



An Analysis of Islamic Inheritance System Under Object-Oriented Paradigm

A. H. M. Sajedul Hoque^(✉), Sadia Tabassum, Rashed Mustafa,
Mohammad Sanaullah Chowdhury, and Mohammad Osiur Rahman

University of Chittagong, Chittagong, Bangladesh
{hoque.cse,rashed.m,s.chowdhury,drosi}@cu.ac.bd

Abstract. The wealth of deceased should be distributed precisely and quickly according to inheritance laws in order to avoid the unrest situation in the society. A professional software solution can serve this purpose. The professional software can be developed through analyzing, designing, building and testing of the manual system. As the analysis is the first and crucial stage of software development life cycle, this paper focuses on the analysis of Islamic Inheritance System (IIS). This analysis can be performed employing either structured or object-oriented approach. The structured analysis depends on the functions of the IIS, while objects are the main part in the object-oriented analysis technique. The one demerit of structured analysis is that the IIS is modeled using Data Flow Diagram (DFD) which is not standard diagram in software engineering community. In addition, the structured analysis phase of IIS does not provide any knowledge of internal modules. In order to address these problems, this paper concentrates on the object-oriented analysis of IIS where standard UMLs are utilized describing the system. The object-oriented analysis is accomplished by generating use case diagram, activity diagram and domain class diagram of IIS which can be easily used in the design and implementation phase of software development process.

Keywords: Islamic Inheritance System · Object-Oriented Analysis · Activity Diagram · Domain Class Diagram

1 Introduction

People may be crazy for earning assets either in right or wrong way. This sort of tendency increases the crimes in the society. This problem is seriously visible after a man or woman is died. The way of distributing of the assets of deceased among relatives is a crucial task. The distribution laws vary from country to country or religion to religion. This article focuses on the Islamic Law. So, it is felt that a computerized Islamic Inheritance System (IIS) is necessary for easing the computation to avoid unrest in the society. In order to make a professional

IIS, the system should be analyzed, designed, implemented and tested respectively. Among those phases, this article emphasizes on the analysis phase which determines all necessary requirements to build a robust IIS and this phase can be performed employing either structured or object-oriented technique. The structured analysis depends on developing Data Flow Diagrams (DFDs) over involved processes in IIS done by S. Tabassum et al. [10]. Those DFDs are transformed into structure chart as Architectural Design consisting of functions as module. However, the constructed DFDs in the structured IIS are not standard diagrams which may arise confusion among software engineering community. Secondly, although DFDs of IIS show the name of processes, a clear idea about its modules are not found. Finally, as the processes of IIS in DFDs cannot be extended, any types of changes or modifications on those DFDs are not easy. Considering these three problems, this paper focuses only on the object-oriented analysis phase for IIS. In this phase, the objects of IIS from its description are used to form a conceptual class diagram from which a class-based architectural design can be built in the Object-Oriented Design phase. This paper consists of five sections. The Sect. 2 describes the related works. The overall methodology of object-oriented analysis to determine the domain class diagram is illustrated in Sect. 3. Section 4 presents the result and discussion of the paper. Finally, the paper is concluded in conclusion section.

2 Related Works

Software makes the professional life of human being straightforward and it has become increasingly part of all aspect of daily life since 1960. From then on, it is a great challenge to produce a high-quality software being maintainable, usable, reliable and acceptable. Apportioning the shares of dead among live heirs according to inheritance laws is monumental business process in the society to prevent riots and the process of computation varies from religion to religion. This article basically focuses the Islamic Inheritance Law. In order to computerize the business process, software engineering techniques can be used to build the professional software solution named Inheritance system. So far, few works have been done on Islamic law. Firstly, H. Alshahad et al. presented a concept for calculating the shares of heirs in 2015 where a decision table having all inheritance related Islamic rules is employed [3]. Then, an expert system for the computation integrating 43 rules on Islamic law using CLIPS language was proposed by Alaa N. Akkila et al. (2016) [2]. In 2018, Zulkifli et al. applied Rapid Application Development (RAD) model for developing an android application of inheritance system [14]. Then, a family ontology based architecture of Islamic Inheritance System named AraFamOnto was provided by S. Zouaoui et al. where the ontology indicated family relationship [13]. Mathematical models for each heir should be developed for the business logic of each module of Islamic Inheritance System. In 2017, K. Babalola gave all mathematical models for computing the desired shares of live heirs of deceased [5]. But these expressions were not well-expressed for building the Inheritance system. In 2019, S. Tabassum et al.

