more AMA are involved in a deep coordination in a selected adaptive goal or they are involved in cooperation in large number of respective adaptive goals.

The taxonomical framework systematically informed the construction of a taxonomy delineating cooperation levels within cooperative AMA. The resulting taxonomy encompasses seven distinct levels of cooperation, each characterized by specific behavioral and structural criteria. Figure 3 presents the complete taxonomical structure, while the corresponding classification criteria and theoretical foundations are elaborated in the subsequent discussion.

Figure 1 Taxonomy for cooperation levels of AMA

Level 0: independent-AMAs are unaware of each other and implement self-adaptation independently. It is characterized by the absence of communication mechanisms (both adaptive and non-adaptive), lack of context sharing capabilities, and no coordinated adaptive actions among participating applications.

Level 1: aware at functional level; at this level, AMAs recognize each other's existence and functional capabilities, but lack awareness of adaptive mechanisms, preventing cooperative adaptation.

Level 2: passive information sharing; AMAs are aware of each other's existence and may passively share some basic, non-critical contextual information (e.g., presence, general capabilities). They operate at a complete autonomy.

Level 3: context sharing; AMAs react to explicit requests or offers from peers for sharing context information. The relationships between them are usually transitory. They share selective or considerable information about each other's adaptive mechanisms. There is an opportunity for partial alignment of adaptive goals.

Level 4: collaborative learning; this involves a more committed relationship of context information about not only contextual information regarding to the environment but also adaptive mechanisms of individual AMAs. This can enable a consolidated view of context not available to each AMA at a particular time and the chance to optimize adaptive mechanisms.

Level 5: conflict resolution; this level necessitates a greater level of cooperation compared to context sharing and utilizes available information to address situations where adaptive actions from AMAs counteract each other. AMA in this level can have either active or proactive mode of engagement. Moreover, the level requires a medium to long term duration of cooperation. The adaptive action of an AMA can result in the imminent or possible undesirable context state of another AMA. It also deals with the cases where Adaptive actions of one AMA counteracts the effects of another AMA.

Level 6: complementary adaptation; AMAs engage in proactive coordinated adaptive actions. They align their adaptive actions in a manner that is beneficial to the whole ecosystem by collaboration and coordination of their adaptive mechanisms. This can involve negotiation, role assignment and synchronized actions. This level of cooperative adaptation entails the use of ontology models that provide shared, formal, and machine-interpretable representation of knowledge, allowing diverse AMAs to understand each other's context, capabilities, intentions, and tasks, which is foundational for this level of cooperation [29]. AMAs could also pool their adaptive mechanisms l (e.g., context sensing and reasoning engines) to execute more effective and efficient adaptation [30]. This includes the potential of combining models for AMAs that base their analysis and planning of adaptation on ML. The AMAs that are cooperating at complementary adaptation level should be have the ability to:

– Anticipate needs: understand and predict situations where another AMA's advantage in engaging in cooperation formulate sub-goals for the specific cooperation need.
– Negotiate: agree on adaptive task allocation and dynamically determine "who will do what?" in the collaboration.
– Synchronize actions: coordinate the timing and sequence of adaptive mechanism that guarantee cooperation.
– Maintain a shared understanding: keep consistent view in regards to the identified sub-goals making reliable updates spanning the cooperating AMAs.

Level 7: integrated goal; this is the apex of cooperation level where AMAs act as a cohesive unit to pursue an overarching goal. This involves complex planning, unitary understanding of global context. Each AMA loses significant portion of its own independent adaptive mechanism for global benefit. AMAs work together to maintain a global goal of adaptation and synergistically combine their adaptive mechanisms to optimize the global view of the overall system. A process of unified data collection and sensor fusion is the first step for an ecosystem at this level of adaptive cooperation. Explicit data, implicit data and contextual data are the major data types collected across the AMAs in this level. The collected raw data should be processed and inferred to generate higher-level attributes. The temporal and spatial aspects of all data should be established by discovering relationships at a larger scale. Advanced data processing techniques and machine learning (ML) algorithms are utilized to accomplish this objective. A dynamic model for each for each identified goal is constructed and refined during the uptime of the AMA ecosystem. The models can be stored in accessible format that enables the updating of models by AMAs. These models are used to formulate more stable global models that characterize the user's interactions with the AMAs. We can leverage the global models to propose coherent adaptation by the AMAs in the level.

Overall, the proposed taximony intends to delinate the evolutionary stages of cooperative behavior, from independent adaptation mechanisms to sophisticated consolidated goal, thereby providing a comprehensive classification scheme for understanding the spectrum of cooperative adaptation capabilities within mobile application contexts. Table 1 summarizes the major characteristics of each identified levels. To assess the theoretical validity and practical applicability of this taxonomy, section 4 evaluates its constructs through case studies of two AMAs-ethio-djibouti railway application (EDRapp) and Jirtaa employing descriptive analysis and mathematical formalizations to demonstrate how their adaptive mechanisms and outputs align with the proposed levels.

