Learn Your Way through Functions

(SOCIAL OGMENTED - TEAM 3)

Ajit Ramesh Mali
School of Computing, Informatics & Decision System
Arizona State University
armali@asu.edu

Rashmi Hari Prasad
School of Computing, Informatics & Decision System
Arizona State University
frashmih@asu.edu

Chandni Shrivastava
School of Computing, Informatics & Decision System
Arizona State University
cshrivas@asu.edu

Prashant Gonarkar
School of Computing, Informatics & Decision System
Arizona State University
pgonarka@asu.edu

Sagar Patni
School of Computing, Informatics & Decision System
Arizona State University
shpatni@asu.edu

1. INTRODUCTION

The features of Augmented Reality (AR) has always been a wonderment to both users and developers. An AR system can basically supplement real world object visible to the naked eye to coexist with the virtual objects created on the screen on the same space provided on the screen, which maybe a cellphone, tablet, laptop or even as big as a television screen. The three main features an AR system should consist of include - (1) capability to combine virtual world objects with real world objects in a real environment, (2) continuous interaction between both the objects existing in the same space and (3) can add and register the real world object to communicate effectively with the virtual objects on the screen [1].

AR system are now being developed and used extensively to understand how objects similar to reality can look in the real world, for example, a civil engineer can now check how a building might look at a specific place in the city, by just creating the building as a virtual object and placing it in the real world position before the start of the construction. This can help in improvement, understanding how it can fit in, and also get feedback from the builders. Also, AR has been a great part of video and cellphone games since a decade. It helps in interactively communicating with the user to learn new skills, educate the user based on the game usability or merely for fun and as a stress buster. It communicates with the virtual objects or a "avatar" of the user on the screen, giving the feel to the user as to experience the game attributes in reality [2].

The interactive and extensive features of AR games can help teach advanced skills to students and learners such as interpretation, thinking on multiple levels, problem-solving, information management, teamwork, flexibility etc.,[3]. Here, the system focuses on implementing a simple social puzzle that helps its users learn various basic programming concepts. Users can learn and understand object-oriented programming concepts such as different types of functions, recursion along with data structures such as linked lists, trees, graphs in a better way through an interactive game. In addition to that, users are free to create their own functions.

The user interface of the application has a simple navigational layout such that a first-time user of the application would not require prior knowledge of the system and it is easy for any novice programmer or an expert to understand the operations of the game. Once the user logs into the application, he encounters various levels that test his logical abilities in order to successfully solve them. User is free to take as much time as he wants on a level and can attempt a single level multiple times. Moreover, if at some point, user feels that he is stuck on a level and is unable to crack it, then the system takes care of this hurdle by suggesting the user with hints based on the user's interactions with the system earlier in the game. Additionally, our application also tracks user's progress and in order to help him learn better, compares his progress with other users; thereby providing a fun way to learn concepts ranging from simple to complicated.

2. MOTIVATION

The motivation for this project comes from the fact that with rapid advances in technologies these days, it has become of paramount importance that younger generation should have basic knowledge of computer science. However, most of the younger generation has little to no knowledge of simple basic programming concepts. It could seem a behemoth domain to a person who is starting to learn the basic concepts in computer science and programming. So where to start? The internet has a huge collection of resources and a person might get confused with them. Moreover, it is always more cognitive to combine textual content with visual content. This helps in understanding and learning the concepts faster; just a like a kid learning different words and objects with figures.

To design a system that could help the users learn programming concepts in a visual manner, we undertook this project to build a platform using Unity3D and Augmented Reality. The focus was to allow the user to navigate its way through the system in the manner of data structures traversals. It is pretty straightforward to understand and learn the vital parts of functional programming in fun way. It is for certain that such kind of system is believed to be helpful not only to novice programmers but also to experts who may want to refresh their concepts, or better yet, contribute towards enhancing this learning platform.

3. METHODOLOGY

The design of the application is not complex, so it's easier for new users to interact with the system to make better use of it. The combined concepts of augmented reality along with navigational features to make the user move around the screen, run functions, complete levels and also compete with his friends and other users of the system across the domain, make it an adaptive bundle.