presented well-expressed computational models for heirs [10]. Few prominent web applications for inheritance Calculators are found. Firstly, Halis Aydemir developed an inheritance calculator named “Division of Inheritance” where the results are displayed in percentage, fraction and bar chart [4]. In addition, a web application entitled ShariahStandards.org depicts the allocated shares of heirs in a circle with different shades of colors [9]. Moreover, the lubnaa is another inheritance calculator which is capable of computing shares of nine heirs in fraction [6]. Furthermore, the Access to Information (a2i) project of the office of the prime minister of Bangladesh is running a web application named Uttaradikhar based on the amended Islamic laws of Bangladesh constitution [11]. Over and above, Najeeb built another web application named Islamic Inheritance Calculator adding all Islamic laws considering 29 legal heirs over five generations of heirs [8]. Finally, the most comprehensive Islamic Inheritance System developed by Muhammad Waqas integrated all possible conditions of Islamic inheritance laws [12]. Although all aforementioned systems are available to use, their architectural designs are not published. Thus, the knowledge about the modules of those system is unknown which makes difficult to extend or maintain the system with extended rules. The precondition of developing an architectural design is to analyze the system for finding the requirements. Although S. Tabassum et al. [10] did the structured analysis approach for Islamic Inheritance System (IIS), there is no works for object-oriented analysis for IIS. For this reason, the main focus of the article is to analyze the IIS employing Object-Oriented paradigm.

3 Methodology

A very meticulous procedure of this work has to be followed in order to achieve the intended outcomes. The procedure will include several steps befitting to the objectives of this research. At the starting of the process, the context of the system is analyzed to find out the important services to be given to the user. According to the object-oriented technique, the determined services of the system are modeled using a UML diagram known as use case diagram. Then, the scenarios of those use cases are modeled where the sequence of interactions between system and user are described step by step. Next, the activities under the sequence of interactions are modeled using another UML diagram called Activity Diagram. After that, a conceptual class diagram using Unified Process methodology is built after finding the responsible autonomous agents or objects from the textual description of fixed use cases. The detail description of the methodology is presented in the following subsequent sections.

3.1 System Context Modeling

An application software without any useful service is meaningless to the user. Thus, every software system should have one or more services or use cases. In order to find those use cases, the context of the desired software system should be analyzed thoroughly. This paper focuses on one use case to distribute