Table 1. Comparison of cooperation level in AMA

Level	Name	Awareness of other AMAs	Context sharing	Adaptive mechanism sharing	Conflict resolution	Goal alignment	Duration of relationship
0	Independent	None	None	None	None	None	None
1	Aware at functional level	Presence and basic functions only	None	None	None	None	None
2	Passive information sharing	Presence + general capabilities	Basic, non-critical, passive	None	None	None	None
3	Context sharing	Full presence + adaptive mechanisms	Selective or substantial sharing	Explicit request/offer	Partial disclosure of adaptation logic	None	Short term
4	Collaborative learning	Full + deep insight into adaptation logic	Environment + adaptive mechanisms	Committed, ongoing sharing	None	Indirect (better individual models)	Medium-term
5	Conflict resolution	Full + impact awareness	Full relevant context	Full disclosure where needed	Active detection and resolution (reactive or proactive)	Local goals preserved, conflicts avoided	Medium- to long-term
6	Complementary adaptation	Full + intentions and capabilities	Full context + pooled mechanisms (sensing, ML models, reasoning)	Full disclosure + pooling	Proactive prevention + resolution	Strong alignment of sub-goals for ecosystem benefit	Long-term, dynamic
7	Integrated goal	Complete unity (act as single entity)	Unified sensor fusion and global context	Complete merging of adaptation logic	Systemic prevention	Single overarching global goal	Permanent during ecosystem lifetime

4. RESULTS AND DISCUSSION

This section presents a preliminary evaluation of the proposed taxonomical framework through the application of case study descriptions and mathematical formalizations. The evaluation methodology employs both descriptive case analysis and formal mathematical representations to assess the taxonomy's theoretical validity and practical applicability within the domain of AMA.

This preliminary evaluation examines two hypothetical AMA that demonstrate distinct adaptive mechanisms within transportation and educational domains. The EDRapp is an AMA that leverages multi-dimensional context monitoring to enhance the passenger experience. It tracks temporal, spatial, ambient, user profile, and device-specific data (e.g., connectivity and power) to dynamically adapt services throughout the travel journey. EDRapp intelligently manages resources by adjusting the user interface (UI) based on battery levels and network quality, and it proactively downloads data in anticipation of network blackouts. The application also provides personalized features, such as seat recommendations based on user states like fatigue and noise preferences.

The second AMA, Jirtaa, is a pedagogical, self-adaptive mobile system designed to teach the Afan Oromo language widely spoken in Ethiopia. The system uses a mobile device's sensors to provide contextually relevant educational content. For example, it can generate location-specific lessons, such as vocabulary related to water bodies when the user is near a lake. Jirtaa's interface also adapts its modalities based on the user's activity, environmental conditions, and the device's connectivity status, ensuring a personalized and effective learning experience.

Both applications are used to demonstrate the taxonomical framework's applicability across diverse domains while illustrating varying degrees of context sensitivity, adaptive granularity, and external service integration capabilities. We used mathematical formalism in tandem with the use cases of cooperations between the EDRapp and Jirtaa to evaluate the taxonomy for level of cooperation between different AMA.

The subsequent formalism employed for evaluation purposes is predicated upon the following foundational notations. Let:

$$A = A_1, A_2 \dots, A_n \quad (1)$$

Be a finite set of $n \geq 2$ mobile applications deployed in a shared computing infrastructure utilized by a single user.

Let $C(t)$ represent the shared contextual state vector at time t under a common schema of context information described for A. Since, the applications reside on the same device, they have access to a common set of context parameters. This vector comprises a set of measurable environmental and internal conditions, such that:

$$C(t) = (c_1(t), c_2(t), \ldots, c_m(t)) \quad (2)$$

is driven from [31] where $c_j(t)$ is the value of the j-th context parameter at time t, and m is the total number of context parameters relevant to the applications in A.

Let $C_i(t)$ be the context relevant to application A_i. We define,

$$C_i(t) \subseteq C(t) \quad (3)$$

meaning that each application uses a subset of the available context.

Let $S_i(t)$ represent the internal state of application A_i at time t, encompassing its configuration, resource allocation, and operational parameters. Let:

$$F_i: C_i(t) \times S_i(t) \times M_i(t) \rightarrow \Delta S_i(t+1) \quad (4)$$

be the adaptation function of application A_i.
Where: $M_i(t)$ represents the set of messages received by application A_i from other applications in A at time i, adapted from similar concepts in MAS [32] where:

$$M_i(t) = M_{ij}(t) | A_j \in A, j \neq i \quad (5)$$

each message $M_{ij}(t)$ contains information relevant to adaptation, such as the contextual state, internal state, and/or adaptation actions of application A_j.

