

Scaffolding computer programming languages learning with tailored English vocabulary based on learners' performance states

Muhammad Adnan^{1,*}, Asad Habib¹, Jawad Ashraf¹, Shafaq Mussadiq¹, Arsalan Ali Raza²

¹Institute of Computing, Kohat University of Science and Technology, KUST, Kohat, Pakistan

²Comsats University Islamabad, Vehari Campus, Pakistan

Abstract

Due to the ubiquitous nature of mobile devices, they are now considered as an emerging platform for facilitating both teaching and learning experiences. In this paper, we presented a tailored mobile learning system, namely the Integrated English and Programming Language Learning System (IEPLS), which aims at learning English vocabulary before studying programming language concepts. The IEPLS supports programming language learning in three ways; (a) Recommending to learn specific English vocabulary used in programming language concepts (b) Adaptation to the learning flow of the learner and (c) Motivating and encouraging learners to learn items based on individual learner's performance. The IEPLS was used by one hundred and fifty undergraduate students for six months. Evaluation results revealed the attainment of IEPLS in supporting learners in learning programming languages backed by precise English vocabulary.

Received on 03 July 2019; accepted on 09 September 2020; published on 18 September 2020

Keywords: Programming languages learning, Algorithms, Self-regulated learning, Mobile learning, Mobile devices, Adaptive learning

Copyright © 2020 Muhammad Adnan *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi:10.4108/eai.26-5-2020.166350

1. Introduction

For computer science students, learning programming languages is a challenging task [1]. Authors in [2] and [3] had identified several key factors that make programming difficult. These factors include the inherent sensitivity of programming languages, students aptitude towards learning programming languages, learning style, motivation, interest, learning pace and teaching style, etc. These factors make programming languages difficult to learn in those countries where English is the mother tongue. It is vital that complexities and difficulties must be thoroughly identified in learning programming languages in those countries where English is considered as a second language (ESL) or a third language (ETL).

In Asia, English is considered as a second language and interestingly, also considered as a third learning language [4], [5]. In Asia and particularly in south-east Asian countries, students first learn their native language i.e. mother tongue. As Chinese, Urdu, Hindi, Arabic, etc. are the national languages of Asian countries, therefore, students also have to learn them as a second language and are taught as mandatory subjects in primary, secondary, and intermediate classes. Lastly, students also have to learn English as it is the official communication language [6]. Due to a lack of proper academic setting and lack of English learning environment, most of the students are not able to speak, write and understand English after completing secondary education (12 years of schooling) [7]. Because of weak English proficiency, students face difficulties in comprehension of courses taught at the university level because most of the books are written in English and courses are taught in English. For students, learning programming languages and facing difficulties is no exception. Learning a language involves two major

*Corresponding author. Email: Adnankust@gmail.com

steps: 1) building a significant amount of vocabulary and 2) learning the grammatical rules or syntax of that language. Therefore, in this research, we have proposed a system called Integrated English and Programming Language Learning System (IEPLS) that takes advantage of the ubiquitous nature of smartphones in helping undergraduate students to learn programming language-specific English vocabulary and then understand and practice programming language concepts. Details of IEPLS is presented in section 3.

The contemporary high-level programming languages such as JAVA, PHP, C#, Python, Objective-C, Ruby, C, and C++, etc. are very close to the English language which means that the keywords and language constructs found in these languages have almost the same meanings as in English language [8]. The syntax, language constructs and keywords found in one high-level programming language have almost the same meaning in another high-level programming language [9]. These characteristics of high-level programming languages infer that learning one programming language makes it easier to learn other programming languages. Rather than stressing beginner to learn different high-level programming languages simultaneously in a short academic period, it would be better to encourage and guide them in learning standard programming concepts (keywords, the syntax in pseudo-code, program execution steps) that apply to all high-level programming languages. These common programming concepts are expressed in Computer Algorithms [10]. The abstraction level of Algorithms is higher than high-level programming language in that they hide unnecessary details of step-wise computer program execution and low-level machine details. Our proposed IEPLS guides learners in learning basic programming concepts found in most programming languages. This system first encourages learners to learn algorithms and related English vocabulary. Based on the interest, performance, and preferences of learners, IEPLS allows them to choose a particular programming language for learning and doing programming exercises. We have used Android smartphones supported by web services as an assistive learning tool in learning algorithms, programming languages, and related English vocabulary [11].

Recent advancement in educational technologies has made learning personalized, tailored, mobilized, and learning at any time, anywhere. Moreover, progress in smartphones has changed the spectrum of learning from a one-size-fits-all approach to adapted and personalized learning [12], [13]. Inherently, smartphones are light weighted, compact, and more interactive as compared to laptops and desktop computers. These features of smartphones make them the perfect platform for accessing and sharing learning contents, especially in rural areas [14].

2. Literature Review

Computer programming is considered as the most exciting, challenging, and one of the most sought jobs of the 21st century. Yet, new programmers feel irritated and frustrated thinking that where they have landed up into. Most new programmers initially start questioning themselves whether programming is meant for them and whether they should continue doing it? The situation becomes more serious when new programmers have spent a few months doing programming and they can't see any useful outcome. It is a challenge for educational institutions to motivate new programmers to keep up with their decision and guide them in knowing applications of programming in solving real-world problems. The challenges for educational institutions become more severe when the medium of learning is English and students' mother language is not English. Therefore, it is challenging for educational institutions, having English as a second learning language, to represent computer programming in such a way that does not hesitate or frustrate new programmers.

Previously, numerous studies have reported the causes that make learning programming languages a daunting task. McDonald and Carlton pointed out that the main reason behind perceiving programming language learning is the lack of general problem-solving abilities in students [15]. Because students do not know how to create Algorithms, they fail to solve programming problems.

Several other reasons pointed out researchers in [16], [17] and [18] for making programming difficult are:

- Lack of personalized teaching
- One-size-fits-all type teaching approach that does not support students' learning style
- Lack of teaching dynamic programming language contents
- Lack of programming language expertise
- Weak mathematical and logical knowledge
- Lack of patience and persistence
- Lack of critical thinking, abstraction and knowledge accumulation
- Complex programming language syntax and structures
- Lack of students' motivation

Programming is a thinking process. It requires some abilities and patience in students that go beyond learning simple programming language syntax.

In solving programming problems, consolidated and broad programming language knowledge is required that students can apply when facing new programming problems. Several studies had been conducted to investigate and explore why learning to program is complicated and how modern tools and technologies can help students in making programming interesting, easy and pedagogically rich [19], [20], [21].

Similar to programming language problems, solutions and research studies, there are numerous studies related with English learning. Chen and Chung proposed personalized mobile learning system that facilitates learners in learning English based on item response theory and learning memory cycle [22]. Their mobile learning system was successfully implemented on Personal Digital Assistant (PDA) for personalized English learning. Experimental results revealed that the proposed system, due to its flexible and personalized functionality, was very successful in developing learners interest in learning English.