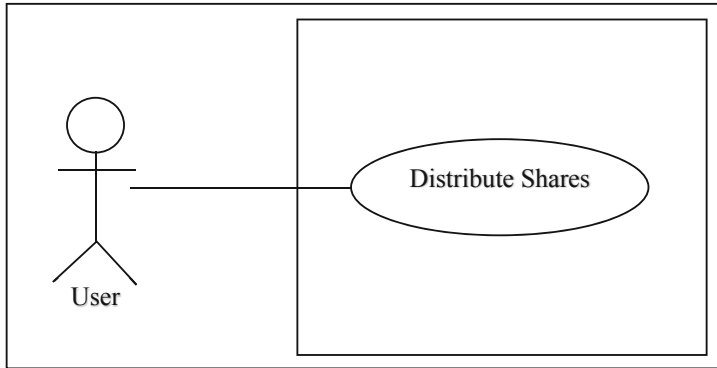


Fig. 1. Use Case Diagram

the wealth of deceased among live relatives illustrated in Fig. 1 using UML use case diagram. The distribution process varies from creed to creed. In Islam, the way of distribution is mentioned in the holy Quran and Hadith [1, 7]. At the beginning of the process, the **Total Effective Shares (TES)** for allotting are computed by deducing his or her loan, will and funeral costs from the remaining shares. Then, the **TES** is distributed among the prescribed heirs: **Spouse (Husband or Wives), Daughter, Father, Mother, Grandfather, Paternal Granddaughter, Sister(full), Paternal Grandmother and Maternal Grandmother** who are specified with their deserved portion in the Holy Quran. Among those relatives, **Spouse** and **Mother** must get shares without any condition, if they are alive. The rest of the prescribed relatives have one or more preconditions. For example, the **Father** will be considered for prescribed shares, if the deceased has no offspring. The Table 1 shows all preconditions of eligible prescribed heirs and their respective computational model for computing the portions [10].

The acronyms of computational model in Table 1 are as follows:

1. hNC() : hasNoChildren()
2. NW : number of Wives
3. hSD() : hasSingleDaughter()
4. ND : number of Daughters
5. hNMS() : hasNoMultipleSiblings()
6. hNMGM() : hasNoMaternalGrandmother()
7. hNF() : hasNoFather()
8. hPGM() : hasPaternalGrandmother()
9. hSGD() : hasSingleGranddaughter()
10. NGD : number of Grand Daughters
11. hSS() : hasSingleSister()
12. NS : number of sisters

Table 1. Summary of Prescribed Heirs

No	Heirs	Preconditions	Computational Model
1	Husband	Confirmed	$\frac{TES}{4} (1 + hNC())$
2	Wives	Confirmed	$\frac{TES}{8} (1+hNC())$
3	Daughters	Deceased has no son	$\frac{TES}{2} + \frac{TES}{6} \frac{NW}{ND} (1-hSD())$
4	Father	Deceased has offspring	$\frac{TES}{6}$
5	Mother	Confirmed	$\frac{TES}{6} (1 + (hNC() \text{ AND } hNMS()))$
6	Paternal Grandfather	Deceased has no father and has offspring	$\frac{TES}{6}$
7	Paternal Grandmother	Deceased has no mother and father	$\frac{TES}{12} (1 + hNMGM())$
8	Maternal Grandmother	Deceased has no mother	$\frac{TES}{12} (1 + (1-(hNF() \text{ AND } hPGM()))$
9	Paternal Granddaughters	Deceased has no son, grandson and at most one daughter	$\frac{TES}{6} \cdot hSD() + \frac{TES}{2} (1-hSD()) + \frac{TES}{6} (1-hSGD())$ NGD
10	Sisters (full)	Deceased has no daughter, son, paternal grandson, paternal granddaughter, father, paternal grandfather and brother	$\frac{TES}{2} + \frac{TES}{6} \frac{(1-hSS())}{NS}$

Every computational model has one or more predicates which return integer value 1(one) whenever those are true. However, the models for precondition of **Father** and **Grandfather** proposed by S. Tabassum [10] have not been covered completely. The modified models are shown in Table 1 where if deceased has offspring, the **Father** will be eligible for prescribed shares. Similarly, the condition for **Grandfather** is not to have **Father** and offspring. After distributing the prescribed shares, one out of four cases may happen. Firstly, the **Total Allotted Prescribe Shares(TAPS)** is equal to **TES**. Then, the earned prescribed shares will be final shares. Secondly, there are still some remaining shares. Then, the given list of heirs of deceased is checked if there are eligible residual heirs. The matching conditions for each eligible residual heir are illustrated in Table 2.