This function F_i maps the relevant contextual state, the current internal state, and received messages to a change in the internal state $S_i(t+1)$, leading to a new internal state:

$$S_i(t+1) = S_i(t) + \Delta S_i(t+1) \quad (6)$$

Level 0: EDRapp and Jirtaa deployed in the mobile phone of the user without any awareness of each others operative and adaptive mechanism.

Level 1: EDRapp and Jirtaa deployed in the mobile phone of the user where there is awareness of the services provided by each other. We can copy text values from EDRapp to Jirtaa and vice versa based on the permission granted by the user.

Level 2: EDRapp and Jirtaa deployed in the mobile phone of the user and there is broadcast of presence messages; discovery protocols; read-only access to public context.

Level 3: EDRapp and Jirtaa deployed in the mobile phone of the user and sharing context information in adhoc request and response manner. EDRapp sharing the context of $User.Activity = \{“Walking”\}$ in response to Jirtaa’s request.

Level 4: EDRapp and Jirtaa deployed in the mobile phone of the user and negotiating to share context information regarding User. Activity for an extended period of time. EDRapp can collect information on some aspect of the user’s activity and Jirtaa can collect context information about user’s activity not tracked by EDRapp. A consolidated context from this long-term period data can inform both AMAs. EDRapp can also inform about the effectiveness of changing a location tracking mechanism after a change in location to Jirtaa. Enabling Jirtaa to apply a more effective location tracking method.

Level 5: Jirtaa can propose to extend the time of a battery intensive learning module depending the focus and progress of the user profile. This can deplete the battery of the mobile device restricting the use of contacting emergency services using the EDRapp. A conflict resolution strategy based on prioritization can propose for Jirtaa to suspend the use of the power intensive module.

Jirtaa proposing a new module which is high in multimedia content based on the progress of the user. The Adaptive action uses a high amount of battery. The adaptive action can be described as (focusing on the unique values for the action):

$$\alpha' = \langle JIRTA, Functional, \{Import = \text{Module F}, Source = \text{"}Server\ Address\text{"}, \{U_J(t).Progress = \text{"Level 2"}, U_J(t).Activity = \text{"Focused"}\}, \{A_J(t).batteyUsage = \text{"High"}, A_J(t).Structure = \text{"}MultiMedia1\text{"}\}, timestamp, resource\_cost \rangle \quad (7)$$

where:
- $JIRTA$ is the AMA involved in adaptive action.
- $Functional$ is a type of adaptive action that adds new functional capabilities.
- $Import =$ Module F $Source =$ "$Server\ Address$ are the parameters for the function implementing the adaptive action.
- $U_J(t).Progress$ is the user's context of Jirtaa describing his/her progress in the education application.
- $U_J(t).Activity =$ "Focused" is the user's context describing his/her activity status.
- $A_J(t).Structure =$ "$MultiMedia1$" denotes to the Application context of Jirtaa that involves a multimedia module.
- $A_J(t).batteyUsage =$ "High" describes the Application context of EDRapp that denotes the battery usage amount.

A conflict arises when the execution of α' would cause the shared device's battery level to drop below a critical threshold for the emergency module of EDRapp or if the rate of consumption jeopardizes EDRapp. If $E\_device\ (t).battery\_level$ is shared battery level of the device on which both apps might be running observed by EDRapp, and BatteryImpact(α',E_device (t)) be the predicted battery level after executing α'.

Then a conflict condition as a result of α' can be formulated [32] as:

$$Conflict(\alpha', E, t) \iff (BatteryImpact(\alpha', E_{device}(t)) < A_E\ (t).criticality\_threshold) \wedge (A_E\ (t).operational\_status = active \vee A_E\ (t).operational\_status = standby) \quad (8)$$

where:
- $Conflict(\alpha', E, t)$ denotes the conflict situation on EDRapp as a result of α'.
- $A_E\ (t).criticality\_threshold)$ is an application context for EDRapp that satisfies a battery level below which EDRapp's functionality is critically impacted.
- $A_E\ (t).operational\_status$ is an application context for E describing the running mode of EDRapp.

This means that executing α' reduces the device's battery below the EDRapp's critical threshold while the EDRapp is operational. A conflict resolution mechanism R can be considered as a function that takes the conflicting action of Jirtaa on EDRapp in the Ecosystem context and proposes a revised set of actions.

$$R(\alpha', C(t)) \to \alpha'' \quad (9)$$

α'' is a set of new actions proposed to counter conflict. $C(t)$ describes the overall context of the Ecosystem. Possible actions to mitigate context conflict can be using prioritization and dropping the request for new module by Jirtaa or loading a less energy intensive module $A_J(t).Structure =$ "MultiMedia2" whose $A_J(t).batteyUsage =$ "Medium".