Wu et al., conducted a systematic quantitative analysis of the outcome of integrating mobile devices with learning and teaching on students' learning performance [23]. These studies found that overall effect of using mobile phones for studies is more efficient than using desktop computers. Through this meta-analysis, it was revealed that inquiry-oriented learning was more effective than the use of long online lectures, cooperating learning and game based learning. Students primarily consider mobile devices for communication and entertainment purpose; therefore, informal learning environments were more effective than their formal counterpart. Similarly, short-term interventions were superior to long-term intervention in encouraging students for improving their learning.

Chen and Li pointed out that in learning English, vocabulary learning is very vital as vocabulary makes the building block of English language [24]. Furthermore, the authors have stressed that situational i.e. context based language learning is important in enhancing learners interest and efficiency in learning English. Societal, cultural and life contexts in learning process makes vocabulary learning more effective and meaningful. For empirical proof, Personalized Context-aware Ubiquitous Learning System (PCULS) was developed that encourages English vocabulary learning based on learners location, leisure time, English language abilities. Experimental result showed that students who used context aware vocabulary learning system (PCULS) showed greater performance in English vocabulary learning as compared to students who have used it without context awareness.

In non-English speaking countries, Learning English and subsequently applying the English in academic settings and research studies becomes challenge to

the learners [25], [26]. The weak English-language speaking, writing, listening and reading skills make it harder to programming language students to learn programming language concepts and apply them in software development process. While interpreting programming problems, most of the times students are deducing English vocabulary meanings used in programming problems. Often, these non-familiar English vocabulary words create hindrance in learning programming. Therefore, in this research, through Android smartphone app, we have tried to facilitate students in learning programming languages specifics English vocabulary words and then applying those learned words in solving programming problems.

3. IEPLS Architecture and Design

This section describes the architecture and design of IEPLS and how it helps students in learning Algorithms supported by related English vocabulary words and subsequently helping students in learning programming languages using their smartphones. Internal architecture and working of IEPLS is presented in section 3.1. Section 4 explains how students interact with IEPLS smartphone app. IEPLS evaluation and acceptance by students is explained in section 5. Section 6 concludes our research work along with future directions.

3.1. IEPLS Working Architecture

Integrated English and Programming Language Learning System (IEPLS) uses Representational State Transfer (RESTful) based web service architecture integrated with Android smartphone app (Application) to facilitate students in learning algorithms, related English vocabulary and programming languages [27]. The RESTful based IEPLS architecture along with smartphone app components and functionality is presented in figure 1. IEPLS smartphone app consists of four modules namely algorithm learning module, programming language learning module, evaluation module and learning performance database module. IEPLS web server consists of learner' algorithm and English vocabulary performance database, learner' programming language performance database, RESTful web services, Algorithms, English vocabulary words and programming language skills synchronizer. IEPLS web services manipulate and hold records of students progress, performance and new recommendations.

Students interact with IEPLS system through their Android smartphone app. As IEPLS system was developed and tested in that part of the world where English is considered as second or third learning language, therefore, smartphone app was designed keeping in mind its perceived usefulness and perceived ease-of-use factors to the students [28], [29]. For

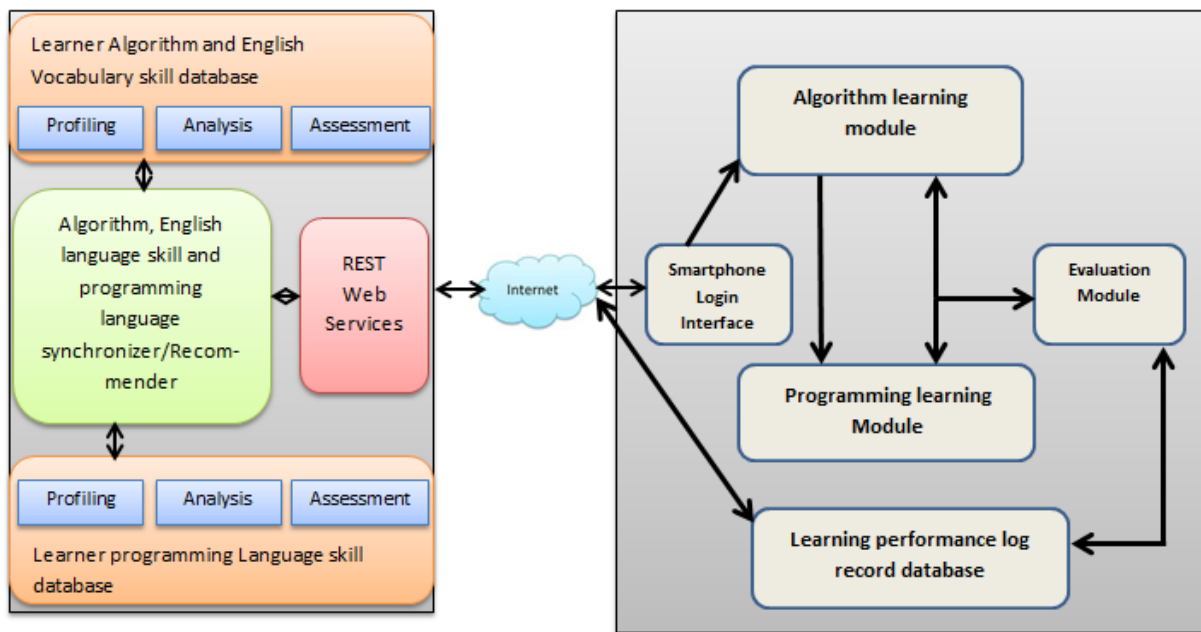


Figure 1. Working and architecture of Integrated English and Programming Language Learning System (IEPLS)

students, the use of smartphone app was made very simple. Students just need their university registration number and any desired password to log and use the smartphone app.

Algorithm Learning Module. Algorithm learning module is the primary module that a student interacts. Algorithm learning module supports algorithm learning in two ways.

- Showing the English meaning of words that students encounter during algorithm learning.
- Showing meaning of words and how they applied or used in algorithms i.e. how a concept or a word is conceived from an algorithmic point of view.

In Algorithm learning module, students are presented with learning levels that start from simple problems interpretation and then gradually progresses through more detail and specific problems that are typical in programming. During algorithm learning, as students encounter high learning levels, they are presented with new challenges that demand elaborated solutions requiring detailed algorithmic procedures. The main idea consists of making algorithm learning attractive and stimulating besides evaluating and promoting students in different algorithm learning levels.

The Algorithm learning module recommends new level to a student based on his/her performance in the previous level. At the end of each level, students are evaluated by evaluation module through multiple choice quizzes, pseudo code challenges and finding errors in algorithms. The primary objective of

Algorithm learning module is to encourage students in doing a lot of practice with algorithms concepts. Once students are confident about the knowledge they have gained, they will not hesitate in doing coding in high-level programming languages.

