Development of Word Game Algorithm for Learning Javanese Script

Ozzi Suria

Program Studi Sistem Informasi, Fakultas Teknologi Informasi,
Universitas Mercu Buana Yogyakarta, Yogyakarta 55501, Indonesia
Email: ozzisuria@mercubuana-yogya.ac.id

Abstract—The students consider learning Javanese script to be difficult particularly in distinguishing and memorizing *Carakan*, and memorizing *Sandangan* and *Pasangan* with its writing rules. This work intends to develop a supporting medium for learning Javanese script. The development process is started by defining the game functionalities by using the use-case diagrams, and then, the activity diagram is created to describe the workflow of the game algorithm. The database to support the game is also created and displayed by using the physical data model. Afterward, the game algorithm script is created using JavaScript so that the game can be played through a web browser. There are 27 respondents requested to test the game and to fill in questionnaires about the web application. The results suggest that 100% of respondents agree that the web application is necessary and useful to learn Javanese script. The application provides positive benefit to the users such as students who still need to learn Javanese script in schools with 97% average success rate to run the game

Index Terms—Javanese Script, Word Game, Web Learning, Multimedia Application

I. INTRODUCTION

NOWADAYS, there are many forms of word games. Those are used for language learning such as online games of English proficiency test for young learners in Japan [1], a game based tasks for foreign language instruction [2], online vocabulary games for learning Arabic [3], and an augmented reality mobile learning tool for learning French [4]. Using the word game as a helper tool for language learning can improve learners' vocabulary and motivate them on learning language [5]. It also makes learners interested in engaging more in the learning process [6]. Since word games are proven to be effective in enhancing language learning process, it will be utilized as a medium to learn Javanese script in this research.

Javanese script is a part of Javanese language material learned by students in elementary, junior, and senior high schools. It has been acknowledged as part

of the local language and already included in school curriculum in Central Java, East Java, and Yogyakarta. Hence, it is essential to understand that Javanese script should be learned and preserved as local cultural heritage [7]. Learning process of Javanese script consists of highly repetitive reading and writing. At the beginning of the reading process, students must master the basic twenty scripts called *Aksara Carakan* [8]. After that, they can continue to learn other scripts like *Sandangan, Pasangan, Aksara Wilangan, Aksara Swara, Aksara Rekan*, and *Aksara Murda* [9]. With the traditional method of teaching in the class, Javanese script is considered difficult to learn by students even though they have been learning it since elementary grade [10]. The things that make the learning difficult are:

1) distinguishing and memorizing *Carakan* scripts when reading because they look similar to one another,
2) memorizing *Sandangan* and the rules to write it in words that have è or é sound, or words with syllable *ré, lé, ra, kré,* and others, and
3) memorizing *Pasangan* and the rules to write it in words that have the double consonant in the middle, last, and others.

Referring to those three problems, the objective of this research is to develop Javanese script word game to make students easier and enjoy learning Javanese script. The game is developed in the web platform. Based on the previous research on vocabulary learning [11], one of the benefits of developing an application in a web platform is to make it easy to use for students. Additionally, using web-based multimedia combined with the conventional method of teaching in class is more effective and can increase students' cognitive learning [12]. The utilization of web-based learning application is also more useful compared to a mobile application that makes learner have to deal with the real world and digital world at the same time [13].

The uniqueness of this game is that the question

should be answered by inputting the translation of the word in Javanese script by using a custom keyboard on the screen. This research discusses more the game workflow, the interface, and its algorithm which are used to run the game. Before starting the game, players can select the level and duration. The level and duration preferences are used as a challenge to make the learning process become more effective [14] and also to make players excited to get a higher score before the time runs out [15]. Players interact with the game using custom Javanese script keyboard on the screen. The keyboard uses Javanese script typeface so that players can learn to memorize the letter and get used to Javanese script.

The game algorithm is built using JavaScript language. It works with PHP script and AJAX to communicate with the database server asynchronously. JavaScript is used as a base to develop the algorithm because it is natively supported by the browser [16–18]. It can also be used as a resource for future development because most of app development tools use JavaScript as part of its programming code [19–21].

In general, the algorithm handles all functions to control the game elements such as level, time, scores, question, custom keyboard, inputted answer, and result from the submitted answer. These elements work together simultaneously to run the game and to handle user's interaction with the game. For example, before submitting the answer, users can interact with Javanese script keyboard to input the answer. Afterward, the algorithm checks the submitted answer. If it is correct, the algorithm will tell the time controller to add more time and the score controller to add the score. However, before adding the score, the algorithm checks the level preference to add score based on it.

Compared to the previous research [9], this research presents a different view to make learning Javanese script more interesting. The previous research discussed a game to learn Javanese script by matching a picture with paragraphs which were written in Javanese script. However, it was more focused on analyzing the result of implementing the game and only presented a few details about the game. The game was also played traditionally without using technology. On the other hand, the present research discusses the making of the game algorithm from a technical perspective by using Unified Modelling Language (UML) diagrams to model the system requirement and evaluate the result of system implementation.

This previous research [22] also presented a game which focused on introducing the Javanese script. It was developed using Construct 2 and used Waterfall method to develop the game. However, there was no method described to develop the algorithm inside the game. The algorithm of the game was not written although there was a step to gather the requirement in the first development phase. Compared to the previous research, the current research tries to model the system functionalities and the game algorithm which later can be used in the development process. The UML diagrams are combined with Waterfall method to produce a robust game algorithm. It can be developed later in the future.

The main contribution of this research is to develop a web application and game algorithm for learning Javanese script through a game by using UML diagrams. It models the algorithm functionalities which are presented as use-case and activity diagram and Waterfall method during the development process. This web application is expected to be useful for students who want to keep learning Javanese script and give a positive benefit to them in competing with other students to get the highest score in the game. The performance of the algorithm is tested to make sure that the algorithm can work well during the game.

II. RESEARCH METHOD

The word game uses a word as the object to be guessed, arranged, or sorted based on the predefined rules to play. The Latin alphabets are commonly used in most of the word games. For example, Latin alphabets are used on the word game to learn Italian [23, 24], help users to increase their vocabulary, and improve their lexical accuracy of Italian. In this research, the type of word game is guess-word game. This game is usually played by answering the question using the correct word or spelling in Latin alphabets. In this case, the game is modified by using Javanese script to answer the question that is displayed in Latin alphabets. To support the case, a word game algorithm is required.

A word game algorithm is an algorithm to run the whole process of the game. In this case, it is specifically used to support the Javanese script characters in the word game. In the application, the algorithm works as a part of the main layer to run the whole process of the word game and to support layer to handle the input answer in Javanese script character. Hence, the development of this algorithm supports the Javanese script word game.

