

Development of AR-based Mobile Learning Application for Ruminant Anatomy

Koo Mei Qi¹, Mohd Norasri Ismail^{1*}

¹Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: <https://doi.org/10.30880/aitcs.2022.03.02.036>

Received 19 July 2022; Accepted 26 October 2022; Available online 30 November 2022

Abstract: The use of augmented reality technology in education is growing in popularity because it makes learning less tedious and increases learning. Anatomy has always been considered difficult for medical students to learn and memorize since there are so many anatomical terms. Also, mobile augmented reality (AR) learning application is limited. Thus, the objective of this project is to design a mobile learning application for ruminant anatomy entitled “Ruminant Anatomy (AR)” using marker-based AR on the Android platform, to develop an interactive learning application using marker-based AR and to perform functional and user acceptance testing on the developed application for the target users. Users can view, move, resize, and rotate the 3D models and videos in the AR session, learn the anatomical information in the learning module and take quizzes in the quiz module. The methodology utilized in the application development is Multimedia Mobile Content Development (MMCD). The user acceptance test conducted among a random target audience of 30 IVM students shows that the target user satisfaction is 80% and the application is functioning at its optimum stage. This application is expected to provide an interactive way for users to assist in grasping ruminant anatomy quickly and efficiently.

Keywords: Ruminant Anatomy, Marker-based AR, Mobile learning

1. Introduction

Augmented reality (AR) is an experience in which designers employ computer-generated input to augment parts of the user's physical world [1]. The usage of augmented reality technology in education is becoming more common, as the immersion of augmented reality reduces the boredom of learning and improves learning. Thus, the objectives of this project are to design an AR-based mobile learning application for ruminant anatomy entitled “Ruminant Anatomy (AR)”, to develop an interactive learning application using marker-based AR, and to perform functional testing and user acceptance testing on the developed application to the target user.

Anatomy is a science that allows us to understand the inner structure of life on earth, and it is because of anatomy that the inner structure of most plants and animals is understood. However, anatomy has long been considered difficult for medical students to learn and memorize because there are so many

anatomical terms [2]. In addition, in the teaching process, teachers often adapt traditional teaching methods based on classroom teaching [3], while the existing experimental teaching links are weak, and experiments mostly rely on static observation, which makes the course boring.

Nowadays, improvements in augmented reality technology are allowing students to learn ruminant anatomy on their mobile devices, addressing these concerns. As a result, this project proposes Ruminant Anatomy (AR), an interactive application for users to assist them in grasping ruminant anatomy quickly and efficiently. The application not only utilizes augmented reality technology to change the way users learn by providing an immersive experience but also has a variety of rich multimedia elements to make learning more entertaining and interesting. Therefore, users can choose to use this application to learn about ruminant anatomy without having to go to the dissection lab.

2. Related Work

Three applications on Google Play related to proposed topics such as animal anatomy are selected for review. The next section will go over a comparative analysis of existing and proposed applications.

2.1 Domain Case study Background

The fundamental distinction between ruminants and non-ruminants is that there are four compartments in the stomach of ruminants, namely, the rumen, reticulum, omasum, and abomasum. Ruminants have unique anatomical and physiological features that enable them to make significant contributions to a variety of sectors, and their output has played an important role in the success of communities all over the world for ages [4].

Augmented reality technology plays a crucial role in the anatomy learning process. After all, in the absence of physical objects, students can learn animal anatomy through augmented reality technology at any time and everywhere with just one smartphone.

2.2 Marker-based AR

Marker-based AR, also known as image recognition or recognition-based AR, works by recognizing markers. Markers are required to trigger features in marker-based AR. Markers are distinct patterns that are easily identified and processed by the camera and are visually independent of the surrounding environment, and they can be made of paper or real-world objects. Markers must be distinctive, and stock photographs should be avoided because they may have previously been used in other projects or applications [5]. To avoid such issues, it is best to gather unique information or construct a custom design.

2.2 Comparative Analysis of Existing and Proposed Applications

This section discusses the comparison between the three reviewed applications which are Animal anatomy physiology [6], Anatomia Veterinária [7], Veterinary Anatomy [8], and the proposed application, Ruminant Anatomy (AR) based on several features. The comparison is tabulated in Table 1.

Table 1: Comparison between reviewed applications and proposed application

Features	Animal anatomy physiology	Anatomia Veterinária	Veterinary Anatomy	Ruminant Anatomy (AR)
Augmented reality (AR)	AR feature is not provided.	AR feature is not provided.	AR feature is not provided.	Marker-based AR feature is provided.
Content focus	Animal health topics.	Horse skeleton.	The skeleton of three animals.	Basic information, body parts, skeleton anatomy, muscle anatomy, and digestive system of six ruminants.
Learning module	The learning module is text-based.	The learning module is image-based.	The learning module combines images and 3D models.	The learning module combines text, images, videos, and 3D models.
Quiz module	The quiz module is not provided to users.	The quiz module is not provided to users.	The quiz module is provided to users.	The quiz module is provided to users.
Language	This is an English language app.	This is a Portuguese language app.	This is an English language app.	This is an English language app.
Operating system	Android version 4.1 or later is required.	Android version 4.1 or later is required.	Android version 5.0 or later is required.	Android version 7.0 or later is required.
Ads	There are no ads in the app.	There are ads in the app, and ads do not affect the use of the app.	There are ads in the app that can only be turned off after 5 seconds.	There are no ads in the app.

3. Methodology

Multimedia Mobile Content Development (MMCD) approach was utilized in this project, which aids developers in achieving their objectives in a short amount of time. The approach consists of five stages which are application idea creation, analyzing the structure, designing the process, developing main functions, and testing, all of which are used to ensure that the application development process runs smoothly and on time. The MMCD model is shown in Figure 1.

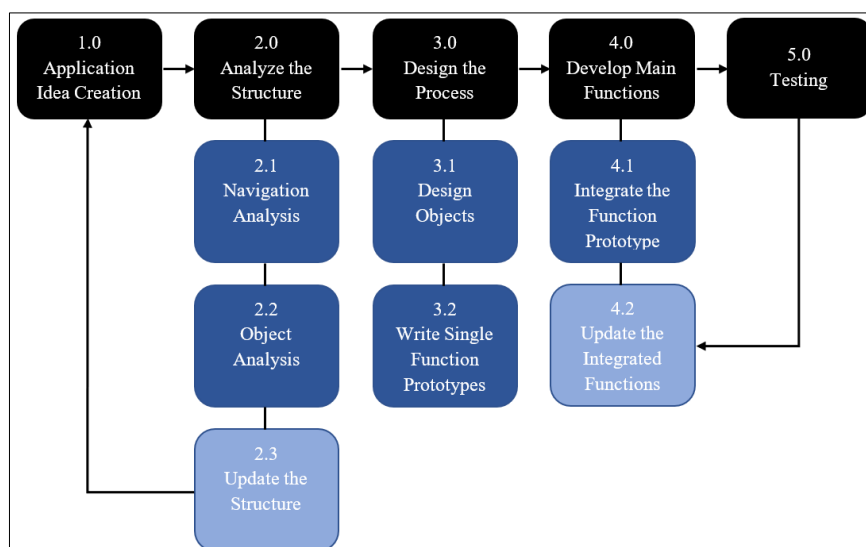


Figure 1: MMCD Model [9]

3.1 Application Idea Creation Stage

The first stage of the MMCD approach is the application idea creation, which prepares all the necessary information before moving on to the design and development stage of the application [9]. The application idea creation checklist is shown in Table 2, while the user requirement analysis is shown in Table 3.

