

Let's Code: A Kid-friendly Interactive Application Designed to Teach Arabic-speaking Children Text-based Programming

Tahani Almanie¹, Shorog Alqahtani², Albatoul Almuhan³, Shatha Almokali⁴, Shaima Guediri⁵, Reem Alsofayan⁶
Information Systems Department
King Saud University, Riyadh, Saudi Arabia

Abstract—Programming is the cornerstone for the development of all of the technologies we encounter in our daily lives. It also plays an important role in enhancing creativity, problem-solving, and logical thinking. Due to the importance of programming in combination with the shortage of Arabic content that aims to teach children programming, we decided to develop *Let's Code*, an interactive mobile-based application designed for Arabic-speaking children from 8 to 12 years old. The application focuses on the basics of programming such as data types, variables, and control structures using the Python programming language through a simple, attractive, and age-appropriate design. The application presents its users with an interesting storyline that involves a trip to space with “*Labeeb*”, a robot character designed to explain programming concepts to the child throughout the trip. Each planet represents a level in the application and introduces a programming concept through a set of lessons and exercises. The application can be used by educational institutions and parents to teach programming and will provide an opportunity through which Arabic-speaking children can keep up with the development and dissemination of technology.

Keywords—Edutainment; mobile application; interactive application; Arabic; children; programming; coding for kids; Python

I. INTRODUCTION

Programming skills are becoming increasingly important in our digital era. In recent years, many programming education initiatives have arisen in Arab countries. In December 2017, the MiSK foundation launched “Saudi Codes,” an initiative encouraging Saudi students to learn coding skills and participate in programming culture [1]. In October 2017, Mohammed bin Rashid Global Initiatives (MBRGI) launched a huge initiative to teach programming called “One Million Arab Coders.” The program aims to empower one million Arabs with different programming skills via free online courses [2]. These initiatives, both of which are sponsored by governments, are positive signs of the increased recognition of the importance of programming in the Arab world. However, it is important to recognize that any attempt to build a new generation with good programming skills and technological abilities must start with children.

Learning how to code can help children enhance their creativity, logical thinking, and problem-solving skills that can

be applied in both their personal and professional lives. Liao and Bright found that programming increases brain abilities involved in learning and progress [3]; in their study, students with experience in the field of programming achieved 16% higher on cognitive ability tests than students who did not have programming experience.

On this basis, we built an edutainment application that aims to teach Arab children the basics of Python, which is a simple object-oriented programming language. We chose Python because of its ease of use; Python is designed to be more readable and easier to understand than any other scripting language. Python scripts are one-third to one-fifth the size of the equivalent code in C++ or Java [4]. We considered that using Arabic as the primary language in the application would help Arab children understand programming concepts without having to additionally learn the terminology in a foreign language.

In this project, we hope to help Arab children take a step beyond their unproductive use of the Internet by teaching them programming, one of the most powerful and essential tools for digital productivity. We aim to develop their “Digital Fluency”: the ability to design, create, and invent with new media [5], which is far more than the activities in which children are currently engaged such as texting, playing, browsing, and interacting online.

This paper is organizing in the following structure. Section II presents the literature review that discusses related studies and applications. The definition of the problem, the proposed solution, and the project objectives are outlined in Section III. Section IV outlines the application development process used in our project. Section V discusses the limitations and future work, and Section VI concludes the paper.

II. LITERATURE REVIEW

We begin this section with a literature review to present previous studies that discuss topics related to the concept addressed in our project. Next, we provide a comparison of similar applications based on differing features. Finally, we present a discussion that summarizes the results and outcomes.

A. Related Studies

Teaching programming is challenging, especially when the learners are children [6], since programming concepts are difficult to understand and explain. In this literature review, we

This research is fully sponsored by the Deanship of Scientific Research at King Saud University through the Undergraduate Research Support Program, project no. (URSP – 4 –19-89).

argue that even though programming is considered a difficult skill to learn, success in teaching children programming depends on the ways in which programming concepts are introduced.

In their 2011 study [7], Saeli et al. concluded that difficulties encountered in teaching students programming can be overcome by teaching a simple programming language such as Python or Scratch. In addition, a study conducted by Kalelioğlu and Gülbahar [6] in which the authors applied different methodologies to teach students programming found that teaching programming to fifth grade students using Scratch as a simple environment enhanced the participants' "self-confidence in their problem-solving abilities" and improved their desire to continue learning programming.