The research flow is displayed in Fig. 1. There are three phases: design phase, development phase, and evaluation phase. The first phase focuses on designing the fundamental needs for the system. It starts with analyzing system requirement. This process is used to map the functionalities and data that may be required
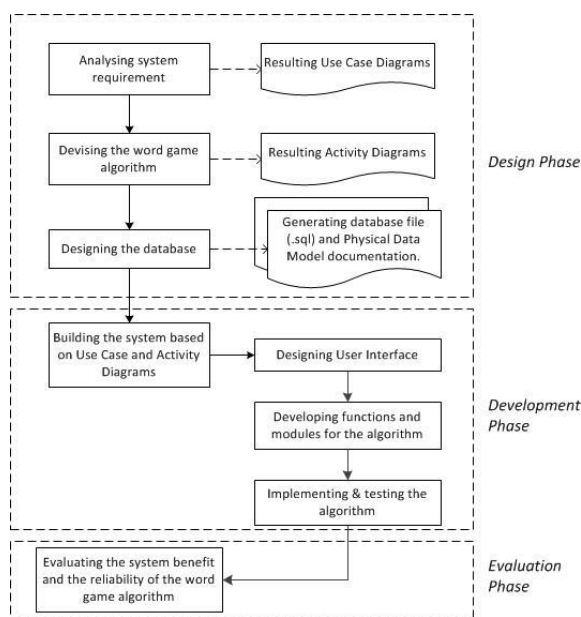
Fig. 1. Flow of research.

for the algorithm. Then, the word game algorithm is devised, and the relational database is designed. Once the first phase is done, the results can be used in the second phase.

The second phase is the development phase. This phase discusses the development process consisting of building the right User Interface (UI), developing functions and modules for the algorithm, and implementing and testing the algorithm. The last phase is the evaluation phase. The evaluation is done to test the reliability of the game algorithm and know the benefit of the system.

The process begins by analyzing the requirement of an application to develop the word game. The first step of the analysis is done by using use-case model to identify the important functionalities of the system at early phase [25]. In this step, use-case diagrams describe the interaction between actors and system based on the system functionality [26–30]. There are two use-case diagrams designed to distinguish interaction between player as an actor with system and system as an actor with itself, which is coded as the algorithm. The second step is creating an activity diagram. The activity diagram is used to model workflow between functions or process [31] in accordance with the functions that are defined in use-case diagram [32]. It has three components, namely users, systems, and interactions [33]. These components can be used to illustrate sequential activities that have parallel activities [34–36]. It is very useful to develop the game algorithm. This activity diagram is a reference to create the algorithm script.
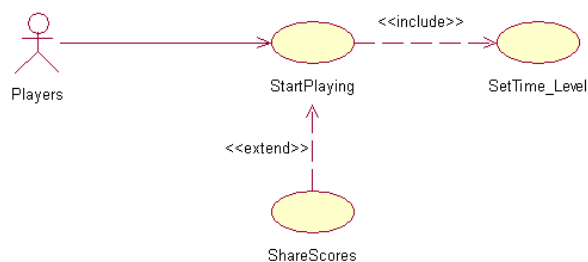

Fig. 2. Use-case diagram of players and system.

The last step is designing the database model to save data for the game elements. For example, there are level options, time options, rules to add bonus time and score, players' information, and playing history.

A. Use-Case Diagram

The use-case diagram for interaction between players as an actor with the system is shown in Fig. 2. There are three defined use-cases.

1) `SetTime_Level`. This use-case describes players' action to set time and level before starting the game. In Fig. 2, this use-case is associated with `StartPlaying` use-case with 'include' stereotype. It means that this use-case must have been done first before use-case of `StartPlaying` is executed.

2) `StartPlaying`. The `StartPlaying` use-case describes players' action to start playing the game until the game finishes. It has two associations with other use-cases. One is already mentioned in the previous point. The other one is associated with `ShareScores` use-case with extend stereotype.

3) `ShareScores`. This use-case describes players' action to share scores after finishing the game. The association between this use-case and `StartPlaying` use-case means that this use-case is optional and players can choose whether to share their scores on social media or not.

As mentioned before, the interaction between systems is displayed using use-case diagram in Fig. 3. The purpose of creating this use-case diagram is to understand the main action that is executed by the system automatically when the game is in progress.

As shown in Fig. 3, there are five defined use-cases.

1) `CountdownTime`. This use-case describes an action to control the countdown timer during the game. Time can also be modified by adding additional time when players' answer is correct. The added time value depends on the selected level preference.
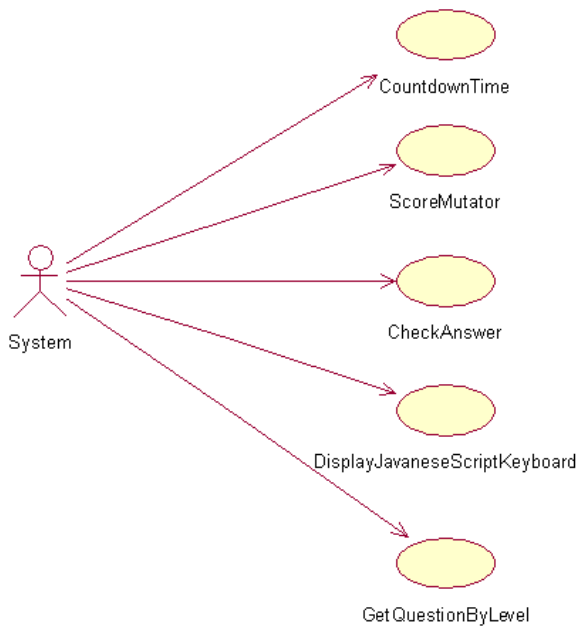
Fig. 3. The use-case diagram of a system with itself.

2) `ScoreMutator`. This use-case illustrates action to modify scores value. The action to add or deduct scores is based on the players' answer. If the answer is correct, some points are added to the scores. Otherwise, some points will be taken from the score. Total points that will be added or taken from the scores depend on the selected level preference.

3) `CheckAnswer`. This use-case is an action to check the submitted answer with the answer key from the database.

4) `DisplayJavaneseScript Keyboard`.
This use-case shows the action to display the Javanese script keyboard and how to process the inputted character in the answer before being checked.

5) `GetQuestionByLevel`. This use-case illustrates action to load questions from database and displays them in the question field to be answered by the players. Retrieved questions are based on selected level preference.

B. Activity Diagram

The activity diagram of the algorithm is shown in Fig. 4. Meanwhile, the workflow in the activity diagram is based on the use-case diagram in Fig. 3.

The execution process is begun after players click the start button. During the execution process, the algorithm has three primary conditions. They are initial condition, playing condition, and final condition. In

initial condition, the system loads level and time preferences. The players have already inputted its before clicking the start button. These data are used by the algorithm to adjust game variables to be played by the players.

In the playing condition, the system starts to enable Javanese script keyboard to be used by players. Before the game is started, the Javanese script keyboard has already been displayed but not in active mode. Then, score controller sets the score to zero. The question controller tells time controller to start counting down.

The counting down is an iterative process which does repeated activities. These activities are as follows.

1) Check the availability of time. This is the main activity that controls the repeated process of next activities. This activity checks the countdown time. If the time has already reached zero, all activities are terminated, and the process is continued to final condition. Otherwise, the process may continue to its next activity. As long as the time has not run out, players may use Javanese script keyboard to input the answer to the question.