Table 2: Application idea creation checklist [9]

Item	Note
Type of application	Mobile learning application
Target device	Android mobile device
Target users	Students of Institut Veterinar Malaysia (IVM)
Graphical user interface (GUI)	Homepage interface, Learning module, Quiz module, About This App module
Text	Explanation in the learning module, the questions and answers in the quiz module, and acknowledgment in the about this app module.
Images	Background images, images in the learning module, icons, buttons
Video	Real ruminant videos, application tutorial video
Audio	Background music, answer sound effects
Augmented reality (AR)	The ruminants' 3D models and videos are displayed with maker-based AR.
Application synopsis	Ruminant Anatomy (AR) is an augmented reality-based English mobile learning app designed specifically for IVM students to help them grasp the anatomy of ruminants fast and effectively.

Table 3: User Requirement Analysis

Stakeholder	Role	Design Implications	Action Needed
Subject Matter Expert (SME)	Content consultant expert in related field	Based on the interview, <ul style="list-style-type: none"> • Easy to use 	<ul style="list-style-type: none"> • Apply different buttons for different purposes. • Provide a user guide.
		<ul style="list-style-type: none"> • Easy to navigate 	<ul style="list-style-type: none"> • The application should contain back, previous, next, redo, AR, home, and exit buttons. • Apply buttons of a consistent size. • Avoid complex navigation structures.
		<ul style="list-style-type: none"> • Simple user interface design 	<ul style="list-style-type: none"> • Use icons instead of text. • Use appropriate font sizes and types in the content.
		<ul style="list-style-type: none"> • Reliable content 	<ul style="list-style-type: none"> • The application should provide correct information about ruminant anatomy.
General User	End-user of the system	Based on the questionnaire, <ul style="list-style-type: none"> • Free to download 	<ul style="list-style-type: none"> • Create an application that is free to download and use.
		<ul style="list-style-type: none"> • Can be accessed anytime and anywhere 	<ul style="list-style-type: none"> • Create an application that does not require an Internet connection.
		<ul style="list-style-type: none"> • Easy to use 	<ul style="list-style-type: none"> • Create an application that provides navigation that is easy to understand.

3.2 Analyze the Structure Stage

The second stage of the MMCD approach is to analyze the structure. The two subcomponents to be analyzed in this stage are navigation and the application's objects. Since navigation and objects are crucial characteristics of a well-structured MMCD, these two subcomponents must be thoroughly examined to avoid negatively impacting the design stage and development process. The content structure checklist is shown in Table 4.

Table 4: Content structure checklist [9]

Item	Note
Layers design	Layer 1: Background images Layer 2: Content Layer 3: C# Scripting for Unity
Frame design	Frame 1: Homepage interface (choose module) Frame 2: Family selection interface (choose ruminant family) Frame 3: Topic selection interface (choose the topic) Frame 4: Topic 1 (Basic information)

Table 4: (Continued)

Item	Note
Frame design	Frame 5: Topic 2 (Body parts) Frame 6: Topic 3 (Skeleton anatomy) Frame 7: Topic 4 (Muscle anatomy) Frame 8: Topic 5 (Digestive system) Frame 9: Quiz module Frame 10: Score interface Frame 11: About interface (choose acknowledgment/tutorial) Frame 12: Acknowledgment interface Frame 13: Tutorial interface Frame 14: Exit warning interface
Menu and navigation	<ul style="list-style-type: none"> • Home button • Back button • Previous and next button • AR button • Redo button • Exit button
Number of main GUI	<ul style="list-style-type: none"> • Application logo
Images	<ul style="list-style-type: none"> • Background images • Images in the learning module • Icon and buttons
Video	<ul style="list-style-type: none"> • Real ruminant videos • Application tutorial video
Audio	<ul style="list-style-type: none"> • Background music • Correct and wrong answer sound effects (Quiz module)
AR	<ul style="list-style-type: none"> • The 3D models and videos are displayed with maker-based AR.

3.3 Design the Process Stage

The third stage of the MMCD approach is to design the process. The primary purpose of this phase is to prepare all of the items described in Table 4 and designing objects and writing single-function prototypes are the two subcomponents of this stage. The first prototype is finished at the end of the process since it is based on graphic and object design, object positioning on stage, and a single script in each frame. Finally, write the main function script to conclude the application's construction and ensure that it runs well. In this project, design software such as Adobe Photoshop and Canva are utilized for visual design, Blender is used to create 3D models, and Unity is used to build, merge, and integrate all assets required for the script process. The button design is shown in Table 5, while the structure of the application such as flowchart, navigational structure, and content structure is shown in Appendix A.

Table 5: Button Design


Button	Description
	<ul style="list-style-type: none"> • This is the learning button. • It will navigate the user to the family selection interface.

Table 5: (Continued)














Button	Description
	<ul style="list-style-type: none"> This is the quiz button. It will navigate the user to the quiz module.
	<ul style="list-style-type: none"> This is the about this app button. It will navigate the user to the about interface.
	<ul style="list-style-type: none"> This is the acknowledgement button. It will navigate the user to the acknowledgment interface.
	<ul style="list-style-type: none"> This is the tutorial button. It will navigate the user to the tutorial interface.
	<ul style="list-style-type: none"> This is the family selection button. This button will have a constant size with different words and colours to represent each family. It will navigate the user to the topic selection interface.
	<ul style="list-style-type: none"> This is the topic selection button. This button will have a constant size with different words to represent each topic. It will navigate the user to the related topic interface.
	<ul style="list-style-type: none"> These are yes and no buttons. Their purpose is to confirm whether the user wants to exit.
	
	<ul style="list-style-type: none"> This is the exit button. It will navigate the user to the exit warning interface.
	<ul style="list-style-type: none"> This is the home button. It will navigate the user to the exit warning interface.
	<ul style="list-style-type: none"> This is the back button. It will navigate the user to the previous interface.
	<ul style="list-style-type: none"> This is the redo button. It will navigate the user to redo the quiz.
	<ul style="list-style-type: none"> This is the AR button. It will navigate the user to the AR Instruction interface.

