Personalized Al-Quran Memorization Testing System Using Group Decision Support System

Rian Adam Rajagede1,*, Yuanda Hanif Hisyam1, Muhammad Ichlasul Amal Yulianto1, Farid Amin Ridwanto2, Alfian Try Putranto2, Muhammad Rifqi Fatchurrahman Putra Danar2

1Department of Informatics, Universitas Islam Indonesia, Sleman, Indonesia; 2BeHafizh, Yogyakarta, Indonesia;
*Corresponding author email: rian.adami@u ui.ac.id

Received: October 26, 2021
Accepted: December 7, 2021
Online: December 30, 2021

Abstract – Memorizing Al-Quran is one of the most important acts of worship for Muslims. After memorizing some parts of the Al-Qur’an, the hafiz or Al-Qur’an’s memorizer is recommended to repeat or muraja’ah their memorization to strengthen it. This process is usually done in pairs by listening to each other’s memorization or testing by asking questions about Al-Quran. This study proposes a system that can help memorizers test their memorization independently without a partner. The system will perform a memorization test to support the user’s process of memorizing the Al-Quran. The system records and analyzes user data and uses it to personalize memorization testing from time to time. The system was made using the Group Decision Support System (GDSS) approach with the help of several Al-Quran memorizers as decision-makers. The GDSS algorithm used combines Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) and Weighted Geometric Mean to rank surahs based on provided user data. The evaluation was conducted with the help of human evaluators, and the evaluators showed 78% agreement with the system decision.

Keywords: Al-Quran memorization test, Personalized Quiz, GDSS, TOPSIS.

Introduction

Al-Quran is the holy book of Muslims which consists of 114 surahs or chapters. Memorizing the Al-Quran is one of the most important acts of worship. In memorizing the Al-Qur’an, one of the stages is muraja’ah, which is repeating the memorization to strengthen. The method that is often used in Muslim-majority countries such as Indonesia and Malaysia is tasmi’. (Ariffin et al., 2013; Aziz et al., 2019; Nik Md Saiful Azizi et al., 2019; Mercellina et al., 2020). The tasmi’ method requires a partner. The partner’s task here can be to correct the reading, remind, or test the memorization. For example, when a memorizer of the Al-Qur’an has a partner who is good at reading, the partner can also help in correcting the tajweed of the recitation. A good partner will also support the memorization process by paying attention to the condition of the memorizer, understanding which surahs are difficult or have not been memorized. However, for some people, this is not easy. Daily busyness and geographical conditions make it difficult for someone to find a partner to memorize the Al-Quran. For example, the current pandemic condition that forces us to stay at home can be a barrier to finding a partner. Also, another common problem is finding the right schedule. Adjusting the schedule between the memorizer and his partner during daily activities can be quite difficult. Therefore, we need a system that can help the process of testing the memorization of the Al-Qur’an.

Like a real human partner, the Al-Qur’an memorization testing system should pay close attention to user behavior. The system should be able to create a test scheme that depends on who is being tested (personalized). Several studies have been conducted to create a personalized test system in learning, sometimes called adaptive
Some show that adaptive quizzes can help participants learn (Simon-Campbell et al., 2018; Soltanpoor et al., 2018; Lin, 2020). Some show that adaptive quizzes can help participants learn (Simon-Campbell et al., 2018; Soltanpoor et al., 2018). Lin (2020) modeled the question selection problem as the Bernoulli Bandit Problem, where the agent must estimate the probability of success of each question. The Thompson Sampling Algorithm is proposed to select a problem that has been modeled as the Bernoulli Bandit Problem.

Group Decision Support System (GDSS) is a system created to assist more than one decision-maker in determining the best alternative. This technology has been around since the 1980s (Gray, 1987) but until now, GDSS is still developing and used in various cases. GDSS can be utilized in various domains where several decision-makers have to choose between available alternatives, such as in health (Arifin et al., 2016; Hsu et al., 2021), education (Mohammed et al., 2017; Saraswati et al., 2019), or business and management (Ogiana et al., 2017; Dewi, 2019; Yap et al., 2019). Various methods can be used for GDSS. In the health sector, GDSS with the Z-DEMATEL method is used to see the influence of technology on the health industry in Taiwan (Hsu et al., 2018). The group used to consider the study consisted of several people with very diverse backgrounds. The Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) method is also widely used in GDSS. TOPSIS can be combined with the Analytical Hierarchy Process (AHP) method (Saraswati et al., 2019) or Geometric Mean (Mohammed et al., 2017). Both studies used TOPSIS on the GDSS in the education domain. In GDSS, two types of approaches can be taken to deal with more than one decision-maker, namely Aggregating Individual Judgments (AIJ) and Aggregating Individual Priorities (AIP) (Ossadnik et al., 2016). AIJ is used when groups of decision-makers have a common goal and are assumed to act together (Galo et al., 2018), while AIP is used when the group is assumed to consist of different individuals. The process of unifying decisions in both groups can be done using the geometric mean (Arifin et al., 2018; Saraswati et al., 2019).