2) Get Question by Level Preference & Display Question. This activity retrieves a question from the database. Then, the question is checked. If it is already issued, this activity will be done again. Otherwise, the question may be displayed.

3) Check the submitted answer. The submitted answer from players is not directly matched with the answer key. A function is used to replace specific characters into the text to be matched with the answer key. After the process is done, the answer is checked with the answer key. If it is a correct answer, it will trigger the activity controlled by score controller to add scores and trigger time controller to add more time. If the answer is wrong, it will trigger the score controller to deduct scores. The process is repeatedly done and terminated when the time has run out. The final condition is when the time has run out. The system compiles the final scores and offers players to share it on social media. When players choose to share it, a dialog box opens and asks players' confirmation before posting it to the public.

C. Physical Data Model

This system uses the database to save information such as questions and answer key, player information, level options, time options, level setting, time setting, and playing history. The database is implemented in the system using MySQL. Using a physical data model, the database is represented in Fig. 5
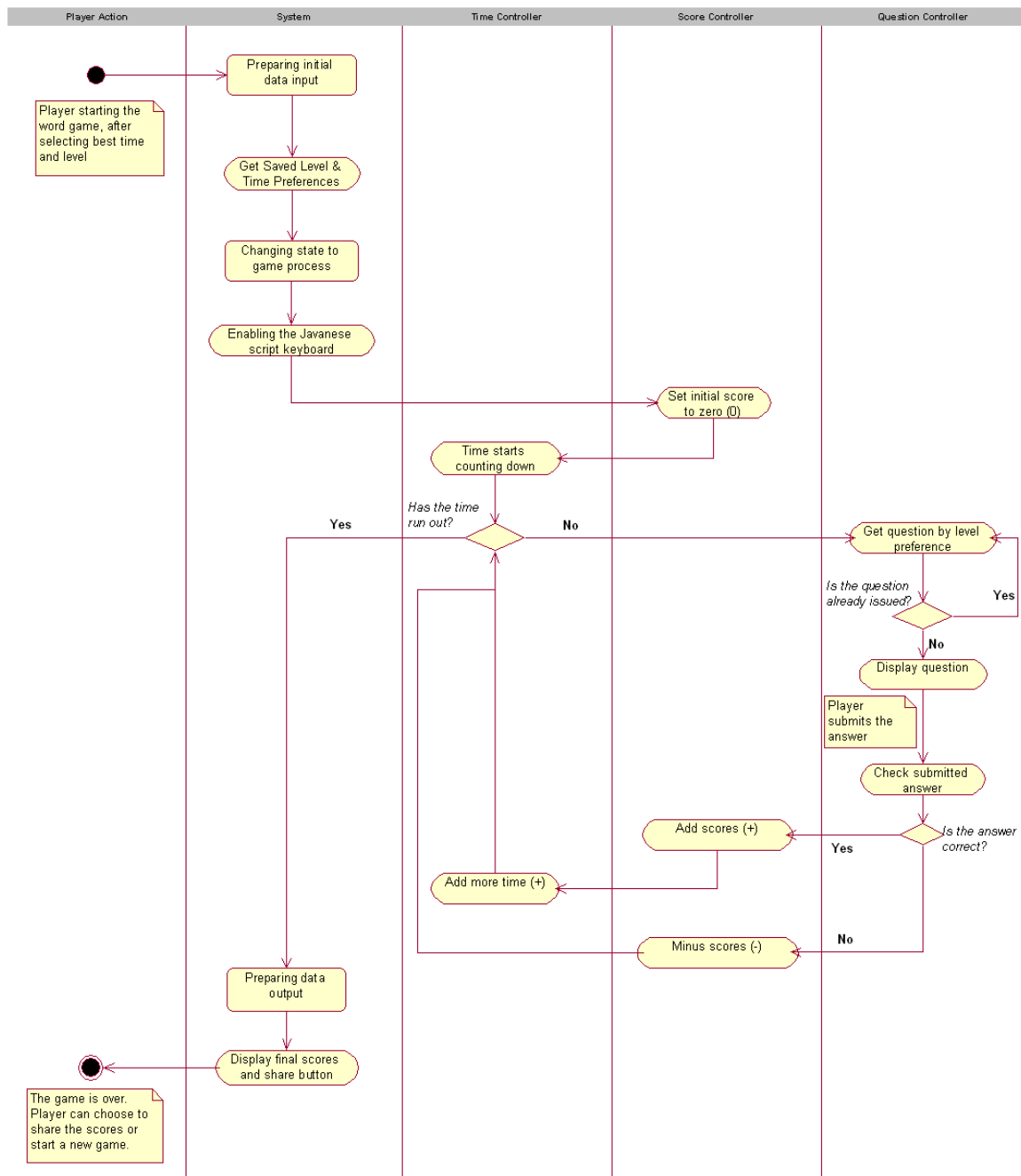
Fig. 4. The activity diagram of word game algorithm.

1) `master_user`. This table is used to save players' record after they have finished playing the game.
2) `master_level`. This table contains level options to be selected by players before they start the word game. For example, the level options are *Carakan, Carakan* with *Sandangan, Carakan,*
Sandangan, and *Pasangan.*
3) `master_time`. This table contains time options to be selected by players before they start the word game. For example, the time options are 15 seconds, 30 seconds, and 60 seconds.
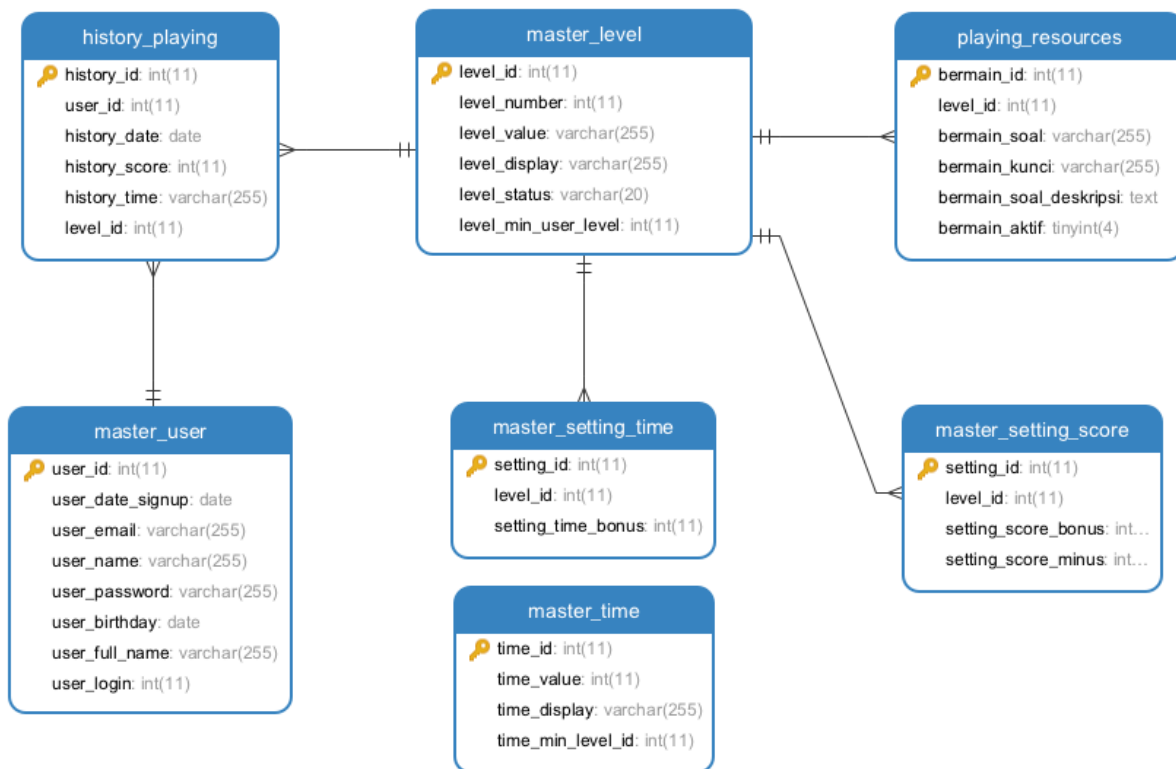4) `master_setting_time`. This table contains bonus time in accordance with data from

