



TRƯỜNG ĐẠI HỌC FPT

MINISTRY OF EDUCATION AND TRAINING

# FPT UNIVERSITY

## Capstone Project Document

### Bird Training System

GFA23SE03	
<b>Group Members</b>	Nguyễn Thành Trung – SE151134 Phạm Nhật An – SE151210 Trịnh Nam Anh – SE151264 Hoàng Đình Thông – SE151092 Nguyễn Thọ Thái Bảo – SE150653
<b>Supervisor</b>	Mr. Đỗ Tấn Nhàn
<b>Ext Supervisor</b>	
<b>Capstone Project code</b>	FA23SE049

- Ho Chi Minh, December 10 2023 -

## Contents

Acknowledgement .....	5
Definition and Acronyms .....	6
I. Project Introduction .....	7
1. Overview .....	7
1.1 Project Information.....	7
1.2 Project Team .....	7
2. Product Background.....	7
3. Existing Systems .....	7
3.1 Parrot Life:.....	7
3.2 Animal-Clinic:	7
4. Business Opportunity.....	7
5. Software Product Vision .....	8
6. Project Scope & Limitations .....	8
6.1 Major Features.....	8
6.2 Limitations & Exclusions .....	10
II. Project Management Plan .....	11
1. Overview .....	11
1.2 Project Risks .....	13
2. Management Approach .....	14
2.1 Project Process.....	14
2.2 Quality Management .....	15
2.3 Training Plan.....	15
3. Project Deliverables .....	16
4. Responsibility Assignments.....	16
5. Project Communications.....	17
6. Configuration Management.....	17
6.1 Document Management .....	17
6.2 Source Code Management.....	17
6.3 Tools & Infrastructures .....	17
III. Software Requirement Specification .....	18
1. Product Overview .....	18
2. User Requirements .....	19
2.1 Actors .....	19
2.2 Use Cases .....	19
3. Functional Requirements.....	24

3.1 System Functional Overview.....	24
3.2 Bird Training Online Course .....	35
3.3 Workshop.....	38
3.4 Bird Training Academy .....	41
4. Non-Functional Requirements.....	71
4.1 External Interfaces .....	71
4.2 Quality Attributes.....	71
5. Requirement Appendix .....	72
5.1 Business Rules .....	72
5.2 Application Messages List .....	73
IV. Software Design Description .....	80
1. System Design .....	80
1.1 System Architecture.....	80
1.2 Package Diagram.....	81
2. Database Design.....	88
2.1 Table Descriptions.....	88
2.2 Attribute Data Dictionary.....	89
3. Detailed Design .....	103
3.1 Authentication/Authorization.....	103
3.2 Training Course Management Feature .....	107
3.3 Training Management Feature .....	111
3.4 Customer Feature In Training Course .....	116
3.5 Send Consulting Ticket Feature.....	119
3.6 Confirm Consulting Ticket Feature.....	121
3.7 Submit Consulting Evidence Feature .....	125
3.8 Payment Feature.....	127
3.9 Enroll Online Course Feature .....	129
3.10 Workshop Management Feature.....	136
3.11 Workshop Class Management Feature.....	147
3.12 Online Course Management Feature.....	159
V. Software Testing Documentation.....	170
1. Scope of Testing .....	170
1.1. Test model.....	170
1.2. Test levels.....	170
1.3. In scope .....	171
1.4 Out of Scope.....	173

2. Test Strategy .....	173
2.1 Testing Types.....	173
2.2 Test Levels.....	173
2.3 Supporting Tools .....	173
3. Test Plan.....	173
3.1 Human Resources .....	173
3.2 Test Environment.....	174
3.3 Test Milestones .....	174
4. Test Cases.....	176
<b>VI. Release Package &amp; User Guides.....</b>	<b>177</b>
1. Deliverable Package .....	177
2. Installation Guides .....	177
2.1 System Requirements .....	177
2.2 Setup Files .....	179
2.3 Installation Instruction .....	179
3. User Manual.....	181
3.1 Overview .....	181
3.2 <Guest> Register and login for authentication and authorization .....	187
3.3 <Customer> Customer Service.....	188
3.4 <Trainer> Trainer timetable.....	210
3.5 <Manager><Staff> Service Management .....	212
3.6 <Administrator> Administration Management .....	232

## Acknowledgement

Firstly, we want express our profound gratitude to our esteemed supervisor, Mr. Đỗ Tấn Nhàn, whose guidance has been instrumental in shaping the success of this capstone project.

Our special thanks extend to the revered lecturers of FPT University, including Mr. Nguyễn, Mrs. Thanh Vân, Mr. Minh Sang, and Mr. Khánh Phương, for generously sharing their knowledge and offering invaluable insights.

Again, special appreciation goes to Mr. Nhàn for his dedicated mentorship, particularly in navigating the intricacies of technology and business.

This project owes much of its refinement to the constructive feedback and guidance received from our respected teachers. We acknowledge that mistakes are an inherent part of the learning process, and we sincerely appreciate your understanding and forgiveness.

To our cherished family and friends, your unwavering support has been our pillar of strength. We are truly grateful for the encouragement and patience you have extended throughout this journey. Thank you for being the driving force behind our success.

Thank you for all your support. Regards!

## Definition and Acronyms

Acronym	Definition
AWS	Amazon Web Services
BA	Business Analysis
BR	Business Rule
ERD	Entity Relationship Diagram
GUI	Graphical User Interface
PM	Project Manager
SDD	Software Design Description
SPMP	Software Project Management Plan
SRS	Software Requirement Specification
UAT	User Acceptance Test
UC	Use Case
API	Application Program Interface

# I. Project Introduction

## 1. Overview

### 1.1 Project Information

- Project name: Bird Training System
- Project code: FA23SE049
- Group code: GFA23SE03
- Software type: API, Web Application, Mobile Application

### 1.2 Project Team

Full Name	Role	Email	Mobile
Đỗ Tấn Nhàn	Lecturer	nhandt35@fe.edu.vn	0903056041
Nguyễn Thành Trung	Leader	trungntse151134@fpt.edu.vn	0908941761
Phạm Nhật An	Member	anpnse151210@fpt.edu.vn	
Trịnh Nam Anh	Member	anhtnse151264@fpt.edu.vn	
Nguyễn Thọ Thái Bảo	Member	baonttse150653@fpt.edu.vn	0909290138
Hoàng Đình Thông	Member	thonghdse151092@fpt.edu.vn	

## 2. Product Background

Nowadays, people are embracing the joy and wonder of owning birds as pets. Birds are intelligent, social creatures that bring beauty and charm to our lives. However, properly caring for and training birds can be a complex and challenging endeavour. Users need to know detailed information about bird training center and training services so that they can be consulted and choose the right training course for their favourite bird.

## 3. Existing Systems

### 3.1 Parrot Life:

- Parrot Life primarily provides in-home behaviour consultations to companion parrot owners. The business took off quickly, and now offers online video consultations worldwide, and runs regular courses, seminars and workshops.

- Link to the system: [Parrot Life Behaviour and Training | Parrot Help](#)

### 3.2 Animal-Clinic:

- Animal Clinic is a service-website which allows user to book the appointment for their birds. Therefore, Bird Training Center System uses the same booking as Animal Clinic

- Link to Animal Clinic: <https://www.animal-clinic.co.uk/online-booking/#vs-xc-appointment-type>

## 4. Business Opportunity

Bird Training Center System is a comprehensive online platform designed for one Center in an area to facilitate the training and education of bird and bird owners. The system enables users to purchase various courses, including in-person workshops, online courses, consultations, and specialized training programs. Users have the opportunity to send their birds to trainers for personalized training or engage in self-paced online learning. The system also provides a feature that allows users to track the progress of their birds' training.

## 5. Software Product Vision

For bird lovers who want to make a deep understanding with their birds as well as have the best behaviour to them, the Bird Training Center System is an Internet-based and smartphone-enabled application that will provide the online courses to the customers about many bird-living aspects such as habitat, diet, health care, etc to their birds. Also, the Center has the ability to support the customer at any time with Advice Consulting Service which can accept the help ticket from customer and reply or professional support with trainer. The Center hosts the workshop meeting in period for the customers joining and have many activities with their bird. Moreover, the Center has a lot of training courses for customer's birds with various skills and options, one by one training with professional bird trainer.

## 6. Project Scope & Limitations

### 6.1 Major Features

Feature Code	Description
FE-01	[Customer] <u>explore</u> [Online Course] from the website
FE-02	[Customer] <u>view</u> [Online Course] detail
FE-03	[Customer] <u>enroll</u> [Online Course] with online payment
FE-05	The system <u>track the progress</u> of each [Section] and [Lesson] studied by [Customer] in [Online Course]
FE-06	[Customer] receive [Certificate] on completing the [Online Course] - all the [Section] progression is done;
FE-07	[Customer] view and share [Certificate] on internet
FE-08	[Customer] send [Consultant ticket] to center - [Consultant ticket] must be select offline or online appointment; select available [Trainer]; datetime must be within [Slot] provided by the center, specified price must be detailed on center website
FE-09	[Customer] view workshops on center website - specified price must be detailed for each [Workshop]
FE-10	[Customer] view workshop classes in workshop.
FE-11	[Customer] register to workshop class with online payment.
FE-12	[Customer] explore [Training Course] - on mobile app - specified price must be detailed for each [Training Course]
FE-13	[Customer] register [Bird] to [Training Course] - on mobile app - stricken with [Bird] specialty
FE-14	[Customer] view [Training Course Detail] - of the in-training [Bird] - to check progression
FE-15	[Customer] register joining [Workshop] - then pay for the charge amount of the [Workshop] to the center

FE-16	Customer's [Bird] receive [Bird Certificate] - by completing the [Training Course]
FE-17	[Staff] view [Consultant ticket] from [Customer]
FE-18	[Staff] choose free [Trainer] to assign to [Consultant Ticket] if [Consultant Ticket] is not assigned
FE-19	[Staff] manage [Workshop Class] in workshop.
FE-20	[Staff] assign [Trainer] to workshop class slot.
FE-21	[Staff] check attendance of customers in workshop class.
FE-22	[Staff] confirm [Bird] registration to [Training Course] - confirming [Customer] has charged the amount of [Training Course]
FE-23	[Staff] receive [Bird] from [Customer] - [Customer] must be at the center to send [Bird]
FE-24	[Staff] assign [Trainer] by [Slot] - to train the [Bird] for each [Skill] in [Training Course] - [Trainer] must have relevant [Skill] to train
FE-25	[Staff] confirm [Bird] completing the [Training Course] - by checking all the progression in each [Training Course Detail]
FE-26	[Staff] notify [Customer] - to receive [Bird] - [Customer] must be at the center to receive [Bird]
FE-27	[Manager] manage [Online Course], [Section] and [Lesson]
FE-28	[Manager] manage [Workshop]
FE-29	[Manager] manage [Training Course].
FE-30	[Trainer] view assigned [Advice Ticket] by [Staff] or [Customer]
FE-31	[Trainer] submit [Evidence] from the advice appointment - after completing the appointment
FE-32	[Trainer] confirm getting paid from [Customer] by given information from [Billing Form]
FE-33	[Trainer] check assigned workshop class
FE-34	[Trainer] check attendance of customers in assigned workshop class.
FE-35	[Trainer] check assigned [Training Course]
FE-36	[Trainer] submit progression to [Training Course Detail] - for each training [Slot]
FE-37	[Administrator] manage accounts.
FE-38	[Administrator] top up customer account.