Level 6: Jirtaa that is intending to use an external server anticipates that EDRapp will use the same external server, when nearing a particular location. The sub-goal for this collaboration is to decrease the cost of using External server by the user. Based on the works in the field of MAS regarding beliefs, desires and intentions [32] we can define a belief function $B_J(needs(E, S, future)) = true$ (Jirtaa believes EDRapp will need server S in the future).

The cooperation is defined by the objective:

$$min(Cost(Collaboration)) = min(P(T_{lease})) \quad (10)$$

Such that the combined needs of Jirtaa and EDRapp are met, where:
- T_{lease} be the total time (in minutes) for which the server is leased.
- P is the pricing function.
- $Cost(Collaboration)$ refers to the cost function with the impact of collaboration.

Jirtaa will negotiate with EDRapp on who will contract the leasing and for how long time the external service provider's service will be acquired. As per negotiation Jirtaa will lease the services of the server taking into consideration the needs of EDRapp. For a particular time, EDRapp will refrain from negotiating the same service. Collaboration Goal can be formulated as [33]:

$$min\left(P\left(\left\lceil\frac{t_J+t_E^{anticipated}}{T_{batch}}\right\rceil\right)\right) \quad (11)$$

where:
- t_J is Jirtaa's required service time ($t_J = 45$).
- $t_E^{anticipated}$ is the anticipated lease time of EDRapp.

Level 7: Jirtaa, EDRapp and a set of other applications can group together to develop and maintain a global model of the user, particularly his/her interaction. Let U denote the state of the user, and T is the current time: a user state vector $S_U(T)$ encapsulate the user's property at time T. This vector can combine:
- Preferences (P): a vector of learned preferences. $P = [p_1, p_2, \dots, p_m]$ where $p_j$could be a score for a content category, a preferred notification frequency, and a UI density preference.
- Context (C): a vector of inferred contextual attributes. $C = [c_1, c_2, \dots, c_k]$, where $c_i$could be current activity, e.g. $Sleeping, Driving, Focused, Walking$; location type e.g., $home, work, transit$]; and device state $battery\_low, network\_slow$.
- Goals/intentions (G): a vector representing the inferred current goal or task. $G = [g_1, g_2, \dots, g_l]$, where g_i could be the probability of engaging in a specific task (e.g., [$Learn, visit\ museum, order\ meal$]
- Cognitive load (L): A scalar or vector representing the inferred cognitive load or attentional state of the user [34] $L \in [0,1]$.

hence,

$$S_U(T) = (P, C, G, L) \quad (12)$$

defines the user state vector aligning with our previous representation of context in (2).
If $O(T) = (SensorData(T), InteractionLogs(T), Feedback(T))$ is a vector of raw observation collected at time T driven from user interaction modelin [35], feedback driven adaptation [36] and sensor based data fusion [37]. Where:
- $SensorData(T)$ captures sensor readings, e.g. GPS coordinates, accelerometer data, and light levels.
- $InteractionLogs(T)$ interaction logs with AMAs, e.g. Click, scroll, text input, and time in app.
- $Feedback(T)$ logs explicit feedback from the user U.

The model state $S_U(T)$ can be updated based on the new observations founded in dynamic system updates [38]: This yields:

$$S_U(T + \Delta T) = f(S_U(T), O(T + \Delta T), \theta) \quad (13)$$

where:
- f is a model update function, e.g., Kalman filter [38], Bayesian inference [39], and neural network [40].
- θ represent the model's learned parameters.

Considering the state of each AMA let $A = \{A_1, A_2, \dots, A_M\}$ be a set of M mobile applications in the cooperation level 7 and each with its own internal state $S_{A_j}(T)$. The adaptive output for each application A_j based on $S_U(T)$ is given by $O_{A_j}^{adapt}(T)$.

The adaptive output can be calculated by:

$$O_{A_j}^{adapt}(T) = g_j\left(S_{A_j}(T), S_U(T), \phi_j\right) \quad (14)$$

where;
- g_j is the adaptation logic function for application A_j (e.g., a recommendation engine, UI layout algorithm, notification scheduler).
- ϕ_j are A_j's specific adaptation parameters.

This ensures coherence across applications, as they all draw from the same $S_U(T)$. For example in the running case if the user is currently sleeping, all AMAs (including Jirtaa and EDRapp) fallback to a mode that uses less frequent notification. This is further illustrated in Figure 4.

Finally, we can envision $Lcoop$ be a variable representing the level of cooperation, taking values from a predefined ordered set:

$$Lcoop \in \{L0, L1, L2, L3, L4, L5, L6, L7\} \quad (15)$$

where $L0 < L1 < L2 < L3 < L4 < L5 < L6 < L7$ correspond to the qualitative levels defined above (independent, aware at functional level, context sharing, collaborative learning, conflict resolution, complementary, and integrated goal).