Students learning procedure with smartphone app interface connected with backend IEPLS system is summarized as follows:

Algorithm learning module has several levels starting from basic Algorithm concepts and moving towards advanced Algorithm concepts and applications. A student is promoted to next level based on his/her performance evaluated by evaluation module. Each Algorithm learning level as associated programming language learning level. If a student has mastered concepts in Algorithm learning level and has been cleared by evaluation module for that particular level then s/he can learn programming languages in corresponding programming language module level. Three programming languages are recommended to students in programming language module namely JAVA, PHP and C#. Out of these three programming languages, students can select preferred language for learning and are further evaluated in the same programming language.

The learning performance database module collects students performance results from evaluation module and stores it inside smartphone client side database. Furthermore, learning performance database module is also responsible for record synchronization between client side database and web server side database. During the learning process, different students can

have distinct learning and performance states. Students are free to learn based upon their learning interest, at anytime and anywhere. This freedom of learning reassures personalized learning behavior in students.

Evaluation Module. After learning each algorithm concept, students are evaluated in two ways. First students are given the task of writing their algorithm for assigned problem using their Android smartphone. With the help of simple code editor, students can write algorithm with it. The code editor embedded in IEPLS client algorithm learning module helps students in only writing algorithm code while the code is analyzed and assessed at web server side by the team of instructors and lab engineers. The IEPLS code editor resembles a very simple version of TouchDevelop code editors for mobile devices [30]. While using IEPLS code editor, students are supported with built-in common algorithmic statements that are building blocks of algorithms and any high-level programming languages.

IEPLS evaluation module checks students' algorithm problem-solving skills through several multiple choice questions (MCQ's). The result of MCQ's together with student algorithm code is stored and further sent by learning performance log record database to backend web service module for evaluation and assessment. To track students' progress in English vocabulary words learning and in English words meaning about algorithms, they are evaluated with multiple choice questions (MCQ's) tests with different test levels. The test levels start from beginner's skills and gradually move towards advanced expertise. To measure proficiency in English vocabulary words meaning and English words meaning about algorithm, students can:

- Take tests at any time and from anywhere from their smartphones
- Can take the same test multiple times for improvement
- Can share test scores with colleagues
- Can review their test scores data from a database record.

Students' English vocabulary skills and English words meaning about algorithms skills information is stored in learning performance log record database. The learning performance database also includes statistics about:

- Students logins into the system
- Students algorithm learning level
- Number of times students have attempted English vocabulary skill test and English words meaning about algorithms skill test.

- The number of times students have submitted solution (algorithmic) to the assigned problem

For information consistency, learning performance log record database and web services at backend carries out information synchronization process several times on daily basis. Algorithm learning levels are tethered to their corresponding programming language learning levels. Unless and until a student clears his/her particular algorithm learning level, he/she cannot attempt corresponding programming language learning level. This correspondence between algorithm learning levels and programming language learning levels encourage step by step learning and make students learning more controlled and personalized.

Programming Learning Module. Like Algorithm learning levels, programming language learning module also has several learning levels. At the end of each level, a student is evaluated with interactive multiple choice questions (MCQ's). For the interest of students, programming language learning module is embedded with three separate code interpreters called **CodeIt** for three programming languages (**CodeIt_JAVA**, **CodeIt_PHP**, **CodeIt_C#**). **CodeIt** allows students to write and run code right within their smartphone and subsequently view the output result. Similar to Algorithm learning module, the result of students evaluation in programming learning module is initially stored in learning performance log record database and later on sent to web service for assessment by instructors.

Web Services Module. IEPLS system uses Spring-based RESTful Web Services module for information exchange with IEPLS smartphone client module[31]. The information exchange includes programs, messages, documents and different user interaction statistics with IEPLS smartphone client module. Mainly two databases are maintained at web service module which stores learners skill information regarding algorithms learning, English vocabulary learning, and programming languages learning. At web service side, students information is analyzed and assessed by instructors' team from time to time. Appropriate learning recommendations are disseminated to students on their smartphones by instructors using web services. For information persistence, all usage data related to students performance are logged at web server side.

4. IEPLS Client Side Interaction

Figure 2a shows student's login interface of IEPLS client-side module. First, a student logs into the system by using his/her registration number and password. Subsequently, a student is directed towards Algorithm learning module.

Figure 2b displays learning levels related to basic and advanced algorithm learning concepts.

After learning each algorithm learning level, a student is evaluated by Evaluation module from four different perspectives to examine students' algorithm concepts learning ability. These four evaluation perspectives are:

- Examining student's English vocabulary words learning skills through multiple-choice questions which ask a question from a student and gave four answers of which one is right.
- Examining student's algorithm learning abilities through multiple choice questions which ask a question from a student and gave four answers of which one is correct. Figure 2c demonstrates algorithm quiz activity of android app.
- Examining student's English vocabulary words meaning about algorithms abilities through multiple choice questions which ask a question from a student and gave four answers of which one is correct.
- Asking students to write an algorithm to a problem similar to which he/she has understood in algorithm learning level.

Figure 2d shows one of algorithm writing activity performed by a student in Algorithm learning module. The results of first two perspectives are immediately revealed to a student for possible assessment.

After taking acceptable score in English vocabulary words test and English vocabulary words about algorithms test, a student is promoted to next higher learning level and is allowed to take the corresponding programming learning module learning level.

In programming learning module, a student is once again evaluated by evaluation module to examine student's programming language learning abilities from three perspectives. The three evaluation aspects include:

- Examining student's programming language learning skills through multiple-choice questions which ask a question from a student and gave four answers of which one is correct. Figure 2e demonstrates programming quiz activity of android app.
- Asking a student to write program code for a problem using **CodeIt** editor and immediately showing the result of program execution to a student for possible assessment and error correction.
- Asking a student to correct a code presented to him/her in **CodeIt** editor for checking student's programming abilities.

The results of all the three evaluation perspectives are immediately shown to a student and further sent to the web server for possible evaluation to instructors. Figure 2f shows computer program writing activity in JAVA language performed by a particular student.