If the list of residual heirs is not empty, the remaining shares are distributed among male and female candidates employing the ratio 2:1. If the number of male and female candidates are **NM** and **NF** respectively, the portion of **Male Residual Shares (MRS)** and **Female Residual Shares (FRS)** are computed as Eq. 1 and 2 [10].

$$MRS = 2 \cdot \frac{TES - TAPS}{2 \cdot NM + NF} \quad (1)$$

$$FRS = \frac{TES - TAPS}{2 \cdot NM + NF} \quad (2)$$

Then, the individual earned shares in prescribed and residual category are summed and this is the final shares of heirs of deceased. Thirdly, in spite of having remaining shares, there is no residual heirs of deceased. In this case, the **Earned Prescribed Shares (EPS)** are proportionately increased except

Table 2. Summary of Residual Heirs

No	Relatives	Preconditions
1	Son	Confirmed
2	Daughter	Has Son
3	Paternal Grandson	Has no Son
4	Paternal Granddaughter	Has no Son AND Has Paternal Grandson
5	Father	Has no (Son AND Grandson)
6	Grandfather	Has no (Son AND Grandson AND Father)
7	Brother	Has no (Son AND Grandson AND Father AND Grandfather)
8	Sister	Has no (Son AND Grandson AND Father AND Grandfather) AND Has (Brother OR Daughter OR Granddaughter)
9	Brother's Son	Has no (Son AND Grandson AND Father AND Grandfather AND Brother AND Daughter AND Granddaughter)
10	Son of Brother's Son	Has no (Son AND Grandson AND Father AND Grandfather AND Brother AND Daughter AND Granddaughter AND Brother's Son)
11	Paternal Uncle	Has no Son AND Has no (Grandson AND Father AND Grandfather AND Brother AND Daughter AND Granddaughter AND Brother's Son AND Son of Brother's Son)
12	Paternal Cousin	Has no (Son AND Grandson AND Father AND Grandfather AND Brother AND Daughter AND Granddaughter AND Brother's Son AND Son of Brother's Son AND Paternal Uncle)

Spouse, whereas S. Tabassum considered **Spouse** in her model [10]. This situation is called **Radd** case [8]. Finally, the **TAPS** is greater than **TES** which is known as **Awal** case [8]. In this situation, the **EPS** are proportionately decreased. For both cases, the **Final Shares(FS)** of **heir** in the list of eligible prescribed heirs is determined using the Eq. 3, where **RS** and **TAPS** are **Remaining Shares** and **Total Allotted Prescribed shares** respectively.

$$FS(\text{heir}) = EPS(\text{heir}) \left(1 + (1 - \text{isSpouse}(\text{heir}) \cdot \text{isPositive}(\text{RS})) \frac{\text{RS}}{\text{TAPS}} \right) \quad (3)$$

Here the function **EPS(heir)** returns the Earned Prescribed Shares of **heir** and **isSpouse(heir)** and **isPositive(RS)** are two predicates. The former predicate returns one if the **heir** is spouse, while the later predicate gives one for being positive value of **RS**. Otherwise, both predicate provides zero. It has been seen that when both predicates are 1, which indicates the heir is spouse and the sum of distributed prescribed shares is less than **TES**, the shares will not be increased. Otherwise, it will be increased. However, the whole scenario of the use case in Fig. 1 is structured in Fig. 2. This scenario consists of several activities. These activities and the flows among activities are shown in Fig. 3.

3.2 Domain Class Modeling

In object-oriented approach, domain modeling means finding the embodied autonomous agents or objects and their relationships from the scenario of use