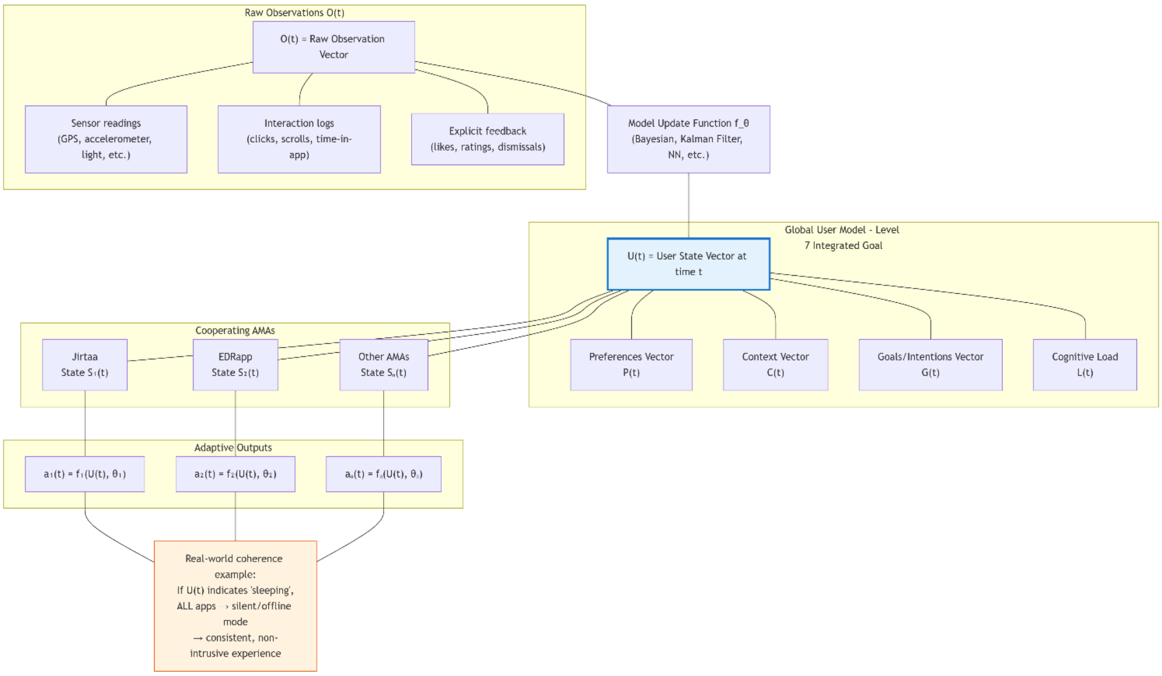


Figure 4. Major processes of Level 7 integrated goal example

5. CONCLUSION

This study employed the proposed taxonomical framework through application to a hypothetical configuration of interacting AMAs. Overall, the framework covers the full spectrum of cooperative adaptation in mobile applications from independence to integration, providing comprehensive coverage. Level 0-3, characterize situations where there is no significant cooperative adaptation among AMA. Level 1 indicates the situation between EDRapp and Jirtaa where there is only recognition of each other's functional capability with limited awareness of adaptive mechanism awareness. Level 2 cooperation between the use case AMAs is characterized with minimal cooperation engagement and passive sharing of information without active solicitation. Level 3 provides an insight on how EDRapp and Jirtaa can cooperative in active context sharing in response to explicit requests. In level 4 of cooperation EDRapp and Jirtaa can engage in committed information sharing that provides an opportunity to optimize adaptive mechanisms. On an immediate higher level, i.e, Level 5, describes management of undesirable context states caused by conflicting adaptations between the two AMAs and possible resolution mechanisms through rigorous formalism. Advanced cooperation levels are addressed in Level 6 and Level 7. The prominent behaviors of proactive coordinated adaptive actions through negotiation and role assignment is covered in the level 6 of the framework. Edrapp and Jirtaa's requirements of need anticipation, adaptive task negotiation and active synchronization is demonstrated by the level. The highest level of cooperation, Level 7, delineates the need for cohesive unit of operation between EDRApp and Jirtaa while pursuing global optimization for specific concerns. This entails global context understanding and dynamic model constructions using advanced data processing ML.

