



## Semarak International Journal of STEM Education

Journal homepage:  
<https://semarakilmu.my/index.php/sijste/index>  
ISSN: 3030-5152



# Design and Pilot Evaluation of a Mobile-First Gamified Android Application for Introductory Programming Education

Mohamad Izzat Azmi<sup>1</sup>, Suhaila Khalip<sup>1</sup>, Ahgalya Subbiah<sup>1,\*</sup>

<sup>1</sup> Department of Information Technology, Faculty of Information Sciences and Engineering, Management & Science University, Selangor, Malaysia

### ARTICLE INFO

#### Article history:

Received 3 September 2026  
Received in revised form 15 September 2026  
Accepted 17 September 2026  
Available online 21 September 2026

#### Keywords:

Android learning application;  
gamification; mobile learning;  
programming education; self-regulated  
learning; system usability

### ABSTRACT

Introductory programming can be difficult for novice learners when instruction depends on desktop-based tools, multiple software installations, and fragmented learning resources. This study designed and pilot-evaluated a mobile-first, gamified Android application that integrates introductory Java, C++, and Python learning in a lightweight, offline-capable environment. The application was developed iteratively using Agile methods, with Cognitive Load Theory and Self-Regulated Learning used as design principles rather than tested constructs. A four-week quantitative pilot involved 51 higher-education learners with predominantly no or limited programming experience. A structured questionnaire adapted from Technology Acceptance Model and System Usability Scale concepts was analysed using descriptive statistics and internal-consistency reliability. Results showed that 67% of participants primarily used smartphones or tablets for learning, 72% preferred interactive and visual learning, and 69% reported difficulty with traditional coding approaches. Interest in mobile-based coding was high (90% interested or very interested); 76% agreed that gamification increased motivation, 74% preferred curriculum-based learning, 71% preferred game-based activities over traditional quizzes, and 82% rated the system useful or very useful. Reported Cronbach's alpha values were 0.81 for usability and 0.84 for user satisfaction. The findings support the feasibility and perceived acceptability of a mobile-first programming-learning approach while avoiding causal claims that cannot be supported by the available pilot data.

## 1. Introduction

Programming and computational literacy are increasingly important across engineering, computing, and other higher-education disciplines. However, novice learners often encounter substantial entry barriers when early programming instruction depends on complex desktop environments, multiple installations, syntax-heavy interfaces, and fragmented learning resources. Cognitive Load Theory explains why poorly structured problem-solving environments can consume working-memory capacity that would otherwise support schema acquisition [1]. At the same time,

\* Corresponding author.

E-mail address: [ahgalya\\_subbiah@msu.edu.my](mailto:ahgalya_subbiah@msu.edu.my)

Self-Regulated Learning emphasizes the value of goal setting, self-monitoring, feedback, and learner control in developing independent learning behaviours [2].

Mobile learning offers one practical route to reduce access friction because personal devices can extend learning beyond formal laboratories and fixed schedules. A meta-analysis by Sung *et al.*, [7] found an overall positive effect of mobile-device integration on learning performance, while Bunjaku *et al.* [9] emphasized that digital tools in engineering education can support engagement, inclusivity, and the connection between theory and practice. Recent reviews also show that technology-enhanced learning can improve flexibility and access, although infrastructure, digital-divide, and implementation barriers remain important [12,13]. Therefore, the value of a mobile platform depends not only on portability but also on how well the interface, learning sequence, and feedback are designed.

Usability and perceived usefulness are especially important in technology-supported learning. Davis [3] identified perceived usefulness and perceived ease of use as central determinants of technology acceptance, while the System Usability Scale introduced by Brooke [4] provides a practical framework for evaluating perceived usability. Engagement is similarly important because learner-to-content interaction and purposeful learning activities are associated with stronger online-learning experiences [8]. However, gamification should not be treated as automatically beneficial. Longitudinal evidence shows that reward-heavy gamification can, in some settings, reduce intrinsic motivation or satisfaction over time [6]. Digital game-based learning research also highlights the importance of sustained user behaviour and retention rather than initial enthusiasm alone [10].

The present study addresses a practical gap in introductory programming education by integrating mobile-first access, multi-language learning content, offline availability, progress tracking, quizzes, and gamified progression within a single Android application. The study does not attempt to establish causal relationships among psychological constructs; instead, it evaluates whether the proposed design is feasible and positively received during an initial pilot. The objective is to design, develop, and pilot-evaluate an Android application supporting introductory Java, C++, and Python learning for higher-education learners with predominantly no or limited prior programming experience. The study contributes a technically lightweight design and a statistically defensible descriptive evaluation that can guide larger, controlled validation studies.