## 6.2 Limitations & Exclusions

### 6.2.1 Limitations

Limitation Code	Description
LI-01	Online Course Certificate template is unmodified
LI-02	Bird Training Course Certificate template is unmodified
LI-03	Cannot verify email and phone of accounts in the system
LI-04	Does not have real-time pushed notifications
LI-05	Mobile API does not implement authorization, authentication
LI-06	Customer cannot preview advice consultation booking price as staff must contact customer to notify
LI-07	The mobile application is supported on Android OS only
LI-08	Administrator account cannot be added

### 6.2.2 Exclusions

Exclusion Code	Description
EX-01	The system does not support feedback function
EX-02	The system does not support refund function for any purchase
EX-03	The system does not support logging off time for the center
EX-04	The system currently does not support cancelling online course
EX-05	The system currently does not support multi-language (English only)
EX-06	The system does not support any currency other than VND
EX-07	Training course does not have bird certificates for each skill in course
EX-08	Skill in training course cannot be removed after being in operation once
EX-09	The system does not support distance calculating from third party
EX-10	Training request cannot auto cancel when customer send bird late
EX-11	The system does not support adding new membership rank

## II. Project Management Plan

### 1. Overview

#### 1.1 Scope & Estimation

#	WBS Item	Complexity	Est. Effort (man-days)
<b>1</b>	<b>Requirement analysis</b>		<b>20</b>
1.1	Define background and context of product	Complex	6
1.2	Find problems of users	Medium	5
1.3	Existed system analysis	Complex	7
1.4	Business opportunity analysis	Simple	2
<b>2</b>	<b>Technology training</b>		<b>54</b>
2.1	Learn .NET Core 6	Simple	3
2.2	Learn Git tool	Medium	5
2.3	Learn ReactJS library	Medium	7
2.4	Learn React Native framework	Complex	7
2.5	Learn AutoMapper .NET library	Simple	3
2.6	Learn JWT Authentication concepts and implementation	Medium	5
2.7	Learn basic query with Odata	Simple	3
2.8	Learn integrating Firebase (Storage)	Complex	7
2.9	Learn integrating Online Payment (Stripe)	Complex	7
2.10	Learn CQRS pattern for backend implementation	Complex	7
2.11	Learn CI implementation with Docker	Complex	7
<b>3</b>	<b>Workspace setup</b>		<b>6</b>
3.1	Setup resources storage workspace	Medium	4
3.2	Setup resources and version control	Simple	2
<b>4</b>	<b>Initialize database</b>		<b>8</b>
4.1	Design database	Complex	7
<b>5</b>	<b>Implementation</b>		<b>352</b>
5.1	Authorization/Authentication features	Complex	7
5.2	Management features for workshop	Complex	25

5.3	Management features for advice consultation	Complex	30
5.4	Management features for training course	Complex	40
5.5	Management features for user accounts	Complex	20
5.6	Trainer features in timetable	Complex	20
5.7	Online payment integrating with Stripe API	Complex	20
5.8	Firebase storage integrating in backend service	Medium	10
5.9	Email sender implementation in backend service	Medium	10
6.0	Dashboard implementation	Complex	20
6.1	Management features for online course	Medium	20
6.2	Customer features for workshop	Complex	20
6.3	Customer features for advice consultation	Complex	20
6.4	Customer features for training course	Complex	40
6.5	Customer features for online course	Complex	20
6.6	Membership application in the system	Complex	10
6.7	Certificate generator in the system	Complex	10
6.8	Customer features for profile management	Medium	10
<b>7</b>	<b>Deployment</b>		<b>9</b>
7.1	Deploy backend project	Simple	3
7.2	Deploy frontend project	Simple	3
7.3	Deploy database	Simple	3
<b>8</b>	<b>Testing</b>		<b>4</b>
8.1	Integration Test	Complex	7
8.2	System Test	Complex	7
<b>9</b>	<b>Compose report</b>		<b>56</b>
9.1	Compose project introduction	Simple	3
9.2	Compose project management plan	Complex	7
9.3	Compose Software Requirement Specification (SRS)	Complex	12
9.4	Compose System Design Description (SDD)	Complex	12
9.5	Compose System Implementation and Test (SIT)	Complex	12
9.6	Compose User Guides report	Medium	5

9.7	Compose Final Project Report	Medium	5
10	<b>Closing</b>		<b>12</b>
10.1	Maintenance	Medium	5
10.2	Thesis defense slides preparation	Complex	7

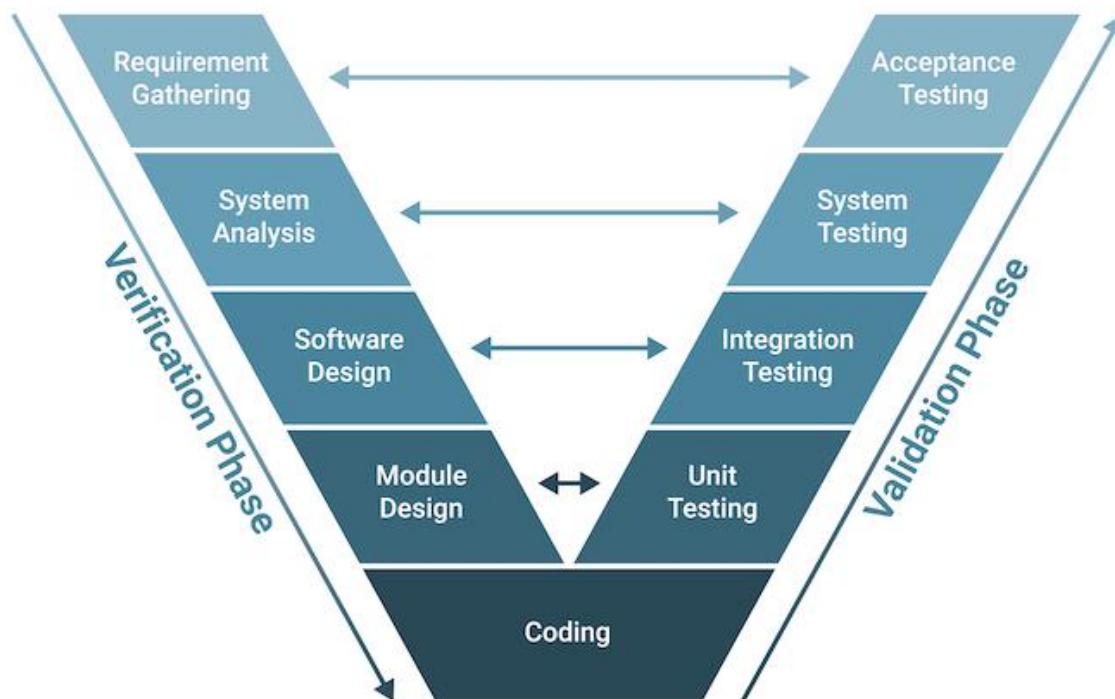
*Total Estimated Effort (man-days)*      **520**

## 1.2 Project Risks

#	Risk Description	Impact	Possibility	Response Plans
1	Lack of human resource on frontend tasks	Critical	Medium	Assign more members and spend more time doing frontend tasks.
2	Member did not complete the tasks on-time	Critical	Medium	Alert to all members, project leader encourages member and provide solution if needed.
3	Mismatch requirements with the clients	Critical	High	Ask supervisor for help, organize meetings with supervisor to give advice.
4	Member did not log task daily	Critical	High	Alert to all members, project leader encourages member and provide solution if needed.
5	Need training for industry business rules before implementation	Critical	High	All team members will have to work overtime to meet the deadlines. Members should invest time into researching the industry during the initiation phase.
6	A member of the team is forced to leave the team or leave the team by personal reasons	Critical	Low	The remaining team members will have to work overtime to meet the deadlines.

## 2. Management Approach

### 2.1 Project Process



- Requirement Gathering: understand the product requirements as laid out by the instructor and existing systems.
- System Analysis: analyze and interpret the business requirements of the proposed system by studying the user requirements document.
- Software Design: define the functionality of each module, the interface relationships, dependencies, database tables, architecture diagrams, technology detail and more. The integration testing model is developed in this phase.
- Module Design: break down the system into small modules and specifies the detailed design of each module.
- Coding: implement the design,
- Unit Testing: execute unit test plans to identify errors at the code or unit level before publishing to the live server.
- Integration Testing: execute integration test plans defined previous in Software Design phase.
- System Testing: execute system test plans developed during the System Analysis phase.
- Acceptance Testing: test the software product in the user environment to identify compatibility issues.

In the validation phase, when a test fails at a certain point, it should be traced back to the corresponding phase in the verification process for thorough examination and correction.

## 2.2 Quality Management

To ensure our product quality, we apply these rules:

- *Design pattern:*
  - For backend API: apply CQRS design pattern referring to <https://learn.microsoft.com/en-us/azure/architecture/patterns/cqrs>
- *Coding convention:*
  - Class names, variable names, constant names, function names, and file names must be written in English correctly and meaningfully.
    - C# code conventions: <https://github.com/dotnet/runtime/blob/main/docs/coding-guidelines/coding-style.md>
    - JavaScript conventions: [https://www.w3schools.com/js/js\\_conventions.asp](https://www.w3schools.com/js/js_conventions.asp)
    - ReactJS conventions: <https://handsonreact.com/docs/code-organization-conventions>
    - React Native conventions:
      - <https://reactnative.dev/docs/tutorial>
      - <https://www.waldo.com/blog/react-native-project-structure>
  - Class names, function names, variable names, and constant names of each programming language must be written in the right way based on the document (camelCase, snake\_case, PascalCase, etc.).
- *Comment code:* comment out the code instead of delete unused code to avoid instant mislead; delete unused code after testing phase.
- *Source code:*
  - Always create new branch, new pull request when doing new task.
  - Branch's name must follow the formation "BE\_NameOfDeveloper\_NameOfTask\_MinorDetail".
  - Pull request must be reviewed by team leader before merged.
- *System testing*
  - Test the system as a whole to confirm that every feature of the application functions correctly
  - Everything tested must be within software requirements specification (SRS) and after integration testing

## 2.3 Training Plan

Training Area	Participants	When, Duration	Waiver Criteria
.NET 6	TrungNT, AnhTN, BaoNTT	Week 1, 7 days	Mandatory
Git	All members	Week 1, 5 days	Mandatory
ReactJS	All members	Week 2, 7 days	Mandatory
ReactNative	ThongHD	Week 2, 7 days	Mandatory
AutoMapper	TrungNT, AnhTN, BaoNTT	Week 2, 3 days	Mandatory
JWT Authentication	TrungNT	Week 1, 5 days	Mandatory
Odata concepts	All members	Week 1, 3 days	Mandatory

Firebase (Storage)	TrungNT	Week 2, 7 days	Mandatory
Online Payment (Stripe)	ThongHD, TrungNT	Week 5, 7 days	Mandatory
CQRS design pattern	TrungNT, AnhTN, BaoNTT	Week 3, 7 days	Mandatory
Deployments	TrungNT	Week 3, 7 days	Mandatory

### 3. Project Deliverables

#	Deliverable	Due Date	Notes
01	Project Plan Document	20/09/2023	Must define project plan before gathering requirements
02	Use case specification	01/10/2023	Functional requirements and non-functional requirements.
03	Design documents	15/10/2023	Architecture Design, Basic Design, Detail Design and Database Design.
04	API Server	01/11/2023	Code & System test cases
05	Web application	15/11/2023	Code & System test cases
06	Mobile application	20/11/2023	Code & System test cases
07	Test Report	30/11/2023	System testing and acceptance testing.

### 4. Responsibility Assignments

D~Do; R~Review; S~Support; I~Informed; <blank>- Omitted

Responsibility	TrungNTSE151134	AnPNSE151210	ThongHDSE151092	AnhTNSE151264	BaoNTTSE150653
Project Planning & Tracking	D	R	R	R	R
Prepare Project Introduction Document	D	S	R	R	S
Prepare SRS Document	D	I	R	S	S
Prepare SDS Document	D	I	I	R	S
Prepare Test Documentation	D	I	R	S	I
Prepare figma (Web)	I	D	S	R	R
Prepare figma (Mobile)	I	S	D	R	R
Backend project initiating	D	R	R	I	I
Frontend project initiating	I	D	S	I	I
Backend project implementing	D			D	D
Frontend project implementing (Web)	I	D	S	S	S
Frontend project implementing (Mobile)	I		D		

Prepare Software User Guides	S	S	D	S	S
Prepare Final Project Report	D	S	S	S	S

## 5. Project Communications

Communication Item	Who/ Target	Purpose	When, Frequency	Type, Tool, Method(s)
Project instruction	NhanDT35	Gather requirements and reviewed	Weekly	Offline
Team meeting	All members	Working in group	3 per week	Offline
Discussion	All members	Notifying/Discussing	Anytime	Online

## 6. Configuration Management

### 6.1 Document Management

- Mega: having syncing function such as OneDrive but shared among teammates.
- Google Drive: log tasks and relevant information,  
<https://docs.google.com/spreadsheets/d/1IZVUfqQAzwKaAAiQphjXRUvehEehkhB48udJLeo5Vig/edit?pli=1#gid=94136293>

### 6.2 Source Code Management

- GitHub

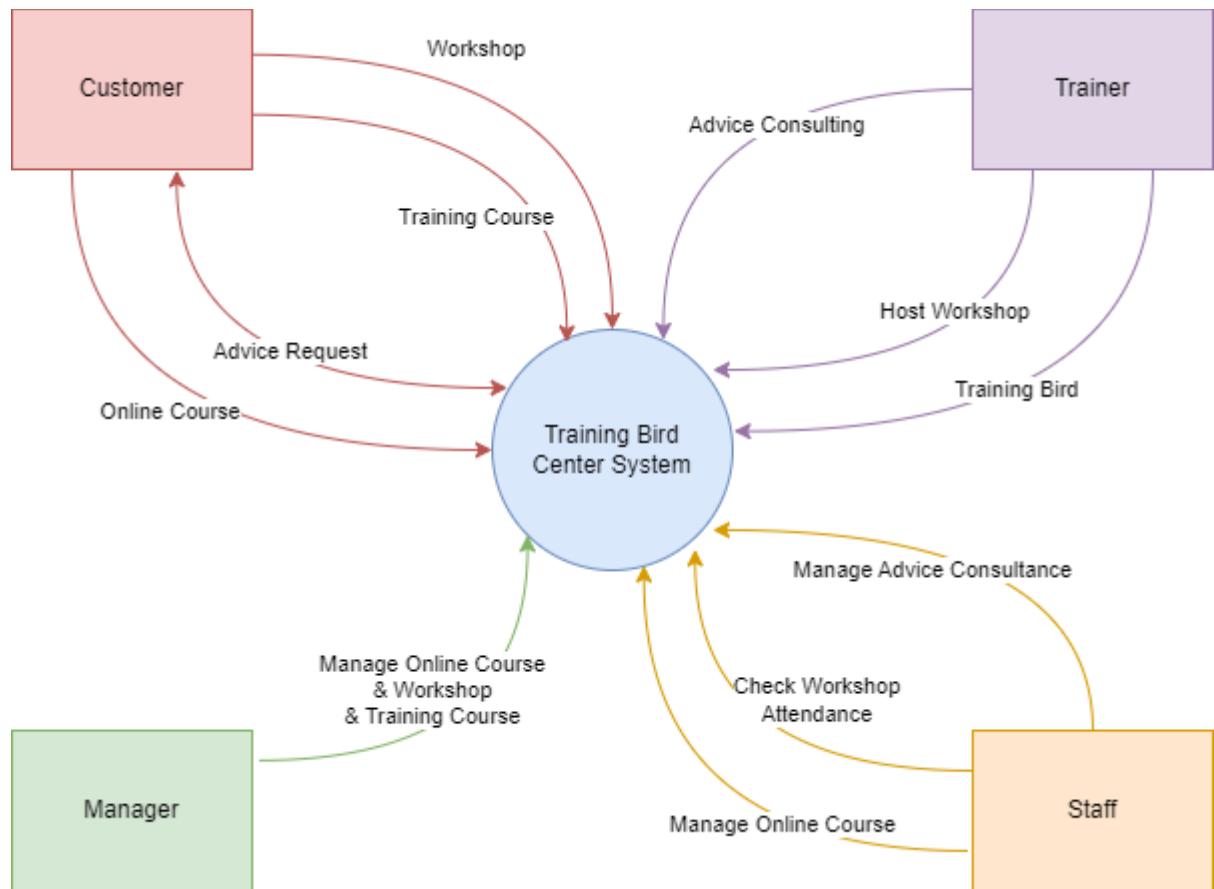
### 6.3 Tools & Infrastructures

Category	Tools / Infrastructure
Technology	ReactJS (Front-end), React Native (App), .NET 6 (Back-end)
Database	PostgreSQL
IDEs/Editors	Visual Studio 2022, Visual Studio Code, Android Studio
Diagramming	Draw.io, Visual Paradigm
Documentation	MS Office, Google Docs/Sheets/Slides
Version Control	GitHub (Source Codes), MEGA (Documents)
Deployment server	AWS LightSail (Amazon)
Project management	Google Sheets (Schedule, Tasks, Defects)