At first, when the user opens the application, he will see a login window. User can choose username and password of his choice and register in the system. User is also asked to provide additional biographic information such as his age, gender, his email address, his level of knowledge in computer science which will later be used to give him feedback on his performance. The idea here is to make each user unique to the system to implement the aspects of socially communicating with other users. There is also a basic guided tutorial to help the user understand how he can run the application. The user interface is designed considering that this application will also be used by children in preliminary school.

For every level, the screen consists of two aspects: the puzzle to be solved on the left side and the commands to be provided to the user on the right. The right side of the screen includes various options that the user can use to interact with the puzzle. It includes well-known programming functions like nested functions and recursion. User can use these 'commands' with any other commands to create his own functions.

Based on a user's performance on a particular level, the system shows how the user has fared as compared to other users with the help of visualizations. The application also gives a ranking of the user describing where he stands in the competition amongst all other users of the system. We used the following computation to find the rank of a user:

$$\sum_{i=0}^{number\ of\ levels} \frac{number\ of\ attempts\ to\ succeed}{total\ number\ of\ attempts * (optimal\ number\ of\ attempts - user's\ optimal\ attempt - 1)}$$

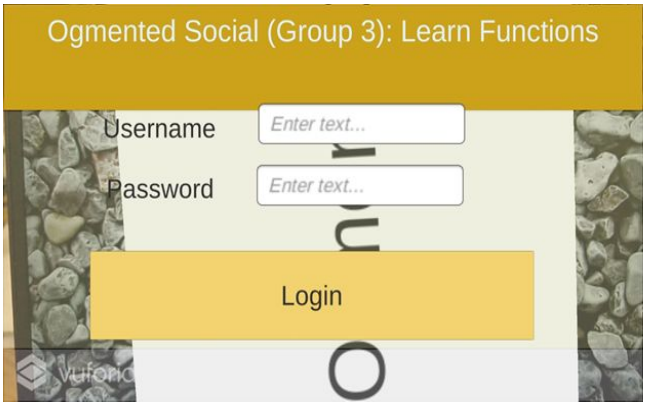
If a user feels that he is stuck on a level and can't solve it, then he can take help of our application's recommendation feature. When a user is stuck, he can simply choose to get hints. Hints are generated using collaborative filtering based on other users' interaction with the system for the same level. The system finds the most optimal way of solving a level and suggests it to the user if he is unable to complete the particular level. Additionally, user can choose to skip the level and come back later with a fresh enthusiasm to solve it.

The system gives the user, total freedom to interact with the system the way he wants to, which defines the flexibility of the system and its adjustability to the current and the future users. For the very first users of the system, we have categorized levels into three different types namely easy, medium and hard; based on its difficulty as determined by the level of knowledge it requires to crack it. Each level is associated with a certain 'threshold number' which represents the maximum number of attempts it should take to solve the level. If a user is unable to solve the level within the threshold number of attempts, then he can opt to get hints. The interesting thing about this threshold number is that it will not be constant for that level throughout the game and it will keep changing adaptively to the user's performance. If many users find the level easier then it is anticipated, the threshold number shall decrease; otherwise it is formulated to increase, thereby making the system adaptive for the user.

Since the system is highly focused on user's need to find learning more interesting, easy and fun, it is designed to adhere to their knowledge and skill levels. Users will find it interesting and challenging at the same time. It has been found that children learn much better through interactive systems and in a challenging environment. Our systems make these two essential things available to the user. Also, through the feedbacks, ranking and visualization, user will know exactly which parts of object-oriented programming he needs to work harder for and where his standing is as compared to other users; thus making him/her aware about the things that he needs to improve.

4. SYSTEM DESIGN & IMPLEMENTATION

The social augmented reality system implemented, is an adaptive application used to teach young students and learners basic programming concepts through an interactive approach using the features of augmented reality, adaptive learning, collaborative filtering, recommendation ideology and with a social blend.