Table 5: (Continued)

Button	Description
	<ul style="list-style-type: none"> • These are the previous and next buttons. • It will navigate the user to the previous or next page of the content in the learning module since the content is more than one page.
	<ul style="list-style-type: none"> • This is the image button. • This button will represent each video with different ruminant images. • It will allow the user to watch the real ruminant video.
	<ul style="list-style-type: none"> • This is the image target button. • This button will have a constant size with different image targets to represent each AR session. • It will navigate the user to the AR session.
<p data-bbox="320 994 699 1061"><i>Click here to download all image targets.</i></p>	<ul style="list-style-type: none"> • This is the text button. • It will allow the user to download all the image targets.
	<ul style="list-style-type: none"> • These are answer buttons that are used in the quiz module. • The user needs to choose one of them to proceed to the next question.

3.4 Develop Main Functions Stage

The fourth phase of the MMCD approach is to develop main functions. Navigation between the homepage, family selection interface, topic selection interface, topic interface, and AR session are the main functions of this application. To function, all of these navigation buttons require scripting. The script in the learning module focuses on the ability to transition from one interface to the next, as well as from the interface to the AR session. For the quiz module, the script concentrates on randomly showing questions, verifying for proper responses, moving on to the next question, and scoring.

3.4.1 Frontend Development

User interface development is the process of designing and building an interface, focusing on the appearance or style of the interface. In this project, Canva is used in user interface design.

3.4.2 Backend Development

For backend programming and automation in Windows environments, the C# language is the chosen architecture. Therefore, the C# language is used for backend development in this project.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5
6  public class LoadSceneScript : MonoBehaviour
7  {
8      public void LoadScene(string sceneName)
9      {
10         SceneManager.LoadScene(sceneName);
11     }
12 }

```

Figure 2: LoadSceneScript.cs script

UnityEngine.SceneManagement is the library that must be enabled in order to write a single function prototype. This library manages the SceneManager object with the LoadScene behaviour and is used to load the scenes for the application. The LoadSceneScript.cs script is shown in Figure 2.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ExitWarningScript : MonoBehaviour
6  {
7      public GameObject exitPanel;
8
9      void Start()
10     {
11         exitPanel.SetActive(false);
12     }
13
14     public void ShowExitPanel()
15     {
16         exitPanel.SetActive(true);
17     }
18
19     public void ToggleResume()
20     {
21         exitPanel.SetActive(false);
22         Time.timeScale = 1;
23     }
24
25     public void ExitApp()
26     {
27         Application.Quit();
28     }
29 }

```

Figure 3: ExitWarningScript.cs script

The ShowExitPanal() is used to open the Exit Warning interface via the exit button on the Homepage or the home button in the learning module. The Start() is used to close the Exit Warning interface at the beginning, while the ExitApp() is used to exit the application. The ToggleResume() is used to allow the user to continue to use the application or continue to do the quiz. The ExitWarningScript.cs script is shown in Figure 3.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class EmbedLink : MonoBehaviour
6  {
7      public void LinkMarker()
8      {
9         Application.OpenURL("https://drive.google.com/file/d/1_7fBgMEY-i0V8LwKOTeCiVbGhqB1hqmm/view?usp=sharing");
10     }
11
12     public void LinkImages()
13     {
14         Application.OpenURL("https://drive.google.com/file/d/1eEv6kygs8g17ErP6E1VMaH5Q2gRWbhG/view?usp=sharing");
15     }
16 }

```

Figure 4: EmbedLink.cs script

The LinkMarker() and the LinkImages() are used to open the link of Google Drive in the browser via the application's button. Thus, the user can download all the image targets with the LinkMarker() and all the images used in the application with the LinkImages(). The EmbedLink.cs script is shown in Figure 4.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class BackgroundMusic : MonoBehaviour
6  {
7      private static BackgroundMusic instance = null;
8      private AudioSource audio;
9
10     private void Awake()
11     {
12         if (instance == null)
13         {
14             instance = this;
15             DontDestroyOnLoad(gameObject);
16             return;
17         }
18         if (instance == this) return;
19         Destroy(gameObject);
20     }
21
22     void Start()
23     {
24         audio = GetComponent<AudioSource>();
25         audio.Play();
26     }
27 }

```

Figure 5: BackgroundMusic.cs script

The BackgroundMusic class is used to play the background music of the application at the beginning, and prevent it from stopping while navigating to another interface. The BackgroundMusic.cs script is shown in Figure 5.

```

17     public void Answer()
18     {
19         if (isCorrect)
20         {
21             GetComponent<Image>().color = Color.green;
22             Debug.Log("Correct Answer");
23             quizManager.correct();
24         }
25         else
26         {
27             GetComponent<Image>().color = Color.red;
28             Debug.Log("Wrong Answer");
29             quizManager.wrong();
30         }
31     }
32 }

```

Figure 6: The part of the AnswerScript.cs script

The Answer() is used to show the correct answer in green colour and the wrong answer in red colour. The part of the AnswerScript.cs script is shown in Figure 6.

```

70     public void correct()
71     {
72         // One point for the correct answer
73         score += 1;
74         // Play right audio
75         quizAudio.PlayOneShot(rightAudio, 1.0f);
76         // Avoid repeating questions
77         QnA.RemoveAt(currentQuestion);
78         StartCoroutine(waitForNext());
79     }
80
81     public void wrong()
82     {
83         // Play wrong audio
84         quizAudio.PlayOneShot(wrongAudio, 1.0f);
85         // Avoid repeating questions
86         QnA.RemoveAt(currentQuestion);
87         StartCoroutine(waitForNext());
88     }

```

Figure 7: The part of the QuizManager.cs script

The correct() and wrong() are used to avoid repeating questions and to check the user's answers. For the correct response, the correct() is used to add one point and play the right audio, while for the wrong response, the wrong() is used to play the wrong audio. The part of the QuizManager.cs script is shown in Figure 7.

```

35 void generateQuestion()
36 {
37     if (QnA.Count > 0)
38     {
39         // Make the question random
40         currentQuestion = Random.Range(0, QnA.Count);
41         // Set the number of the question
42         questionNum++;
43         questionNumber.text = questionNum + "/" + totalQuestions;
44         // Set the question
45         quizQuestion.text = QnA[currentQuestion].Question;
46         setAnswers();
47     }
48     else
49     {
50         Debug.Log("Out of Questions");
51         gameOver();
52     }
53 }
54
55 void setAnswers()
56 {
57     for (int i = 0; i < options.Length; i++)
58     {
59         options[i].GetComponent<Image>().color = options[i].GetComponent<AnswerScript>().startColor;
60         options[i].GetComponent<AnswerScript>().isCorrect = false;
61         options[i].transform.GetChild(0).GetComponent<Text>().text = QnA[currentQuestion].Answers[i];
62
63         if (QnA[currentQuestion].CorrectAnswer == i + 1)
64         {
65             options[i].GetComponent<AnswerScript>().isCorrect = true;
66         }
67     }
68 }

```

Figure 8: The part of the QuizManager.cs script

Generating questions and answers is a basic function required by the application's quiz module. The generateQuestion() function is used to count the number of questions in the list by QnA value, display the number of questions, and generate 20 questions in random order for the user to answer. The setAnswer() is used to set the answer to the corresponding question. The part of the QuizManager.cs script is shown in Figure 8.

```

96 void gameOver()
97 {
98     scorePanel.SetActive(true);
99     quizScore.text = score + "/" + totalQuestions;
100 }
101
102 public void redoQuiz()
103 {
104     SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
105 }

```

Figure 9: The part of the QuizManager.cs script

The gameOver() is used to calculate the user's quiz score and display the score in the Score interface, while the redoQuiz() is used to allow the user to redo the quiz. The part of the QuizManager.cs script is shown in Figure 9.