5. IEPLS Evaluation and Acceptance

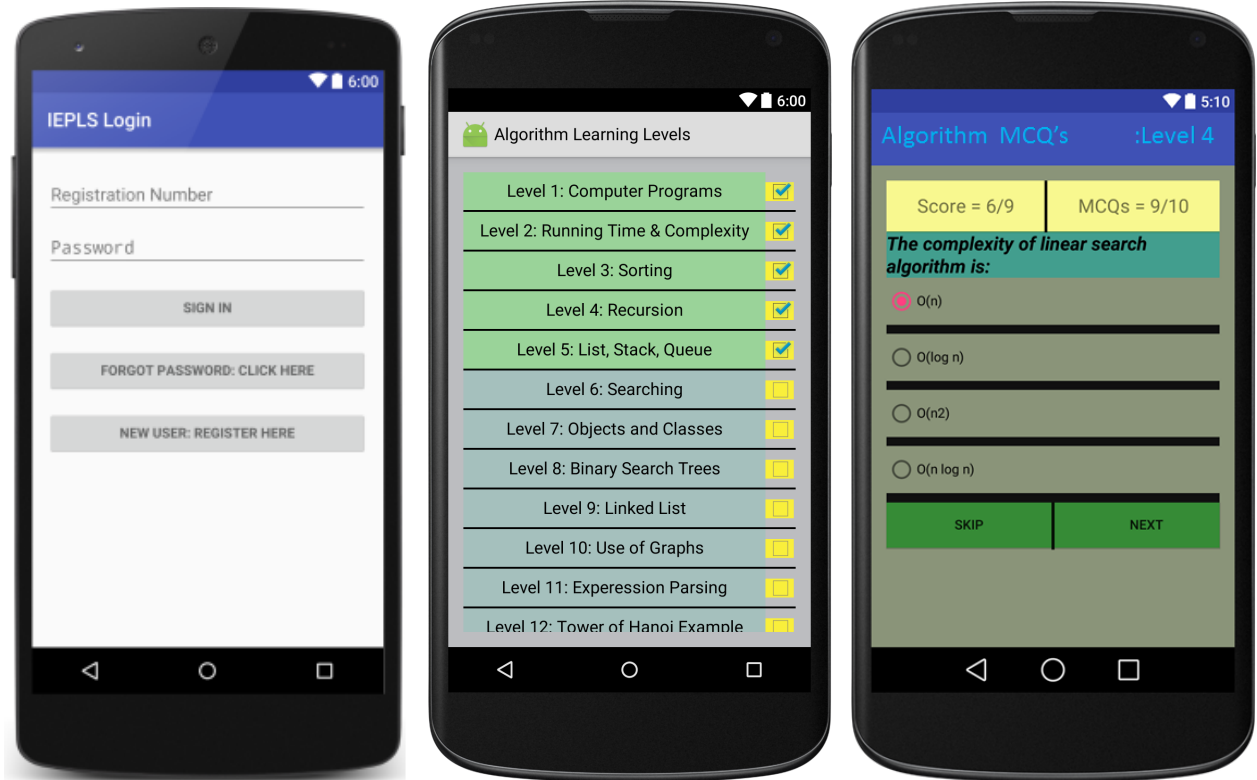
IEPLS evaluation and its impact on increasing learning performance of students were assessed by analyzing data retrieved from IEPLS database server. Students adoption and acceptance of IEPLS were determined by Technology Acceptance theoretical Model (TAM) which includes four factors namely perceived ease of use (PEOU), Perceived Usefulness (PU), Attitude (AT) and Behavioral Intention (BI) [32].

5.1. IEPLS Statistical Analysis

Table 1 displays IEPLS information and students usage statistics retrieved from backend web server database. Total numbers of logins made during six months were 80882, an average of 539.21 logins per students which indicates that most students were interested in using IEPLS system. The statistical data in table 1 does not reveal how much English vocabulary words learning, algorithm and programming skills of students were enhanced. Therefore, a more elaborated analysis was performed on students tests activities performance in each month.

Total points for each multiple choice quiz, algorithm assignment, programming assignment and programming code correction were ten. Figure 3a, 3b, 3c, 3d, 3e and 3f show average students scores per month for English vocabulary meanings tests, English vocabulary meanings about algorithm learning tests, algorithm assignments, programming languages MCQ's, programming assignments and programming code correction tests for a period of six months. The graphs in figure 3a, 3b, 3c, 3d, 3e and 3f clearly show that in every succeeding month, on average, learning performance of students improved. These figures show that on average the entire performance of students in the last learning month was much superior to learning performance in first learning month. Therefore, IEPLS was efficacious in encouraging and assisting students in improving the English vocabulary and programming language skills.

Next, we ran paired sample t test using SPSS 22.0 on learning performance results of the first month and sixth-month tests to find the statistical evidence of the impact of the IEPLS on learning efficiencies of students. To simplify the statistical results, we have shown paired sample t-test results of English vocabulary meaning MCQ's in algorithm learning module, English vocabulary meaning about algorithm learning mcq's in algorithm learning module and programming learning MCQ's in programming learning module. Table 2 shows



(a) Student login Interface

(b) Algorithm learning levels

(c) Algorithm MCQ's quiz



(d) Algorithm writing activity

(e) Programming MCQ's quiz

(f) Computer program writing activity

Figure 2. Students' IEPLS interaction activities

Table 1. IEPLS System usage statistics retrieved from log record

S.no	Items	Total Numbers	Average
1	Total number of students participated in Evaluation and Survey	150	
2	Evaluation period	6 months	
3	Total numbers of logins to IEPLS system	80882	539.21
4	Total numbers of learning levels in Algorithm Learning Module	20	
5	Total numbers of learning levels in Programming Learning Module	20	
6	Total number of English Vocabulary words MCQ's tests conducted in Algorithm learning module by 150 students	11877	79.18 MCQ's tests per student
7	Total number of English vocabulary words w.r.t Algorithm learning MCQ's tests conducted in Algorithm learning module by 150 students	8927	59.51 MCQ's tests per student
8	Algorithm assignment attempts by 150 students	6711	44.74 attempts per student
9	Total number of Programming MCQ's tests conducted in programming learning module by 150 students	10637	70.91 MCQ's tests per student
10	Programming Assignment attempts by 150 students	5839	38.92 attempts per student
11	Programming code correction assignments by 150 students	6111	40.74 attempts per student

the result of paired sample t-test for English vocabulary meaning MCQ's in algorithm learning module whereas table 3 and table 4 indicates the result of paired sample t-tests for English vocabulary meaning about algorithm learning MCQ's and programming learning MCQ's in programming learning module respectively. From the results in table 2, we found out that:

- English Vocabulary MCQ's tests conducted in the first month and sixth month with means of 3.38 and 7.22 respectively, were weakly correlated (0.022) with a positive significance value of .785. It revealed that on average students were having low scores in the first month of IEPLS usage, but IEPLS helped them a lot to increase their English vocabulary meaning skills and students scored relatively high in the sixth month of MCQ's tests.
- Moreover, data for *paired differences* revealed that having 149 degrees of freedom and t value of -36.72, the result of IEPLS learning performance is significant with significance value i.e. $P=000 < 0.05$. The value of *mean* = -3.84 also revealed that overall learning performance of each student increased by 3.84.

Similarly, the paired sample t-test result in table 3 for English vocabulary meaning about algorithm learning MCQ's for the first month and the sixth month shows that students English vocabulary meaning skills about algorithm learning abilities were promoted from 3.81 to 6.75. Moreover, the paired sample correlation result in table 3 shows that correlation is not significance, indicating that IEPLS helped students during six month evaluation period in improving their abilities. The comparison result in table 3 shows that there is significant difference in students first month abilities and sixth month abilities (Mean = 2.94, $t = 25.54$, $p = .000$). That is, most students performances were clearly improved after using IEPLS system. Table 3 also shows that mean testing score escalates with 2.94 points for every student.