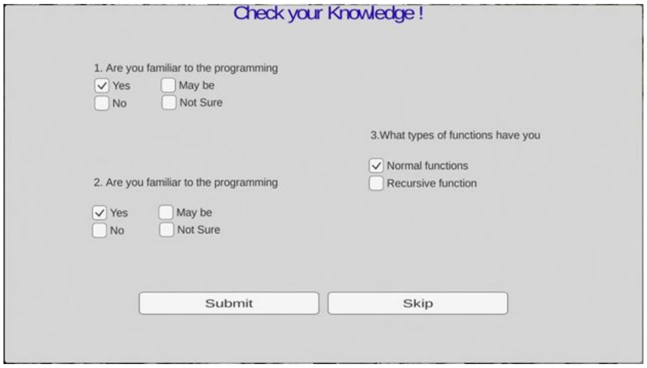
Ogmented Social (Group 3): Learn Functions

Username Enter text...

Password Enter text...

FIGURE - 1: Login or register

In order to accurately predict recommendations and make our system adaptive, it is necessary to collect some basic information about the user. As described before, the user can log into the system or if not can create an account to register with the system. While creating an account, the user is prompted to answer basic biographical information such as adding his account name, gender, email address, his location along with his level of knowledge of computer science and programming concepts as shown in figures 1 and 2. This basic information collected explicitly is then used to customize the order of levels that will appear for that specific user in our system.



Check your Knowledge !

1. Are you familiar to the programming

☑ Yes □ May be
□ No □ Not Sure

2. Are you familiar to the programming

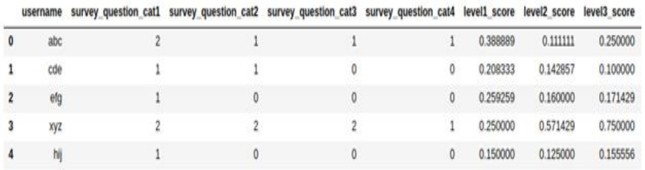
☑ Yes □ May be
□ No □ Not Sure

3.What types of functions have you

☑ Normal functions
□ Recursive function

FIGURE - 2: Understanding user's knowledge

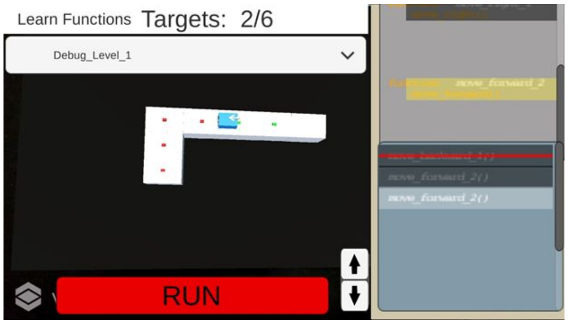
In addition to this, we also collect information about the user implicitly by the way he interacts with our system. This implicit information about the user contains information mainly the number of attempts the user is taking to the levels. This information, along with the explicitly collected information is then used to create a user profile. Based on this information that we have obtained from the user, we created an open user model. User modeling is used to interact with the user and build a conceptual model of the user as shown in snapshot in figure 3 below. It mainly resides on our backend server which communicates with our application to collect user statistics.



	username	survey_question_cat1	survey_question_cat2	survey_question_cat3	survey_question_cat4	level1_score	level2_score	level3_score
0	abc	2	1	1	1	0.388889	0.111111	0.250000
1	cde	1	1	0	0	0.208333	0.142857	0.100000
2	efg	1	0	0	0	0.259259	0.160000	0.171429
3	xyz	2	2	2	1	0.250000	0.571429	0.750000
4	hij	1	0	0	0	0.150000	0.125000	0.155556

FIGURE - 3: Open social user model

In open user modeling, the user gets to control the data that a system can use about the user. It will keep the user informed about the data being collected implicitly and explicitly. Three aspects of building the open user model are: Transparency, controllability and visualization. For transparency, the implicit and explicit data collection methodologies is described in a privacy policy document, which the user is asked to sign while setting up the application. In this way, the application keeps the user informed as to which data is been captured for giving personalized experience. Through this approach the data used is transparent and the user is well aware and the system has consent from the user to use the information which making the model more transparent. For controllability, during the initial setup of the application we prompt a dialogue which asks for required permissions like the use of motion capture, location, contacts, camera etc. Through this option user is able to control the data that is been collected and analyzed. If user is not fine with allowing one of the medium, he can easily deny the access of that medium.