### III. Software Requirement Specification

#### 1. Product Overview

Bird Training Center is a website that provides online learning, consulting, and bird training services. Besides, there is also a user management system, finance and functions on the website



## 2. User Requirements

### 2.1 Actors

#	Actor	Description
1	Staff	A Staff is a person that manages workshop classes, manages customer advice consultation requests
2	Manager	Manager is a role that manages online courses, trainer timetable
3	Trainer	Trainer is a role that do bird training, consulting customers
4	Customer	Customer is a role that study online courses, book training and consulting
5	Administrator	Admin manage user accounts, and manage crud for bird training program
6	Guest	Guest can view courses, workshops and register account

### 2.2 Use Cases

#### Admin:

- Manage User Accounts: Admin creates, modifies, and oversees user accounts.
- Control Access Permissions: Admin sets access rights to ensure system security.
- Monitor User Activities: Admin observes and reviews user interactions for system oversight.
- Manage Slot time: Admin creates, modifies, and oversees slot time.
- Manage Membership Rank: Admin creates, modifies, and oversees Membership Rank.

#### Manager:

- Manage Online Courses: Manager creates, modifies, and oversees online courses.
- Organise Workshop Class: Manager plans and schedules workshops for customer participation.
- Analyse Sales Reports: Manager reviews reports to gauge the success of courses and workshops.
- Manage Training Course: Manager creates, modifies, and oversees Training Courses.
- Manage Trainer Skill: Manager creates, modifies, and oversees Trainer Skills.
- Manage Price Policies of Workshop and Training Course: Manager creates, modifies, and oversees these Price Policies to change the way calculate money.

#### Trainer:

- Conduct Consultations: Trainer provides valuable bird-related advice during customer consultations.
- Lead Training Sessions: Trainer guides and leads effective bird training sessions.
- Document Training Progress: Trainer records and evaluates the progress of each bird's training.

#### Staff:

- Manage Workshop Attendance: Staff records and tracks customer attendance at workshops.

- Support Training Sessions: Staff assists trainers in ensuring the smooth flow of training sessions.

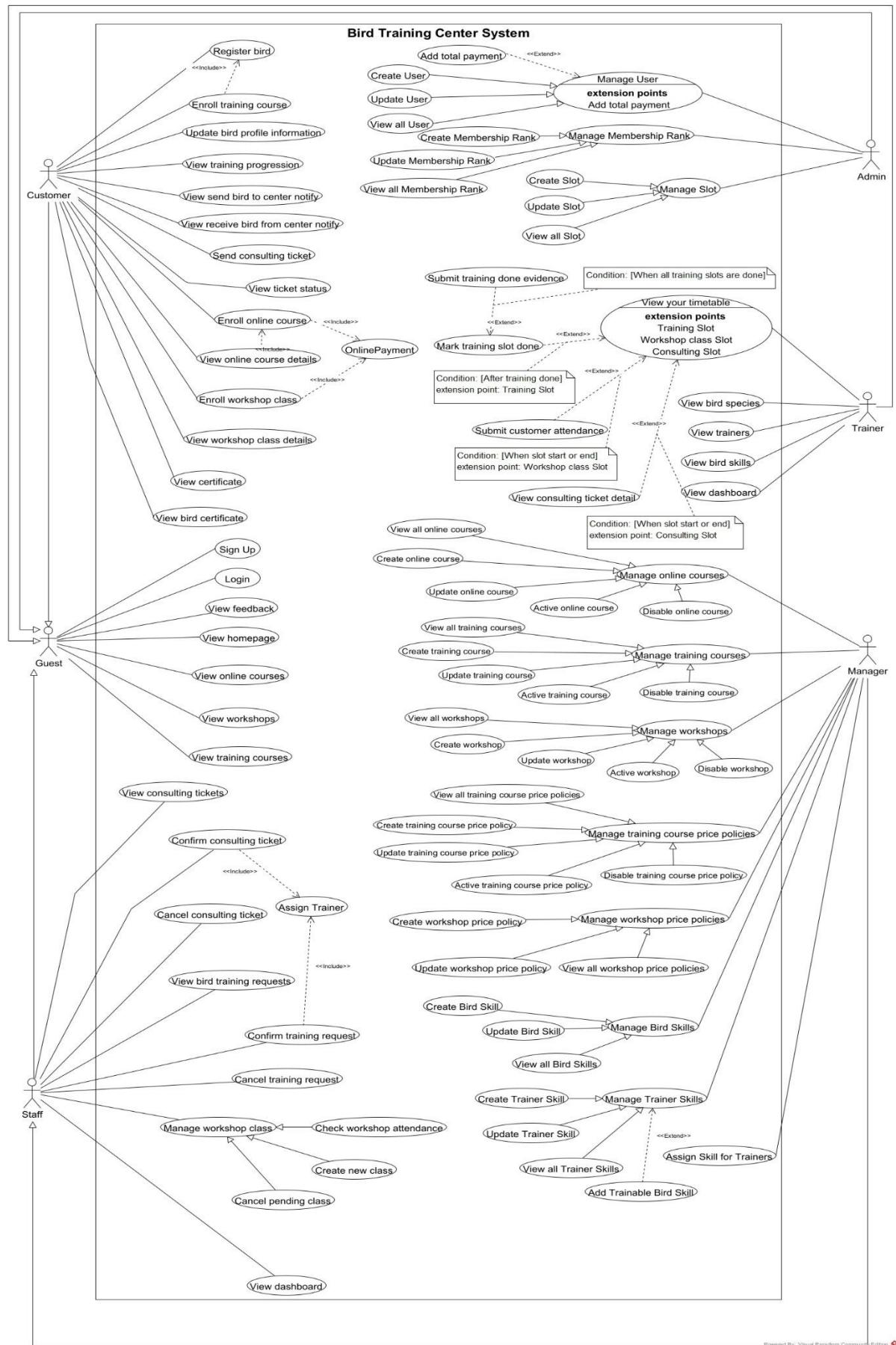
**Customer:**

- Enrol in Online Courses: Customer registers and participates in online bird training courses.
- Participate in Workshops: Customer signs up for and attends informative workshops.
- Schedule Consultations: Customers book appointments for personalised consultations with trainers.
- Track Training Progress: Customer monitors and reviews their bird's training progress through the mobile app.
- Manage Account Information: Users log in and update their personal information.
- Explore Course and Workshop Details: Users view information about available courses and workshops.
- Register and Unregister for Events: Users easily sign up for or withdraw from courses and workshops.
- Stay Informed with Notifications: Users receive timely updates and notifications from the system.

**Guest:**

- Login: Guest use account email and password to access more feature.
- Register: Guest can register new Customer account using email, password and phone number.
- Home page over view: Guest can view some content showing on home page and login if want to use some services.

## 2.2.1 Diagram(s)



### 2.2.2 Descriptions

ID	Use Case	Actors	Use Case Description
1	Sign Up	Guest	Create account Customer to access more features
2	Login	Guest	Use account email and password to gain access to system
3	View homepage	Guest	View home page content
4	View feedback	Guest	View feedback of others
5	View online courses	Guest	View active online courses list
6	View training courses	Guest	View active training courses list
7	View workshops	Guest	View workshops list
8	Register bird	Customer	Register bird information
9	Update bird profile information	Customer	Update bird profile
10	Enrol training course	Customer	Enrol Training Course for Bird
11	View training progression	Customer	View training progress after enrol training course for bird
12	View send bird to center notify	Customer	View notify information about send bird to training
13	View receive bird from center notify	Customer	View notify information about receive bird after training done
14	Send consulting ticket	Customer	Choose when and how to have a consultation
15	View ticket status	Customer	View progression after send consulting ticket
16	OnlinePayment	Customer	Pay online for some services
17	Enrol online course	Customer	Must pay online before enrol online course
18	View online course details	Customer	View sections and lessons details in online course
19	Enrol workshop class	Customer	Must pay online before enrol workshop class
20	View workshop class details	Customer	View details of workshop class before enrol
21	View certificate	Customer	View certificate after finish online course
22	View bird certificate	Customer	View certificate after finish training course for bird
23	View all User	Admin	View all user account in system
24	Create User	Admin	Create account with specific role
25	Update User	Admin	Update role of account or status of account
26	Add total payment	Admin	Add total payment for customer archive membership rank
27	View all Slot	Admin	View all slot time in system
28	Create Slot	Admin	Create new slot time
29	Update Slot	Admin	Update slot start time, end time
30	View all Membership Rank	Admin	View all membership in system
31	Create Membership Rank	Admin	Create new membership

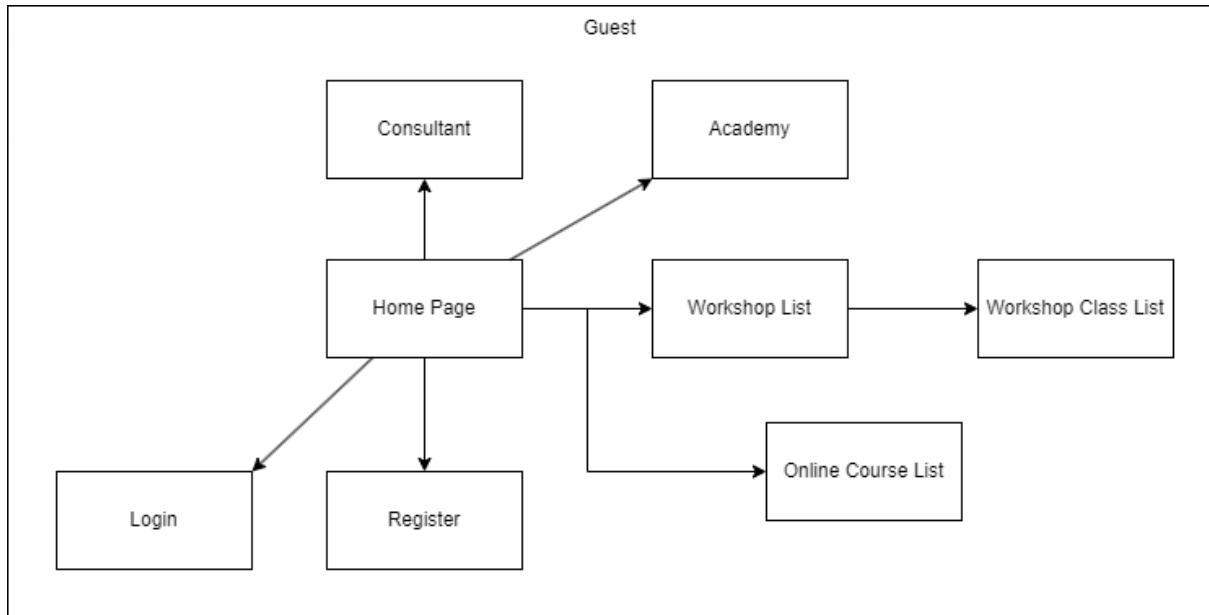
32	Update Membership Rank	Admin	Update membership requirement, discount
33	View all online courses	Manager	View list all online courses
34	Create online course	Manager	Create online course
35	Update online course	Manager	Modify sections and lessons in course
36	Active online course	Manager	Active online course
37	Disable online course	Manager	Disable online course
38	View all training courses	Manager	View list all training courses
39	Create training course	Manager	Create training course
40	Update training course	Manager	Modify skills in course
41	Active training course	Manager	Active training course
42	Disable training course	Manager	Disable training course
43	View all training course price policies	Manager	View list all training course price policies
44	Create training course price policy	Manager	Create training course price policy
45	Update training course price policy	Manager	Modify name and rate in policy
46	Active training course price policy	Manager	Active training course price policy
47	Disable training course price policy	Manager	Disable training course price policy
48	View all workshop price policies	Manager	View list all workshop price policies
49	Create workshop price policy	Manager	Create workshop price policy
50	Update workshop price policy	Manager	Modify name and rate in policy
51	View all Bird Skills	Manager	View list all Bird Skills
52	Create Bird Skill	Manager	Create Bird Skill
53	Update Bird Skill	Manager	Modify name and description
54	View all Trainer Skills	Manager	View list all Trainer Skills
55	Create Trainer Skill	Manager	Create Trainer Skill
56	Update Trainer Skill	Manager	Modify name and description
57	Add Trainable Bird Skill	Manager	Provide information which Trainer Skill can train which Bird Skill
58	Assign Skill for Trainers	Manager	Add trainer skill for trainers
59	Assign Trainer	Staff, Manager	Assign trainer to work
60	View consulting tickets	Staff, Manager	View list all consulting tickets
61	Confirm consulting ticket	Staff, Manager	Confirm consulting ticket after assign trainer
62	Cancel consulting ticket	Staff, Manager	Cancel consulting ticket
63	View bird training requests	Staff, Manager	View list all consulting tickets
64	Confirm training request	Staff, Manager	Confirm training request after assign trainer
65	Cancel training request	Staff, Manager	Cancel training request
66	Check workshop attendance	Staff, Manager	Check customer attendance
67	Create new class	Staff, Manager	Create new workshop class
68	Cancel pending class	Staff, Manager	Cancel when specific problem
69	View dashboard	Staff, Manager	View diagrams of data in system