3.4.3 AR Development

The most critical part of this project is AR development because the AR session in this project can make learning ruminant anatomy easier, and more interesting.

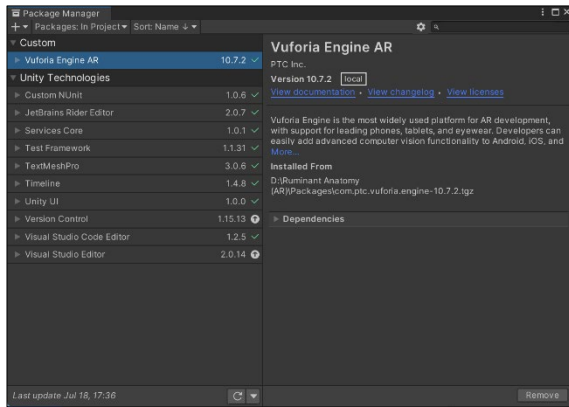


Figure 10(a): Vuforia Engine AR package

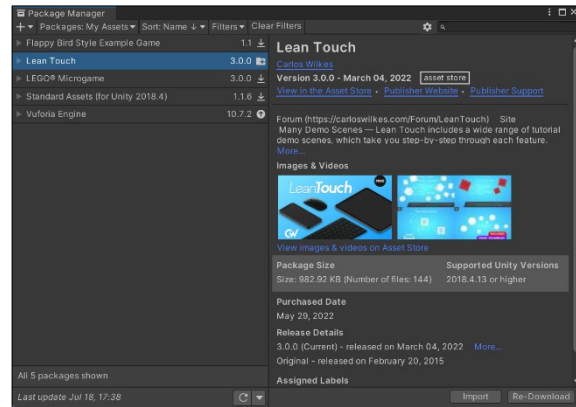


Figure 10(b): Lean Touch package

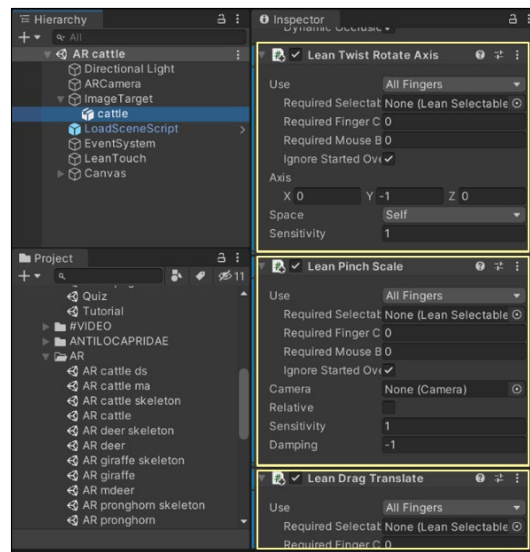


Figure 11: Scripts imported from Lean Touch package

The AR session of this project is implemented by using the built-in package in Unity which is Vuforia Engine AR. The interactive features of moving, resizing, and rotating 3D models and videos in the AR session are implemented by attaching scripts imported from the Lean Touch package. The built-in packages for AR and interactive implementations in Unity are shown in Figures 10(a) and 10(b), while the scripts imported from the Lean Touch package is shown in Figure 11.

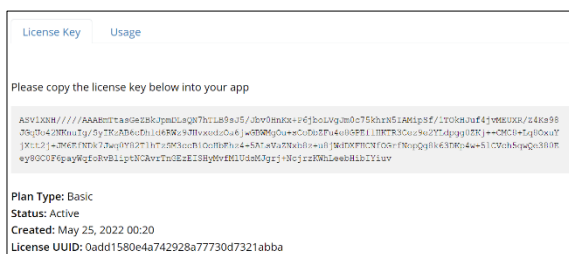


Figure 12(a): License Key

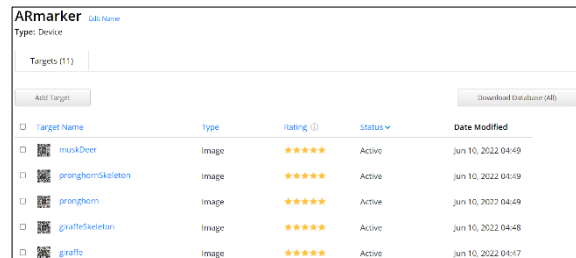


Figure 12(b): ARmarker database

Figure 12(a) and Figure 12(b) show the license key and ARmarker database are created in Vuforia Developer Portal. For star rating, although low-rated image targets such as 1 or 2 stars can usually be detected and tracked, the image targets with 4 or 5 stars are the best results. Therefore, 11 image targets created in the ARmarker database are all 5 stars for the users to detect easily as shown in Figure 12(b).

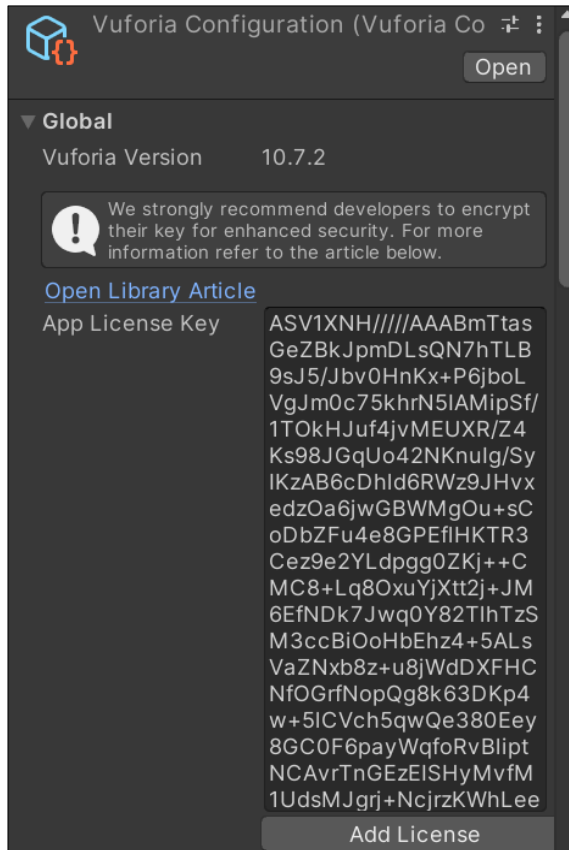


Figure 13(a): License Key (Unity)