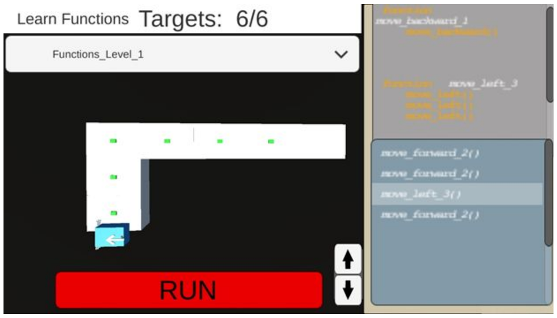
```
function    move_forward_2
    move_forward()
```

```
move_backward_1()
move_forward_2()
move_forward_2()
```

FIGURE - 4: Snapshot showing debug level 1

The figure - 4 above shows one of the concept in programming - debugging and a level in the game related to it. Here there are 6 targets and all the user needs to do is to use functions on the left hand side to move the box on the user to reach all 6 targets by running a series of commands such as add, move_left, move_right, delete etc., These commands facilitate easy usage of the app to navigate across the screen and the user to complete that particular level.

Each user has multiple levels to complete, to learn a single programming concept, such as debugging, data structures, recursion, functions etc., Figure - 5 below shows another programming concept to be learnt, functions and level - 1 of the game. The snapshot shows the successful completion of the level where the user has reached all the 6 targets on the screen. The target turns green from red once the user completes it. One advantage the user has is, that he can either use the functions provided to him by the application or he can build and run custom built functions that the user can create on his own. This helps the user understand both the simple inbuilt functions and also write his own functions with a different layout.



```
function
move_backward_1
    move_backward()

function    move_left_3
    move_left()
    move_left()
    move_left()
```

```
move_forward_2()
move_forward_2()
move_left_3()
move_forward_2()
```

FIGURE - 5: Snapshot of functions - level 2

If a user is unable to understand the next move, or if he is stuck in a single level for a longer time, the system provided him with hints in order to solve the level faster. The application provides recommendations to the user to complete a level faster, based on the moves and improvements of users who have completed the same level faster and in less number of attempts.

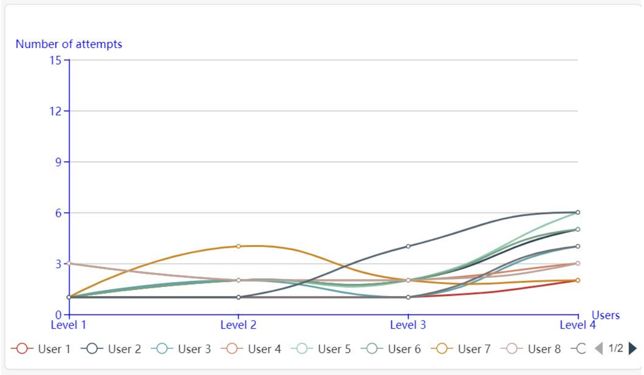


FIGURE - 6: Line graph showing # attempts of 10 users

For visualization, we are providing the user with graphs and charts of his performance for each and every level as well as for his performance compared with the other users. For this we have created a web dashboard where the user's live actions are fed in a recommendation engine. The dashboard consists of 3 visualizations shown in figures 6, 7 and 8. Figure 6 shows a line graph of the number of attempts of 10 users for four different levels. It shows that most users take less than 3 attempts to clear level 1. Figure 7 shows the ranks of the users and each user can check his rank when compared to the other users in the system. The scores are based on the average score considering it to be 10 for each level and reducing 1 point for a failed attempt till the score reaches 5.