## 2. Methodology

A quantitative descriptive pilot design was combined with iterative Agile software development. Cognitive Load Theory and Self-Regulated Learning informed the design of the learning environment, particularly the use of short, structured lessons, visible progress, immediate feedback, and self-paced progression [1,2]. These theories were used as design principles; cognitive load and self-regulation were not directly measured as latent constructs in the present pilot.

### 2.1 Application Design and Development

The application was developed in Java using Android Studio. The design consolidated introductory Java, C++, and Python learning resources into a single mobile interface and used a local SQLite database for lesson storage, progress tracking, and offline access. Development followed iterative Agile cycles involving requirements review, interface prototyping, implementation, functional testing, and refinement. The design emphasized simplicity for novice learners and reduced reliance on institutional computers or continuous internet connectivity.

**Table 1**  
Functional components of the proposed application

| Component                        | Implementation                                         | Educational purpose                                                                         |
|----------------------------------|--------------------------------------------------------|---------------------------------------------------------------------------------------------|
| Mobile-first Android interface   | Java / Android Studio                                  | Provides a consistent learning environment on personal smartphones and tablets              |
| Multi-language learning content  | Java, C++, and Python modules                          | Allows beginners to access several introductory programming pathways within one application |
| Offline lesson access            | Local SQLite database                                  | Supports learning when continuous Wi-Fi or mobile data are unavailable                      |
| Progress tracking                | Local learner-progress records and progress indicators | Makes advancement visible and supports self-monitoring                                      |
| Interactive quizzes and feedback | Short activities embedded within lessons               | Provides reinforcement and immediate formative feedback                                     |
| Gamified progression             | Structured pathways and micro-rewards                  | Encourages continued participation while keeping learning tasks short and goal-oriented     |

## 2.2 Study Setting and Participants

The pilot was conducted at Management and Science University, Shah Alam, Malaysia. A non-probability sample of 51 higher-education learners participated. Recruitment emphasized learners at the introductory end of the programming-experience spectrum rather than imposing a strict zero-experience criterion. This wording is consistent with the observed profile: 58% reported no prior programming experience, 34% reported basic or beginner experience, and 8% reported intermediate or higher experience. Participants used the application over a four-week period before completing the post-use questionnaire.

## 2.3 Measurement Instrument

A structured self-administered questionnaire was used to capture programming background, device access, preferred learning mode, perceived difficulty with traditional coding, satisfaction with existing platforms, interest in mobile-based coding, perceptions of gamification, curriculum preference, preference for game-based activities, and overall perceived usefulness of the proposed system. The instrument also incorporated usability and acceptance items adapted from the Technology Acceptance Model [3] and System Usability Scale [4]. Responses to attitudinal items were recorded on a five-point Likert scale ranging from Strongly Disagree to Strongly Agree. The source study reports Cronbach's alpha values of 0.81 for the usability subscale and 0.84 for overall user satisfaction, indicating acceptable internal consistency for this pilot.

## 2.4 Data Analysis

The final analysis focused on statistics that are directly supported by the available study record. Frequencies and percentages were used to summarize participant characteristics, access patterns, learning preferences, and acceptance indicators. Internal-consistency reliability was reported using the two available Cronbach's alpha coefficients. Inferential structural-equation, mediation, common-

method-bias, and predictive-relevance claims were not retained because the participant-level item data and corresponding reproducible outputs required to verify those analyses were not available. This decision avoids introducing coefficients, p-values, confidence intervals, AVE, CR, HTMT,  $R^2$ ,  $Q^2$ , or bootstrap results that cannot be independently supported by the available data.

## 2.5 Ethical Considerations

Participation was voluntary, informed consent was obtained, and survey data were processed anonymously. The study record reports institutional research approval and no conflict of interest. Data were used only for academic evaluation of the prototype and were reported in aggregate form.

## 3. Results

### 3.1 Participant Profile, Access, and Existing Learning Experience

Table 2 summarizes the participant profile and baseline learning context. The cohort was predominantly novice: 92% reported either no programming experience or only basic/beginner experience. Mobile-device access was prominent, with 67% identifying a smartphone or tablet as their primary learning device. Interactive and visual learning was preferred by 72% of participants, whereas only 18% preferred text-based learning and 10% preferred video-only learning. Difficulty with traditional coding approaches was common, with 69% agreeing or strongly agreeing that they had trouble. Satisfaction with existing learning platforms was comparatively weak: only 24% were satisfied, while 38% were neutral and 38% were unsatisfied.