Currently, there are not many systems that focus on helping the memorizers of the Qur’an from the side of the muraja’ah process. Several memorization management systems have been created before, either on the web (Mohamed Elhadj, 2010; Suryana et al., 2021) or mobile platforms (Pradhana et al., 2019). However, these systems primarily focus on monitoring, not on memorization testing. The memorization test is still in the form of a static quiz, not personalized using artificial intelligence. In fact, research has shown the importance of personalizing quizzes in the success of their students (Simon-Campbell et al., 2018; Soltanpoor et al., 2018). This study proposes an intelligent system that can be a partner in testing the memorization of the Al-Quran. The system can reflect a memorization partner in making personalized exams that pay attention to the user’s condition. The proposed method is the Technique for Others Reference by Similarity to Ideal Solution (TOPSIS) algorithm with the Aggregating Individual Priorities Group Decision Support System (AIP-GDSS) approach using the Weighted Geometric Mean. The results of this study are expected to make it easier for users who are still constrained in finding partners in the process of memorizing the Quran.

Materials and Methods

System analysis

We developed a personalized Al-Quran memorization testing system using Group Decision Support System (GDSS) approach. In general, the system flow is shown in Figure 1. The system takes the criteria obtained from the surahs that have been memorized and user history data when doing a memorization test. By using TOPSIS and Geometric Mean algorithms, the system will calculate the weight of the surah to be tested based on the judgment of the decision-makers. The system will output the top five surahs to be tested.

![Figure 1. Interaction between the user, method, and decision-makers in the proposed system](image-url)
In more detail, the process of choosing a surah is shown in Figure 2. We divide the criteria into two types of criteria, internal and external. Internal criteria are unique to each user, while external criteria are unique to each surah in the Quran. The criteria values obtained from the user and the memorized surahs are combined with the weights from the decision-makers to calculate which surah should be tested.

![Figure 2. Surah scoring process based on criteria values and weights](image)

In GDSS, there are two approaches when facing more than one decision-maker, namely Aggregating Individual Judgments (AIJ) and Aggregating Individual Priorities (AIP). AIJ is used when groups of decision-makers have a common goal and are assumed to act together, while AIP is used when the group is considered to consist of different individuals. The AIP approach was used in this study because DMs do not come from the same organization, so it is not easy to assume they have the same goal. One of the differences between AIP and AIJ is the aggregated value. In AIP, the aggregated value results from the TOPSIS calculation of each DMs, while in AIJ, the weights of all DMs are aggregated.

<table>
<thead>
<tr>
<th>ID</th>
<th>Criteria Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>Choose a surah that recently tested</td>
<td>Internal</td>
</tr>
<tr>
<td>C2</td>
<td>Choose a surah that has not been tested recently</td>
<td>Internal</td>
</tr>
<tr>
<td>C3</td>
<td>Choose a surah based on incorrect answer frequency</td>
<td>Internal</td>
</tr>
<tr>
<td>C4</td>
<td>Choose a surah based on its difficulty</td>
<td>External</td>
</tr>
<tr>
<td>C5</td>
<td>Choose a surah based on its length</td>
<td>External</td>
</tr>
<tr>
<td>C6</td>
<td>Choose a surah that was recently marked as memorized</td>
<td>Internal</td>
</tr>
<tr>
<td>C7</td>
<td>Choose a surah that has long been marked as memorized</td>
<td>Internal</td>
</tr>
<tr>
<td>C8</td>
<td>Choose a surah that is rarely tested</td>
<td>Internal</td>
</tr>
<tr>
<td>C9</td>
<td>Choose a surah that is often tested</td>
<td>Internal</td>
</tr>
</tbody>
</table>

**Data collection**

We surveyed seven people who had experience in the Al-Quran memorization program to determine the considerations in choosing the surah to be tested. We refer to these seven people as decision-makers (DMs). To increase the data variation, we tried to select DMs from various educational backgrounds and have different
levels of memorization. We divide it based on the number of juz memorized: 30 juz (100% memorized), less than 20 juz, less than ten juz, and less than five juz but still at least two juz memorized.