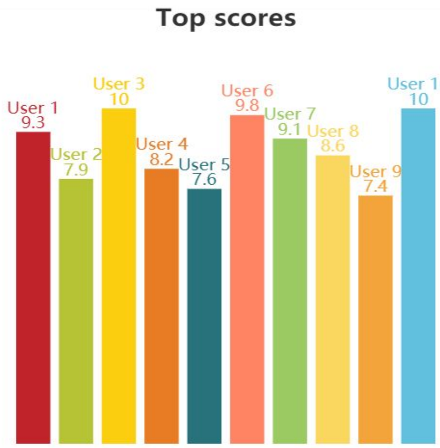


FIGURE - 7: Ranking system of users

Figure 8 shown below is a bar graph represent the users individual progress in the game. It shows the number of successful and failed attempts of the user which shows how a user is improving with each level.

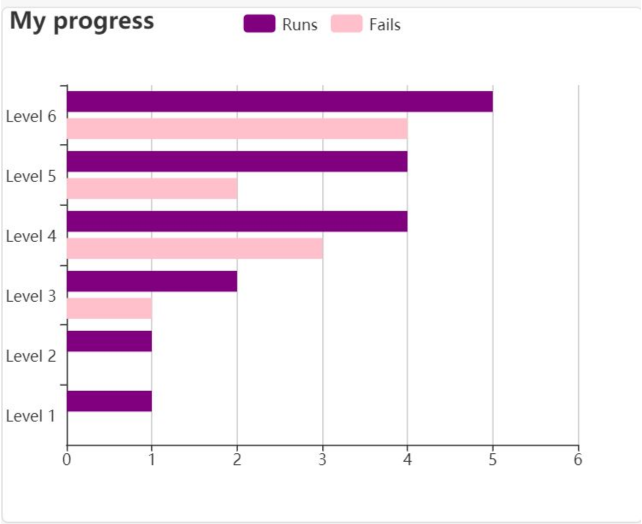


FIGURE - 8: User’s individual progress

5. EVALUATION PLAN

The project scheduled to begin in August 2018 was concluded in November 2018. We performed both formative and summative evaluation of our project. For formative evaluation we concentrated on the planning and implementation stages of our project. Whereas for summative evaluation we focused on the outcomes of the project. Upon completion of each stage of the project we reviewed the progress and determined if we have successfully met all the requirements set for that stage of the project.

5.1 Project Planning

In this phase, we started with learning and understanding the platform of Unity and Vuforia provided to us by our Professor and the research guide. After comprehensive research of the application and its use cases, we formulated a use case of our own to build an adaptive system. We tried to incorporate most of the concepts learnt in our lectures and make use of them effectively in our application.

Our team came up with several questions during the design discussions and we tried to address those in our implementation.

- who would be the users of our system?
- what problem solving techniques are we going to use?
- what should be the target of the user’s problem solving and learning goal?
- how many difficulty levels should be included?
- how many attempts should be allowed for the user?
- how much time can a user spend in every level?
- what kind of explicit and implicit feedbacks should be captured by the system?
- how to formulate the comparison component between users?
- what all feedbacks and suggestions should the system give to the user?
- what data should these recommendations be based on?
- how many users’ data should be collected?
- how to guide the users in the system?

As we wanted to build a system for novice learners, we started with designing the user interface layout that covers most of the navigational features of a common application and is also simple enough for the user to understand. Then we decided different modules for our system namely - dashboard, tutorial sections, puzzle view, function controls, user statistics visualization, user model. Once we had a rough layout of the system, we went ahead with developing each of these one-by-one.

5.2 Project Implementation

In the project implementation phase, we started by building all the individual components of the system that were determined in the planning phase. Each of them were tested one-by-one with other modules that were completed along the development process. Once all the components were implemented, we started integrating them. Ou next step was to build and collect data for the visualizations in our open user model. We decided on a specific number of users, the number of times they would use the system etc. to get a comprehensive data that would show appropriate statistics. Then we designed our user model interface and constructed the graphs that would show the user content. All the individual components developed initially were tested with dummy data and worked without any issues with our data dump. Since we were using the Unity and Vuforia platform, we tried to implement all our application’s features on its layout itself. We went ahead with incorporating the visualizations in the application but faced difficulties while integrating it with our application’s data. We decided on building a separate interface for the visualizations that would work using a backend server which will collect data from our application dynamically. The final integration worked well and was close to our initially proposed model.