UC1	: Distribute Shares
Actors	: User
Preconditions	: The Home Screen of the system should be displayed
Main Success Scenario	: <ol style="list-style-type: none"> 1. Home Screen prompts user to select the live heirs and to enter the number of heirs, total shares and the bequest shares of the deceased 2. User selects the live heirs and gives the number of heirs, total shares and bequest shares of the deceased. 3. System takes the given information of deceased from the Home Screen. 4. System makes the list of heirs for prescribed shares measuring the eligibility of the heir. 5. System computes the portions of shares for all eligible heirs of prescribed shares from the total shares of deceased. 6. System sums all of computed prescribed shares. 7. System calculates the rest of shares subtracting the sum of prescribed shares from the total shares of deceased. 8. System makes the list of heirs for residual shares measuring the eligibility of the heir. 9. System computes the portion of residual shares for eligible heirs from the rest of the shares. 10. System computes the final shares of individual summing his or her got portions in prescribed and residual shares.
Post conditions	: The Output Screen of the system shows the computed individual prescribed, residual and total final shares.
Alternative Courses	: <ol style="list-style-type: none"> 2.a. The number of Husband, Father, Mother, Paternal Grand Father, Paternal Grand Mother, Maternal Grand Mother is more than One <ol style="list-style-type: none"> 2.a.01. System will show a warning message as “The number is not possible”. 2.a.02. Resume @ step 1 2.b. The number of wives is more than four <ol style="list-style-type: none"> 2.b.01. System will show a warning message as “The number is not possible”. 2.b.02. Resume @ step 1 7.a. The rest of the shares is equal to zero <ol style="list-style-type: none"> 7.a.01. System will finalize the Computed Prescribed Shares as Final Shares 7.a.02. Resume @ step 10 7.b. The rest of the shares is less than zero <ol style="list-style-type: none"> 7.a.01. System will be scaled down all Computed Prescribed shares under Total shares of deceased 7.a.02. Resume @ step 10 8.a. The list is empty <ol style="list-style-type: none"> 8.a.01. System will be scaled up all Computed Prescribed shares under total shares of deceased 8.a.02. Resume @ step 10

Fig. 2. Use Case Description

cases. For this purpose, it is necessary to retrieve three types of classes: Entity Classes, Control Class and Boundary Classes. After analyzing the steps in Fig. 2, six entity classes, two boundary classes and one control class are modeled. The six entity classes are **DeceasedModel**, **PrescribedShares**, **PrescribedHeirs**, **ResidualShares**, **ResidualHeirs** and **FinalShares**. Among these entity classes, steps 1, 2 and 3 in scenario of Fig. 2 implies **Deceased-Model** class in order to deal the information of deceased including list of heirs, number of each heir and the total effective shares. The responsibilities of the **PrescribedShares** and **PrescribedHeirs** found from step 4 to 7 are to determine