### 3. Functional Requirements

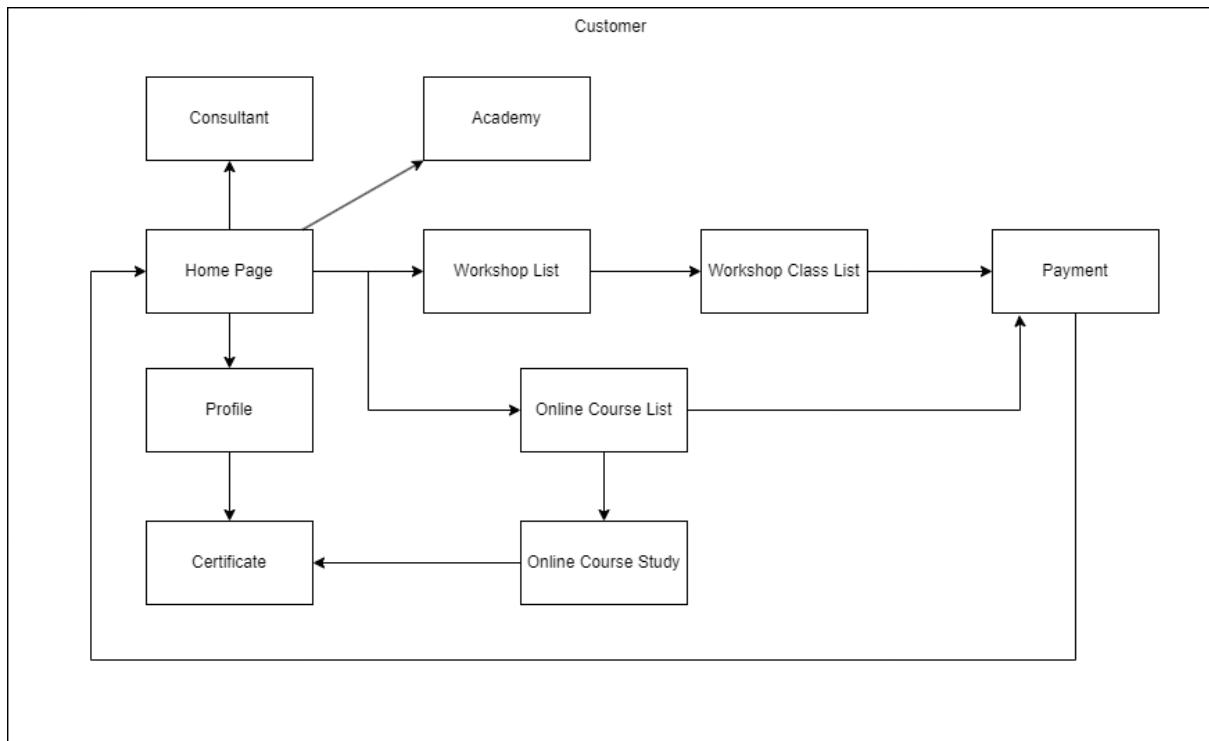
#### 3.1 System Functional Overview

##### 3.1.1 Screens Flow

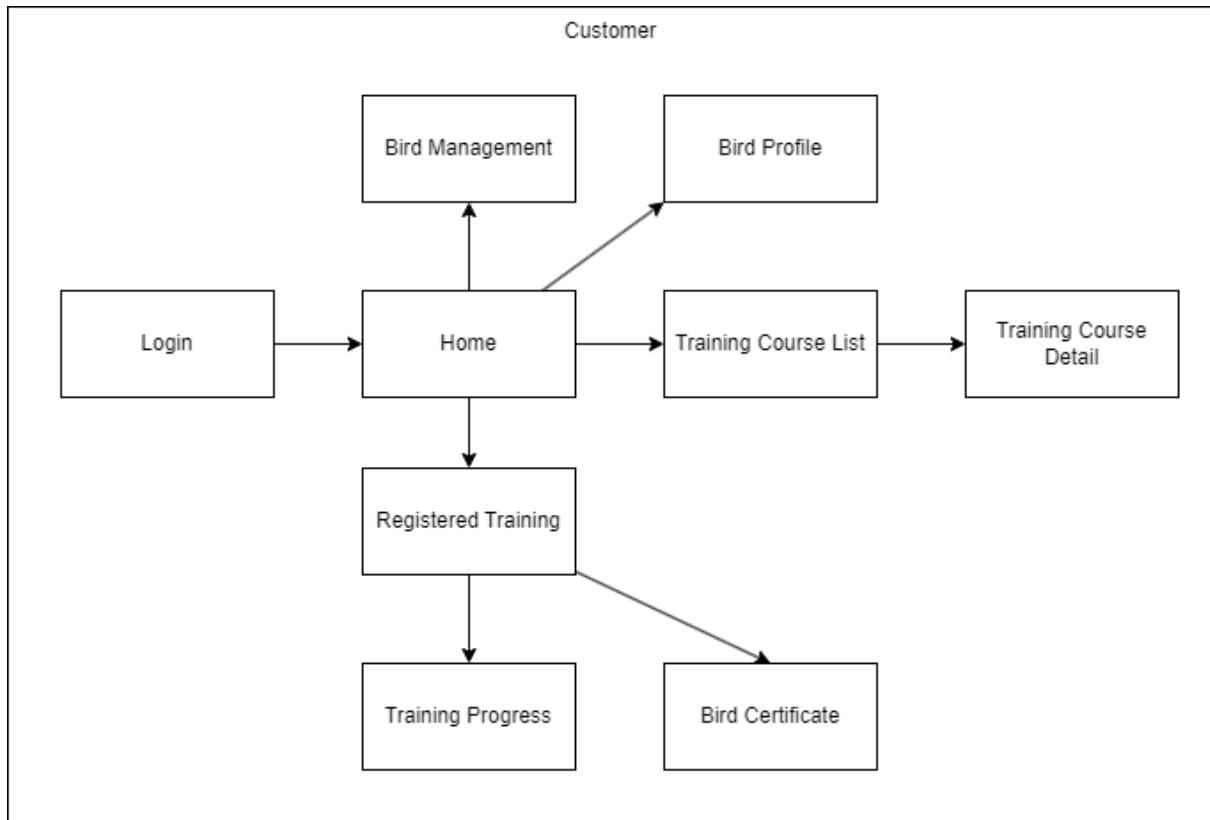
###### 3.1.1.1 Guest Screen Flow



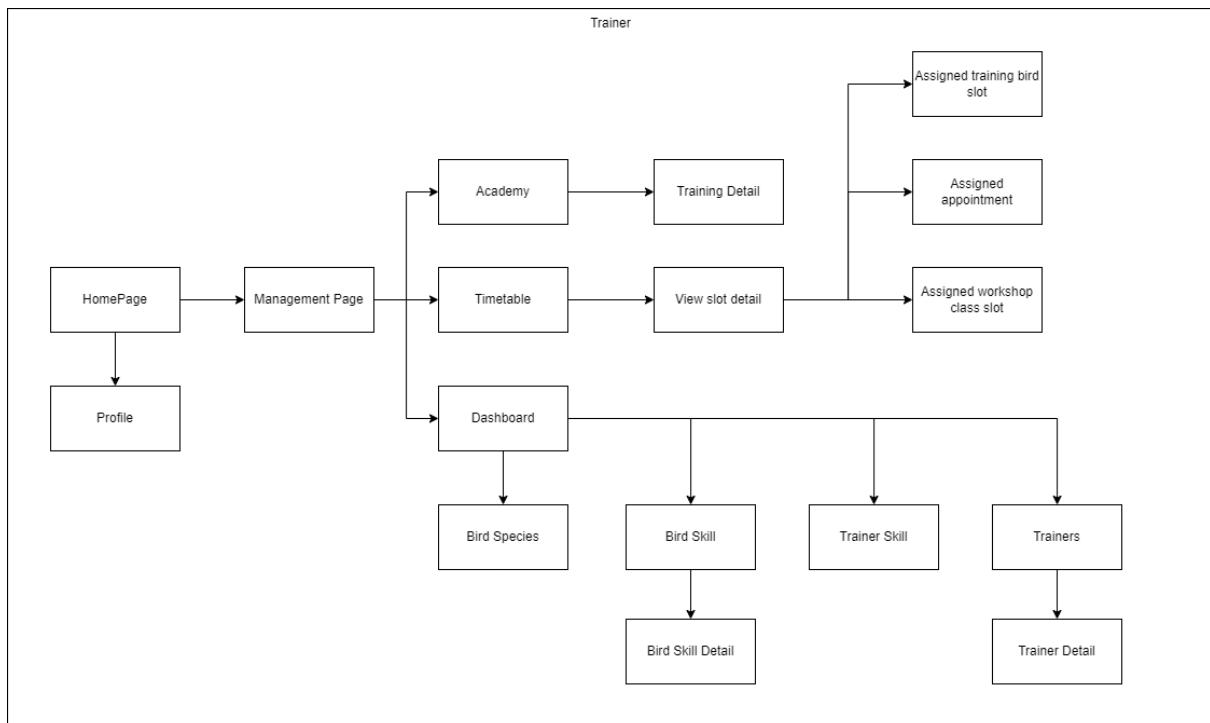
###### 3.1.1.2 Customer Screen Flow



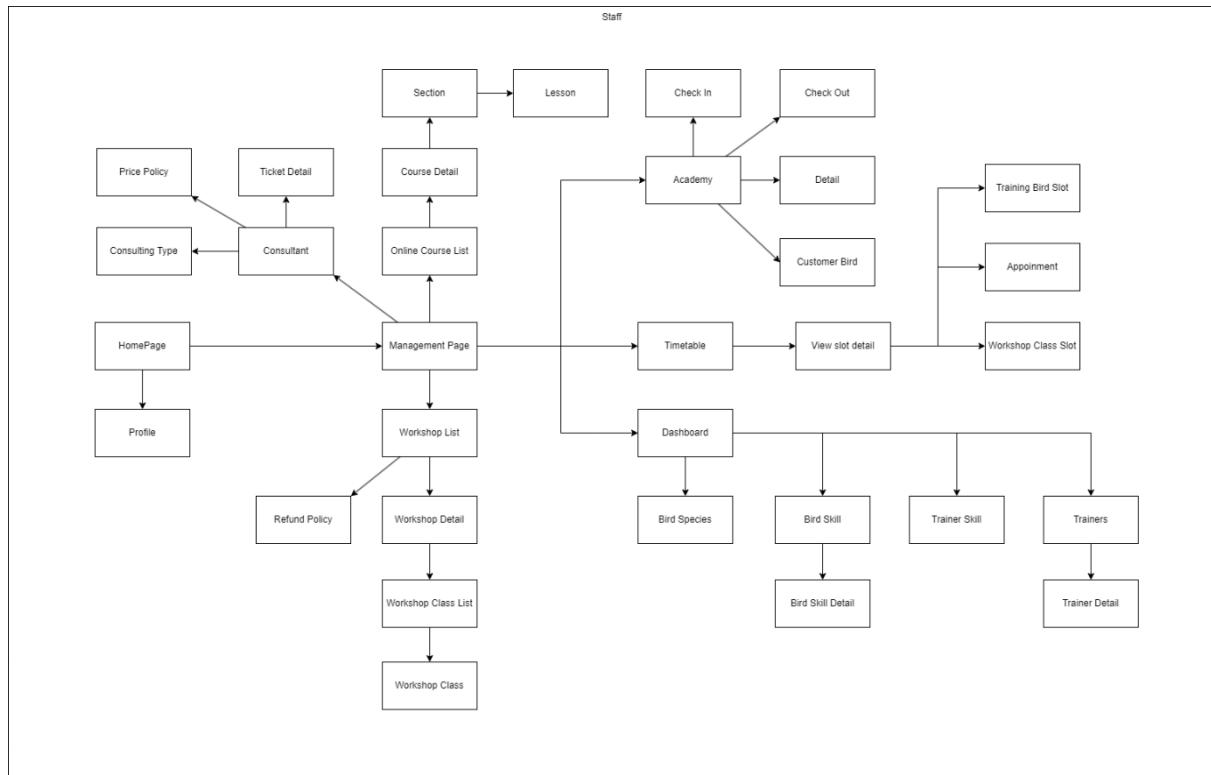
### 3.1.1.3 Customer Mobile Screen Flow



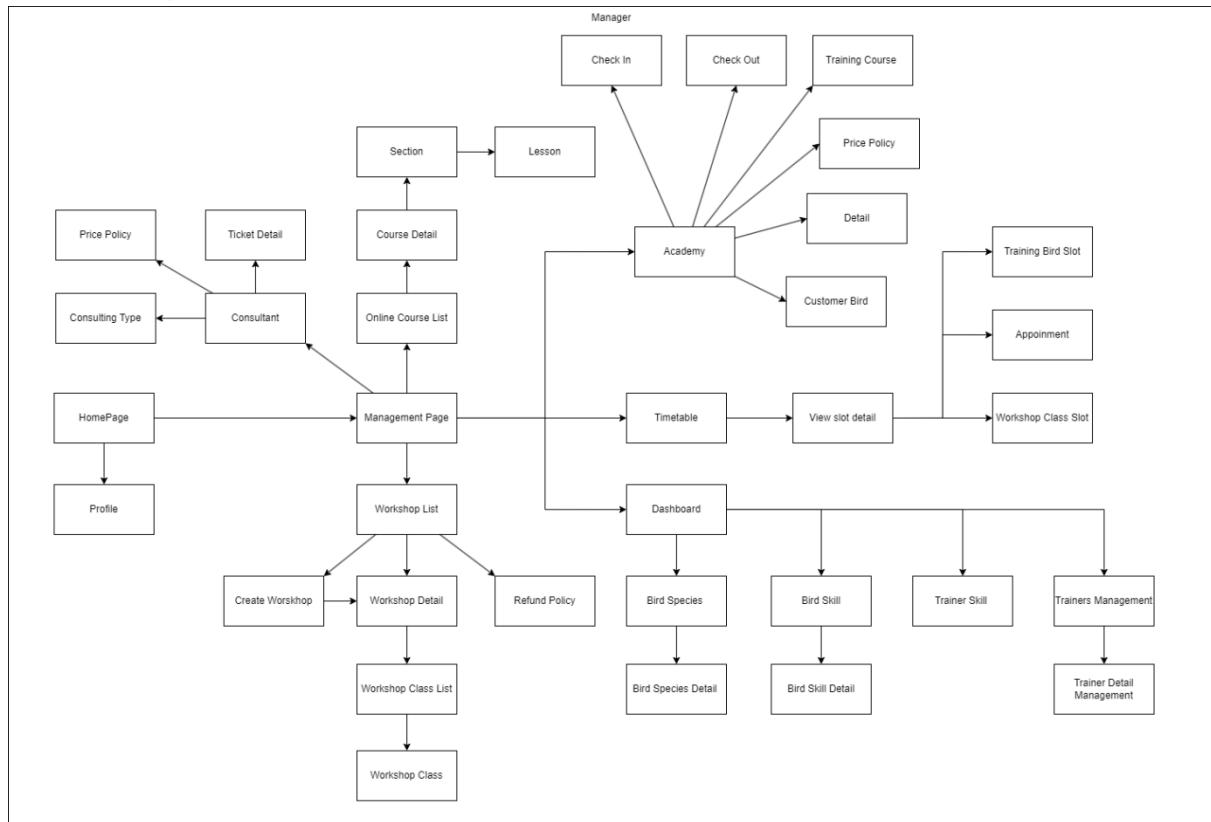
### 3.1.1.4 Trainer Screen Flow



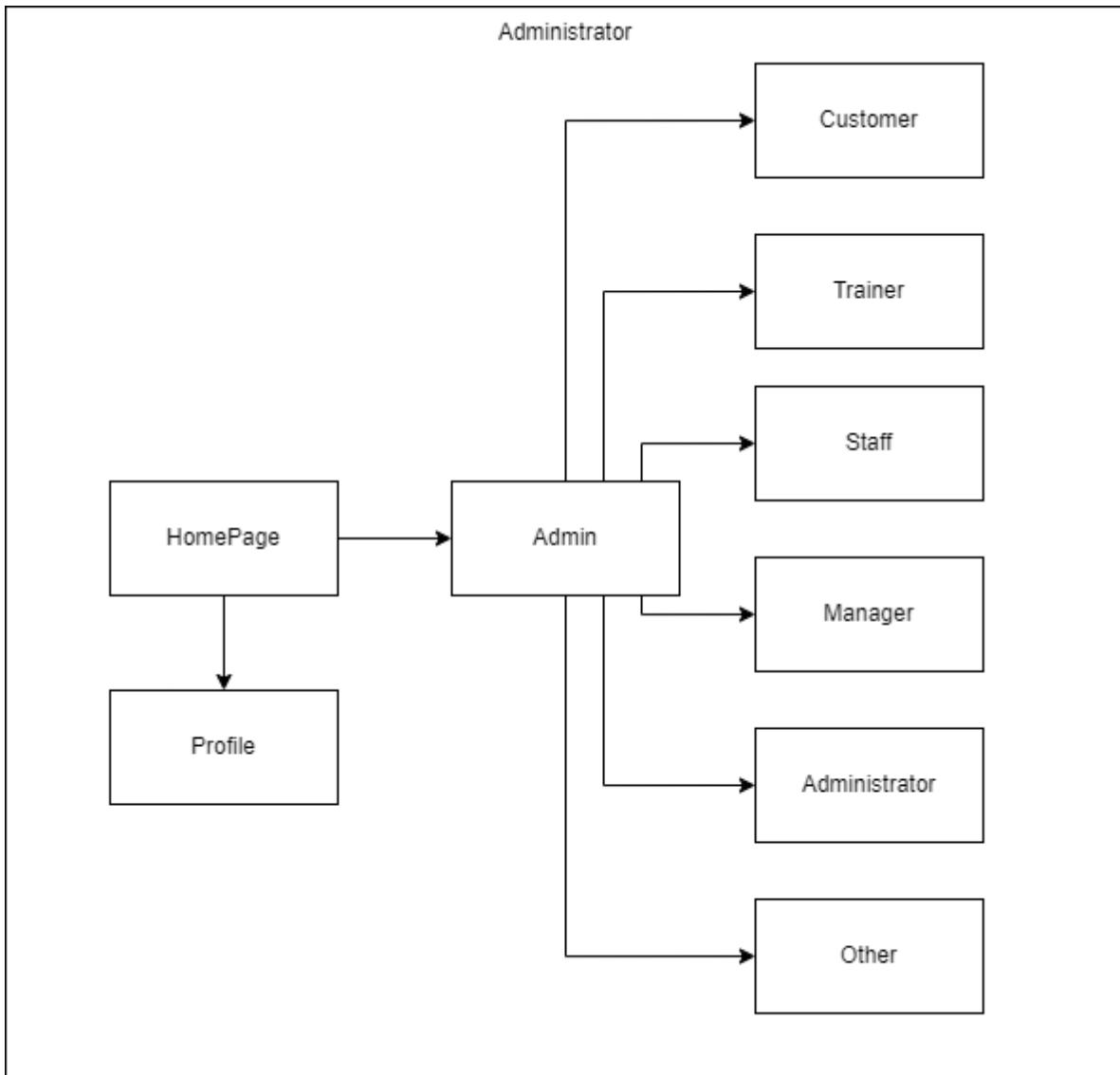
### 3.1.1.5 Staff Screen Flow



### 3.1.1.6 Manager Screen Flow



### 3.1.1.7 Administrator Screen Flow



### 3.1.2 Screen Descriptions

#	Feature	Screen	Description
01	Login	Login	Guest login into the application with an existing account
02	Register	Register	Guest registers new account in the system
03	View consultation detail	Consultant	View consultation detail
04	Send consultation ticket	Consultant	Submit form to send advice consultation ticket
05	Download mobile application	Academy	Embed mobile application APK for downloading
06	View workshops	Workshop List	View workshops those are available
07	View workshop classes	Workshop Class List	View workshop classes within workshop
08	View workshop class	Workshop Class List	Click a class to view popup detail
09	Register workshop class	Workshop Class List	From popup detail, click enrol button
10	View online course list	Online Course List	View online courses those are available
11	View online course detail	Online Course List	Click a course to view detail
12	Enrol online course	Online Course List	From course detail, click enrol button
13	Study online course	Online Course Study	From course detail, click study button. Only enrolled course can show study button Customer study the course through sections and lessons
14	View certificate	Certificate	Customer view completed course's certificate
15	Manage profile	Profile	Authenticated user can manage profile
16	Online payment	Payment	Customer accesses this page via registering workshop class or enrolling online course When payment done, back to home page
17	View timetable	Timetable	View trainer's timetable As staff and manager, can freely choose trainer to view
18	View slot detail	View slot detail	From timetable, click on an occupied slot to view the slot detail Slot detail may be the workshop class slot, consultation appointment, bird training slot
19	Manage bird training slot	Training Bird Slot	Manage the bird training slot
20	View appointment detail	Appointment	View the appointment detail of a consultation ticket
21	Manage workshop class slot	Workshop Class Slot	Manage the workshop class slot Submit attendance form
22	View tickets	Consultant	View tickets from consultation service sent by customers
23	View tickets detail	Ticket Detail	View the detail of the ticket
24	Manage ticket	Ticket Detail	Manage the ticket Trainer can upload evidence for the ticket

			Staff and Manager can assign trainer to ticket or do the late payment for customer
24	View dashboard	Dashboard	View statistics data
25	View bird species	Bird Species	View bird species those are available in the system
26	Manage bird species	Bird Species	Manage bird species (add, update) (only Manager)
27	View bird skill	Bird Skill	View bird skills those are supported in the system
28	Manage bird skill	Bird Skill	Manage bird skill (add, update)
29	Manage bird skill with trainer skill	Bird Skill Detail	Manage trainer skills in selected bird skill (only Manager)
30	View trainer skill	Trainer Skill	View trainer skills those are supported in the system
31	Manage trainer skill	Trainer Skill	Manage trainer skill (add, update) (only Manager)
32	View trainers	Trainers	View trainers' basic information
33	View trainer's trainer skills	Trainer Detail	View trainer's trainer skills of selected trainer