Fig. 5. The physical data model of the database system.

master_level respectively. The data are retrieved when players give the right answer and are added to the players' current playing time.

5) master_setting_score. It contains bonus score in accordance with data from master_level. When players give the right answer, bonus score from setting_score_bonus is added to the players' current score. Otherwise, the players' score is deducted with the score from setting_exp_minus.

6) playing_resources. The question data and its answer key are saved in this part. The question data are retrieved before the countdown starts, and it is displayed to players during the game. The players' answers are checked with the answer key whether it is correct or wrong. In Fig. 5, every question data has level_id information. Therefore, questions based on the selected level can be retrieved.

7) history_playing. It is used to save records of every game session from all players. Every record contains the date of play, scores, playing time, and selected level.

These use-case diagram, activity diagram, and physical data model are used as a base to develop algorithm script of the Javanese script word game. The code logic, functions, and its implementation will be explained further.

III. RESULTS AND DISCUSSION

The core system is developed using PHP CodeIgniter framework. The database system uses MySQL engine. Along with developing the core system, the web pages and its UI are designed using HTML, CSS, and JavaScript. Although UI design is more optimized to be viewed in desktop screen than in mobile screen, the development uses responsive design code. Therefore, all pages can still be viewed on the mobile screen properly.

The word game page has more objects to be displayed than other pages. This page with its logic function is designed to be used and viewed in 13-inch screen size with minimum width resolution of 1366 px. The objects are arranged with specific position and size to give players the best user experience when they play the game. Although players can still view and play the game with a screen less than 13-inch, they will not get good experience since the objects will be arranged in vertical position and they will have to deal with page's

scrolling over and over. Therefore, it is recommended to play the game in the device which has at least 13-inch screen. Figure 6 shows the display of the game page in 13-inch screen. Then, Fig. 7 illustrates the display on the mobile screen.

The game development consists of several stages. Those are designing the UI as shown in Fig. 6, creating custom keyboard as input tool to produce a letter in Javanese script, and creating the script to control and run the game.

A. User Interface

The game UI with its elements is as shown in Fig. 6. The screen is divided into two parts. The left part about 16.7% of the screen is used to place level and time options, start button, time controller, and scores. The right part about 83.3% of the screen is used to display the question box, answer box, and keyboard buttons from top to bottom. The right part needs a bigger size than the left because 74 letters of Javanese script should be displayed on the keyboard area. To start the game, players must go to the left side to select the preferred option for level and time. Next, the start button will appear, and players may start playing. Players may see the question and answer box on the right side with an order from top to bottom. Reference [37] showed that about 79% of eye movement started from left to right and continued from top to bottom. Hence, the left, right, top, bottom order is used in this page design. The complete look of UI from the left side is shown in Fig. 8.

B. Javanese Script Keyboard

As shown in Fig. 6, there are three kinds of Javanese script displayed on the page as a keyboard. They are *Carakan, Pasangan,* and *Sandangan*. A specific CSS class is placed in keyboard container code to transform every character's typeface in button into Javanese script typeface named Hanacaraka. The sample of Hanacaraka typeface used in this system is shown in Fig. 9.

In Fig. 9, each character represents the Javanese script. For example, the keyboard is already styled using Hanacaraka typeface. If the inputted character is 'a', the letter will become ꦲ, if the inputted character is 's', the letter will become ꦱ and so on. The mapping of Javanese script with UTF-8 character is listed in Tables I and II. Table I lists all UTF-8 characters for *Carakan* and *Pasangan* and its Javanese script typeface respectively.

As shown in Table II, there are some special characters from UTF-8 used to display some Javanese script letters. The problem with these particular characters

TABLE I
LIST OF JAVANESE SCRIPT OF CARAKAN AND PASANGAN WITH UTF-8 CHARACTER.

Letter	Carakan		Sandangan	
	UTF-8 Char.	Javanese Script	UTF-8 Char.	Javanese Script
ha	a	ꦲ	H	
na	n	ꦤ	N	
ca	c	ꦕ	C	
ra	r	ꦫ	R	
ka	k	ꦏ	K	
da	f	ꦢ	F	
ta	t	ꦠ	T	
sa	s	ꦱ	S	
wa	w	ꦮ	W	
la	l	ꦭ	L	
pa	p	ꦥ	P	
dha	d	ꦝ	D	
ja	j	ꦗ	J	
ya	y	ꦪ	Y	
nya	v	ꦚ	V	
ma	m	ꦩ	M	
ga	g	ꦒ	G	
ba	b	ꦧ	B	
tha	t	ꦛ	T	
nga	z	ꦔ	Z	

is that they cannot be stored in the database. Consequently, the answer key should not contain any special characters and need to be replaced by word with an alphanumeric character. This process may also prevent SQL injection attack that uses special characters [38] before the players' answer is compiled as SQL query and checked with answer key from the database. The Javanese script which has special characters is in Table III. These characters will be replaced with its substitute word before the answer is checked with the answer key from the database. The function that executes this process and checks players' submitted answer is discussed in algorithm script.