A robust taxonomical framework for software artifacts should demonstrate discriminative power, predictive validity, practical applicability, theoretical contribution and extensibility. Our assessment of the taxonomical framework with regards to the above criteria is given as follows: (i) discriminative power: the framework provides clear progression from independence to integration, making it easy to classify existing AMAs and identify improvement pathways. Each level has concrete characteristics that can be implemented and measured in real-world applications; (ii) predictive validity: the framework accommodates both simple awareness mechanisms and complex ecosystem-level coordination. However, the framework lacks explicit guidance on how AMAs transition between levels; (iii) practical applicability: the taxonomy is able to be used as a utility to model cooperation of AMA in real life scenarios helping developers; (iv) theoretical contribution:

the taxonomical framework contributes to the advance understanding of the field because we have applied it in the conceptual modeling of cooperative adaptive mobile applications (CAMAS) in our upcoming work; and (v) extensibility: the framework doesn't provide a mechanism to accommodate new levels of cooperation in AMAs without major restructuring of levels. This is due to the complex nature of cooperative adaptation and lack of stable knowledge that fosters predictions.

The taxonomical framework also has the following limitations. It doesn't address security mechanism that are important in the higher levels. There is also a lack of resilience mechanisms built in the taxonomical framework that guides graceful degradation to lower levels. It also doesn't provide for transition where the deletion of AMA and relationships that are common in the mobile application usage imact the level of cooperation in a dynamic manner. While theoretically sound, the framework would benefit from implementation guidelines and transition mechanisms. Finally, more empirical case validations could identifying edge cases that occur in cooperative adaptation in mobile context.

This study introduced a taxonomy of cooperative adaptation levels for AMAs, ranging from independent adaptation to fully integrated goal-oriented cooperation. The taxonomy offers a structured framework for classifying and designing cooperative behaviors among AMAs. Validation through case studies and formal models demonstrated its discriminative power and practical relevance. However, limitations remain, including the absence of explicit transition mechanisms between levels, insufficient treatment of security concerns, and lack of resilience strategies for failure scenarios. Future research should extend the taxonomy with transition guidelines, security integration, and empirical validation across diverse mobile application ecosystems. Despite these challenges, this work provides a foundational contribution to the systematic study of cooperative adaptation in mobile computing.

ACKNOWLEDGMENTS

The Authors would like to thank Dr Abdi Zenebe, CEO of Ethio-Djibouti Railway and Mr Amir Abdulsattar, Ethio-Djibouti Railway Director of Digital Operations for their valuable contribution.

FUNDING INFORMATION

This research received no external funding.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Berhanyikun Amanuel Gebreselassie	✓	✓		✓	✓	✓			✓	✓	✓			
Nuno M. Garcia		✓		✓	✓	✓	✓		✓	✓		✓	✓	✓
Dida Midekso	✓			✓		✓	✓			✓		✓	✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] C.-C. Chang, "Exploring mobile application customer loyalty: The moderating effect of use contexts," *Telecommunications Policy*, vol. 39, no. 8, pp. 678–690, Sep. 2015, doi: 10.1016/j.telpol.2015.07.008.