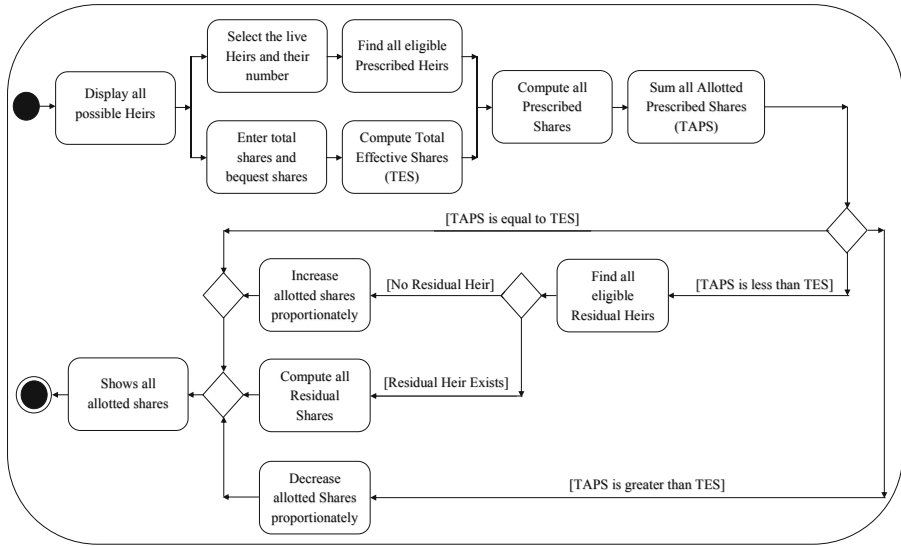


Fig. 3. Activity Diagram

the eligible prescribed heirs and the portion of those heirs. The **Prescribed-Shares** class holds the list of all eligible prescribed heirs of deceased and sum of distributed prescribed shares whereas the **PrescribedHeir** class includes type and number of heirs and respective portion of shares. Similarly, the step 8 and 9 hints about the two other entity classes named **ResidualShares** and **ResidualHeir** to find the eligible residual heirs and the portion of shares of those heirs respectively. The last final entity class called **FinalShares** having two attributes (list of heirs and list of shares) combines the earned portion in prescribed and residual category according to the special rules of Islamic Inheritance. Furthermore, two boundary classes named as **HomeScreen** and **OutputScreen** are seen in step 1 and precondition of Fig. 2 respectively. The object of **HomeScreen** class will take the list of heirs of deceased, their number and the amount of assets, whereas the **OutputScreen** type object shows the computed prescribed, residual and final shares of each heir. Finally, the whole business rules for distributing shares are conducted by a control class named **DistributeShares**. This class receives all necessary information of deceased from **Homescreen** type object, computes prescribed, residual and final shares employing six aforementioned entity classes and at the end shows those shares utilizing the object of **OutputScreen** class. The domain class diagram for distributing shares among heirs of deceased is shown in Fig. 4.

4 Result and Discussion

The purpose of the analysis phase for developing a professional software is to retrieve the required artifacts for design phase. S. Tabassum et al. used struc-

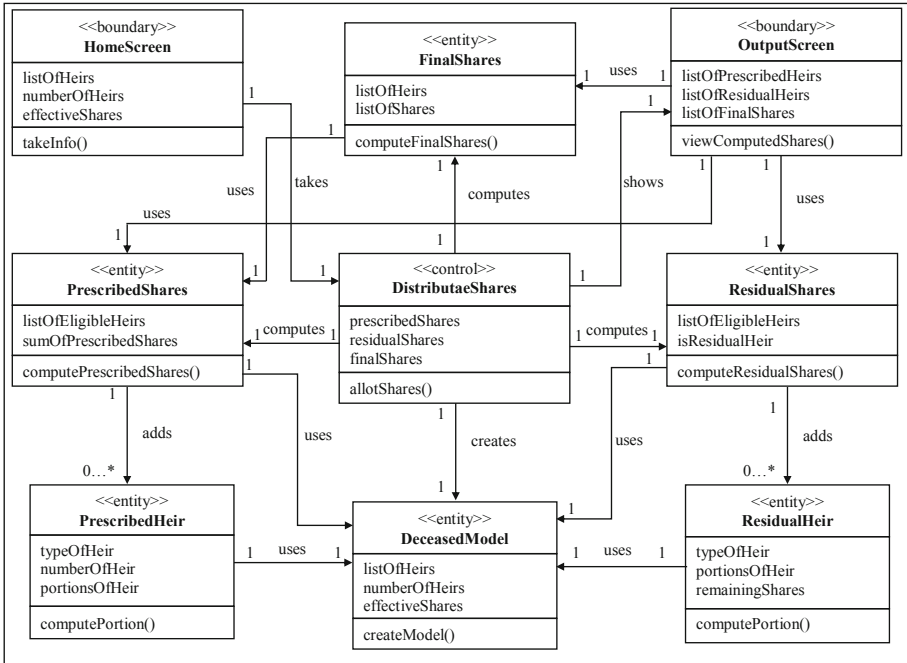


Fig. 4. Domain Class Diagram

tured analysis technique to analyze the Islamic Inheritance system employing level 0,1 and 2 DFDs [10]. Compared to the structured analysis approach for IIS, this work is more robust because of using object-oriented analysis technique. In this analysis, some standard UML artifacts including **Use Case Diagram** in Fig. 1, **Activity Diagram** in Fig. 3 and **Domain Class Diagram** in Fig. 4 are produced, whereas the structured analysis phase generates only DFDs which are not UML. So, this work is affluent in terms of diagram standardization. Secondly, there is no way to get idea about the module of system in structured analysis phase and we have to wait until design phase for structure chart having functions as module. On other hand, the Domain class diagram in Fig. 4 in object-oriented analysis phase consists of classes which are considered as modules for Architectural Design. Finally, classes in class diagram can be inherited to create new classes and functions inside the classes can also be overridden which makes this work very powerful. Therefore, it can be said that object-oriented analysis is superior to structured analysis for IIS.