C. Algorithm Script

The algorithm script consists of functions which run specific command to control all elements used in the

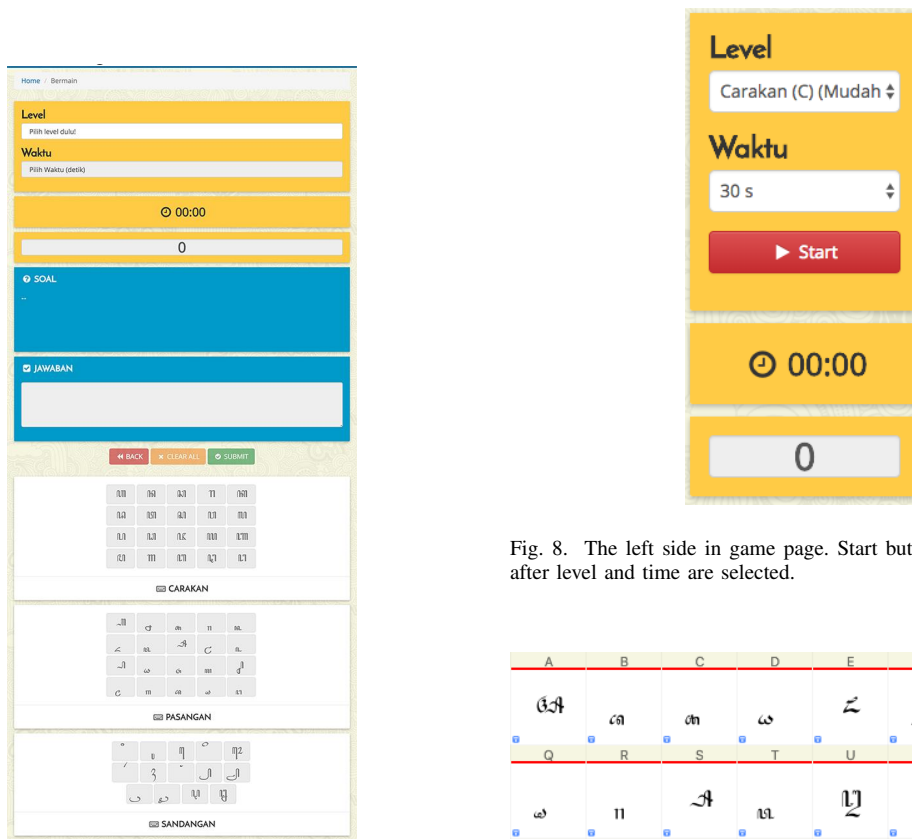Fig. 6. Display of game page in the 13-inch desktop screen.

Fig. 7. Full display of the game page in the mobile screen (7-inch). The elements are adequately displayed but in vertical.

Fig. 8. The left side in game page. Start button will be displayed after level and time are selected.

Fig. 9. The sample of Hanacaraka typeface.

TABLE II
LIST OF JAVANESE SCRIPT LETTER OF SANDANGAN WITH UTF-8 CHARACTER.

Letter	Sandangan	
	UTF-8 Chara.	Javanese Script
i (wulu)	i	ᤠ
u (suku)	u	ᤢ
é (taling)	[	ᤣ
è (pepet)	e	ᤤ
o (taling-taroeng)	[o	ᤥ
r suffix(layar)	/	ᤦ
h suffix (wignyan)	h	ᤧ
ng suffix (ceceg)	=	ᤨ
any consonant suffix in the last word of sentences (pangkon)	\	ᤩ
'dy' syllable in the last word of sentences (pengkol)	-	ᤪ
r infix (cakra)	p	ᤫ
r infix followed by è pepet (cakra keret/cekre)	d	᤬
ré syllable with pepet (pa cerek)	x	᤭
lé syllable with pepet (nga lelet)	X	᤮

TABLE III
JAVANESE SCRIPT WITH SPECIAL CHARACTER.

Javanese Script	Char.	Word Substitute
	=	ceceg
ᤩ	\	pangkon
ᤦ	/	layar
ᤣ	[	taling
ᤫ	]	cakra
᤬	}	cekre

game. The script is mainly created using JavaScript. In some functions, it also involves PHP and AJAX to retrieve data from the server. Details for every function, including the function code, are explained following the activity diagram in Fig. 4.

There are three primary conditions used to group the script functions. Those are as follows.

1) Initial Condition. When the game page is loaded, by default, some game elements are already set without calling a function. Few lines of code in JavaScript `$(document).ready()` function is executed to disable the Javanese script keyboard on the screen. The following is a code sample to disable the keyboard.

```
$(".control :button").attr("disabled", true);
$(".carakan :button").attr("disabled", true);
$(".sandangan :button").attr("disabled", true);
$(".pasangan :button").attr("disabled", true);
```

2) Playing Condition. The first function that is triggered in this condition is start function. At first, the keyboard is enabled. Then, selected time and level options are saved to local variables. The start and share buttons are hidden and the score is set to zero. A local variable is used to convert selected time to seconds and `secondPassed` function starts the countdown timer. The code for the process is as follows.

```
$(".control :button").attr("disabled", false);
$(".carakan :button").attr("disabled", false);
$(".sandangan :button").attr("disabled", false);
$(".pasangan :button").attr("disabled", false);
time = document.getElementById("select_time");
level = document.getElementById("select_level");
document.getElementById('share').style.display = 'none';
document.getElementById('start').style.display = 'none';
document.getElementById("score").value = 0;
seconds = time.options[time.selectedIndex].value;
countdownTimer = setInterval('secondPassed()', 1000);
```

The `secondPassed` function uses `setInterval` with interval 1000 ms. After it is done, the time will run until `clearInterval` function. So, inside of `secondPassed` function, there are codes to do three things. First, there is a code to calculate minutes from total time (in second) which has already been saved before. The minutes of calculation are only used to display the time on screen. Second, there is a condition code to check the remaining total time. If the remaining is above 0, it will deduct the time and the whole process will be repeated. Otherwise, it calls `clearInterval` function to stop the iterative process which has been created before by `setInterval` function and display the time in 00:00 including the share button. A function of reset is also called to return all game elements to initial condition. In general, this `secondPassed` function is called in repetitive like a recursive function and stops when the remaining time is zero. The following is the code of `secondPassed` function.

```
var minutes = Math.round((seconds - 30) / 60);
var remainingSeconds = seconds % 60;
document.getElementById('countdown').innerHTML = minutes + ":"
    + remainingSeconds;
if (seconds === 0) {
    clearInterval(countdownTimer);
    document.getElementById('countdown').innerHTML = "00:00";
    document.getElementById('share').style.display = 'block';
```

```
    if (document.getElementById("score").value > 0) {
time.options[time.selectedIndex].value,
level.options[level.selectedIndex].value);
    }
    reset();
} else {
    seconds--;
}
```

Moreover, the following is the code of reset function.

```
$(".control :button").attr("disabled", true);
$(".carakan :button").attr("disabled", true);
$(".sandangan :button").attr("disabled", true);
$(".pasangan :button").attr("disabled", true);
document.getElementById('countdown').style.color = 'black';
document.getElementById('start').style.display = 'inline-block';
document.getElementById("question").innerHTML = '';
document.getElementById("typebox").innerHTML = '';
oldquest = [];
```