**Table 2**

Participant profile and baseline learning context (N = 51)

| Indicator                           | Response                      | Percentage (%) |
|-------------------------------------|-------------------------------|----------------|
| Programming experience              | No experience                 | 58%            |
| Programming experience              | Basic / beginner              | 34%            |
| Programming experience              | Intermediate or above         | 8%             |
| Primary learning device             | Smartphone / Tablet           | 67%            |
| Primary learning device             | Laptop / Desktop              | 33%            |
| Preferred learning method           | Interactive & visual learning | 72%            |
| Preferred learning method           | Text-based learning           | 18%            |
| Preferred learning method           | Video-only learning           | 10%            |
| Difficulty with traditional coding  | Agree / Strongly Agree        | 69%            |
| Difficulty with traditional coding  | Neutral                       | 21%            |
| Difficulty with traditional coding  | Disagree                      | 10%            |
| Satisfaction with current platforms | Satisfied                     | 24%            |
| Satisfaction with current platforms | Neutral                       | 38%            |
| Satisfaction with current platforms | Unsatisfied                   | 38%            |

### 3.2 Acceptance of the Proposed Mobile Learning Approach

Participants showed strong interest in the proposed learning approach (Table 3). Ninety percent were interested or very interested in a mobile-based coding application. Gamification was viewed positively by most participants, with 76% agreeing or strongly agreeing that it increased learning motivation. A structured curriculum was preferred by 74% compared with language-by-language learning, and 71% preferred game-based activities over traditional quizzes. Overall perceived usefulness was also favorable: 82% rated the system useful or very useful, 13% were neutral, and 5% rated it not useful.

**Table 3**

Acceptance and learning-preference indicators for the proposed approach (N = 51)

| Indicator                           | Response                    | Percentage (%) |
|-------------------------------------|-----------------------------|----------------|
| Gamification increases motivation   | Agree / Strongly Agree      | 76%            |
| Gamification increases motivation   | Neutral                     | 17%            |
| Gamification increases motivation   | Disagree                    | 7%             |
| Interest in mobile-based coding app | Very interested             | 61%            |
| Interest in mobile-based coding app | Interested                  | 29%            |
| Interest in mobile-based coding app | Not interested              | 10%            |
| Curriculum structure                | Prefer curriculum-based     | 74%            |
| Curriculum structure                | Prefer language-by-language | 26%            |
| Learning activity format            | Prefer game-based learning  | 71%            |
| Learning activity format            | Prefer traditional quizzes  | 29%            |
| Perceived usefulness of system      | Useful / Very useful        | 82%            |
| Perceived usefulness of system      | Neutral                     | 13%            |
| Perceived usefulness of system      | Not useful                  | 5%             |

### 3.3 Internal Consistency

The available reliability results indicate acceptable internal consistency for the two reported scales. As shown in Table 4, the usability subscale produced a Cronbach's alpha of 0.81 and the overall user-satisfaction dimension produced an alpha of 0.84. Both values exceed the commonly used 0.70 criterion for early-stage research instruments. These coefficients support the consistency of the reported pilot measures, although a larger validation study should additionally report item-level loadings, composite reliability, convergent validity, and discriminant validity if latent-variable modelling is later undertaken.

**Table 4**

Reported internal-consistency reliability

| Scale                     | Cronbach's alpha | Interpretation                  |
|---------------------------|------------------|---------------------------------|
| Usability subscale        | 0.81             | Acceptable internal consistency |
| Overall user satisfaction | 0.84             | Acceptable internal consistency |

### *3.4 Discussion*

The results provide a coherent descriptive picture of why a mobile-first programming environment may be useful for this participant group. The high proportion of smartphone and tablet users supports the practical relevance of mobile access, while the strong preference for interactive and visual learning suggests that a purely text-centric approach may not align with the learning preferences of many beginners. This interpretation is consistent with broader evidence that thoughtfully integrated mobile devices can support learning performance and flexible access [7] and that digital tools can strengthen engagement when they are linked to appropriate pedagogy [9,12].