5 Conclusion

A professional software solution for Islamic Inheritance System (IIS) has great demand to distribute shares of deceased among his or her heirs. The starting

phase of developing the professional software is analysis which is the main target of this work. In this paper, the analysis of the IIS has been done successfully by creating some artifacts including Use Case Diagram, Activity Diagram and Domain Class diagram. This work has valuable impact in the software engineering community, as the generated artifacts can be used to develop architectural design, detail design and build the desired IIS. In future, this work can be extended integrating other religions to build general inheritance system. The great limitation of this work is that complete object-oriented architecture of IIS is not developed. Thus, another future work is to develop the architectural design and detail design modeling sequence diagram, collaboration diagram and state machine diagram of the IIS from the produced domain class diagram.

References

1. Adelina Zuleika, N.P.D.: Islamic Inheritance Law (Faraid) and its economic implication. *Tazkia Islamic Finance Bus. Rev.* **8**(1), 97–118 (2013)
2. Akkila, A.N., Abu Naser, S.S.: Proposed expert system for calculating inheritance in Islam. *World Wide J. Multidisciplinary Res. Dev.* **2**(9), 38–48 (2015)
3. Alshahad, H.F., Abutiheen, Z.A.: Computation of inheritance share in Islamic Law by an expert system using decision tables. *Q. Adjudicated J. Nat. Eng. Res. Stud.* **1**(No. 1 and 2), 105–114 (2015)
4. Aydemir, D.H.: Division of Inheritance. https://kurandersleri.net/miras/en/Miras_en.html. Accessed 11 Aug 2021
5. Babalola, K.O.: The Inheritance of Legal Heirs: Mathematical Presentation. University of Ilorin, Department of Mathematics, September 2017
6. Lubnaa - Inheritance Calculator. <https://www.lubnaa.com/money/InheritCalc.php>. Accessed 10 Aug 2021
7. Muhammad, B.J.: The Islamic Law of Inheritance: Introduction and Theories. Abu Aisha Publishing (2020)
8. Najeeb: Islamic Inheritance Calculator (2014). <https://inheritance.ilmsummit.org/projects/inheritance/home.aspx>. Accessed 10 Aug 2021
9. *ShariahStandards.org*. <https://www.inheritancecalculator.net/>. Accessed 10 Aug 2021
10. Tabassum, S., Hoque, A.H.M.S., Twahura, S., Rahman, M.O.: Developing an Islamic Farayez system applying software engineering. *Jurnal Kejuruteraan* **31**(1), 25–38 (2019). [https://doi.org/10.17576/jkukm-2019-31\(1\)-04](https://doi.org/10.17576/jkukm-2019-31(1)-04)
11. Uttarahdihkar: Access to information (A2I), prime minister's office of Bangladesh (2016). <https://xn-d5by7bap7cc3ici3m.xn-54b7fta0cc/index.php?lang=en>. Accessed 10 Aug 2021
12. Waqas, M.: Islamic Inheritance Calculator (2017). <https://www.inheritance-calculator.com/index.php>. Accessed 11 Aug 2021
13. Zouaoui, S., Rezeg, K.: Islamic inheritance calculation system based on Arabic ontology (AraFamOnto). *J. King Saud Univ. Comput. Inf. Sci.* **33**(1), 68–76 (2018). <https://doi.org/10.1016/j.jksuci.2018.11.015>
14. Zulkifli, A.N., Batiha, Q.A., Qasim, M.M.: Design and development of M-Faraid: an Islamic inheritance mobile app. *J. Adv. Res. Dyn. Control Syst.* **10**(10 Special Issue), 1569–1575 (2018)