Finally, the paired sample t-test result, correlation and descriptive statistics for programming learning MCQ's are listed in table 4. The students mean score of the first month and sixth months in programming learning MCQ's tests are 5.01 and 7.67, respectively. Table 4 lists that the mean scores of the first month and sixth month have a strong positive correlation which is confirmation of the fact that if a student gets a high score the first month test then s/he should also

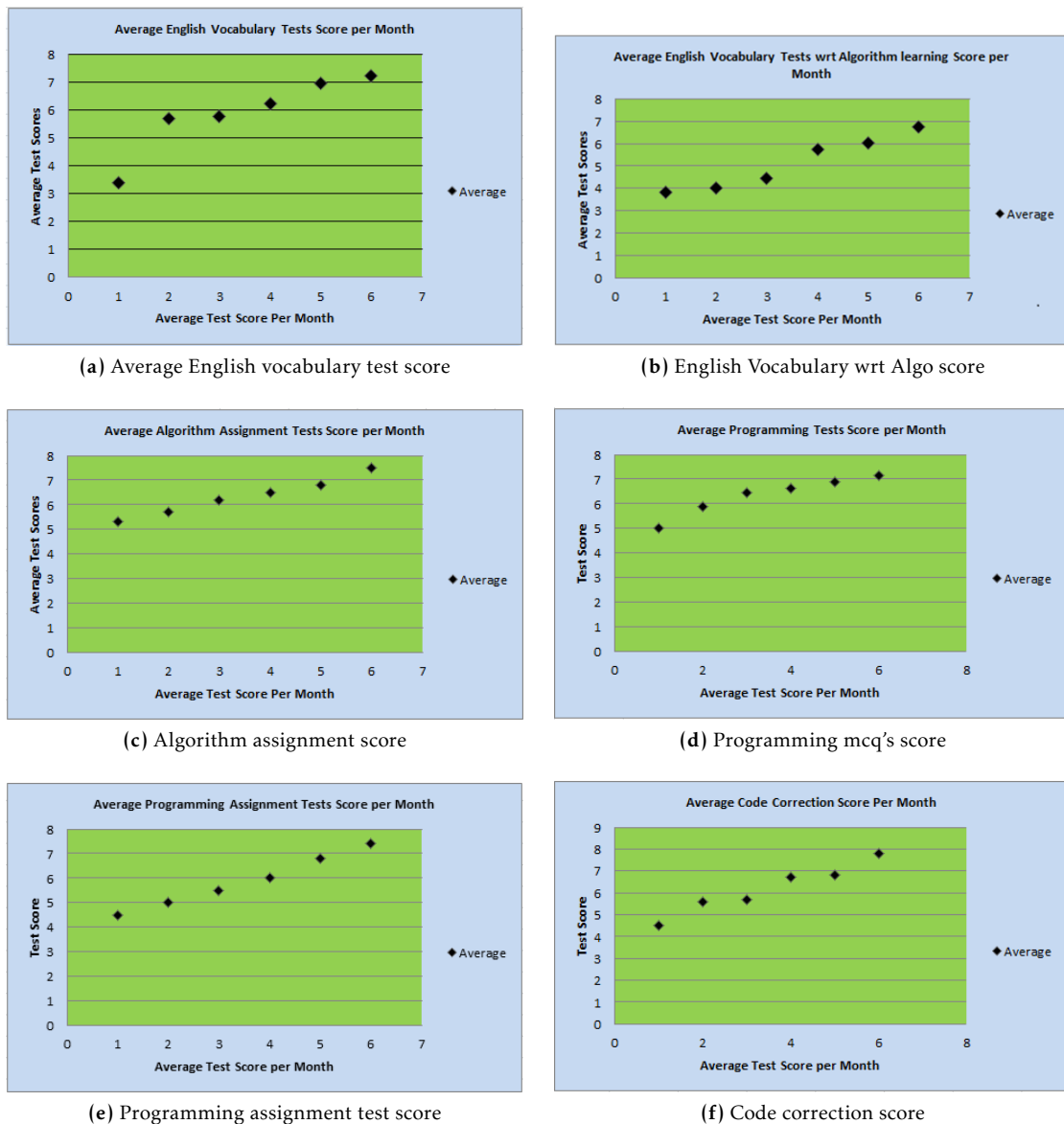


Figure 3. Students' scores in IEPLS activities

gets a high score in sixth month test. The moderate correlation between the first month and sixth month tests in programming learning module indicated that corresponding algorithm learning module was able to make strong programming foundation for students. In addition, table 4 shows significant (2-tailed) p-value of .000 with 149 degrees of freedom and t statistics = -24.473. Since $p < 0.05$, with the difference of 2.66 in mean test score, there is substantial evidence ($t = -24.474$, $p = .000$) that IEPLS intervention in students learning process improved their learning performance in programming language learning. The results also display that, on average, learning performance of students increased by 2.66 points from first learning month to sixth learning month.

5.2. Analyzing Students Acceptance and Adoption of IEPLS based on TAM

In the last stage of research, we used Technology Acceptance Model (TAM) to measure students acceptance of IEPLS and adopting it or similar software systems in future [32]. Presented first by Davis in 1989, TAM is information system framework used to measure how users accept and use information technology systems. TAM is composed of two independent variables called Perceived ease-of-use (PEOU) and Perceived usefulness (PU). Also, there are two dependent variables in TAM called Attitude towards using (AT) and users behavioral intention (BI). According to TAM, actual information system usage is determined by users behavioral intention and behavioral intention is dependent on users

Table 2. Paired Samples Test of English vocabulary meaning MCQ's for first and sixth month

		Paired Differences							
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	First MonthTests - Six MonthTests	-3.84000	1.28073	.10457	-4.04663	-3.63337	-36.721	149	.000
Paired Sample Statistic									
		Mean	N	Std. Deviation	Std. Error Mean				
Pair 1	First MonthTests	3.3873	150	.64561	.05271				
	Sixth MonthTests	7.2273	150	1.12070	.09150				
Paired Sample Correlation									
		N	Correlation	Sig.					
Pair 1	First MonthTests & Six MonthTests	150	.022	.785					

Table 3. Paired Samples Test of English vocabulary meaning wrt algorithm learning mcq's for first and sixth month

		Paired Differences							
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	First MonthTests - Six MonthTests	-2.9413	1.41047	.11516	-3.16890	-3.63337	-2.71377	149	.000
Paired Sample Statistic									
		Mean	N	Std. Deviation	Std. Error Mean				
Pair 1	First MonthTests	3.8153	150	.57902	.04728				
	Sixth MonthTests	6.7567	150	1.30160	.10627				
Paired Sample Correlation									
		N	Correlation	Sig.					
Pair 1	First MonthTests & Six MonthTests	150	.027	.747					

attitude towards the system, perceived usefulness and Perceived Ease of Use (PEOU) of the system. Perceived ease-of-use (PEOU) is defined as "the degree to which a person believes that using a particular system would be free of efforts". Perceived Usefulness (PU) is defined as "the degree to which a person believes

that by using a particular system will increase his aptitude and job performance".