The combination of 69% reporting difficulty with traditional coding and only 24% being satisfied with existing platforms indicates an opportunity for simplified, consolidated learning environments. The proposed application addresses this by reducing tool fragmentation and placing multiple introductory languages, progress tracking, quizzes, and offline content within one interface. From a Cognitive Load Theory perspective, this design is intended to reduce avoidable interface, and navigation demands rather than to reduce the intrinsic complexity of programming itself [1]. Because cognitive load was not measured directly, the present results should be interpreted as evidence of perceived feasibility and acceptability, not as proof that cognitive load was statistically reduced.

The positive response to gamification should also be interpreted carefully. Seventy-six percent of participants perceived gamification as motivational and 71% preferred game-based activities, which supports continued use of structured rewards and progress cues in the prototype. However, longitudinal research demonstrates that gamification can lose effectiveness or even reduce motivation when rewards become controlling or superficial [6]. Research on digital game-based learning likewise shows that sustained engagement and retention are distinct from initial positive reactions [10]. Future evaluations should therefore examine repeated use over a full semester and include objective application logs.

The 82% usefulness rating is consistent with the central role of perceived usefulness in technology-acceptance research [3], while the 90% level of interest in mobile-based coding suggests a favourable initial adoption climate. Similar research on educational technology emphasizes that learner acceptance depends on both perceived utility and the design of the learning experience [11, 15]. Nevertheless, the present pilot does not establish actual long-term adoption, improved programming achievement, or causal relationships among usability, engagement, and satisfaction. Those questions require a larger design with participant-level data, pre/post learning measures, and appropriate inferential analysis.

The study also avoids assuming that younger learners are automatically technologically proficient. Bennett et al. [5] caution against broad “digital native” assumptions, which is relevant here because device familiarity does not necessarily translate into programming competence. The predominantly beginner profile therefore remains important even though most participants had access to mobile technology. Structured guidance, visual progression, and explicit learning pathways remain necessary to support self-regulated learning [2].

### *3.5 Limitations and Future Work*

The study has four principal limitations. First, the sample was small, drawn from a single institution, and recruited through non-probability procedures, which limits generalizability. Similar concerns about convenience-dominated evidence have been identified in recent educational-technology reviews [14]. Second, the evaluation relied mainly on self-reported perceptions and did not compare the application against a control condition. Third, the analysis did not include objective

achievement data, code-quality measures, or longitudinal application logs. Fourth, the theoretical ideas of cognitive load and self-regulated learning informed the design but were not directly measured. Future work should therefore use larger multi-institution samples, pre/post programming assessments, objective usage logs, and validated construct measures. If structural equation modelling is undertaken, all item-level data, model specifications, reliability statistics, AVE, HTMT, path coefficients, bootstrap confidence intervals,  $R^2$ , and predictive-relevance statistics should be retained and reported reproducibly.

#### 4. Conclusions

This study designed and pilot-evaluated a mobile-first, gamified Android application for introductory programming education. The prototype integrates Java, C++, and Python learning resources with offline access, progress tracking, quizzes, and gamified progression in a single lightweight environment. The participant profile demonstrates a clear practical rationale for this design: 67% primarily used smartphones or tablets for learning, 72% preferred interactive and visual learning, and 69% reported difficulty with traditional coding approaches. Acceptance indicators were favourable, with 90% interested or very interested in mobile-based coding, 76% viewing gamification as motivational, 74% preferring curriculum-based learning, 71% preferring game-based activities, and 82% rating the system useful or very useful. Reported reliability was acceptable for usability ( $\alpha = 0.81$ ) and user satisfaction ( $\alpha = 0.84$ ).

The main contribution of the study is therefore a defensible early-stage demonstration of feasibility and perceived acceptability rather than a claim of causal effectiveness. The results justify further development and a larger evaluation using objective learning outcomes, longitudinal usage data, and reproducible inferential modelling. This evidence-based positioning strengthens the manuscript by aligning its conclusions with the data that are available.

#### Acknowledgement

The author acknowledges Management and Science University (MSU), Shah Alam, Malaysia, for institutional support. The author declares no conflict of interest.