34	Manage trainer skill for trainer	Trainer Detail	Manage trainer skill for trainer (add, update, remove) (only Manager)
35	View price policies of consultation	Price Policy	From Consultant screen, click "Price Policy" tab
36	Manage price policies of consultation	Price Policy	From Consultant screen, click "Price Policy" tab Only Manager can do this feature
37	View consulting types	Consulting Type	View consulting types of consultation service
38	Manage consulting type	Consulting Type	Manage consulting types of consultation service Only Manager can do this feature
39	View workshops	Workshop List	View all workshops in the system
40	Create workshop	Create Workshop	Create new workshop Only Manager can do this feature
41	Create new class	Workshop List	Right click on a workshop then choose "New class"
42	View classes in workshop	Workshop Class List	View workshop classes in workshop
43	Create new class	Workshop Class List	Create new class of current selected workshop
44	View refund policies	Refund Policy	View refund policies of workshop service
45	Manage class	Workshop Class	Manage class information Manage class slot by selecting date, time and assigning trainer
46	View online courses	Online Course List	From Management page, click "Online Course" tab
47	View online course detail	Course Detail	View online course detail, status, sections overview
48	Manage online course	Course Detail	Modify course information, status, delete section Only Manager can do this feature
49	View section detail	Section	View section detail
50	Manage section detail	Section	Modify section detail, delete lesson Only Manager can do this feature
51	Add new lesson	Section	Add new lesson in the section

			Only Manager can do this feature
52	View lesson detail	Lesson	View lesson detail
53	Manage lesson detail	Lesson	Modify lesson detail Only Manager can do this feature
54	View registered training request	Academy	View registered requests from mobile customer
55	Manage training request	Detail	Manage request by assigning trainer
56	Check in customer's bird for training	Check In	Check in customer's bird to the center
57	Check out customer's bird	Check Out	Check out customer's bird in the center
58	Manage training course	Training Course	Manage training course information, skills, slots, etc... Only Manager can do this feature
59	Manage price policies for training course	Price Policy	From Academy page, click "Price Policy" button Only Manager can do this feature
60	View customer's registered birds	Customer Bird	View birds registered by the customer
61	Login	Login (Mobile)	Customer login with existing account
62	Manage birds	Bird Management (Mobile)	Manage birds for registering training course
63	Manage current bird	Bird Profile	Manage bird in current profile
64	View training courses	Training Course List	View training courses available in the system
65	View training course detail	Training Course Detail	View detail of the selected training course
65	View registered training in current profile	Registered Training	View registered training courses of selected bird profile
66	View training progress	Training Progress	View progress of training the bird
67	View bird certificate	Bird Certificate	View bird certificate for completing the course
68	Manage customer account	Customer (Admin)	Manage customer account
69	Manage trainer account	Trainer (Admin)	Manage trainer account
70	Manage staff account	Staff (Admin)	Manage staff account
71	Manage manager account	Manager (Admin)	Manage manager account, switch role between manager, staff and trainer
73	Manage administrator account	Administrator (Admin)	Manage administrator account
74	Manage slot time	Other (Admin)	Manage slot period, number of slot each day is fixed at 8