Survey Methodology. Technology Acceptance Model (TAM) questionnaire consisting of 16 questions was broadcasted online on students email addresses using

Table 4. Paired Samples Test of programming language learning mcq's for first and sixth month

		Paired Differences							
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	First MonthTests - Six MonthTests	-2.6653	1.33386	.10891	-2.88054	-2.45013	-24.473	149	.000
Paired Sample Statistic									
		Mean	N	Std. Deviation	Std. Error Mean				
Pair 1	First MonthTests	5.0120	150	.91394	.07462				
	Sixth MonthTests	7.6773	150	.84783	.06922				
Paired Sample Correlation									
		N	Correlation	Sig.					
Pair 1	First MonthTests & Six MonthTests	150	-.145	.076					

Table 5. Technology Acceptance Model (TAM) Based students' satisfaction evaluation questionnaire

TAM Constructs	Questions (Likert Type Scale was use to answers these questions i.e. 1 = Strong disagree, 2 = Disagree, 3 = Somewhat disagree, 4 = neutral, 5 = Somewhat Agree, 6 = Agree, 7 = Strongly Agree)	Source (citation)
Perceived Ease of Use (PEOU)	Using the IEPLS application would enable me to accomplish tasks more quickly	[29] [33]
	Using the IEPLS application would make it easier to do my job	
	Using the IEPLS application every day will be easy for me	
	I will be able to learn by myself how to use the IEPLS application installed on my smartphone	
Perceived Usefulness (PU)	Using the IEPLS application would enhance my English and programming language skills.	[34]
	Using the IEPLS application would improve my performance in English and programming tests.	
	Using IEPLS application would increase my productivity.	
	I find the IEPLS application useful in undergraduate program.	
Attitude towards Use (At)	Using IEPLS application is a good idea.	[35] [36]
	Using IEPLS application is a wise idea.	
	I think that using IEPLS application in university is a good idea	
	I like the idea of using IEPLS application	
Behavioral Intention (BI)	I intend to use the IEPLS application frequently in the coming year	[37]
	I will use IEPLS application regularly in university in the coming years.	
	I will be a heavy user of the IEPLS application in the coming year	
	I will recommend IEPLS application to other in coming future	

Google form data collection service. The questionnaire is displayed in table 5.

We used correlation analysis, regression analysis, and path analysis as data analysis approach to finding

relations among constructs of Technology Acceptance Model (TAM).

First, we contended bivariate Pearson Correlation on TAM PEOU, IEPLS PU, IEPLS AT and IEPLS BI, which measures the strength and direction of linear relationships between pairs of constructs. Table 6 displays correlation coefficient, r , between all pairs of TAM constructs, the significance value p and 150 students observations. Based on the results of table 6, we can state the following:

- There is statistically significant positive linear relationship between all pair of TAM construct with $p < 0.01$
- The magnitude, or strength of relationship i.e. r , is moderate and strong ($.455 < |r| < .869$).
- All constructs are positively correlated, meaning that greater value of PEOU is associated with a greater value of PU, AT, and BI. Similarly, a greater value of PU is associated with a greater value of PEOU, AT and BI and so on.

Linear Regression and Multiple Linear Regression Analysis.

We conducted linear regression and multiple linear regressions to predict the casual relationship between Technology Acceptance Model (TAM) dependent variables and independent variables.

From table 7, the adjusted R-square value of .491 determines that the two independent variables (PEOU and PU) contribute 49 percent of variability on Attitude towards use (AT) variable. In table 8, with a mean square value of 30.162 and f statistics = 72.971, the significance value (.000) is less than 0.05 which concludes that we have strong evidence that independent variables of PEOU and PU do predict Attitude toward Use (AT) and that regression model has explanatory power and is valid. In table 9, the unstandardized coefficients (Beta) for Perceived Ease of Use and Perceived Usefulness are .225 and .719, which implies that with one unit increase in PEOU and PU, the Attitude towards Use (AT) increases by .225 units for PEOU and .719 units for PU respectively. Table 9 also shows that Perceived Ease of Use independent variable ($\beta = .225$) has slightly smaller impact on Attitude towards Use than Perceived Usefulness independent variable ($\beta = .719$). The significance values in tables 9 clearly indicates that both variables of PEOU and PU added significantly to Attitude towards Use variable with $p < .05$.

Similarly, multiple regression analysis was conducted on the impact of Perceived Usefulness (PU) and Attitude (AT) on Behavioral Intention (BI). In table 10, the adjusted R-square value is .609, which determines that 69 percent of the variance in Behavioral Intention (BI) is contributed by Perceived Usefulness (PU)

and Attitude towards use (AT). In table 11, with a mean square value of 28.380 and f statistics = 117.022, the significance value (.000) is less than 0.05 which concludes that we have strong evidence that independent variables of PU and Attitude do predict Behavioral Intention (BI) and that regression model has explanatory power and is valid. Table 12 shows that, relatively, with one point increase in Perceived Usefulness (PU), the increase in Behavioral Intention (BI) is 1.020 and with one point increase in Attitude towards Use (AT), the increase in Behavioral .149. Therefore, Perceived Usefulness (PU) accounts for more variability in Behavioral Intention (BI) than Attitude towards Use (AT). The significance values for Perceived Usefulness (PU) and Attitude towards Use (AT) are .000 and .018 indicating that both variables significantly contributed to variance in Behavioral Intention (BI).

In Table 13, the value of adjusted R Square (.739) explains that 74 percent of the variance in Perceived Usefulness variable is predicted by Perceived Ease of Use (PEOU). 74 percent of the variance in Perceived Usefulness by Perceived Ease of Use (PEOU) indicates that Perceived Ease of Use (PEOU) is a significant predictor of variance in Perceived Usefulness (PU). The significance value of .000 in table 14 indicated that our regression model for PEOU and PU is a valid and rational model. The significance P value for PEOU in table 15 is .000, which is way less than .05 indicating that PEOU added significantly to PU.

We also ran a simple linear regression on Attitude towards Use (AT) and Behavioral Intention (BI) data to determine how much the dependent variable of Behavioral Intention (BI) is predicted by independent variable of Attitude towards Use (AT). Table 16 shows adjusted R square value of .202 for the predictor variable Attitude towards Use (AT), indicating that a variance of 20 percent is caused in Behavioral Intention (BI) by Attitude towards Use (AT). The significance value of .000 in table 17 indicates that our regression model for Attitude towards Use (AT) and Behavioral Intention (BI) is valid and a rational model. The significance P value for Attitude towards Use (AT) in table 18 is .000, which is way less than .05 indicating that Attitude towards Use (AT) added significantly to Behavioral Intention (BI). Figure 4 shows Technology Acceptance Model, its variables and the unstandardized coefficient weights between its variables.