Difficulties in teaching children programming can also be overcome through the use of educational games. Singh et al. [8] found that educational games are the best method for teaching children programming because games grab their attention and provide an interesting, interactive environment. The authors provided a prototype of a simple 2D game called CQuest that aims to introduce children to basic programming concepts in a format that includes a storyline, challenges, rules, and a reward system. The game received positive feedback after being used by children and school teachers.

Another notable project is "The Game Design Studio," which aims to teach children to design games for other children [9]. This involves transforming the classroom into a game design studio for six months in order to teach students programming and help them design simple games to teach other children about fractions. Kafai argues that one effective way of teaching children to code is through letting them program their own game; she provided samples of children's work in her article.

Robots and Arduino technologies can also be used to introduce programming concepts to children in an interactive manner. Dash and dot, Bee-bot, and CodeaPillar are examples of robots designed to teach children computer science concepts [10].

It is worth mentioning that a huge global initiative called the Hour of Code aims to reach millions of students all over the world to teach them coding for free [11]. The initiative provides one-hour online coding tutorials in over 45 languages targeting different age groups, from children to the elderly. This global initiative has served over 600 million students to date.

B. Related Applications

- Kodable

This is an iOS application that is also available as a website. The application aims to provide a basic curriculum in the form of fun lessons in English to teach the basics of programming in a manner that is suitable for children aged 4–11 years [12]. This application also includes simple games that enable children to learn the basics of the JavaScript language. With a paid subscription, the application provides simplified reports that allow parents and teachers to see how their children are progressing.

- Code Karts

This is a game application available for both iOS and Android. It aims to teach children the fundamentals of coding by allowing them to acquire immediate understanding through a sequence of logical puzzles in a race game [13]. The game targets young children from age four upwards and it does not require any reading skills. Children guide race cars around a path with drag-and-drop code.

- SoloLearn

This is a programming community available on different platforms. It provides courses in various programming languages such as Java, Python, and C++ as well as tutorials and quizzes and gives learners the opportunity to challenge each other [14]. The program allows learners to discuss different issues, see each other's levels and progress, and to share code.

- Mimo

This is an educational iOS application that targets learners over 4 years old. The application teaches beginners the basics of programming through interactive tutorials and quizzes; each tutorial starts with a basic explainer and moves into an interactive question-and-answer section followed by feedback [15]. Mimo helps users learn to develop a website, build an iOS app, develop a game, and become an ethical hacker. It also teaches users to code in many programming languages.

C. Discussion

In the studies that we discussed in the literature review, researchers presented successful examples of teaching programming to children. These studies support our claim that the way programming is presented to children is highly important. We found that most researchers agreed that the best way to teach programming to children is through the use of simple programming languages and interactive environments that attract children to learn complicated concepts. Play and gaming are the most frequently used and successful methods for introducing programming to children. The aforementioned studies support the concept we employ in our project, which relies on introducing the foundations of programming to children in the form of an entertaining application using a simple programming language.

The similar applications that we outlined above helped us understand the features utilized in most successful applications and allowed us to consider the features we should employ to develop a more professional application. It also assisted us in creating an initial prototype of the structure for our application. Table I presents a comparison of the *Let's Code* application with the similar applications described above based on specific features. It is clear that none of the aforementioned applications support the Arabic language, and most of them require a paid subscription.

III. PROBLEM DEFINITION

Despite the importance of digital technologies and the necessity of encouraging children to learn the basics of programming for their future success, it is difficult to attract children to participate in programming education, as they tend

to perceive programming as a difficult skill [16]. Although there are several applications that aim to teach children basic programming concepts, none of them are directed toward teaching Arabic-speaking children text-based coding.