75	Manage membership rank	Other (Admin)	Modify membership rank supported in the system			
----	------------------------	---------------	--	--	--	--

### 3.1.3 Screen Authorization

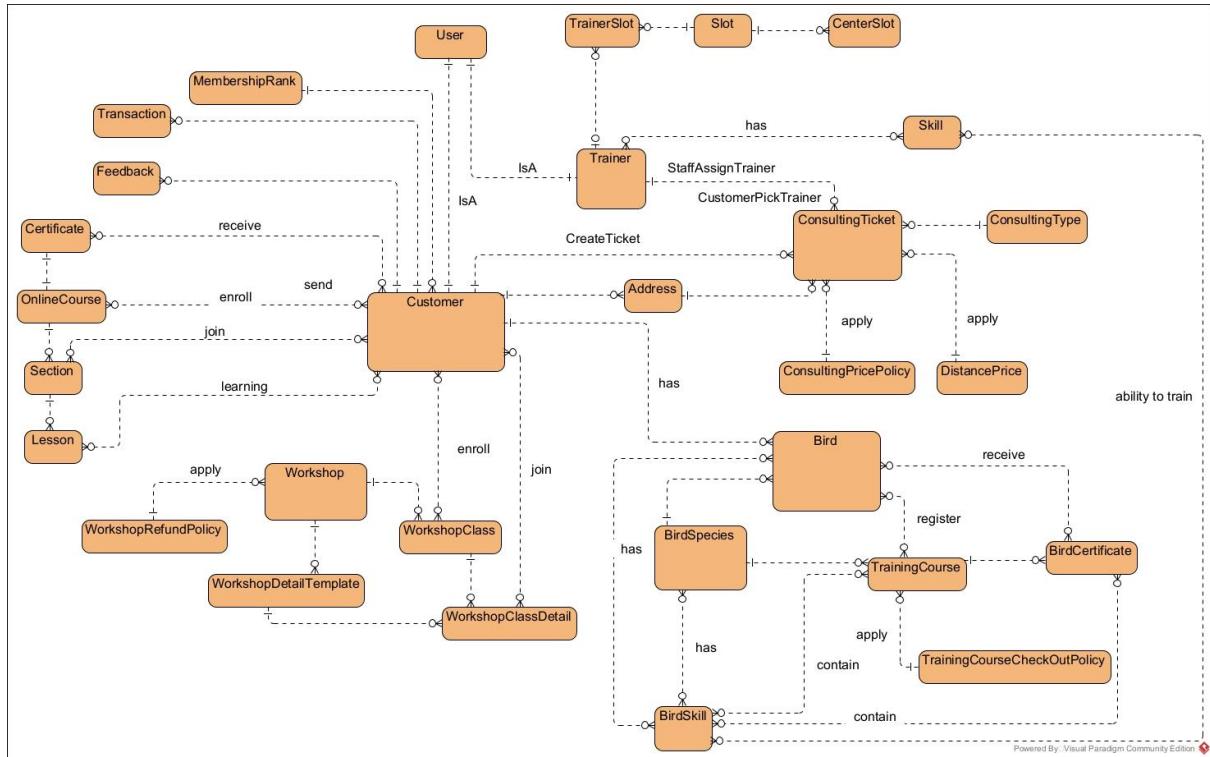
Screen	Customer	Admin	Manager	Staff	Trainer	Guest
Login						X
Register						X
Home Page	X	X	X	X	X	X
Consultant	X					X
Academy	X					X
Workshop List	X					X
Online Course List	X					X
Workshop Class List	X					X
Payment	X					
Online Course Study	X					
Certificate	X					
Profile	X	X	X	X	X	
Management Page		X	X	X	X	
Timetable			X	X	X	
Training Bird Slot			X	X	X	
Appointment			X	X	X	
Workshop Class Slot			X	X	X	
View slot detail			X	X	X	
Academy (Management)			X	X	X	
Training Course			X			
Price Policy (Academy)			X	X		
Check In			X	X		
Check Out			X	X		
Detail			X	X	X	
Customer Bird			X	X		
Training Detail						X
Consultant (Management)			X	X	X	
Ticket Detail			X	X	X	
Price Policy (Consultant)			X	X		
Consulting Type			X	X		
Dashboard			X	X	X	
Bird Species			X	X	X	
Bird Species Detail			X			
Bird Skill			X	X	X	
Bird Skill Detail			X	X	X	
Trainer Skill			X	X	X	
Trainers			X	X	X	
Trainer Detail			X	X	X	

Online Course List (Management)			X	X		
Course Detail (Management)			X	X		
Section			X	X		
Lesson			X	X		
Workshop List (Management)			X	X		
Create Workshop			X			
Workshop Detail			X	X		
Workshop Class List			X	X		
Workshop Class			X	X		
Refund Policy (Workshop)			X	X		
Login (Mobile)						X
Home (Mobile)	X					
Bird Management (Mobile)	X					
Bird Profile (Mobile)	X					
Training Course List (Mobile)	X					
Training Course Detail (Mobile)	X					
Registered Training (Mobile)	X					
Training Progress (Mobile)	X					
Bird Certificate (Mobile)	X					

### 3.1.4 Non-Screen Functions

#	Feature	System Function	Description
1	Send transaction information via email	Use email service to send on each request	Send email after customer's payment
2	Auto-cancel unhandled appointment	Change status of unhandled ticket if overdue	Cancel appointment ticket if its overdue
3	Auto-check completing class	Change status of workshop class if the class is actually completed	Automatically transform status to complete when last slot come by
4	Auto-check attendance on late submitted	Set the attendance status of customer within slot to "Present"	Automatically check attendance after slot
5	Auto-check if trainer miss on bird training slot	Change status if the trainer has not submitted the report when overdue	Automatically check for the report on late submitted

### 3.1.5 Entity Relationship Diagram



### **Entities Description**

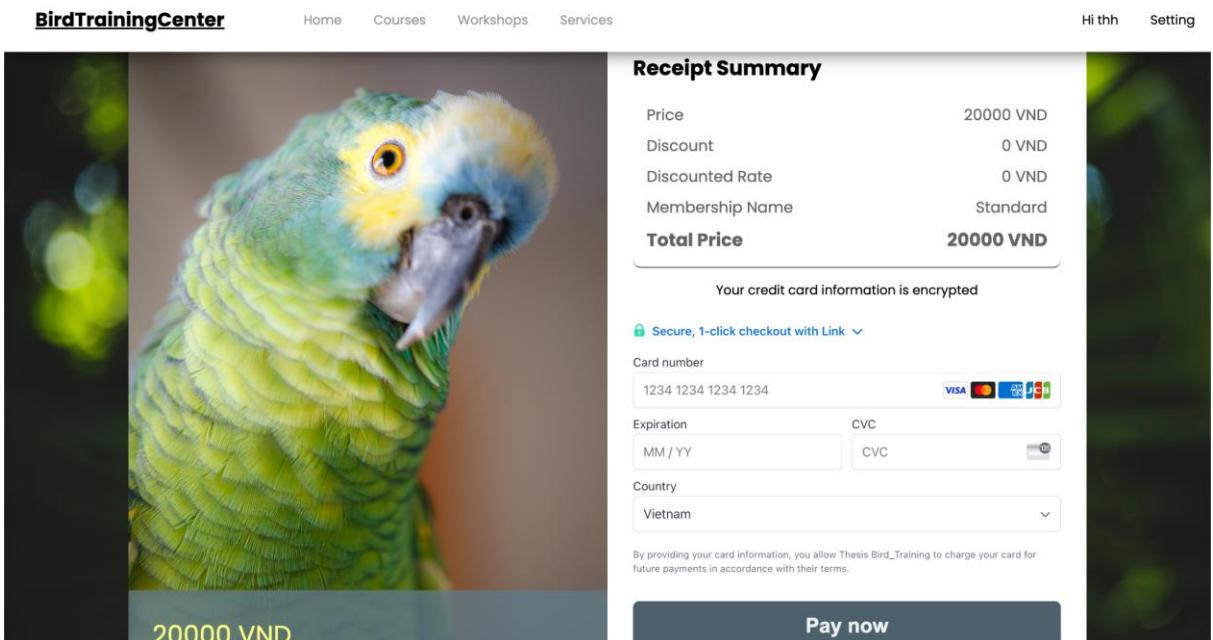
#	Entity	Description
1	User	An entity that represents base information of accounts
2	Customer	An entity that represents extend information of customers
3	Trainer	An entity that represents extend information of trainers
4	TrainerSlot	An entity that represents busy time of trainers
5	Slot	An entity that represents slots timespan
6	CenterSlot	An entity that represents busy time of center
7	MembershipRank	An entity that represents rank of membership to apply discount
8	Transaction	An entity that represents payment information of customer at center
9	Feedback	An entity that represents feedback of customer about center services
10	Certificate	An entity that represents certificate information of online course
11	OnlineCourse	An entity that represents base information about online course
12	Section	An entity that represents information about section in online course
13	Lesson	An entity that represents information about lesson in section
14	Workshop	An entity that represents base information about workshop
15	WorkshopDetailTemplate	An entity that represents information about details in workshop
16	WorkshopRefundPolicy	An entity that represents information about refunding after enrol workshop
17	WorkshopClass	An entity that represents base information about workshop class
18	WorkshopClassDetail	An entity that represents information about details of workshop class
19	Address	An entity that represents information about customer's address
20	ConsultingTicket	An entity that represents information about consulting ticket
21	ConsultingType	An entity that represents information about consulting type
22	DistancePrice	An entity that represents information about distance price apply on ticket
23	ConsultingPricePolicy	An entity that represents information about base price apply on ticket

24	Bird	An entity that represents information about bird
25	BirdSpecies	An entity that represents information about bird species
26	BirdSkill	An entity that represents information about bird skill
27	TrainingCourse	An entity that represents information about training course for bird
29	TrainingCourseCheckOutPolicy	An entity that represents information about payment when check out
30	BirdCertificate	An entity that represents certificate information of training course

### 3.2 Bird Training Online Course

#### 3.2.1 Buy Online Course

- Function trigger: this function triggers when the customer click on buy course
- Function description: only customer can buy online courses
- 



The screenshot shows a website for 'BirdTrainingCenter'. At the top, there's a navigation bar with links for Home, Courses, Workshops, Services, Help, and Setting. The main content area features a large image of a green and yellow parrot. Below the image, a dark overlay displays the price '20000 VND' in yellow text. To the right of the image is a 'Receipt Summary' section containing the following details:

Receipt Summary	
Price	20000 VND
Discount	0 VND
Discounted Rate	0 VND
Membership Name	Standard
<b>Total Price</b>	<b>20000 VND</b>

Below the summary, a note says 'Your credit card information is encrypted' with a lock icon. There's a 'Secure, 1-click checkout with Link' button. The checkout form includes fields for Card number (1234 1234 1234 1234), Expiration (MM / YY), CVC, and Country (Vietnam). A small note below the form states: 'By providing your card information, you allow Thesis Bird\_Training to charge your card for future payments in accordance with their terms.' At the bottom right is a large blue 'Pay now' button.

#### 3.2.2 Study Online Course

- Function trigger: After buying course, user can study course (path: /onlinecourse/study/:courseid)
- Function description: customer who has bought the course has the access to study

**BirdTrainingCenter**

Home Courses Workshops Services Hi Pham Nhat An Setting

## MINI COURSE - Target Training

**Introduction**

**Training Steps**

- Step 1(Completed)
- Step 2(Completed)
- Step 3(Completed)
- Step 4(Completed)
- Step 5(Completed)
- Step 6(Completed)

**Additional Resources**

**testing part 2**

**da report new chua**

**Step 2**

Begin to move your hand to different spots of the cage or play stand, encouraging your bird to follow your hand around the environment.

Click as they take each treat.

0:00 / 0:17

Step 2. Begin to move your hand to different spots of the cage or play-stand, encouraging your bird to follow your hand around the environment. Click as they take each treat.

### 3.2.2 Create Online Course, section, lesson

- Function trigger: path(/management/onlinecourse)
- Function description: Only Manager has the access to create online course, others can view

**Bird Training Center Management**

**Manager**

- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

**Courses**

Image	Title	Short Description	Price (vnd)	Status
	MINI COURSE - Target Training	<a href="#">VIEW DETAIL</a>	10,000(vnd)	<input checked="" type="checkbox"/>
	MINI COURSE - Stationing Training	<a href="#">VIEW DETAIL</a>	50,000(vnd)	<input type="checkbox"/>
	Test Online Course 01	<a href="#">VIEW DETAIL</a>	100,000(vnd)	<input checked="" type="checkbox"/>
	TEST ONLINE COURSE	<a href="#">VIEW DETAIL</a>	100,000(vnd)	<input type="checkbox"/>
	Course Test 1	<a href="#">VIEW DETAIL</a>	20,000(vnd)	<input checked="" type="checkbox"/>
	1	<a href="#">VIEW DETAIL</a>	10,000(vnd)	<input type="checkbox"/>

\*click on the “ADD NEW COURSE” button

The screenshot shows the 'Create New Online Course' screen within the Bird Training Center Management application. The left sidebar displays navigation options: Manager (selected), Dashboard, Consultant, Online Course (selected), Timetable, Workshop, and Academy.