In the survey, we asked the respondents to act as Al-Quran memorization testers who had data on the people to be tested. DMs were asked to fill out a questionnaire containing a list of criteria for choosing surah and determining their priority on a scale of 1-5. The list of criteria for selecting the surah is shown in Table 1, and the Type column shows the criteria as internal or external type. To support the C4 criteria, we also asked what surah were challenging to memorize and later used them as a reference. Details on how to obtain values from C1 to C9 are explained in the Criterion Calculation section.

**Technique for Others Reference by Similarity to Ideal Solution (TOPSIS)**

TOPSIS is one of the algorithms commonly used in Multi-Attribute Decision Making (MADM). TOPSIS algorithm is widely used in individual (Arifin et al., 2017) or group decision making (Dewi, 2019; Saraswati et al., 2019). TOPSIS scores an alternative by comparing its distance to a positive ideal solution (the closer, the better) and a negative ideal solution (the farther, the better). This study performed TOPSIS calculations for each DM before combining Geometric Mean (Aggregating Individual Priorities).

TOPSIS will receive a decision-making matrix \( X \) of size \( n \times m \), where \( n \) is the number of alternatives and \( m \) is the number of criteria. The matrix will be \( n=114 \) in size in this study, representing the number of surahs in the Al-Quran, and \( m=9 \), representing the number of criteria. TOPSIS will receive a criteria matrix of \( n \times m \), where \( n \) is the number of alternatives and \( m \) is the number of criteria.

The steps of the TOPSIS algorithm are as follows:

1. Construct a decision matrix, \( X = [x_{ij}]_{n \times m} \)
   where \( n \) is the number of alternatives and \( m \) is the number of criteria.

2. Create a normalized matrix \( R = [r_{ij}]_{n \times m} \) from the criteria matrix. This process is formulated below.
   The value of \( x_{ij} \) is the value of the \( j \)-th attribute for the \( i \)-th alternative. In this research, \( x_{ij} \) representing the value of the \( j \)-th choosing criterion on \( i \)-th surah
   \[
   r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{n} x_{ij}^2}}
   \] (2)

3. Multiply the normalized \( r_{ij} \) by the weight \( w_j \). The weight \( w_j \) is the weight of the \( j \)-th criterion obtained from DMs. In GDSS, this process is conducted for all DMs. So we will get seven \( Y \) matrices that are calculated using the following formula:
   \[
   y_{ij} = r_{ij}w_j
   \] (3)

4. Calculate the positive and negative ideal solution \( S^+ \) and \( S^- \):
   \[
   S^+ = (y^+_1, y^+_2, ..., y^+_m)
   \] (4)
   \[
   S^- = (y^-_1, y^-_2, ..., y^-_m)
   \] (5)
   The value of \( y^+_j \) and \( y^-_j \) are calculated using:
   \[
   y^+_j = \max_{1 \leq i \leq n} y_{ij}
   \] (6)
   \[
   y^-_j = \min_{1 \leq i \leq n} y_{ij}
   \] (7)
5. Calculate the distance between the alternatives to the ideal solution. The value of $D_i^+$ is the Euclidean distance between the $i$-th alternative to the $S^+$. While the $D_i^-$ is calculated to $S_-$. The steps of calculating the value of $D_i^+$ and $D_i^-$ as follows:

$$D_i^+ = \sqrt{\sum_{j=1}^{m} (y_j - y_j^+)^2} \quad (8)$$

$$D_i^- = \sqrt{\sum_{j=1}^{m} (y_j - y_j^-)^2} \quad (9)$$

6. Calculate the preference value for each alternative. This is done by finding the value of $V_i$ using the following formula:

$$V_i = \frac{D_i^-}{D_i^+ + D_i^-} \quad (10)$$

The preference value $V_i$ is the final score for the $i$-th alternative. It should be noted that in this study because we use GDSS, the $V_i$ the value will be obtained as much as DMs.