Continuing the explanation in the first paragraph, after the `setInterval` run `secondPassed`, the start function creates AJAX request to a specific URL with selected level as its parameter. If the request is handled successfully, the game settings such as bonus time, bonus scores, and minus scores are saved to local variables. Then, `load_question` starts retrieving the question from database. The following is the sample AJAX request code to get the game setting.

```
xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function () {
if(xmlhttp.readyState==4 && xmlhttp.status==200) {
  var val = xmlhttp.responseText.split("-");
  bonus_time = val[0];
  bonus_exp = val[1];
  minus_exp = val[2];
load_question(level.options[level.selectedIndex].value);
}};
xmlhttp.open("GET", "<?php echo site_url('bermain/get_setting_game')
; ?>" + "/"
+ (level.options[level.selectedIndex].value), true);
xmlhttp.send();
```

There is also AJAX request inside of `load_question`. The objective of this function is only to get one question at a time from database. If the request is successful, `check_question` will be done and the retrieved question will be used as its parameter. The code of `load_question` is as follows.

```
xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function () {
if(xmlhttp.readyState===4 && xmlhttp.status===200){
  check_question(xmlhttp.responseText);
}};
xmlhttp.open("GET", "<?php echo site_url('bermain/get_a_question');
  ?>" + "/" + level + "/" + seconds, true);
xmlhttp.send();
```

The `check_question` is used to check the retrieved question to make sure that the question has not been issued previously. The newly retrieved question will be checked with a list of old retrieved questions before. If there is no match with the list, it will be displayed in question box and saved in the list. Otherwise, the `load_question` will be done again until it gets a new question. The sample code of `check_question` is as follows.

```
var check = 0;
for (var i = 0; i < oldquest.length; i++) {
    if (oldquest[i] == val) {
        check = 1;
        break;
    }
}
if (check == 1) {
load_question(level.options[level.selectedIndex].value);
} else {
    oldquest.push(val[0]);
    document.getElementById("question").innerHTML = val[0];
    seconds = parseInt(val[1]);
}
```

A function to check players' answer will be executed when player clicks the submit button. When it is clicked, the answer will be copied to a local variable as string value. Afterwards, this string value is filtered to remove special characters in Table III. Then, AJAX request is created for PHP function on the server to check the answer. This request returns a result whether the answer is correct or wrong. If the request is successful, `update_score` is done. Then, the result will be used as its parameter. The following is the sample code.

```
var temp = document.getElementById("typebox").value;
var quest = document.getElementById("question").innerHTML;
var answer = temp.replace(/'/g, '');
answer = answer.replace(/=/g, 'ceceg');
answer = answer.replace(/\\/g, 'pangkon');
answer = answer.replace(/\//g, 'layar');
answer = answer.replace(/\[/g, 'taling');
answer = answer.replace(/\]/g, 'cakra');
answer = answer.replace(/\|/g, 'cekre');
xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function () {
if(xmlhttp.readyState==4 && xmlhttp.status==200) {
  countdownTimer = setInterval('secondPassed()', 1000);
  updateScore(xmlhttp.responseText);
}}
xmlhttp.open("GET", "<?php echo site_url('bermain/check_answer');
  ?>" + "/" + quest + "/" + answer + "/" + seconds, true);
xmlhttp.send();
```

The `update_score` function is used to add score, deduct score, and add time. To add or deduct scores, the value of current score is parsed to integer first and is added with local variable which has bonus score setting. The additional time is added directly to the local variable which controls the remaining time. After this process completes, `load_question` is called again to issue a new question. The sample code of `update_score` function is as follows.

```
var score = document.getElementById("score").value;
if (string != '') {
  var val = string.split("-");
  var temp;
  seconds = parseInt(val[1]);
  if (val[0] === 'true') {
    temp = parseInt(score) + parseInt(bonus_exp);
    seconds = parseInt(val[1]) + parseInt(bonus_time);
    document.getElementById("score").value = temp;
  } else {
    temp = parseInt(score - minus_exp);
    document.getElementById("score").value = temp;
  }
}
load_question(level.options[level.selectedIndex].value);
```

3) Final Condition. After the game is over, a Twitter share button is shown. Players may choose to share the final scores or start playing again.

D. Implementation and Testing

The system is implemented online to be tested in real time. There are several testing cases used to test the reliability of the game algorithm. This testing process involves 27 respondents. The respondents are from Yogyakarta. Their profiles are varied from high school students, teachers, and lecturers who have already had experience of learning Javanese script at school. Each of them is requested to complete a questionnaire and follow the recommendation rule to test the system. Respondents are recommended to use a laptop or desktop computer with at least 13-inch screen and Google Chrome to access the system. All questions in the questionnaire are marked as a required field, so they have to fill them all.

There are two question groups in the questionnaire. The first group consists of questions about web application as an alternative tool to learn Javanese script and the Javanese script word game in a web application as discussed in this paper. There are three points used to measure the importance of the web application for both of them. On web application as a tool to learn Javanese script, the points discussed are its necessity, usability, and positive benefit. Whereas, in the Javanese script game, the discussion focuses on honing the ability to read, write, and memorize basic material of Javanese script. To answer the question, respondents may choose one from four statements. These statements are strongly agree, agree, disagree, and strongly disagree. The total percentage of all respondents' answer in this question group is displayed in Table IV.

To calculate the final result in Table IV, the assessment is narrowed down into two groups. Those are agree and disagree groups. The agree group consists of strongly agree and agree parameters. Meanwhile, the disagree group consists of disagree and strongly disagree parameters. As displayed in Table IV, all assessments are in between strongly agree and agree. Hence, the numbers in the same column are respectively added to get the total percentage for the agree group and disagree group. The summary of Table IV which has been narrowed down into two groups is displayed in Fig. 10. The result of the total percentage of the agree group is 100% whereas the disagree group is 0%. Therefore, it can be stated that a web application to learn Javanese script is necessary, useful, and beneficial for the users. In addition, the Javanese script game in a web application can be used to hone the ability to read, write, and memorize basic material of Javanese script.

The second question group is about the testing case for the game algorithm. Questions used in this group are based on the algorithm script discussed in the
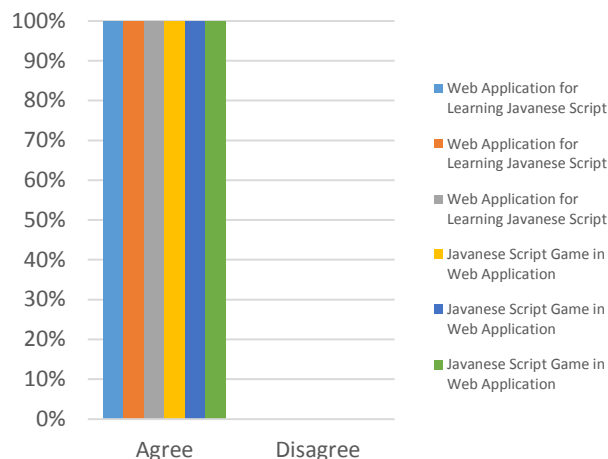

Fig. 10. Summary of assessment on the utilization of web application as an alternative tool to learn Javanese script and the Javanese script word game in a web application.

previous point. There are 10 testing cases tested.