Finally, randomly selected twenty students were interviewed about their experience with IEPLS and their views on IEPLS strengths and weaknesses. On average each interview lasted for five minutes. Twenty out of twenty students thought that their English and programming language abilities were improved after using IEPLS. Students believed that IEPLS was very friendly, simple, elaborated and gave them enough time to learn English vocabulary words and practice their

Table 6. Correlations among TAM constructs

		IEPLS_PEOU	IEPLS_PU	IEPLS_AT	IEPLS_BI
IEPLS_PEOU	Pearson Correlation	1	.861**	.648**	.869**
	Sig. (2-tailed)		.000	.000	.000
	N	150	150	150	150
IEPLS_PU	Pearson Correlation	.861**	1	.700**	.774**
	Sig. (2-tailed)	.000		.000	.000
	N	150	150	150	150
IEPLS_AT	Pearson Correlation	.648**	.700**	1	.455**
	Sig. (2-tailed)	.000	.000		.000
	N	150	150	150	150
IEPLS_BI	Pearson Correlation	.869**	.774**	.455**	1
	Sig. (2-tailed)	.000	.000	.000	
	N	150	150	150	150

** . Correlation is significant at the 0.01 level (2-tailed).

Table 7. Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.706 ^a	.498	.491	.64291

a. Predictors: (Constant), IEPLS_PU, IEPLS_PEOU

Table 8. ANNOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	60.323	2	30.162	72.971	.000 ^b
	Residual	60.760	147	.413		
	Total	121.083	149			

a. Dependent Variable: IEPLS_AT
b. Predictors: (Constant), IEPLS_PU, IEPLS_PEOU

Table 9. Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.294	.433		.679	.498
	IEPLS_PEOU	.225	.149	.174	1.513	.032
	IEPLS_PU	.719	.150	.551	4.797	.000

a. Dependent Variable: IEPLS_AT

Table 10. Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.784 ^a	.614	.609	.49246

a. Predictors: (Constant), IEPLS_AT, IEPLS_PU

programming skills. Students thought that traditional universities programming classes are fast-paced, not allowing all students to practice programming tasks again and again. Contrasts to other subjects, computer

programming subjects need time, patience, thinking process and a lot of practice to learn it.

Based on information collected from students through interviews, surveys and from database log

Table 11. ANNOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	56.760	2	28.380	117.022	.000 ^b
	Residual	35.650	147	.243		
	Total	92.410	149			
a. Dependent Variable: IEPLS_BI						
b. Predictors: (Constant), IEPLS_AT, IEPLS_PU						

Table 12. Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.708	.320		2.213	.028
	IEPLS_PU	1.020	.082	.894	12.457	.000
	IEPLS_AT	.149	.063	.171	2.383	.018
a. Dependent Variable: IEPLS_BI						

Table 13. Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.861 ^a	.741	.739	.35240
a. Predictors: (Constant), IEPLS_PEOU				

Table 14. ANNOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	52.594	1	52.594	423.504	.000 ^b
	Residual	18.380	148	.124		
	Total	70.973	149			
a. Dependent Variable: IEPLS_PU						
b. Predictors: (Constant), IEPLS_PEOU						

Table 15. Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.713	.230		3.103	.002
	IEPLS_PEOU	.854	.041	.861	20.579	.000
a. Dependent Variable: IEPLS_PU						

Table 16. Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.455 ^a	.207	.202	.70367
a. Predictors: (Constant), IEPLS_AT				

records, we conclude the following arguments about how programming abilities of students can be improved

where the English language is considered as second or third learning language.

Table 17. ANNOVA^a

Model	Sum of Squares	df	Mean Square	F	Sig.	
1	Regression	19.127	1	19.127	38.629	.000 ^b
	Residual	73.283	148	.495		
	Total	92.410	149			
a. Dependent Variable: IEPLS_BI						
b. Predictors: (Constant), IEPLS_AT						

Table 18. Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.257	.351		9.276	.000
	IEPLS_AT	.397	.064	.455	6.215	.000
a. Dependent Variable: IEPLS_BI						

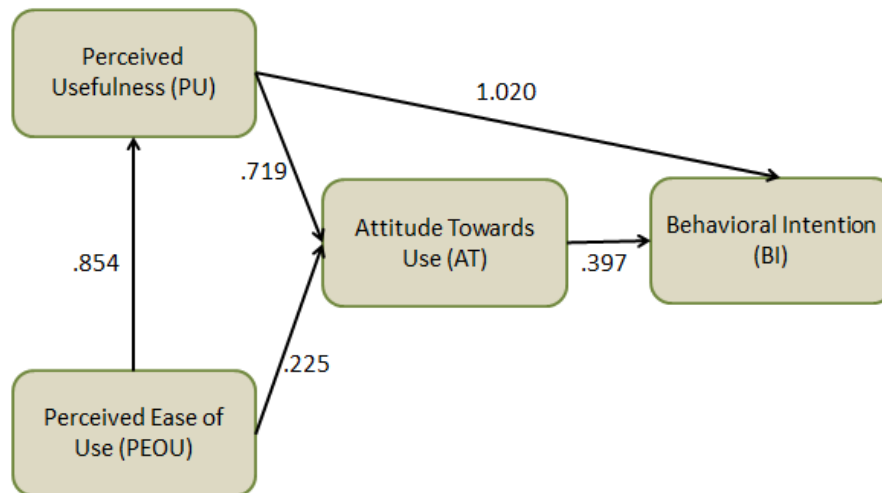


Figure 4. Unstandardized Coefficient weights between Technology Acceptance Model (TAM) variables

- To overcome students frustrations towards learning programming language, learning system or application should integrate English vocabulary words (programming language oriented) help and assistance.
- Learning system should give platform to students where they can freely practice programming activities i.e. providing learning by doing platform.
- Learning system should encourage students to practice programming exercises with diverse programming challenges and activities.
- Learning system should provide self-assessment activity where students make sure that his/her learning was effective after concluding a programming activity.
- Rather than stressing on learning one particular programming language, learning system should motivate students to learn analytical skills by developing algorithm based problem-solving abilities.
- Learning system has to be regular and progressive not compelling students to complete an activity in urgency. Learning programming is different from other subject requiring solving problems from various interpretations, intensive exercises, deep thinking process and analytical skills.
- Programming is not learned by memorizing procedures and formulas. Rather, programming skills

are developed by solving as much programming problems as possible. Programming problems are limitless and challenging. Therefore, programming learning system must provide platform where students can do different experiments solving diverse problems knowing where to apply the right procedure and formula accurately.

6. Conclusion

This research work presents IEPLS developed primarily to increase programming oriented English vocabulary words and computer programming abilities of students living in those areas where the English language is considered as the second and third learning language. IEPLS is based on personalized and self-control learning, where students are gradually promoted to higher learning levels based on their English vocabulary and programming abilities. Experimental results demonstrated that through a simple and attractive smartphone interfaces, IEPLS was successful in increasing the English vocabulary and programming skills of students. The experimental results also revealed that computer programming needs time and ample practice to acquire it efficiently. Additionally, our research showed that smartphones currently and in the future could become a very useful tool in facilitating students in their studies. There are many parts of the world where programming courses are taught in the English language, and programming courses, in general, are considered very challenging. Weak English language skills make programming courses more challenging and arduous. IEPLS was mainly developed for those students who, due to the geographical location have poor English language skills. IEPLS facilitated students in learning programming oriented English vocabulary words concurrently with learning computer programming. Technology Acceptance Model (TAM) based survey and analysis in our research determined that IEPLS was successful in increasing students' programming abilities and their intention towards using smartphones in their learning process. In the future, we plan to develop a similar type of system which can improve students' mathematical, analytical reasoning and logical abilities.