Let's code is our proposed solution. It is an interactive edutainment mobile-based application designed to teach programming to Arabic-speaking children. The application aims to teach the basics of programming in an easy, age-appropriate, and interesting way using the Python language. As Python is one of the simplest programming languages, it is a good place to start when introducing programming concepts. The application content is provided in Arabic to assist Arabic-speaking children in understanding the basics of coding. This aims to solve the problem of the language barrier that currently deters many people from learning programming. Through the development of the *Let's code* application, we are attempting to achieve the following objectives:

- Encourage children to learn programming at an early age.
- Enable Arab children to learn and understand programming concepts without requiring fluency in foreign languages.
- Integrate entertainment with education in order to motivate children to spend more time learning.
- Enhance children’s problem-solving and logical thinking skills.
- Provide a creative space for children to think, analyze, and explore.
- Help parents and educational organizations support children in learning programming.
- Enrich Arabic content in the technical field.

IV. APPLICATION DEVELOPMENT

This section describes the process through which we developed the application, starting with the information gathering techniques used in the study and proceeding to a detailed description of the system design, user interfaces, and system implementation. Finally, we present our evaluation and testing process.

A. Information Gathering

First, we conducted interviews with children aged between 8 and 12 years old on September 28, 2018. The aim was to gather information and understand children’s perspectives regarding education and entertainment that utilize modern technology in general and the concept of programming in particular in order to improve the project’s outcome. Fortunately, we became involved with an initiative concerned with programming that was presented by the MISK foundation, known as “Saudi Codes” [1]. The Saudi Codes initiative encourages young people to learn the basics of programming in an easy and fun environment.

Second, we conducted an online survey that received 219 responses. The target population was Arabic-speaking parents. The results of the survey indicated that 55% of Arab children

spend more than two hours daily using phones, tablets, or computers. More specifically, 90% of the time children spend on those devices is on games and entertaining content while 10% is spent on educational content. We also found that most respondents think that it is important to teach children logical thinking and problem-solving skills at an early age, and about 68% of them had tried to teach their children a certain skill using technology. However, 46% of the respondents believed that foreign languages can be an obstacle to their children learning new skills. On the other hand, we found that about 41% of respondents were interested in programming, although only 8% of the respondents tried to teach programming to their children in different ways such as buying games that teach programming or joining a coding website. About 68% of those who had never tried to teach their children to program stated that they did not know how to teach, 27% stated that there is a lack of content specialized in teaching programming, and 33% of them believed that there is a lack of awareness of the significance of programming. Finally, the respondents mentioned features for educational applications that would attract their children to spend more time learning, such as attractive interfaces, usability, simplicity in explaining information, providing multiple media, and a motivating reward system. We took these responses into consideration while designing our application.

Overall, the interviews and questionnaires we conducted indicated that children spend most of the time on their devices on entertaining content; this should be taken into consideration when developing an application for children. Moreover, we found that many Arab children have difficulty understanding and interacting with English content, which encouraged us to provide the application in Arabic to help them learn programming without struggling with a foreign language. We also considered all of the respondents’ suggestions in order to provide a better solution to increase the awareness and practice of programming.

TABLE I. COMPARISON WITH RELATED APPLICATIONS

| Features\App | <i>Kodable</i> | <i>Code Karts</i> | <i>SoloLearn</i> | <i>Mimo</i> | <i>Let's Code</i> |
|--|----------------|-------------------|------------------|-------------|-------------------|
| Support for Arabic language | ✗ | ✗ | ✗ | ✗ | ✓ |
| Targets children | ✓ | ✓ | ✗ | ✓ | ✓ |
| Designed as an edutainment application | ✓ | ✓ | ✗ | ✓ | ✓ |
| Free of charge | ✗ | ✗ | ✓ | ✗ | ✓ |
| Teach Python | ✗ | ✗ | ✓ | ✓ | ✓ |
| Rewards system | ✓ | ✓ | ✓ | ✓ | ✓ |
| Allows results to be shared | ✓ | ✗ | ✓ | ✓ | ✓ |
| Text-based coding | ✗ | ✗ | ✓ | ✓ | ✓ |

B. System Design

Let's Code is designed as an edutainment application that aims to teach Arab children to code. The application allows children to learn the basics of programming in the Python programming language using a simulated editor for Python. The target audience is users between the ages of 8 and 12 years old. We also target educational institutions that provide educational programs to children of this age. The application is designed for Arabic-speaking children, since the language used in the application is Arabic. While all explanations of programming concepts are provided in Arabic, it is helpful for users to understand the basics of English, which is used in programming syntax. The content may include simple addition and subtraction operations, so users should have a familiarity with basic mathematics. Otherwise, no special knowledge is needed.