**Geometric mean**

The geometric mean is a method for calculating the central tendency of a set of values using multiplication, in contrast to the arithmetic mean, which uses addition. The geometric mean is calculated in the following way, where $k$ is the number of values and $v_i$ is the $i$-th value:

$$\bar{v} = \left(\prod_{i=1}^{k} v_i\right)^{\frac{1}{k}} \quad (11)$$

The geometric mean is used to combine the TOPSIS preference values $V$ from all DMs. In this study, $k$ is the number of DMs and $v_i$ is the vector of preference value from the $i$-th DM. However, in this study, we used the weighted geometric mean because DMs had a different level of Al-Quran memorization. We assigned the weights of DMs based on the amount of their memorization:

1. DM that has memorized 100% of Al-Quran (30 juz) is weighted by 4
2. DM that has memorized less than 20 juz is weighted by 3
3. DM that has memorized less than ten juz is weighted by 2
4. DM that has memorized less than five juz is weighted by 1

The formula is shown below, where $k$ is the number of DMs and $z_i$ is the weight of each DM.

$$\bar{v} = \left(\prod_{i=1}^{k} v_i^{z_i}\right)^{\frac{1}{\sum_{i=1}^{k} z_i}} \quad (12)$$

**Criterion calculation**

Calculating the values of the criteria is carried out when the user is going to do a memorization test. The system will count the values C1 to C9 from 114 surahs. The criteria matrix will be sent via API for server-side processing. We use the following formulas to derive the numerical value of the criteria in Table 1 for each surah:

1. **C1 and C2 formulation**

Score a surah that was recently tested using the formula below. The value of $c$ is the number of memorization tests that have already been taken since the surah appeared in the memorization test last time. We use 30 as the threshold, as it is assumed that the user takes 1 test per day, so it is too long if a surah is not tested in the last 30 days.

$$c1_{score} = \max(1, 30 - c) \quad (13)$$
2. **C3 formulation**
   Whenever the user makes an error in answering a test, the system will track and accumulate how often the user is wrong in answering the respective surah.

3. **C4 formulation**
   The difficulty level of the surah is determined based on the questionnaires. As a result, two surahs were voted by at least two DMs as difficult surahs, namely Al-Muthofifin (surah no. 83) and At-Takwir (surah no. 81). The value of C4 is binary, which represents whether the surah is difficult or not.

4. **C5 formulation**
   This value is obtained from how many verses are in a surah.

5. **C6 dan C7 formulation**
   To find out how long a surah has been memorized, we calculate the value of D1 and D2. D1 is the number of days from the time the application is installed to the time the test is started, and D2 is the number of days from the surah marked as memorized in the system until the time the test is started.

\[
c_{6\_\text{score}} = 1 - \frac{D_2 + 1}{D_1 + 1}
\]  
\[
c_{7\_\text{score}} = \frac{D_2 + 1}{D_1 + 1}
\]

6. **C8 and C9 formulation**
   C8 and C9 values are calculated similarly with C6 and C7. We count the number of tests that are already taken by the user, F1, and count the number of respective surahs that appear in the test, F2, and then calculated using the formulas below.

\[
c_{8\_\text{score}} = 1 - \frac{F_2 + 1}{F_1 + 1}
\]
\[
c_{9\_\text{score}} = \frac{F_2 + 1}{F_1 + 1}
\]

**Implementation**

The algorithm is implemented as API (Application Programming Interface). The API will receive input in a 114 x 9 matrix representing C1 to C9 for 114 surahs on each call. The API will return a list of five recommended surahs based on user data. This API is developed using JavaScript and runs on Heroku servers.

**Evaluation method**

This study focused on developing algorithms for testing Al-Quran memorization, so testing was only carried out on the GDSS algorithm used. Algorithm evaluation is conducted by manually evaluating the system by a human evaluator selected from DMs. We randomly generated ten users’ persona who memorized the surahs in juz 30 and already took 50 tests. We limited to only generating users who memorized surahs in juz 30 to make it easier for human evaluators. The human evaluator will validate five surahs from juz 30 chosen by the system and present them in percentage. We did not use accuracy as there is no ground truth in choosing surahs for memorization tests (Shafinah et al., 2010).