1) Case 1: Player can select level and time before the game is started.
2) Case 2: Countdown timer runs and stops according to the time specified by players.
3) Case 3: The score starts at 0 when the game is started.
4) Case 4: Javanese script keyboard can only be used during the game before countdown timer reaches zero.
5) Case 5: The score will be deducted when the players' answer is wrong.
6) Case 6: The score will be deducted when the players' answer is wrong, and the deduction value is different at every level.
7) Case 7: Additional score and time will be added when the players' answer is correct.
8) Case 8: Additional score and time will be added when the players' answer is correct, and the added value is different at every level.
9) Case 9: All displayed questions are based on the selected level.
10) Case 10: The player can share the final score when the game is over.

The total percentage of all respondents' answer in this question group is displayed in Table V. There are four cases which work perfectly. Those are Case 1, Case 3, Case 4, and Case 7. However, some respondents encounter a problem for the other six cases especially for Case 6 and Case 10 which is filled by 7.4% of respondents.

In Case 6, respondents are asked to give the wrong answer and check the deduction value in the different level. Assuming that respondents have followed the

83

TABLE IV
THE RESULT OF THE FIRST QUESTION GROUP.

Assessment	Learning Javanese Script			Javanese Script Game		
	Necessity	Usability	Positive Benefit	Ability to read	Ability to write	Ability to memorize the material
Strongly Agree	51.9%	70.4%	70.4%	55.6%	51.9%	63%
Agree	48.1%	29.6%	29.6%	44.4%	48.1%	37%
Disagree	0%	0%	0%	0%	0%	0%
Strongly Disagree	0%	0%	0%	0%	0%	0%

TABLE V
THE RESULT OF THE SECOND QUESTION GROUP.

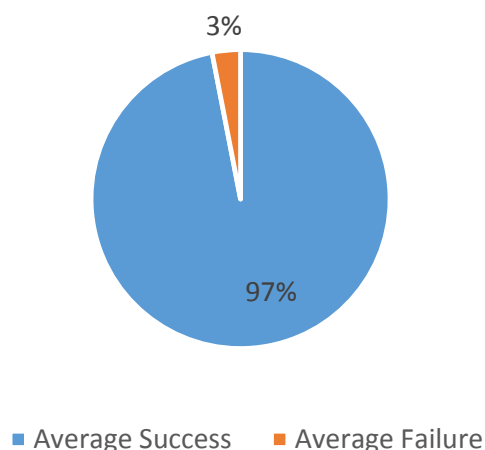| Assessment	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8	Case 9	Case 10
Yes	100%	96.3%	100%	100%	96.3%	92.6%	100%	96.3%	96.3%	92.6%
No	0%	3.7%	0%	0%	3.7%	7.4%	0%	3.7%	3.7%	7.4%

Fig. 11. Summary of the result from respondents in algorithm testing case.

recommended rule to test the case and the internet connection is well established, this problem may happen because the AJAX request in start function fails. The failed request may happen because the server is busy with processing incoming requests. Thus, some requests are rejected, and the current game setting cannot be updated. The new game session will use the old setting from the previous game session.

In Case 10, respondents are asked to test the Twitter share button. With the same assumption used, this problem can occur because the browser configuration may not allow pop-up display. Hence, the pop-up dialog is blocked, and there will be no information shared on Twitter.

Based on the result in Table V, it can be concluded that the average successful performance of the algorithm from 26 respondents is 97%. The summary of Table V showing the average percentage of success and failure of the algorithm is displayed in Fig. 11.

IV. CONCLUSION

Based on every discussion and development activities performed, it can be concluded that web application is necessary and useful to learn Javanese script. It also gives positive benefits to users like students who still need to learn Javanese script in school. In addition, the Javanese script word game is successfully developed. The development of this algorithm is based on use-case diagram and activity diagram design. The algorithm is tested and has an average 97% success rate to run the game. However, to get the best experience during the game, players have to follow the recommendation such as using at least 13-inch screen and Google Chrome browser. Right now, the game is only available in the web platform, but students may play it as many as they can without the need of using a mobile device. The expected result by playing this Javanese script word game is that the students can hone their ability to read, write, and memorize the Javanese script. Thus, they can read and write Javanese script fluently. Moreover, for future study, UI and algorithm in this game can be refined to support more functionalities and developed to run the game in a mobile platform. This game algorithm can also be improved and combined with machine learning algorithm to save players' activities, learn players' activities, and adjust the question level based on players' performance on the previous game session. Another development of the game algorithm to support this game is also beneficial to give more alternatives to learn Javanese script in a fun way.

The development of this word game algorithm can also be beneficial to other word games which may use specific characters besides Latin alphabets. It needs to modify some functions in it to be adjusted with the game requirement without altering the whole algorithm. For example, the researchers may use other

characters that also part of the local heritage in Indonesia such as Balinese script, Sundanese script, and Bataknese script in different word games. Researchers may use the algorithm to develop new functionalities to add more features to the word game. Moreover, the use-case and activity diagram can be used as a resource document to support more algorithm researches which involve word game in the future.

REFERENCES

[1] Y. G. Butler, Y. Someya, and E. Fukuhara, "Online games for young learners foreign language learning," *ELT Journal*, vol. 68, no. 3, pp. 265–275, 2014.

[2] M. Mehregan, "Game-based tasks for foreign language instruction: Perspectives on young learners' vocabulary acquisition." *IAFOR Journal of Language Learning*, vol. 1, no. 1, pp. 1–12, 2014.

[3] M. S. Sahrir and G. Yusri, "Online vocabulary games for teaching and learning Arabic," *GEMA Online® Journal of Language Studies*, vol. 12, no. 3, pp. 961–977, 2012.

[4] B. Perry, "Gamifying French language learning: A case study examining a quest-based, augmented reality mobile learning-tool," *Procedia - Social and Behavioral Sciences*, vol. 174, pp. 2308–2315, 2015.

[5] N. Hidayat, "Improving students' vocabulary achievement through word game," *JEES (Journal of English Educators Society)*, vol. 1, no. 2, pp. 95–104, 2016.

[6] A. Masri and M. Najar, "The effect of using word games on primary stage students achievement in english language vocabulary in Jordan," *American International Journal of Contemporary Research*, vol. 4, no. 9, pp. 114–152, 2014.

[7] T. T. Prabowo. (2017) Mendekatkan pelajar dengan Bahasa dan Aksara Jawa. Accessed: 20-May-2018. [Online]. Available: pendidikan-diy.go.id/dinas_v4/?view=v_artikel&id=54

[8] S. Sutarsih, "Pembelajaran menulis Aksara Jawa anak kelas III Sekolah Dasar," *Aksara*, vol. 27, no. 1, pp. 65–72, 2015.

[9] G. M. Mazida, "pembelajaran membaca paragraf sederhana berhuruf Jawa dengan game berburu gambar pada siswa kelas VII H MTS Negeri Brangsong, Kendal," *Piwulang Jawa: Javanese Learning and Teaching Journal*, vol. 2, no. 1, pp. 1–7, 2013.