The design of the application is important for attracting our target audience. We chose to use a "space" theme for the application and a robot character called "Labeeb" designed to guide children throughout the application. The application's storyline begins with Labeeb looking for recently discovered diamonds in the Milky Way. The child must go on an adventure with the robot, crossing multiple planets in order to find and collect these diamonds.

At the start of the game, the child moves through a series of interfaces that present audio explanations of the basic concepts of programming. Next, a home page showing a series of locked planets that the child must unlock to reach the last planet is displayed. Each planet represents a level in the application and introduces one of the programming concepts. Planets contain lessons, interactive exercises within the lessons, and end with a quiz of multiple questions that measures the child's understanding. A "Hint" option is provided with each question to assist the child if needed.

The application applies a reward strategy to motivate children. Once the child finishes a quiz, feedback is given to show which questions have been answered wrong or right and the total score represented by diamonds. If the gained score is enough, the child will be congratulated and able to move to the next planet. Otherwise, the child will be encouraged to repeat the quiz until the minimum score is achieved. The child's progress will be shown throughout the application.

The child will visit four planets providing the following programming concepts:

- 1) General introduction to Python and programming simple functions such as Print
- 2) Variables and data types
- 3) Boolean and logical statements
- 4) Loops

Finally, when the child completes all the levels successfully, Labeeb will take the child to the planet Earth to receive a certificate of completion as a prize. The certificate

will contain the child's name and total score represented by the total diamonds collected throughout the trip. This certificate can be saved or shared through different social applications. Fig. 1 presents a hierarchy diagram that illustrates the screen flow and navigation through the different interfaces of the *Let's Code* application.

C. User Interfaces

This section presents a sample of screens in *Let's Code* to illustrate the user interface design (Fig. 2 to 9).

D. System Implementation

Let's Code was developed using Android Studio platform, which is the official platform for developing Android applications. We used the Java programming language as the main language to implement the application. In addition, Extensible Markup Language (XML) was used to manage the design of the application's interfaces. The application was developed incrementally; we first designed the interfaces of each level, added functions, and then tested the level to ensure that it worked properly before moving to the next level. To test the execution of the application, we used Android Virtual Device (AVD), which is the official Android emulator provided by the Android Studio platform. Additionally, we tested the application on external devices like Samsung Galaxy 9 and Huawei Mate 9 pro. *Let's Code* uses an SQLite database to save the child's information, such as their name, score, progress, and certificate. In addition, the database is used to store questions answers, keep track of the number of lessons passed, and to update the level's status (locked-unlocked). We also used the database to save videos that explain programming concepts. For the audio files, we created a folder within the application folders called raw and for pictures, we used Drawable folder, which also comes within the application files.

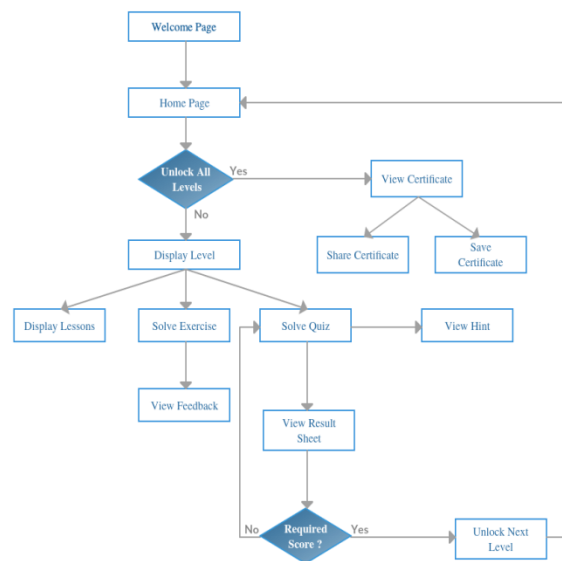


Fig. 1. Screen Flow Diagram.