**Create New Online Course**

**General information**

- Title \* (Text input field)
- Price \* (Text input field)

**Short Description**

A rich text editor toolbar is available above a large text area for the short description.

**Pictures**

An 'UPLOAD IMAGE' button is present, and a message indicates 'No file is provided!'.

**CREATE ONLINE COURSE**

**Course Structure Screen**

The main area shows a course structure with sections: Introduction, Training Steps, and Additional Resources. Each section has an 'UPDATE' button and a 'DELETE' button.

**Add More Section:**

A modal window is open for adding a new section:

- Section Title \***: A text input field.
- Description**: A rich text editor toolbar and a large text area.
- UPLOAD RESOURCE FILE**: A button to upload files.
- No file is provided!**: A message indicating no file has been uploaded.
- CANCEL**: A button to cancel the operation.
- CREATE NEW SECTION**: A button to create the new section.

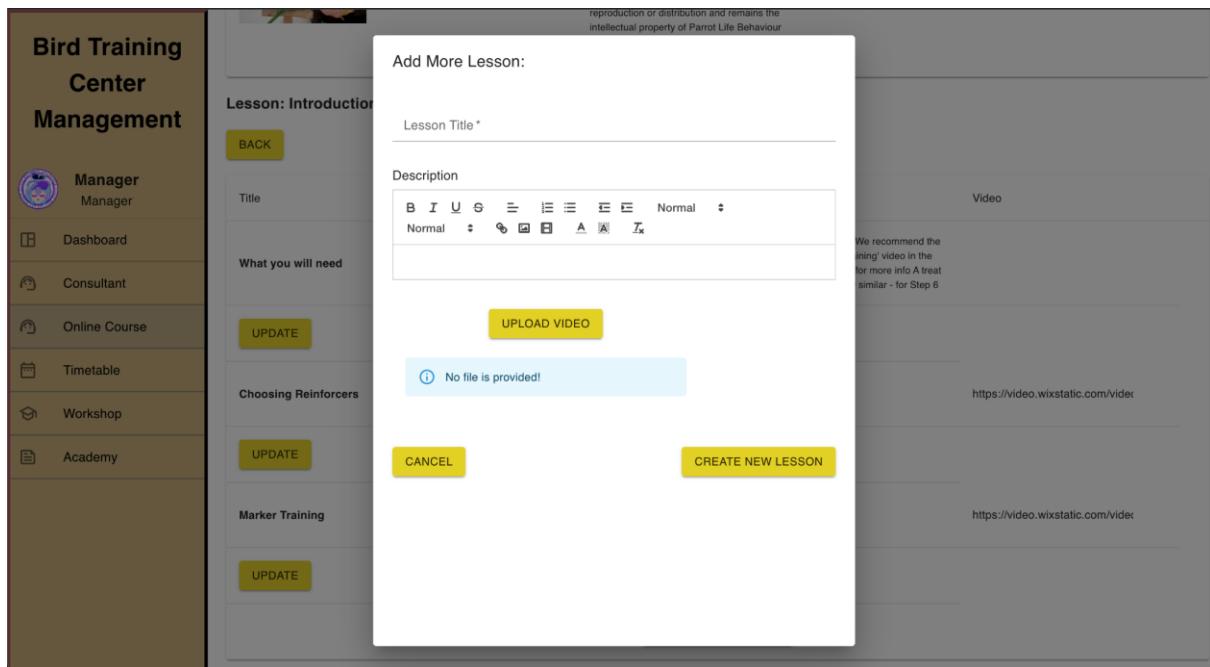
Below the modal, two URLs are displayed:

- https://storage.googleapis.com/birdtrainingcentersystem.appspot.com/online-course-resources/a92f849c-85ca-491d-90ca-312d4016b76c.pdf
- https://storage.googleapis.com/birdtrainingcentersystem.appspot.com/online-course-resources/04be4484-19fb-4005-8903-6a5cc3ae5c79.pdf

Another URL is also visible at the bottom right:

https://storage.googleapis.com/birdtrainingcentersystem.appspot.com/online-course-resources/Birdcon.png

\* After creating the course, click on the course. then click on create section



\* After creating the section, click on the section then click create lesson

### 3.3 Workshop

#### 3.2.1 Enrol Workshop

- Function trigger: trigger when customer click on Enrol now at workshop class
- Function description: only Customer has the ability to access this functionality

**Bird Training Academy 301**

**Description:**  
Learn how to shape behaviour like a pro, create your own training plans, problem solve and train any behaviour you can dream up. Special focus on harness training and 'chained series' of behaviours. Learn about problem behaviours, apply your knowledge to assessing and correcting them.

**Start-Time:** 13:00:00 / **End-Time:** 14:00:00  
**Date:** January 9th, 2024

**Trainer:** Rachel Riley

**Description:**  
Applying this training with live birds.

**Start-Time:** 14:00:00 / **End-Time:** 15:00:00

**Contact Us**

Hotline: 1800 152508

Address: Block E2a-7, D1 Street Saigon Hi-tech Park, Long Thanh My Ward, District 9, Ho Chi Minh City, Vietnam

### 3.2.2 Create Workshop

- Function trigger: path(/management/workshop)
- Function description: Only Manager has the ability to access

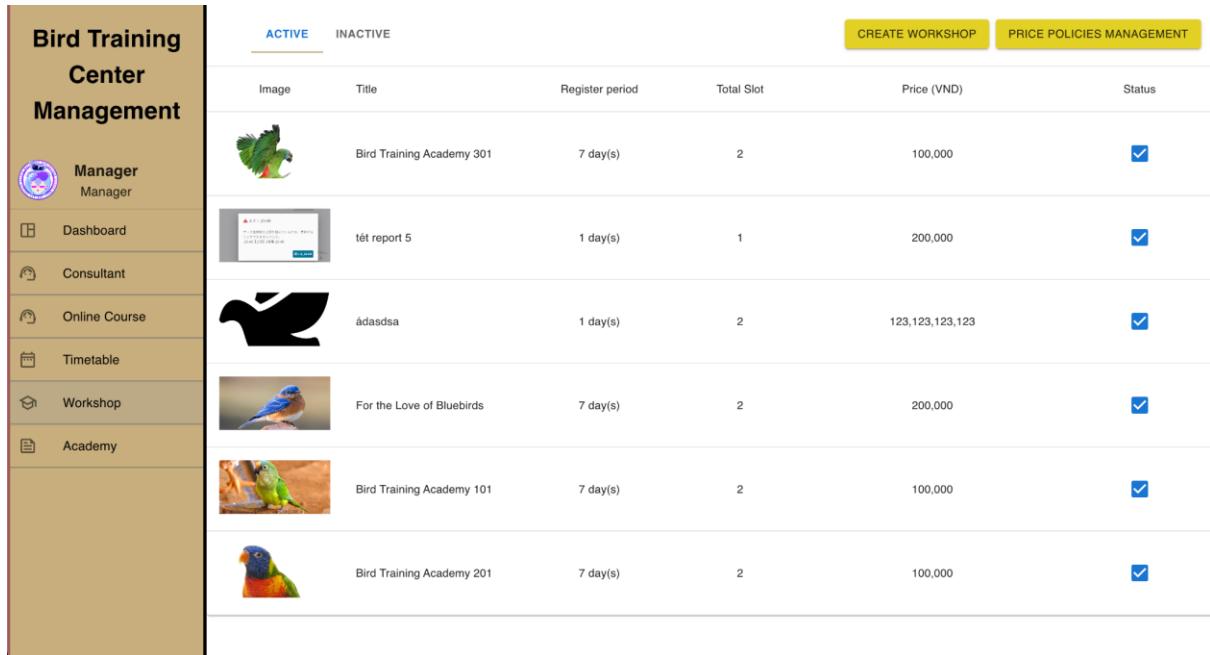
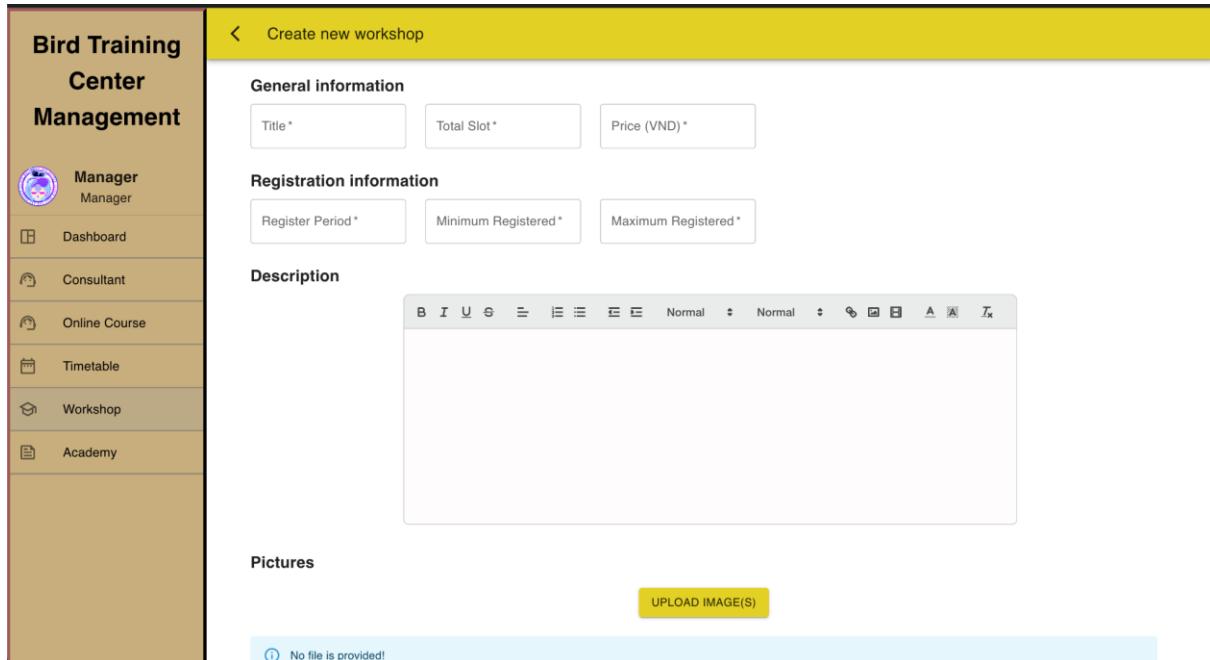


Image	Title	Register period	Total Slot	Price (VND)	Status
	Bird Training Academy 301	7 day(s)	2	100.000	<input checked="" type="checkbox"/>
	tét report 5	1 day(s)	1	200.000	<input checked="" type="checkbox"/>
	ádasdsa	1 day(s)	2	123,123,123,123	<input checked="" type="checkbox"/>
	For the Love of Bluebirds	7 day(s)	2	200.000	<input checked="" type="checkbox"/>
	Bird Training Academy 101	7 day(s)	2	100.000	<input checked="" type="checkbox"/>
	Bird Training Academy 201	7 day(s)	2	100.000	<input checked="" type="checkbox"/>

\* click "CREATE WORKSHOP"



**General information**




**Registration information**




**Description**

**Pictures**

**UPLOAD IMAGE(S)**

No file is provided!

\* After creating the workshop, right click then click on “Workshop Class”

### 3.2.3 Check attendance for a Workshop class

- Function trigger: path(/management/timetable)
- Function description: Staff and Trainer can use this functionality

\* Click on “Host workshop class slot”

The screenshot shows a user interface for managing bird training center users. On the left is a sidebar with a logo, the title 'Bird Training Center Management', and a navigation menu with links: Manager (selected), Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy.

The main area displays a table of 12 users:

	Email	Name	Phone	Action
4	customer1@gmail.com	Pham Nhat An	+84 908941767	<input checked="" type="checkbox"/>
5	alice.johnson@email.com	Alice Johnson	+84 112233445	<input checked="" type="checkbox"/>
6	customer3@gmail.com	Clark Kent	+84 908941769	<input checked="" type="checkbox"/>
7	emma.walker@email.com	Emma Walker	+84 445566777	<input checked="" type="checkbox"/>
8	robert.white@email.com	Robert White	+84 111222333	<input checked="" type="checkbox"/>
9	daniel.nguyen@email.com	Daniel Nguyen	+84 988877666	<input checked="" type="checkbox"/>
10	david.wilson@email.com	David Wilson	+84 554433211	<input checked="" type="checkbox"/>
11	nguyenthothaibao@gmail.com	Nguyễn Thủ Thái Bảo	+84 909290138	<input checked="" type="checkbox"/>
12	christopher.turner@email.com	Christopher Turner	+84 333444555	<input checked="" type="checkbox"/>

A yellow button at the bottom right is labeled 'SUBMIT ATTENDANCE'.