References

- [1] M. Armoni, O. Meerbaum-Salant, M. Ben-Ari, From scratch to Scratch programming, *ACM Transactions on Computing Education (TOCE)* 14 (4) (2015) 25.
- [2] A. Luxton-Reilly, Learning to program is easy, in: *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ACM, 2016, pp. 284–289.
- [3] S. I. Malik, J. Coldwell-Neilson, A model for teaching an introductory programming course using adri, *Education and Information Technologies* 22 (3) (2017) 1089–1120.
- [4] S. M. Gass, *Input, interaction, and the second language learner*, Routledge, 2017.
- [5] K. Hyland, F. Hyland, *Feedback in second language writing: Contexts and issues*, Cambridge university press, 2019.
- [6] R. Kirkpatrick, *English language education policy in Asia*, Springer, 2016.
- [7] B. Fenton-Smith, P. Humphreys, I. Walkinshaw, *English medium instruction in higher education in Asia-Pacific*, Springer, 2017.
- [8] J.-M. Sáez-López, M. Román-González, E. Vázquez-Cano, Visual programming languages integrated across the curriculum in elementary school: A two year case study using Scratch in five schools, *Computers & Education* 97 (2016) 129–141.
- [9] C.-Y. Tsai, Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy, *Computers in Human Behavior* 95 (2019) 224–232.
- [10] D. Zhang, H. Fu, J. Han, A. Borji, X. Li, A review of co-saliency detection algorithms: fundamentals, applications, and challenges, *ACM Transactions on Intelligent Systems and Technology (TIST)* 9 (4) (2018) 38.
- [11] D. Hintze, P. Hintze, R. D. Findling, R. Mayrhofer, A large-scale, long-term analysis of mobile device usage characteristics, *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1 (2) (2017) 13.
- [12] L. Briz-Ponce, A. Pereira, L. Carvalho, J. A. Juanes-Méndez, F. J. García-Peñalvo, Learning with mobile technologies—students' behavior, *Computers in Human Behavior* 72 (2017) 612–620.
- [13] M. Al-Emran, H. M. Elsherif, K. Shaalan, Investigating attitudes towards the use of mobile learning in higher education, *Computers in Human Behavior* 56 (2016) 93–102.
- [14] P. Pruet, C. S. Ang, D. Farzin, N. Chaiwut, Exploring the internet of educational things (ioet) in rural underprivileged areas, in: *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, IEEE, 2015, pp. 1–5.
- [15] C. McDonald, Why is teaching programming difficult?, in: *Higher Education Computer Science*, Springer, 2018, pp. 75–93.
- [16] S. Grover, R. Pea, S. Cooper, Factors influencing computer science learning in middle school, in: *Proceedings of the 47th ACM technical symposium on computing science education*, ACM, 2016, pp. 552–557.
- [17] L. Benton, C. Hoyles, I. Kalas, R. Noss, *Building mathematical knowledge with programming: insights from the scratchmaths project*, Suksapattana Foundation, 2016.
- [18] S. K. Card, *The psychology of human-computer interaction*, CRC Press, 2018.
- [19] M. Alrashidi, K. Almohammadi, M. Gardner, V. Callaghan, Making the invisible visible: real-time feedback for embedded computing learning activity using pedagogical virtual machine with augmented reality, in: *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*,

- Springer, 2017, pp. 339–355.
- [20] R.-S. Shaw, The learning performance of different knowledge map construction methods and learning styles moderation for programming language learning, *Journal of Educational Computing Research* 56 (8) (2019) 1407–1429.
- [21] Á. Santos, A. Gomes, A. J. Mendes, Integrating new technologies and existing tools to promote programming learning, *Algorithms* 3 (2) (2010) 183–196.
- [22] C.-M. Chen, C.-J. Chung, Personalized mobile english vocabulary learning system based on item response theory and learning memory cycle, *Computers & Education* 51 (2) (2008) 624–645.
- [23] Q. Wu, Learning esl vocabulary with smartphones, *Procedia-Social and Behavioral Sciences* 143 (2014) 302–307.
- [24] C.-M. Chen, Y.-L. Li, Personalised context-aware ubiquitous learning system for supporting effective english vocabulary learning, *Interactive Learning Environments* 18 (4) (2010) 341–364.
- [25] J.-J. He, S.-Y. Chiang, Challenges to english-medium instruction (emi) for international students in china: A learners' perspective: English-medium education aims to accommodate international students into chinese universities, but how well is it working?, *English today* 32 (4) (2016) 63–67.
- [26] B. Tomlinson, Current issues in the development of materials for learners of english as an international language (eil), in: *English Language Teaching Today*, Springer, 2016, pp. 53–66.
- [27] Y. Liu, X. Liu, Y. Ma, Y. Liu, Z. Zheng, G. Huang, M. B. Blake, Characterizing restful web services usage on smartphones: A tale of native apps and web apps, in: *2015 IEEE International Conference on Web Services, IEEE, 2015*, pp. 337–344.
- [28] N. Marangunić, A. Granić, Technology acceptance model: a literature review from 1986 to 2013, *Universal Access in the Information Society* 14 (1) (2015) 81–95.
- [29] M. Al-Emran, V. Mezhuyev, A. Kamaludin, Technology acceptance model in m-learning context: A systematic review, *Computers & Education* 125 (2018) 389–412.
- [30] Y. Valsamakis, A. Savidis, Personal applications in the internet of things through visual end-user programming, in: *Digital Marketplaces Unleashed*, Springer, 2018, pp. 809–821.
- [31] R. Johnson, J. Hoeller, A. Arendsen, R. Thomas, *Professional Java Development with the Spring Framework*, John Wiley & Sons, 2009.
- [32] H. Taherdoost, Development of an adoption model to assess user acceptance of e-service technology: E-service technology acceptance model, *Behaviour & Information Technology* 37 (2) (2018) 173–197.
- [33] A. Brandon-Jones, K. Kauppi, Examining the antecedents of the technology acceptance model within e-procurement, *International journal of operations & production management* 38 (1) (2018) 22–42.
- [34] P. Verma, N. Sinha, Integrating perceived economic wellbeing to technology acceptance model: The case of mobile based agricultural extension service, *Technological forecasting and social change* 126 (2018) 207–216.
- [35] R. Scherer, F. Siddiq, J. Tondeur, The technology acceptance model (tam): A meta-analytic structural equation modeling approach to explaining teachers' adoption of digital technology in education, *Computers & Education* 128 (2019) 13–35.
- [36] J. Li, J. Wang, S. Wang, Y. Zhou, Mobile payment with alipay: An application of extended technology acceptance model, *IEEE Access*.
- [37] M. T. Jan, J. W. de Jager, A. M. Ameziane, N. Sultan, Applying technology acceptance model to investigate the use of smartphone advertising in malaysia, *Journal of Economics and Behavioral Studies* 11 (1 (J)) (2019) 202–210.