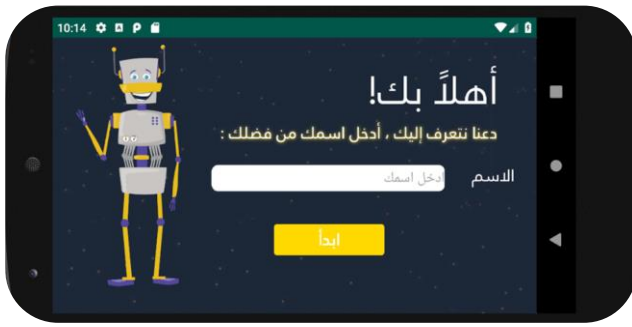


Fig. 2. Initial Screen in which the Child must Enter their Name.

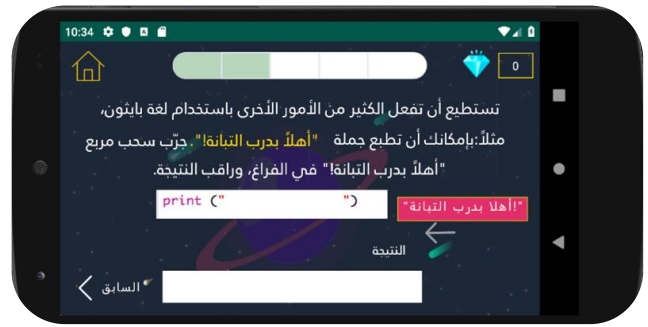


Fig. 6. First Exercise Screen in which the Child Solves the Exercise.



Fig. 3. Welcome Screen in which the Child is Told about the Story used in the Application.



Fig. 7. A Question on the Quiz Screen.

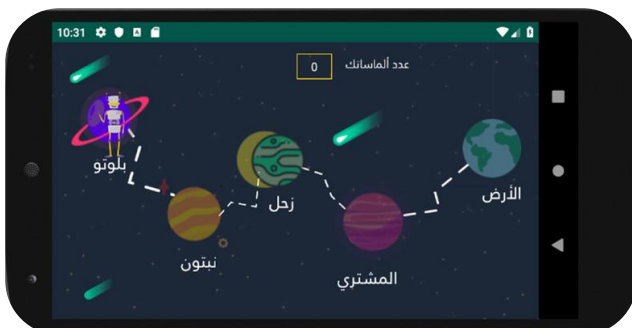


Fig. 4. Home Screen in which each Planet Represents a Level in the Application.



Fig. 8. A Hint Screen that Helps the Child to Solve the Question.



Fig. 5. First Lesson Screen, which Introduces Programming with Python.



Fig. 9. Certificate Screen that Confirms that the Child has Finished all the Levels and Quizzes Successfully.

E. System Evaluation and Testing

In order to detect any errors that might be made during the development phases and to ensure the effective performance of the software application, we applied several test strategies such as unit testing, integration testing, user acceptance testing, and performance testing.

User Acceptance Testing (UAT) is an essential step in which the system is tested by target users to ensure its acceptability and usability. UAT was conducted with six Arab children ranging in age from 8 to 12 years old. Table II displays a sample implementation of the UAT, its results, and user's feedback for *Let's Code*.

TABLE. II. USER ACCEPTANCE TESTING

| User Name | Sara AL Qahtani | | | |
|----------------------|------------------|--------------------------|--|-------------------|
| Task | Number of Errors | msec:min:sec Time Needed | User Feedback | Completion status |
| Enter Name | 0 | 01:05:48 | - | Pass |
| View Level | 0 | 01:17:10 | "What should I do now?" | Pass |
| View Lessons | 0 | 03:22:18 | "Oh Allah! nice" | Pass |
| Solve Exercise | 0 | 01:13:00 | "I did not understand the question" | Pass |
| Solve Quiz | 0 | 04:06:33 | "I feel like my answer is not right" | Pass |
| View Hint | 0 | 00:32:03 | "Oh, now I understand" | Pass |
| Display Result sheet | 0 | 00:26:23 | "Oh my god! The application is nice; it is very smart" | Pass |
| View Certificate | 0 | 00:11:35 | "Wow! I completed it successfully!" (Clapping) | Pass |
| Save Certificate | 0 | 00:23:09 | - | Pass |
| Share Certificate | 0 | 00:56:54 | "I want to send it to my sister; I know her e-mail" | Pass |

Overall, user acceptance testing showed that the application functioned as expected with no errors. It also helped us recognize enhancements that should be made for future work and highlighted contents that were strong or weak. The children were very excited to use the application, asking us "How can we download it on our devices?" It was easy for them to use and interact with the application and they were very happy and proud of the certificate they earned.