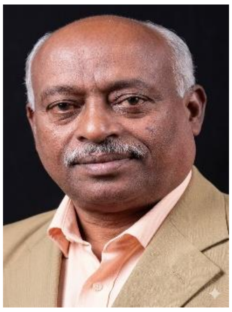
[2] A. Alhaddad, A. Andrews, and Z. Abdalla, “FSMApp: Testing mobile apps,” *Advances in Computers*, vol. 130, pp. 1–65, 2023, doi: 10.1016/bs.adcom.2022.09.001.
[3] D. M. D. Oliveira, L. Pedro, and C. Santos, “The use of mobile applications in higher education classes: a comparative pilot study of the students’ perceptions and real usage,” *Smart Learning Environments*, vol. 8, no. 1, 2021, doi: 10.1186/s40561-021-00159-6.
[4] R. Laddaga, P. Robertson, and H. Shrobe, “Introduction to self-adaptive software: Applications,” in *Self-Adaptive Software: Applications (IWSAS 2001)*, 2003, pp. 1–5, doi: 10.1007/3-540-36554-0_1.
[5] E. M. Grua, I. Malavolta, and P. Lago, “Self-adaptation in Mobile Apps: A systematic literature study,” in *2019 IEEE/ACM 14th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, May 2019, pp. 51–62, doi: 10.1109/SEAMS.2019.00016.
[6] M. Ali, S. U. R. Khan, and S. Hussain, “Self-adaptation in smartphone applications: Current state-of-the-art techniques, challenges, and future directions,” *Data & Knowledge Engineering*, vol. 136, Nov. 2021, doi: 10.1016/j.datak.2021.101929.
[7] M. Ali, S. U. R. Khan, A. Mashkoor, and A. Taskeen, “A conceptual framework for context-driven self-adaptive intelligent user interface based on Android,” *Cognition, Technology & Work*, vol. 26, no. 1, pp. 83–106, 2024, doi: 10.1007/s10111-023-00749-z.
[8] W. F. Passini and F. J. Affonso, “Developing self-adaptive service-oriented mobile applications: A framework based on dynamic deployment,” *International Journal of Software Engineering and Knowledge Engineering*, vol. 28, no. 11n12, pp. 1537–1558, Nov. 2018, doi: 10.1142/S0218194018400168.
[9] S. Dai, M. Li Wang, Z. Gao, L. Huang, X. Du, and M. Guizani, “An adaptive computation offloading mechanism for mobile health applications,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 998–1007, Jan. 2020, doi: 10.1109/TVT.2019.2954887.
[10] D. Liu *et al.*, “HiNextApp: A context-aware and adaptive framework for app prediction in mobile systems,” *Sustainable Computing: Informatics and Systems*, vol. 22, pp. 219–229, Jun. 2019, doi: 10.1016/j.suscom.2018.08.009.
[11] X. Chen, S. Chen, Y. Ma, B. Liu, Y. Zhang, and G. Huang, “An adaptive offloading framework for Android applications in mobile edge computing,” *Science China Information Sciences*, vol. 62, no. 8, Aug. 2019, doi: 10.1007/s11432-018-9749-8.
[12] J. M. Jorquera Valero *et al.*, “Improving the security and QoE in Mobile devices through an intelligent and adaptive continuous authentication system,” *Sensors*, vol. 18, no. 11, Nov. 2018, doi: 10.3390/s18113769.
[13] G. L. Scoccia, M. Autili, and P. Inverardi, “A self-configuring and adaptive privacy-aware permission system for Android apps,” in *2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, Aug. 2020, pp. 38–47, doi: 10.1109/ACSOS49614.2020.00024.
[14] B. Ren, C. Liu, B. Cheng, S. Hong, S. Zhao, and J. Chen, “EasyGuard: enhanced context-aware adaptive access control system for android platform,” in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, Oct. 2016, pp. 458–459, doi: 10.1145/2973750.2985276.
[15] S. Raheel, “Improving the user experience using an intelligent adaptive user interface in mobile applications,” in *2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, Nov. 2016, pp. 64–68, doi: 10.1109/IMCET.2016.7777428.
[16] B. A. Gebreselassie, N. M. Garcia, and D. Midekso, “A systematic review of self- adaptive mobile applications with cooperative dimension,” *IoT*, vol. 7, no. 1, Dec. 2025, doi: 10.3390/iot7010006.
[17] I. Vessey, V. Ramesh, and R. L. Glass, “A unified classification system for research in the computing disciplines,” *Information and Software Technology*, vol. 47, no. 4, pp. 245–255, Mar. 2005, doi: 10.1016/j.infsof.2004.08.006.
[18] S. Vegas, N. Juristo, and V. R. Basili, “Maturing software engineering knowledge through classifications: A case study on unit testing techniques,” *IEEE Transactions on Software Engineering*, vol. 35, no. 4, pp. 551–565, Jul. 2009, doi: 10.1109/TSE.2009.13.
[19] T. W. Malone, *What is coordination theory?* National Science Foundation Coordination Theory Workshop Massachusetts Institute of Technology Cambridge, Massachusettsa, 1988.
[20] M. Deutsch, “A theory of co-operation and competition,” *Human Relations*, vol. 2, no. 2, pp. 129–152, Apr. 1949, doi: 10.1177/001872674900200204.
[21] D. Tjosvold, “Cooperation theory and organizations,” *Human Relations*, vol. 37, no. 9, pp. 743–767, Sep. 1984, doi: 10.1177/001872678403700903.
[22] Y. Li and C. Tan, “A survey of the consensus for multi-agent systems,” *Systems Science & Control Engineering*, vol. 7, no. 1, pp. 468–482, Jan. 2019, doi: 10.1080/21642583.2019.1695689.
[23] D. Maldonado, E. Cruz, J. Abad Torres, P. J. Cruz, and S. del P. Gamboa Benitez, “Multi-agent systems: A survey about its components, framework and workflow,” *IEEE Access*, vol. 12, pp. 80950–80975, 2024, doi: 10.1109/ACCESS.2024.3409051.
[24] K. Sanders, “Cooperative behaviours in organizations,” in *Friends and Enemies in Organizations*, London: Palgrave Macmillan UK, 2009, pp. 101–121, doi: 10.1057/9780230248359_6.
[25] X. Castañer and N. Oliveira, “Collaboration, coordination, and cooperation among organizations: Establishing the distinctive meanings of these terms through a systematic literature review,” *Journal of Management*, vol. 46, no. 6, pp. 965–1001, Jul. 2020, doi: 10.1177/0149206320901565.
[26] Z. Ning and L. Xie, “A survey on multi-agent reinforcement learning and its application,” *Journal of Automation and Intelligence*, vol. 3, no. 2, pp. 73–91, Jun. 2024, doi: 10.1016/j.jai.2024.02.003.
[27] M. Wooldridge, *An introduction to multi-agent systems.* Wiley & Sons, 2009.
[28] D. Weyns, M. U. Iftikhar, S. Malek, and J. Andersson, “Claims and supporting evidence for self-adaptive systems: A literature study,” in *2012 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, Jun. 2012, pp. 89–98, doi: 10.1109/SEAMS.2012.6224395.
[29] A. W. Crapo, A. V. Aragones, J. E. Price, and A. Varma, “Using ontology to drive an adaptive learning interface,” *Systemics, cybernetics and informatics*, vol. 2, no. 5, pp. 44–50, 2025.
[30] D. Weyns, S. Malek, and J. Andersson, “On decentralized self-adaptation,” in *Proceedings of the 2010 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, May 2010, pp. 84–93, doi: 10.1145/1808984.1808994.
[31] A. K. Dey, “Understanding and using context,” *Personal and Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7, Feb. 2001, doi: 10.1007/s007790170019.
[32] M. Wooldridge and N. R. Jennings, “Intelligent agents: Theory and practice,” *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, Jun. 1995, doi: 10.1017/S0269888900008122.
[33] M. Armbrust *et al.*, “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010, doi: 10.1145/1721654.1721672.
[34] B. P. Bailey and J. A. Konstan, “On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state,” *Computers in Human Behavior*, vol. 22, no. 4, pp. 685–708, Jul. 2006, doi: 10.1016/j.chb.2005.12.009.