Here is a brief structure of the implementation process:

- develop the puzzle view layout
- build navigation panels for one level
- construct functions for navigation
- implement algorithms for different function navigations & user ranking
- develop system panels for functions and results
- design dashboard for system and user information
- data collection and processing
- design graphs for various data visualization
- design open user model with these statistics
- integrate the application on android device with the backend server
- test system execution with data population
- test correct functioning of all the modules
- test correct population of data

These are the application development metrics covered :

- Data collection methods: The data used in this application is the data of each user. It is collected both implicitly by understanding the progress of the user with the game and explicitly during user registration by filling out a form. The data collected is transparent to the user and he/she can modify or refuse the consent for the utilization of the data to be used for recommendations.
- Data source and frequency of collection: The data collected of the users is stored on the database on the server. It is updated dynamically whenever a user plays the game. The data from the database is used to update the related stats shown on the dashboard and the corresponding visualization.
- Creating and running the application: The application is built on a library module provided by the professor which uses Unity and Vuforia. It is run on the system and the apk is built using the Vuforia and run on the phone.
- Interactivity among the users: The open social user model of the system shows the different users who are using the system. Each user profile contains information of the user and the user has complete control of it. A recommendation system is used provide recommendations to the user so as to perform better and collaborative filtering is used to provide hints in each level. The level of difficulty for a user is based on the implicit and explicit data of the user.
- Performance outputs of the system: The performance of the system is based on the performance of the users. The user performance is represented on the dashboard in the form of charts. It helps compare a user with one another and also help measure individual performance.

5.3 Outcome Evaluation

In the final phase, we tested and evaluated whether all the components of the application were working smoothly. To verify the design from the user's perspective (beginner and expert), we took the help of our peers and classmates to review our application in terms of all the functional and non-functional aspects. After detail exploration of the dashboard by many peers, we found that our design met almost all the expectations of a first time user, while also covering most of the system designs taught to us by our Professor in class.

6. DISCUSSIONS & FUTURE WORK

Building an application on Unity and Vuforia was a great learning experience for us. As a prototype, we kept the design simple, in a way that is achieved on time and covers all the aspects we had proposed. Our application mainly helped the users learn functional programming concepts by way of using various functions. Given more time, this can be extended further to include more functional and object-oriented concepts such as abstraction, polymorphism, inheritance etc. The system can be enhanced further by adding more types of puzzles to choose from. To implement all these, we would need not just touch input feedbacks, but various others like visual, speech etc. These explicit feedbacks can be used to make the system more interactive and suiting the user's preferences. The dashboard and the open user model can be enhanced further to include more detailed statistics of the users, which can even be made interactive. By adopting Unity features, this can be implemented and integrated with the application, rather than on a separate interface.

7. ACKNOWLEDGMENTS

We thank our professor Dr. Sharon Hsiao for giving us the opportunity to work on an Augmented Reality project. It was a great learning experience for us to get a hands on on building 3D applications combined with the concepts we learned about social recommendations in class, on a platform like Augmented Reality. We also thank our teaching assistant Yihan Lu and the research guide Cheng-Yu for giving us the support in making this project work, solving our doubts and being available for all the questions we had.

8. REFERENCES

[1] Azuma, Ronald, et al. Recent advances in augmented reality. NAVAL RESEARCH LAB WASHINGTON DC, 2001.

[2] Luz, Filipe Costa, Vasco Bila, and José Maria Dinis. "Augmented reality for games." Proceedings of the 3rd international conference on Digital Interactive Media in Entertainment and Arts. ACM, 2008.

[3] Schrier, Karen. "Using augmented reality games to teach 21st century skills." ACM SIGGRAPH 2006 Educators program. ACM, 2006.

[4]https://www.thehealthcompass.org/how-to-guides/how-devel op-monitoring-and-evaluation-plan