V. LIMITATIONS AND FUTURE WORK

As future work, we intend to extend the scope of our application to support a Web application platform and iOS devices. Moreover, we plan to add more levels, lessons, exercises, and quizzes that will contain new and advanced content related to Python. In addition, *Let's Code* is currently limited to Python programming language. Thus, as future work, the application may support more languages such as Java programming language.

VI. CONCLUSION

In this paper, we have provided a comprehensive description of *Let's Code*, is an educational application that enables children to learn coding. Despite the numerous applications in this field, none are directed toward teaching Arabic-speaking children to code. *Let's Code* aims to teach children the fundamentals of text-based programming in an enjoyable manner and intends to enhance children's knowledge and love of learning programming at an early age. Furthermore, it can help educational institutions and parents develop children's skills in logical and analytical thinking, and thus enable them to creatively solve problems and challenges they face. Commercially, the *Let's Code* team seeks to be supported by educational institutions to reach the largest target audience and widely achieve the objectives of the application. Finally, because technology expands and develops every day, we hope that the *Let's Code* project may provide an opportunity through which Arabic-speaking children can keep up with the development and spread of technology.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this

work through the Undergraduate Research Support Program, project no. (URSP – 4 –19-89). The authors also thank the Deanship of Scientific Research and RSSU at King Saud University for their technical support.

REFERENCES

- [1] Saudi Codes, <http://saudicodes.azurewebsites.net>, Microsoft. [Accessed March 17, 2019].
- [2] One Million Arab Coders, <https://mena.udacity.com/one-million-arab-coders/en> [Accessed September 16, 2018].
- [3] Y-K. C. Liao and G. W. Bright, "Effects of computer programming on cognitive outcomes: A meta-analysis," *Journal of educational computing research*, vol. 7, no. 3, pp. 251-268, August 1991.
- [4] M. Lutz, *Learning Python*, 3rd ed. Sebastopol, Calif.: O'Reilly Media, 2013, pp. 3-6.
- [5] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, and Y. Kafai, *Communications of the ACM*, vol. 52, no. 11, pp. 60-67, November 2009.
- [6] F. Kalelioglu and Y. Gulbahar, "The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective," *Informatics in Education*, vol. 13, no. 1, pp. 33-50, 2014.
- [7] M. Saeli, J. Perrenet, W. M. G. Jochems, and B. Zwaneveld, "Teaching programming in secondary school: A pedagogical content knowledge perspective," *Informatics in Education*, vol. 10, no. 1, pp. 73-88, 2011.
- [8] J. Singh, L. L. Wei, M. Shanmugam, S. S. Gunasekaran, and S. K. Dorairaj, "Designing computer games to introduce programming to children," *Proceedings of the 4th International Conference on Information Technology and Multimedia at UNITEN, Malaysia*, 2008.
- [9] Y. Kafai, "Software by kids for kids," *Communications of the ACM*, vol. 39, no. 4, pp. 38-39, 1996.
- [10] F. J. García-Peñalvo, D. Reimann, M. Tuul, A. Rees, and I. Jormanainen, "An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers," *TACCLE3 conference*, 2016.
- [11] Hour of Code, <https://hourofcode.com/us> [Accessed September 26, 2018].
- [12] Kodable, <https://www.kodable.com/> [Accessed October 3, 2018].
- [13] CodeKarts, <https://itunes.apple.com/us/app/codekarts-pre-coding-logic/id1222704761?mt=8> [Accessed September 29, 2018].
- [14] SoloLearn, <https://www.sololearn.com/Courses/> [Accessed September 29, 2018].
- [15] Mimo, <https://www.commonensemedia.org/appreviews/mimo-learn-how-to-code-throughinteractive-tutorials-and-quizzes> [Accessed September 27, 2018].
- [16] A. Idlbi, "Taking kids into programming (contests) with Scratch," *Olympiads in Informatics*, vol. 3, pp. 17-25, 2009.