[35] G. Fischer, "User modeling in human–computer interaction," *User Modeling and User-Adapted Interaction*, vol. 11, no. 1–2, pp. 65–86, Mar. 2001, doi: 10.1023/A:1011145532042.
[36] E. J. Horvitz, J. S. Breese, D. Heckerman, D. Hovel, and K. Rommelse, "The lumiere project: Bayesian user modeling for inferring the goals and needs of software users," *Prepr. arXiv.1301.7385*, Jan. 2013, [Online]. Available: http://arxiv.org/abs/1301.7385
[37] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997, doi: 10.1109/5.554205.
[38] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1960, doi: 10.1115/1.3662552.
[39] Z. Ghahramani, "Learning dynamic bayesian networks," in *Adaptive Processing of Sequences and Data Structures*, 1998, pp. 168–197, doi: 10.1007/BFb0053999.
[40] I. Goodfellow *et al.*, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020, doi: 10.1145/3422622.

BIOGRAPHIES OF AUTHORS

Berhanyikun Amanuel Gebreselassie holds an M.Sc. in Information Science from Addis Ababa University, where he is currently pursuing his Ph.D. in Software Engineering. With over 15 years of experience in higher education, his research interests encompass adaptive systems, software architecture, information systems for developing countries, and data analytics. Notably, he served as the lead researcher for the inaugural Ethiopian National Poll. He can be contacted at email: berhanyihun@gmail.com.

Nuno M. Garcia holds a Ph.D. in Computer Science Engineering from the University of Beira Interior (UBI, Covilhã, Portugal) (2008) and he is a 5-year B.Sc. in Mathematics/Informatics also from UBI (1999-2004). He is Assistant Professor at the Faculty of Engineering Computer Science Department at UBI and Invited Associate Professor at the School of Communication, Architecture, Arts and Information Technologies of the Universidade Lusófona de Humanidades e Tecnologias (Lisbon, Portugal). He was founder and is coordinator of the Assisted Living Computing and Telecommunications Laboratory (ALLab), a research group within the Instituto de Telecomunicações at UBI. He was also co-founder and is Coordinator of the Executive Council of the BSAFE LAB – Law Enforcement, Justice and Public Safety Research and Technology Transfer Laboratory, a multidisciplinar research laboratory in UBI. He is also Chair of the COST Action IC1303 AAPELE – Architectures, Algorithms and Platforms for Enhanced Living Environments (Brussels, Belgium). He is the main author of several international, European and Portuguese patents. He is member of the Non-Commercial Users Constituency, a group within GNSO in ICANN. He is also member of ACM SIGBio, ISOC and IEEE. His main interests include next-generation networks, algorithms for bio-signal processing, distributed and cooperative protocols. He can be contacted at email: nmgarcia@fc.ul.pt.



Dida Midekso is Associate Professor of Computer Science and Software Engineering in Addis Ababa University (AAU). Coordinator of the Software Engineering track in the IT Ph.D. program. He served as head of the Mathematics Department who also oversees the CS undergraduate program at the college on Natural sciences. He specialized in computer-aided learning. His research interests include software engineering, software project management, and pervasive information systems. He can be contacted at email: dida.midekso@aau.edu.et.