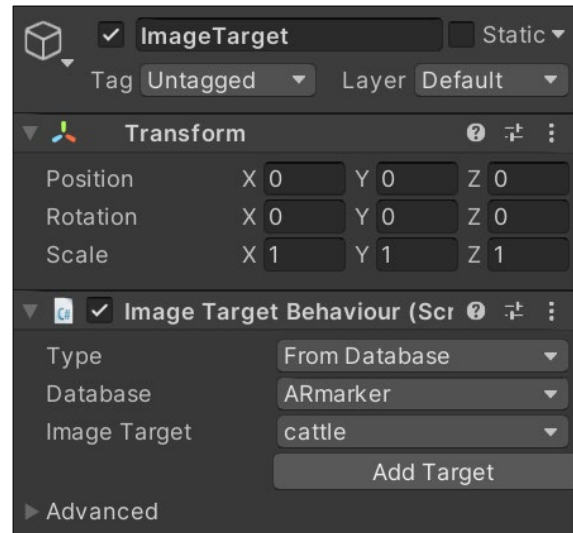


Figure 13(b): ARmarker database (Unity)

The license key created in Vuforia Developer Portal is then copied into the App License Key field of the Vuforia Configuration file in Unity, while the ARmarker database is imported into Unity. Therefore, the developer can use the image targets created in the ARmarker database to implement the AR session. The license key and ARmarker database used in Unity are shown in Figures 13(a) and 13(b).

3.5 Testing Stage

Testing is the final stage in the MMCD approach and a critical component of the development process. This stage involves functional testing by developers to determine if the application will work, and user acceptance testing by the target audience to determine if the target audience will accept it. To ensure that the software is bug-free, developers will conduct functional testing when the complete application has been fully developed. While 30 target users, students in Institut Verterinar Malaysia (IVM), are subjected to user acceptance testing. The major goal of user acceptance testing is to determine whether the generated application is effective and acceptable to the target users by gathering and analyzing feedback and comments from the target users and to determine whether the application should be improved further.

4. Results and Discussion

This session discusses application testing such as functional testing and user acceptance testing.

4.1 Application Testing

The testing process begins when the application is fully developed and ready to run to ensure that the application is running properly on the client side. To do this, functional testing and user acceptance testing are carried out. Black-box testing is used by the developer to test the functionality of the entire application, and user acceptance testing is performed by the target users.

4.1.1 Functional Testing

Functional testing is carried out by the developer, who will use black-box testing to identify all possible bugs in the application. Black-box testing is a software testing method that examines the functionality of an application only without looking at its internal structure. As soon as bugs are found during testing, the developer fixes them immediately to ensure that the application is performing well without bugs. The application is not released to the end-user until it is determined that there are bugs-free in the application. Tables 6 to Table 9 show the black box testing and the results for each test case.

Table 6: Black-box testing for the Homepage interface

Test Case	Expected Result	Actual Result
Learning button	The family selection interface is displayed when the button is clicked.	It works well as expected.
Quiz button	The quiz module is displayed when the button is clicked.	It works well as expected.
About This App button	The about interface is displayed when the button is clicked.	It works well as expected.
Exit button	The exit warning interface is displayed when the button is clicked.	It works well as expected.
Yes button	The user can exit the application when the button is clicked.	It works well as expected.
No button	The user will continue to use the application when the button is clicked.	It works well as expected.

Table 7: Black-box testing for the Learning module

Test Case	Expected Result	Actual Result
Back button	The user can return to the previous interface when the button is clicked.	It works well as expected.
Family Selection button	The topic selection interface is displayed when the button is clicked.	It works well as expected.
Topic Selection button	Learning module content is displayed when the button is clicked.	It works well as expected.
Previous and next buttons	The user can go to the previous or next page of the learning module content when the button is clicked.	It works well as expected.
Image button	The user can watch the real ruminant video when the button is clicked.	It works well as expected.
AR button	AR Instruction interface is displayed when the button is clicked.	It works well as expected.
Text button	The user can download all the image targets when the button is clicked.	It works well as expected.
Image Target button	The user can enter the AR session when the button is clicked.	It works well as expected.

Table 8: Black-box testing for the Quiz module

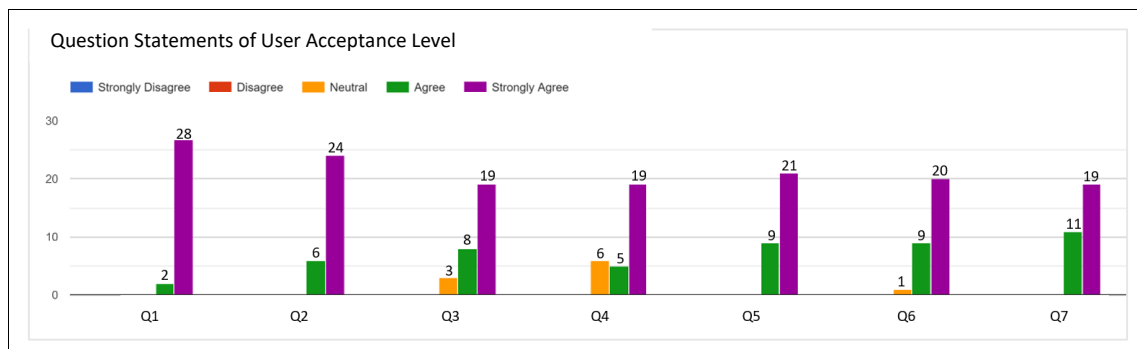
Test Case	Expected Result	Actual Result
Home button	The exit warning interface is displayed when the button is clicked.	It works well as expected.
Yes button	The user can exit the Quiz module and return back to the Homepage interface when the button is clicked.	It works well as expected.
No button	The user will continue to answer the quiz when the button is clicked.	It works well as expected.
Redo button	The user can redo the quiz when the button is clicked.	It works well as expected.
Answer buttons	The next question is displayed when the button is clicked. Correct answers are shown in green with the correct sound, while wrong answers are shown in red with the wrong sound.	It works well as expected.

Table 9: Black-box testing for the About This App module

Test Case	Expected Result	Actual Result
Back button	The user can return to the previous interface when the button is clicked.	It works well as expected.
Acknowledgement button	The acknowledgment interface is displayed when the button is clicked.	It works well as expected.
Tutorial button	The tutorial interface is displayed when the button is clicked.	It works well as expected.