**Results**

**The application prototype**

We created a simple Android application using the Flutter framework as a prototype. Users can use the application to update their memorization data and to conduct self-tests, among other things. In this prototype, the user will be prompted to read and assess the next displayed verse on their own initiative. Flutter is a mobile
application development framework developed by Google for the development of mobile applications for both Android and iOS devices. We choose the Flutter and API implementation for the algorithm implementation in order to make future system development more manageable.

The interfaces for the applications are depicted in Figure 3 below. Using the blue circle on the right-hand side of the application, the user can indicate which surahs have been memorized. Figure 3 illustrates the start screen where the user can indicate which surahs have been memorized. It is important to note that the user has not selected any surah when the application is first opened, and all circles are still empty. This is the original state in which the application is opened. Before taking the test, the user must mark at least five surahs that they have memorized in order to be eligible.

Users can take the test by clicking on the “Take Test” button located at the bottom right corner of the display screen. When the user begins the memorization test, the system will generate a criterion matrix based on the information provided by the user and transmit it to the server via an API call. The server returns the top five surahs recommended for testing based on the algorithm that has been specified.

![Image: Application Prototype Interface in Mobile](image-url)

Figure 3. The application prototype interface in mobile

In this prototype, we only focus on implementing the GDSS algorithm so that the procedure of memorization testing is still simple (Fig. 3 Right). The system will choose one verse randomly from selected surahs and display it to the user. The Arabic text of the verse is obtained through EveryAyah. In the test, the user should recite the next verse after the shown verse without seeing Al-Quran. This is a self-assessment. After reciting, the user will see the correct answer and choose “Correct” if they accurately recite the next verse or choose “Wrong” otherwise.

The TOPSIS algorithm illustration

This section will demonstrate how the TOPSIS algorithm works when scoring surahs in a user’s memorization test. We will only use a subset of the data gained and explained in the previous section for simplification. First, we have collected the weights of criteria C1 to C9 from three DMs based on the results of the survey, as shown in Table 2. We also use one of the generated personas and choose three surahs from it as
an example. The value of each criterion of the persona that we use for the three alternative surahs is shown in Table 3.

Table 2. Decision-makers score for each criterion

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM #1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>DM #2</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>DM #3</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 3. User score for each criterion in three alternative surahs

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surah #1</td>
<td>16</td>
<td>14</td>
<td>2</td>
<td>0</td>
<td>20</td>
<td>0.6</td>
<td>0.4</td>
<td>0.88</td>
<td>0.12</td>
</tr>
<tr>
<td>Surah #2</td>
<td>24</td>
<td>6</td>
<td>4</td>
<td>0</td>
<td>21</td>
<td>0.4</td>
<td>0.6</td>
<td>0.76</td>
<td>0.24</td>
</tr>
<tr>
<td>Surah #3</td>
<td>24</td>
<td>6</td>
<td>2</td>
<td>0</td>
<td>11</td>
<td>0.5</td>
<td>0.5</td>
<td>0.78</td>
<td>0.22</td>
</tr>
</tbody>
</table>

Using the TOPSIS algorithm described previously, a preference vector \( V \) for each DM can be obtained with the weights of the criteria shown in Table 2 and the values shown in Table 3. The preference value of each DM is shown in Table 4. According to the data in the table, it appears that the three DMs have significantly different preference values. DM #3, in particular, has different preferences than DM #1 and DM #2 in terms of Surah #1 score. DM #3 puts Surah #1 in the last place, while DM #1 and DM #2 put Surah #1 in the second.

To combine the values of the three DMs, the geometric mean is used, with the weights of each DM being different depending on how much memorization they have. We give DM #1 and DM #2 a weight of 3.0, while for DM #3, it is 4.0. The results of the combined values of the geometric mean are shown in Table 5. We can conclude that the best surah sequence is Surah #2, Surah #1, then Surah #3.

Table 4. Decision-makers preference score for each surah

<table>
<thead>
<tr>
<th></th>
<th>Surah #1</th>
<th>Surah #2</th>
<th>Surah #3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM #1</td>
<td>0.28</td>
<td>0.32</td>
<td>0.26</td>
</tr>
<tr>
<td>DM #2</td>
<td>0.34</td>
<td>0.36</td>
<td>0.31</td>
</tr>
<tr>
<td>DM #3</td>
<td>0.29</td>
<td>0.35</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Table 5. Decision-makers preference score for each surah

<table>
<thead>
<tr>
<th></th>
<th>Surah #1</th>
<th>Surah #2</th>
<th>Surah #3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final score</td>
<td>0.326</td>
<td>0.368</td>
<td>0.313</td>
</tr>
<tr>
<td>Rank</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

In the actual process, this calculation is carried out on 114 surahs using the weights of the seven DMs. After ranking the overall value of the surah, the system will select the top five surahs to be returned and tested to the user.