#### References

- [1] Sweller, John. "Cognitive load during problem solving: Effects on learning." *Cognitive science* 12, no. 2 (1988): 257-285. [https://doi.org/10.1207/s15516709cog1202\\_4](https://doi.org/10.1207/s15516709cog1202_4)
- [2] Zimmerman, Barry J. "Becoming a self-regulated learner: An overview." *Theory into practice* 41, no. 2 (2002): 64-70. [https://doi.org/10.1207/s15430421tip4102\\_2](https://doi.org/10.1207/s15430421tip4102_2)
- [3] Davis, Fred D. "Perceived usefulness, perceived ease of use, and user acceptance of information technology." *MIS quarterly* (1989): 319-340. <https://doi.org/10.2307/249008>
- [4] Brooke, John, Patrick W. Jordan, Bruce Thomas, Bernard A. Weerdmeester, and I. L. McClelland. "Usability evaluation in industry." (1996): 194. <https://doi.org/10.1201/9781498710411-35>
- [5] Bennett, Sue, Karl Maton, and Lisa Kervin. "The 'digital natives' debate: A critical review of the evidence." *British journal of educational technology* 39, no. 5 (2008): 775-786. <https://doi.org/10.1111/j.1467-8535.2007.00793.x>
- [6] Hanus, Michael D., and Jesse Fox. "Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance." *Computers & education* 80 (2015): 152-161. <https://doi.org/10.1016/j.compedu.2014.08.019>
- [7] Sung, Yao-Ting, Kuo-En Chang, and Tzu-Chien Liu. "The effects of integrating mobile devices with teaching and learning on students' learning performance: A meta-analysis and research synthesis." *Computers & Education* 94 (2016): 252-275. <https://doi.org/10.1016/j.compedu.2015.11.008>

- [8] Martin, Florence, and Doris U. Bolliger. "Engagement matters: Student perceptions on the importance of engagement strategies in the online learning environment." *Online learning* 22, no. 1 (2018): 205-222. <https://doi.org/10.24059/olj.v22i1.1092>
- [9] Bunjaku, Ines Bula, Shejnaze Gagic, and Mary Doyle Kent. "Integrating digital tools in engineering education: Social impact of technological integration." *IFAC-PapersOnLine* 58, no. 3 (2024): 118-122. <https://doi.org/10.1016/j.ifacol.2024.07.136>
- [10] Kiguchi, Mai, Waddah Saeed, and Imran Medi. "Churn prediction in digital game-based learning using data mining techniques: Logistic regression, decision tree, and random forest." *Applied Soft Computing* 118 (2022): 108491. <https://doi.org/10.1016/j.asoc.2022.108491>
- [11] Rof, Albert, Andrea Bikfalvi, and Pilar Marques. "Exploring learner satisfaction and the effectiveness of microlearning in higher education." *The Internet and Higher Education* 62 (2024): 100952. <https://doi.org/10.1016/j.iheduc.2024.100952>
- [12] Valtonen, Teemu, Sonsoles López-Pernas, Mohammed Saqr, Henriikka Vartiainen, Erkkö Tapio Sointu, and Matti Tedre. "The nature and building blocks of educational technology research." *Computers in Human Behavior* 128 (2022): 107123. <https://doi.org/10.1016/j.chb.2021.107123>
- [13] Eltaiba, Nada, Samira Hosseini, and Kingsley Okoye. "Benefits and impact of technology-enhanced learning applications in higher education in Middle East and North Africa: A systematic review." *Global Transitions* 7 (2025): 350-374. <https://doi.org/10.1016/j.glt.2025.06.004>
- [14] Segura-Altamirano, Segundo Francisco, Diana Mercedes Castro-Cárdenas, Merly Liliana Yataco-Bernaola, Amelia Rosa Mendoza-Rejas, Belinda Marleni Navarro-Guerra, Liliana Carmen Campos-Ramirez, Grimaldo Dermal Benavides-Campos, and Angel Almidon-Elescano. "Learning management systems, learning analytics, and student outcomes in engineering and computing higher education: A systematic review." *Social Sciences & Humanities Open* 13 (2026): 103018. <https://doi.org/10.1016/j.ssaho.2026.103018>
- [15] Wijaya, Tommy Tanu, Yiming Cao, Martin Bernard, Imam Fitri Rahmadi, Zsolt Lavicza, and Herman Dwi Surjono. "Factors influencing microgame adoption among secondary school mathematics teachers supported by structural equation modelling-based research." *Frontiers in psychology* 13 (2022): 952549. <https://doi.org/10.3389/fpsyg.2022.952549>
- [16] Jamison, Andrew, Anette Kolmos, and Jette Egelund Holgaard. "Hybrid learning: An integrative approach to engineering education." *Journal of Engineering Education* 103, no. 2 (2014): 253-273. <https://doi.org/10.1002/jee.20041>