4.1.2 User Acceptance Testing

User acceptance testing is conducted among a random target audience of 30 IVM students. The purpose of user acceptance testing is to get feedback from target users after they use the application in order to find as many bugs or usability issues as possible. Therefore, the developer prepared a set of questionnaires via Google Form and sent them to targeted users via social media such as WhatsApp. After that, the data collected by the questionnaire were statistically analyzed.

**Figure 14: User acceptance level**

Q1: I think the Ruminant Anatomy (AR) application is easy to use.
 Q2: I think Ruminant Anatomy (AR) application has a good navigation structure.
 Q3: I think the interface design of the Ruminant Anatomy (AR) application is appealing.
 Q4: I think the Ruminant Anatomy (AR) application provides enough information.
 Q5: I think the AR session of the Ruminant Anatomy (AR) application is very useful.
 Q6: I think the quiz module of the Ruminant Anatomy (AR) application is very useful.
 Q7: I think Ruminant Anatomy (AR) application is well suited as a learning aid for novice veterinarians.

Figure 14: User acceptance level (cont)

Based on Figure 14, there are seven questions to determine the user acceptance level. All respondents agreed that the Ruminant Anatomy (AR) application is easy to use and has a good navigation structure. Besides, they also agreed that the AR session of the Ruminant Anatomy (AR) application is very useful and Ruminant Anatomy (AR) application is well suited as a learning aid for novice veterinarians. Most of the respondents agreed that the interface design of the Ruminant Anatomy (AR) application is appealing, the Ruminant Anatomy (AR) application provides enough information, and the quiz module of the Ruminant Anatomy (AR) application is very useful, respectively 27 respondents (90.00%), 24 respondents (80.00%), and 29 respondents (96.67%). As a result, most of the respondents provided positive feedback on the user acceptance test, among which 71.43% of the respondents strongly agreed with the seven questions in the user acceptance test.

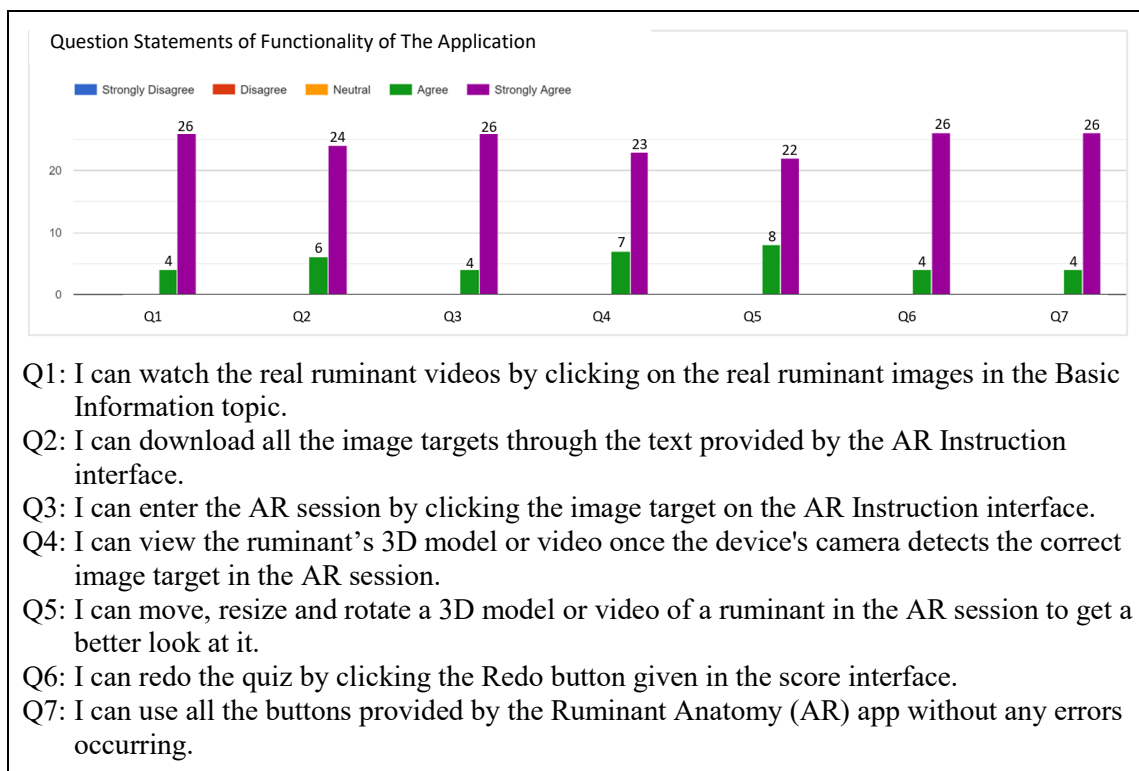


Figure 15: Functionality of the application

Based on Figure 15, there are seven questions to determine the functionality of the application. All respondents agreed on these seven questions, such as they can watch the real ruminant videos by clicking on the real ruminant images in the Basic Information topic. They can download all the image targets through the text provided by the AR Instruction interface, and also can enter the AR session by clicking the image target on the AR Instruction interface. Besides, they can view the ruminant's 3D

model or video once the device's camera detects the correct image target in the AR session, and also can move, resize and rotate a 3D model or video of a ruminant in the AR session to get a better look at it. Next, they can redo the quiz by clicking the Redo button given in the score interface, and also they can use all the buttons provided by the Ruminant Anatomy (AR) application without any errors occurring. As a result, the results showed that Ruminant Anatomy (AR) application is functioning at its optimum stage, with 82.38% of respondents strongly agreeing with the seven questions in the application function test. Overall, Ruminant Anatomy (AR) application can be categorized as successfully meeting the needs of their target users.

5. Conclusion

In conclusion, all the objectives are fulfilled and achieved in this project. The first objective is achieved when all the design phases are completed. The design phase consists of the flowchart, navigational structure, content structure, and button design, which have been discussed in Section 3.3. The second objective is achieved when the developed application can run smoothly on the Android mobile device. The developed application is fully functional with all the modules, which have been discussed thoroughly in Section 3.4. The third objective is achieved when the developed application is functional and the user acceptance test results are satisfactory. The functional testing and user acceptance testing are carried out, and the testing results have been discussed thoroughly in Section 4.

Ruminant Anatomy (AR) application has several advantages, such as the application could be used as a learning aid in veterinary college. This is due to the fact that it is a mobile learning application designed to assist novice veterinarians in quickly and effectively understanding ruminant anatomy. The application provides the basic knowledge of ruminant anatomy and also incorporates a variety of rich multimedia elements to prevent users from becoming bored while learning and to make the process more interesting and enjoyable. Besides, the application replaces traditional gross anatomy by using augmented reality and computer graphics to display 3D models and videos in the AR session. Therefore, the user can use the application to explore and familiarize themselves with 3D models and videos whenever and wherever they choose, without having to go to an anatomy lab. Next, the application includes a quiz module, which provides the user with 20 multiple choice questions in random sequence to assess the user's comprehension and knowledge of the basics of ruminant anatomy.