Evaluation results
The human evaluators selected from DMs were asked to verify the results of the system. Because there is no ground truth, each evaluator is asked to assess whether the surahs given by the system to the ten personas are suitable. The assessment procedure is carried out in the following way:

1. First, we randomly generate ten personas who have memorized juz 30 and have done 50 tests.
2. For each persona, the evaluator will receive the frequency of each surah appearing on the memorization test, the number of the wrong answer made on the surah, and the last time the surah appeared.
3. The evaluator will select five surahs for each persona based on the data.
4. The evaluator will then compare the five selected surahs with the system selection. If there is a different choice of surahs between the system and the evaluator, the evaluator will determine whether the system choice is still acceptable or not.
The percentage of the selected surahs by the system that has been assessed as acceptable by the human evaluators is shown in Table 6. The ten personas are numbered #1 to #10.

Table 6. Evaluation result

<table>
<thead>
<tr>
<th></th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>#5</th>
<th>#6</th>
<th>#7</th>
<th>#8</th>
<th>#9</th>
<th>#10</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluator 1</td>
<td>60</td>
<td>80</td>
<td>80</td>
<td>100</td>
<td>20</td>
<td>80</td>
<td>80</td>
<td>60</td>
<td>80</td>
<td>80</td>
<td>72</td>
</tr>
<tr>
<td>Evaluator 2</td>
<td>100</td>
<td>80</td>
<td>80</td>
<td>100</td>
<td>20</td>
<td>100</td>
<td>80</td>
<td>60</td>
<td>80</td>
<td>60</td>
<td>84</td>
</tr>
<tr>
<td>Average of all evaluators</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>78</td>
</tr>
</tbody>
</table>

**Discussion**

**Evaluation results**

In Table 6, we can see that evaluators agreed with 78% on average of the system’s choice of surahs. Evaluator 1 and Evaluator 2 have a similar agreement despite differing backgrounds. Evaluator 1 and Evaluator 2 agreed on average 72% and 84% of the system’s choice of surahs, respectively. Because evaluators come from DMs who indirectly participate in designing the algorithm on the system, it appears that the system is successfully able to represent DMs.

We can also see that DMs mostly approve the system’s selected surahs from the table. In user persona #4, the evaluator agreed to all the tested surahs and deemed following the user’s needs. Meanwhile, in user persona #5, the evaluator felt that the system chose surahs that did not match the user’s needs. From the results of our analysis, user persona #5 has many surahs that are rarely tested, and evaluators are more likely to choose those surahs, but the system tends to choose surahs that are often wrong.

**Feature extension**

Figure 3 shows that the form of testing available in the current version of the application is still basic, consisting solely of a self-assessment test, as can be seen. It is possible to develop other types of testing methods in the future to generate variations, for example, using gamification.

In the future, this test feature can be added with speech recognition technology such as Al-Quran speech verification (Rajagede et al., 2021) or speech-to-text recognition (Muhammad et al., 2012; Gerhana et al., 2018). This technology will further automate the assessment to improve the testing process than the current version. This is in line with the tasmi’ method, which tests memorization orally, where the memorizer must recite a passage of the Qur’an.

**Conclusion**

We developed a personalized Al-Quran memorization test system based on user data in this research. We use the GDSS method using the TOPSIS algorithm for surah ranking and the weighted geometric mean for aggregation between decision-makers. The algorithm is implemented as an API with every call. The system will receive the criteria matrix and return the top five surahs. We evaluated the system with the help of a human evaluator, and the system showed 78% agreement with the human evaluator. In the future, the memory test can be improved by adding Al-Quran speech recognition and more advanced features.

**Acknowledgment**

This research was supported by the Directorate of Research and Community Service at Universitas Islam Indonesia through a beginner research grant with the number 003/Dir/DPPM/70/Pen.Pemula/III/2021.

**References**