[10] O. Suria and C. W. Kurniawan, "Pengembangan aplikasi multimedia untuk sarana edukasi Aksara Jawa menggunakan framework Codeigniter dan HTML 5," in *Seminar Nasional Teknologi Informasi dan Komunikasi*, Yogyakarta, March 28, 2015, pp. 261–270.

[11] S. H.-J. Liu, Y.-J. Lan, and C. Y.-Y. Ho, "Exploring the relationship between self-regulated vocabulary learning and web-based collaboration." *Journal of Educational Technology & Society*, vol. 17, no. 4, pp. 404–419, 2014.

[12] M. Papastergiou and V. Gerodimos, "Can learning of basketball be enhanced through a web-based multimedia course? An experimental study," *Education and Information Technologies*, vol. 18, no. 3, pp. 459–478, 2013.

[13] H.-C. Chu, "Potential negative effects of mobile learning on students' learning achievement and cognitive load–A format assessment perspective." *Journal of Educational Technology & Society*, vol. 17, no. 1, pp. 332–344, 2014.

[14] T. Zarraonandia, P. Diaz, I. Aedo, and M. R. Ruiz, "Designing educational games through a conceptual model based on rules and scenarios," *Multimedia Tools and Applications*, vol. 74, no. 13, pp. 4535–4559, 2015.

[15] J. T. Kim and W.-H. Lee, "Dynamical Model for Gamification of Learning (DMGL)," *Multimedia Tools and Applications*, vol. 74, no. 19, pp. 8483–8493, 2015.

[16] K. Lange, S. Kühn, and E. Filevich, "Correction: "Just Another Tool for Online Studies" (JATOS): An easy solution for setup and management of web servers supporting online studies," *PloS One*, vol. 10, no. 7, p. e0134073, 2015.

[17] J. Radhakrishnan, "Hardware dependency and performance of Javascript engines used in popular browsers," in *2015 International Conference on Control Communication & Computing India (ICCC)*. Trivandrum, India: IEEE, Nov. 19–21, 2015, pp. 681–684.

[18] N. Jillings, B. De Man, D. Moffat, and J. D. Reiss, "Web audio evaluation tool: A browser-based listening test environment," in *12th Sound and Music Computing Conference*, Maynooth, Ireland, July 26–Aug. 1, 2015.

[19] K. J. Luterbach and K. R. Hubbell, "Capitalizing on app development tools and technologies," *TechTrends*, vol. 59, no. 4, pp. 62–70, 2015.

[20] N. R. Swain, S. D. Christensen, A. D. Snow, H. Dolder, G. Espinoza-Dávalos, E. Goharian, N. L. Jones, E. J. Nelson, D. P. Ames, and S. J. Burian, "A new open source platform for lowering the barrier for environmental web app development," *Environmental Modelling & Software*, vol. 85, pp. 11–26, 2016.

[21] M. Willocx, J. Vossaert, and V. Naessens, "Com-

paring performance parameters of mobile app development strategies," in *IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft)*. Austin, TX, USA: IEEE, May 14–22, 2016, pp. 38–47.

[22] H. Supriyono, R. F. Rahmadzani, M. Adhantoro, and A. K. Susilo, "Rancang bangun media pembelajaran dan game edukatif pengenalan aksara Jawa Pandawa," in *The 4th University Research Colloquium*. Jawa Tengah: STIKES Muhammadiyah Pekajangan, Aug. 27, 2016, pp. 1–12.

[23] M. Roccetti, P. Salomoni, M. Loiseau, M. Masperi, V. Zampa, A. Ceccherelli, C. Cervini, and A. Valva, "On the design of a word game to enhance Italian language learning," in *International Conference on Computing, Networking and Communications (ICNC)*. Kauai, HI, USA: IEEE, Feb. 15–18, 2016, pp. 1–5.

[24] W. Peng, H. Song, J. Kim, and T. Day, "The influence of task demand and social categorization diversity on performance and enjoyment in a language learning game," *Computers & Education*, vol. 95, pp. 285–295, 2016.

[25] N. Thompson and D. Toohey, "Teaching case: Analysis of an electronic voting system," *Journal of Information Systems Education*, vol. 25, no. 1, pp. 13–21, 2014.

[26] J. L. Spears and J. L. Parrish Jr, "Is security requirements identification from conceptual models in systems analysis and design: The Fun & Fitness, Inc. case." *Journal of Information Systems Education*, vol. 24, no. 1, pp. 17–29, 2013.

[27] A.-E. Iordan, "Design with UML of an interactive software for the study of the two surfaces of revolution," *Annals of the Faculty of Engineering Hunedoara*, vol. 12, no. 3, p. 275, 2014.

[28] R. Murali, A. Ireland, and G. Grov, "A rigorous approach to combining use case modelling and accident scenarios," in *NASA Formal Methods Symposium*. Pasadena, United States: Springer, April 27–29, 2015, pp. 263–278.

[29] ——, "UC-B: Use case modelling with event-b," in *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z*. Verlag Berlin, Heidelberg: Springer, May 23–27, 2016, pp. 297–302.

[30] Y. Wautelet, S. Heng, D. Hintea, M. Kolp, and S. Poelmans, "Bridging user story sets with the use case model," in *International Conference on Conceptual Modeling*. Gifu, Japan: Springer, Nov. 14–17, 2016, pp. 127–138.

[31] Y. Jarraya and M. Debbabi, "Quantitative and qualitative analysis of SysML activity diagrams," *International Journal on Software Tools for Technology Transfer*, vol. 16, no. 4, pp. 399–419, 2014.

[32] P. Sapna and A. Balakrishnan, "An approach for generating minimal test cases for regression testing," *Procedia computer science*, vol. 47, pp. 188–196, 2015.

[33] H. Okuda, S. Ogata, and S. Matsuura, "Experimental development based on mapping rule between requirements analysis model and web framework specific design model," *SpringerPlus*, vol. 2, no. 1, p. 123, 2013.

[34] D. Lending and J. May, "Using activity diagrams to model systems analysis techniques: Teaching what we preach," *Journal of Information Systems Education*, vol. 24, no. 2, pp. 91–97, 2013.

[35] M. Huda, K. A. Jasmi, M. I. Mustari, B. Basiron, A. K. Mohamed, W. H. W. Embong, and J. Safar, "Innovative e-therapy service in higher education: mobile application design," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 11, no. 4, pp. 83–94, 2017.

[36] M. Mhereeg and A. G. Tawil, "Analysis and design of a filestream based English language learning system." *Journal of Computers*, vol. 10, no. 4, pp. 268–283, 2015.

[37] J. H. Goldberg, M. J. Stimson, M. Lewenstein, N. Scott, and A. M. Wichansky, "Eye tracking in web search tasks: design implications," in *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*. New Orleans, Louisiana: ACM, March 25–27, 2002, pp. 51–58.

[38] N. P. Singh, "A survey of threats to e-commerce applications," *Research Journal of Science and Technology*, vol. 8, no. 3, pp. 145–149, 2016.