Apart from its advantages, there are also a few limitations in the Ruminant Anatomy (AR) application. One of the limitations of the application is the learning content provided in the learning module is incomplete due to the lack of online resources. Next, there are only a few 3D models and videos provided in the AR session due to technical problems, where these 3D models and videos did not provide labels. The quiz module also provides a limited number of questions.

Based on the overall results of the application, some recommendations and suggestions are put forward in this section with a view to making improvements in the future. One of the suggestions is to provide more ruminant anatomy information in the learning module, and also provide more labeled 3D models or videos related to ruminant anatomy for users in the AR session. There is also a suggestion to provide users with more questions in the quiz module to enhance their comprehension of ruminant anatomy.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Appendix A

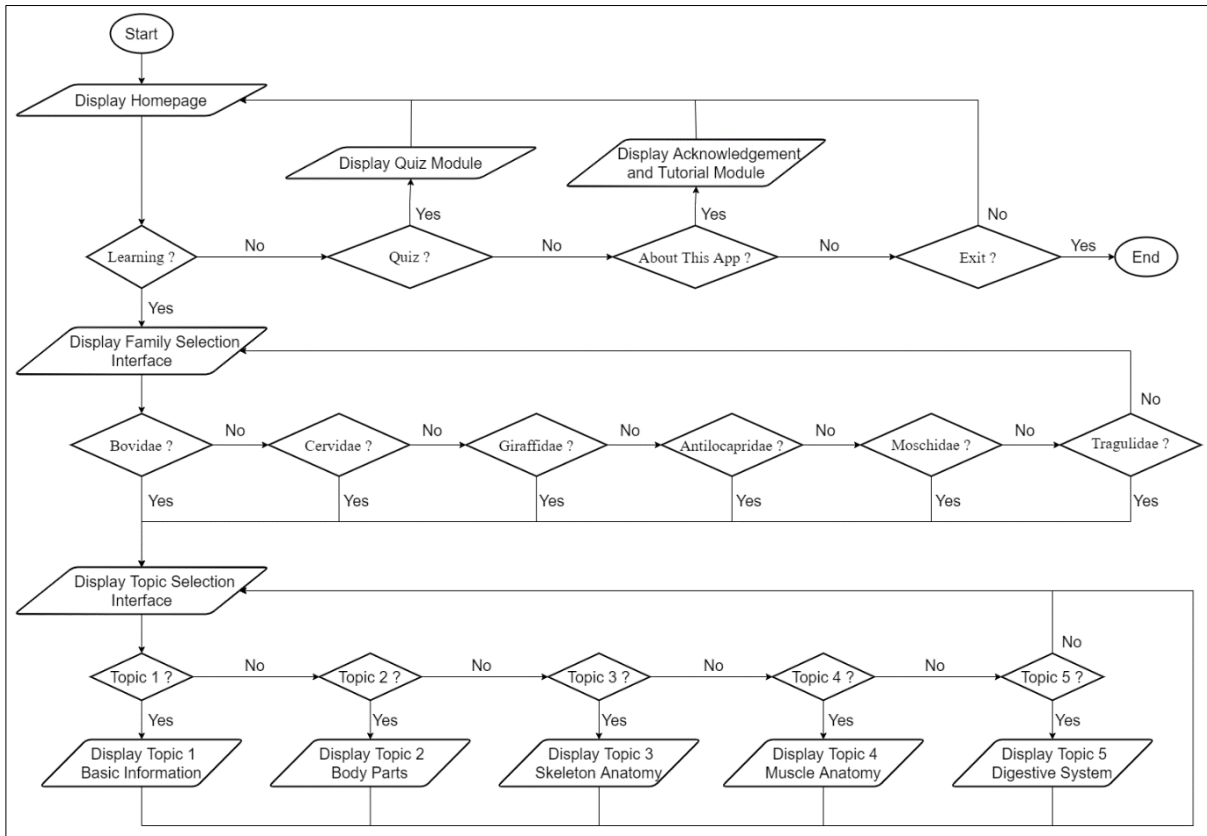


Figure 16: Flowchart

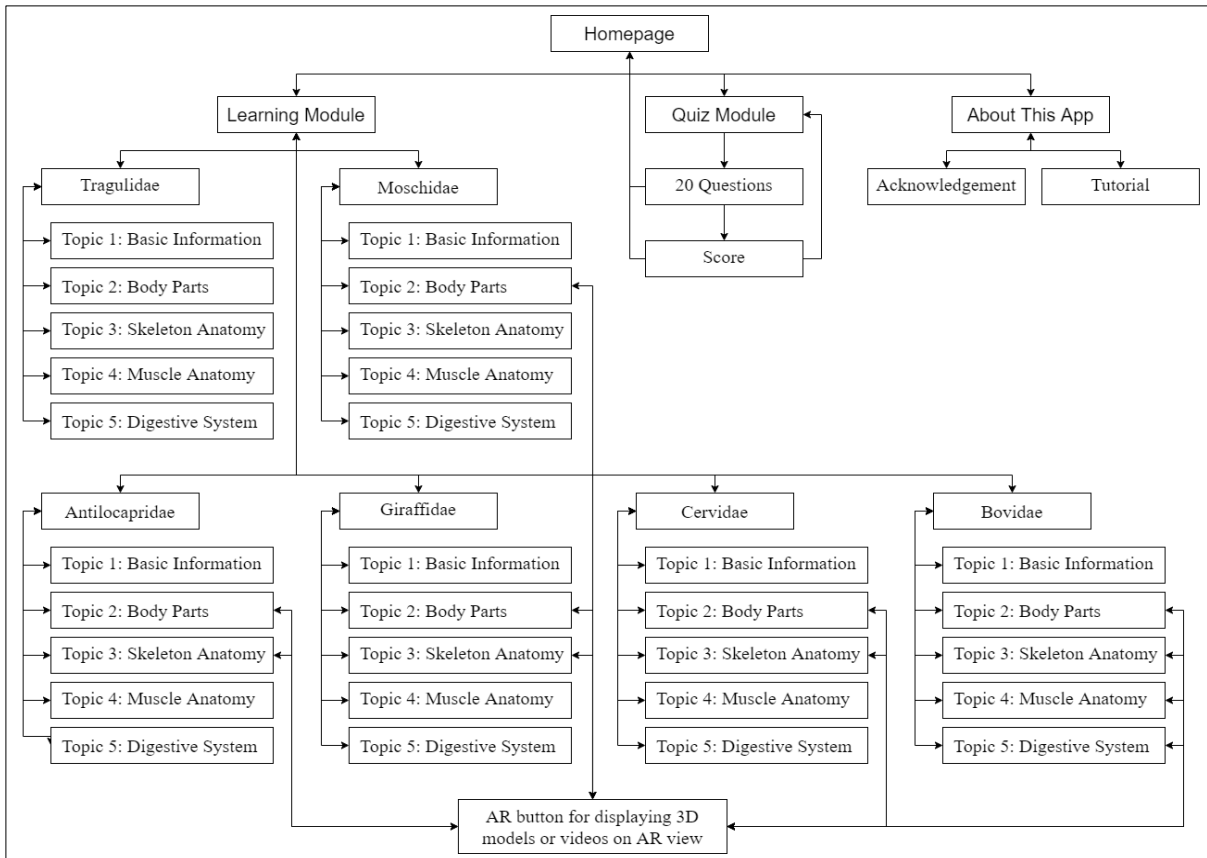


Figure 17: Navigational Structure

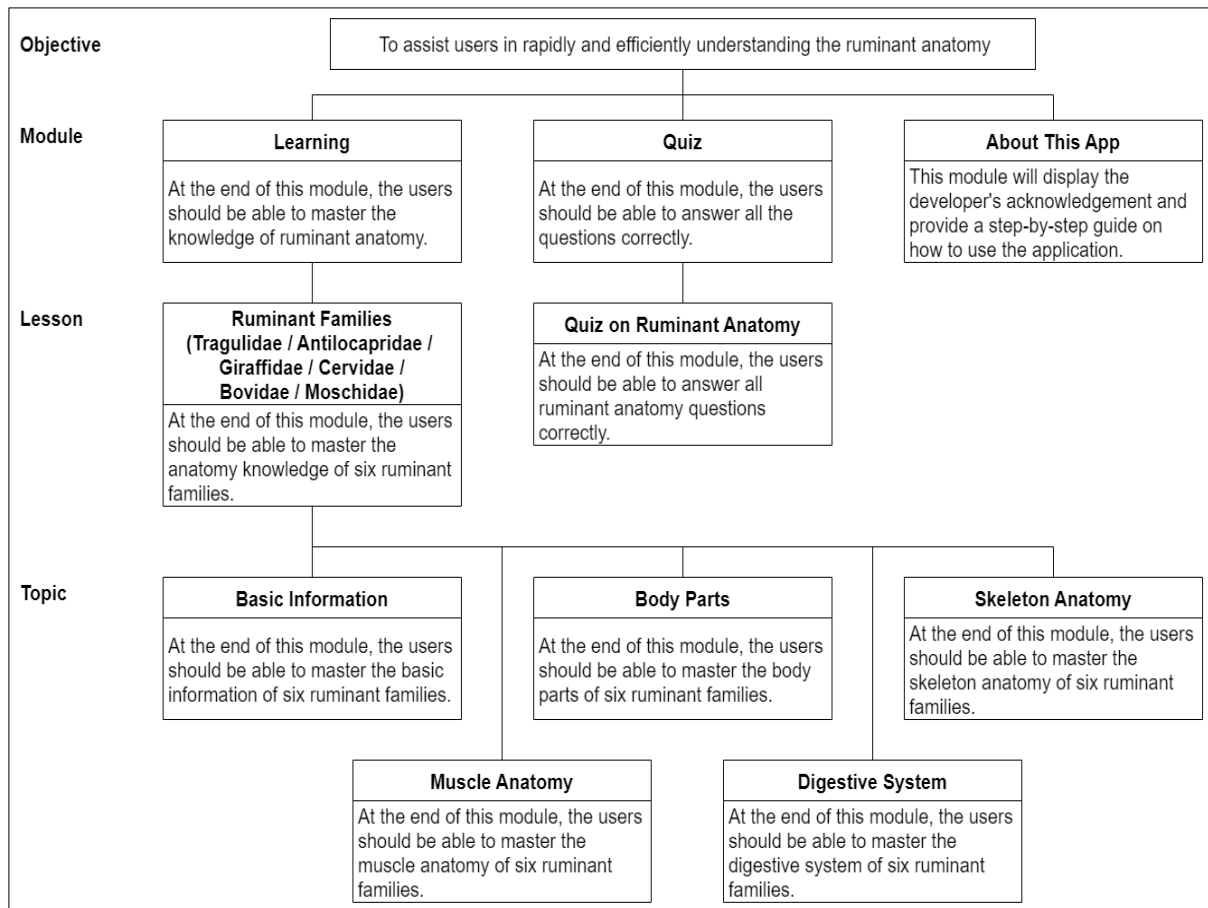


Figure 18: Content Structure

References

[1] “What is Augmented Reality?”, *The Interaction Design Foundation*, 2022. [Online]. Available: <https://www.interaction-design.org/literature/topics/augmented-reality>. [Accessed Oct. 30, 2021].

[2] “Ruhe xuehao jiepouxue [How to learn anatomy well],” Xxmu.edu.cn, 2022. [Online]. Available: <https://www.xxmu.edu.cn/anatomyxxmc/info/1041/1091.htm>. [Accessed Oct. 30, 2021].

[3] M. Song, M. F. Dong, and L. Q. Chai, “Renti zuzhixue yu jiepouxue kecheng jianshe yu gaige [Construction and reform of human histology and anatomy curriculum],” *Anhui nongye kexue*, 44(15), 281-282, 2016.

[4] M. G. Stover, R. R. Watson and R. J. Collier, “Pre- and Probiotic Supplementation in Ruminant Livestock Production”, *Probiotics, Prebiotics, and Synbiotics*, pp. 25-36, 2016. Available: 10.1016/b978-0-12-802189-7.00002-2 [Accessed Oct. 30, 2021].

[5] G. Zvejnieks, “Marker-based vs markerless augmented reality: pros & cons | Overly app”, *Overly*, 2019. [Online]. Available: <https://overlyapp.com/blog/marker-based-vs-markerless-augmented-reality-pros-cons-examples/>. [Accessed Oct. 30, 2021].

[6] (2020). *Animal anatomy physiology (Version 1.1)* [Mobile app]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=com.dhadbadati.apps.the_holy_spirit

- [7] (2020). Anatomia Veterinária (Version 1.8) [Mobile app]. Retrieved from Google Play Store. <https://play.google.com/store/apps/details?id=skins.br.anatomia>
- [8] (2020). Veterinary Anatomy (Version 1.007) [Mobile app]. Retrieved from Google Play Store. <https://play.google.com/store/apps/details?id=com.bilqinm.anatomi>
- [9] W. S. N. S. Saifudin, S. Salam, and M. H. L. Abdullah, "MULTIMEDIA MOBILE CONTENT DEVELOPMENT FRAMEWORK AND METHODOLOGY FOR DEVELOPING M-LEARNING APPLICATIONS", Journal of Technical Education and Training, vol. 4, no. 1, 2012. [Online]. Available:<https://publisher.uthm.edu.my/ojs/index.php/JTET/article/view/481>. [Accessed Oct. 30, 2021].