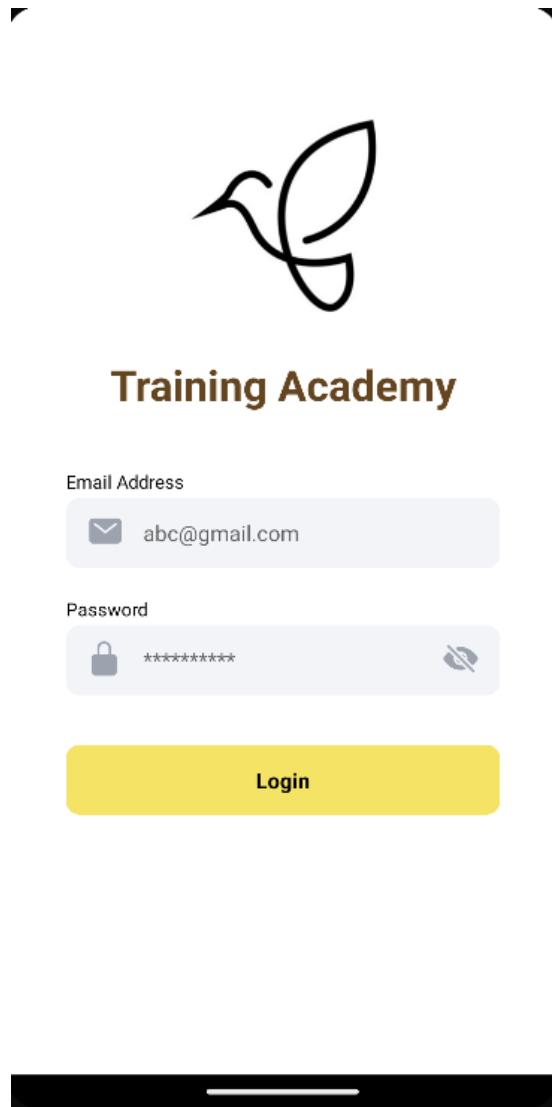
\* Check attendance then click “Submit Attendance” to save

### 3.4 Bird Training Academy

#### 3.4.1 Mobile Application

##### 3.4.1.1 <Guest> Authenticate/Authorization

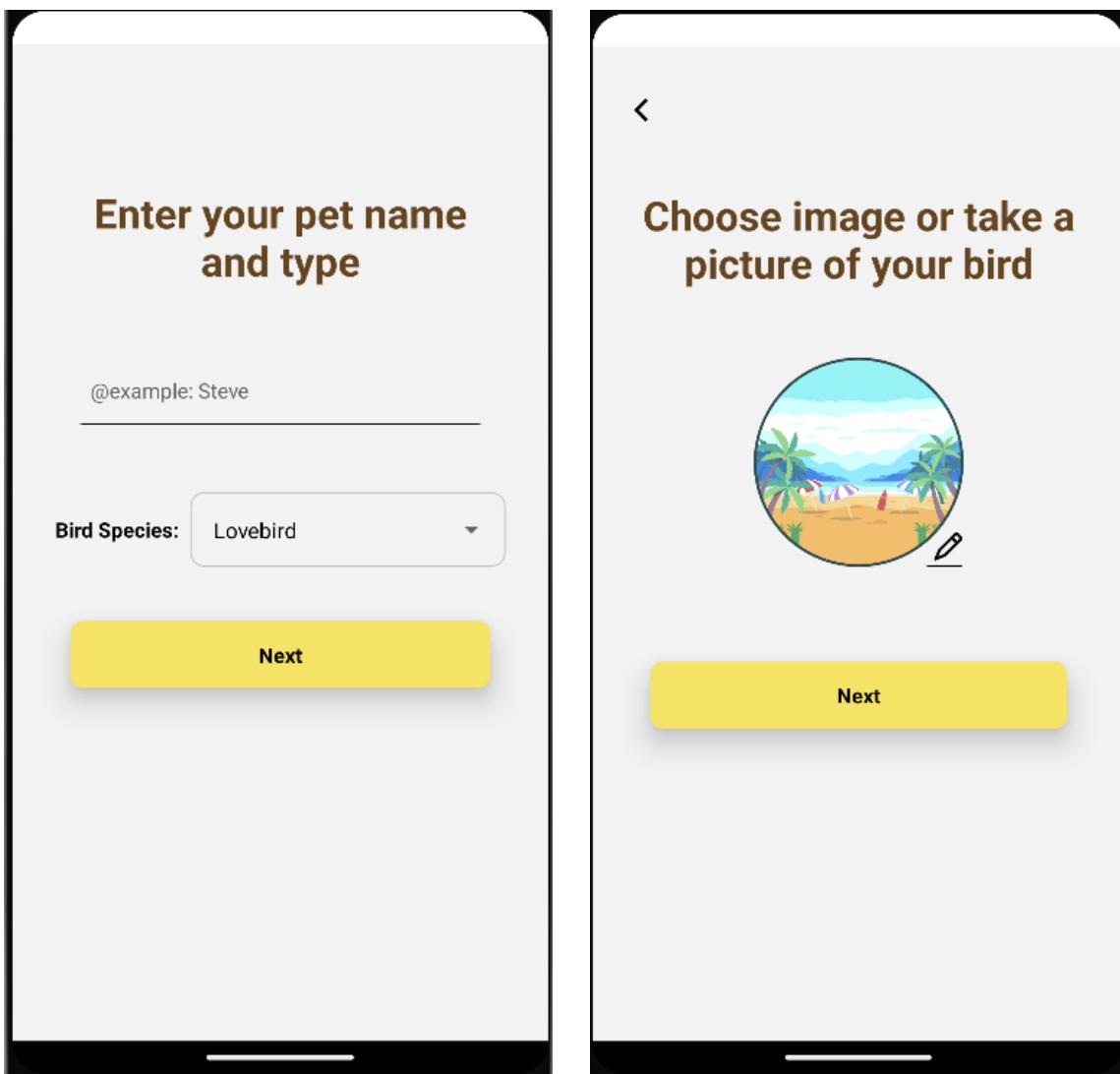
- Function trigger: Guest request to log into the system by Clicking the “Login” button
- Function description: Guest logs in to the system to verify the Customer role and access to the feature for Customer on the mobile application
- Function details:
  - Guest input email and password into the text input fields ( email and password must be create and sign up in Bird Training Center website)
  - Guest taps button “Login”
  - The system queries the email and password combination to verify if the account exists and have the role of Customer
  - The system validates the resolving content:
    - Success: Systems generate a JWT token and return the tokens to the mobile app, the application start loading and navigate to the Home Screen
    - Failure: The systems show an error message and return
- Screen Layout:



#### 3.4.1.2 <Customer> register Bird name and image

- Function Trigger: Customer access to the application navigated to the registered bird name, bird type and image
- Function description: Registered bird detail information including Name, bird's type and image, so as the system automatically provided Training Course based on Type of the bird registered
- Function Details:
  - Customer first time logs into the application will navigate to the register bird name, type and image
  - Customer input "Name" field
  - Customer Choose the field Bird Species
  - The system validates the resolving content
    - Success: Navigate to set Image for your Bird
    - Fail: Show Error that name input is invalid or empty

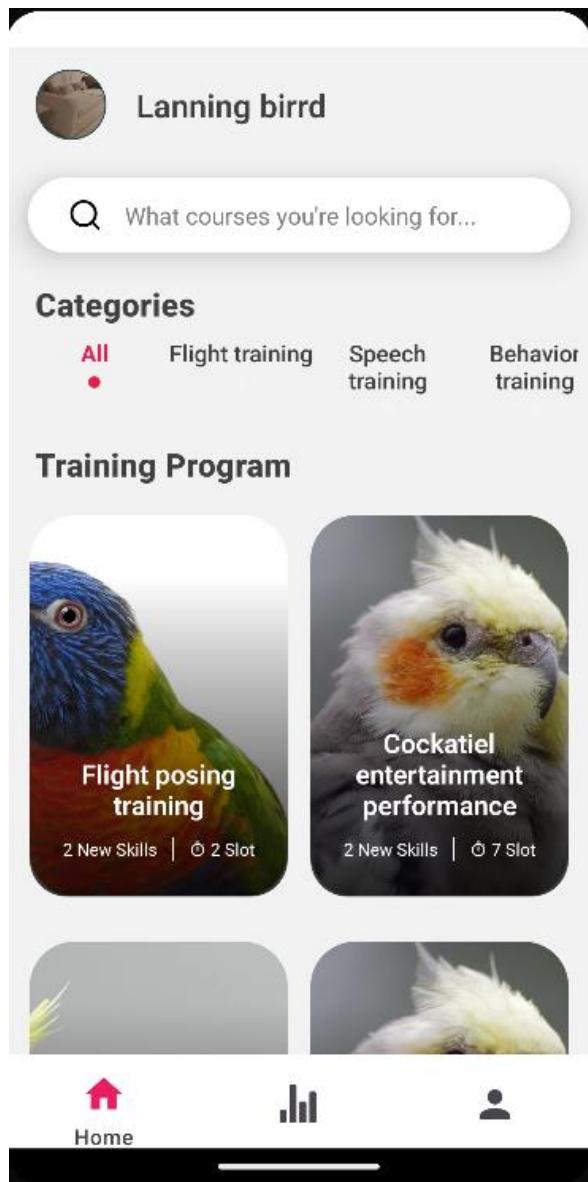
- Customer Choose an image from mobile storage or take a picture to set Image for Customer's bird
- The system validates the resolving content
  - Success: Process and navigate to the Home Screen
  - Fail: Show Error that image input is empty
- Screen layout:



#### 3.4.1.3 <Customer> Can View List and Manage Training Course

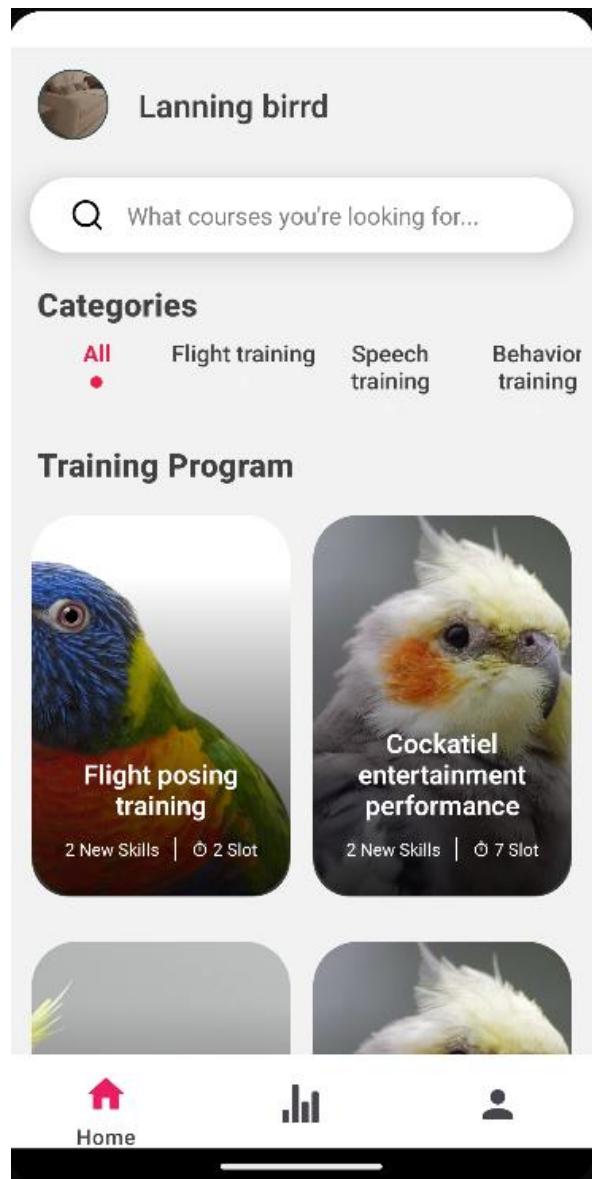
- Function Trigger: Customer Login to the application or After Set Bird Name, Bird Type and Image and is navigated to the Home screen tab
- Function description: Customer can view all training Course including slot training, and all new Skills that your bird will receive after completed that have been assigned automatically base on bird type that Customer registered Bird from the 3.4.1.2
- Function details:
  - After successfully Login or Successfully registered Bird name, type and image, Customer is redirected to the homepage and can view Courses of different skills
  - Customer tab categories option in list base on scroll to the left to select specific skill so to filter course that have that skill

- Customer Can search course base on name of that Course
- Screen Layout:



#### 3.4.1.4 <Customer> Can Manage Course

- Function Trigger: Customer tab on the skill in the Categories List and can search specific Course based on Name of the Training Course
- Function description: Customer can apply categories filter base on skill to show specific course based on skill and search Course base on Name
- Function details:
  - After successfully Login or Successfully registered Bird name, type and image, Customer is redirected to the homepage and can view Courses of different skills
  - Customer tab categories option in list base on scroll to the left to select specific skill so to filter course that have that skill
  - Customer Can search course base on name of that Course
- Screen Layout:



### 3.4.1.5 <Customer> Manage Course

#### 3.4.1.5.1 Register Course

- Function Trigger: Customer tab on the course to view details about the training course
- Function description: Customer can register their bird to training course so as to start training to earn the skill Customer want their bird to have
- Function details:
  - After successfully Login or Successfully registered Bird name, type and image, Customer is redirected to the homepage and can view Courses of different skills
  - Customer tab on the course that they want their bird to study
  - Customer view details of information about the course and tabs Registered
  - Course popup Dialog success and change status of button to Registered and unclickable
- Screen Layout:



## Cockatiel flight racing

Price: 200.000 ₫

### About Course:

Training for racing

### More about course

⌚ 15 Slot

鸟成长 Type: Cockatiel



Flight  
training



Agility  
training

Register



## Flight racing

Price: 300000 VND

### About Course:

Training for racing

### More about course

⌚ 10 Slot

鸟成长 Type: Canary



Flight  
training



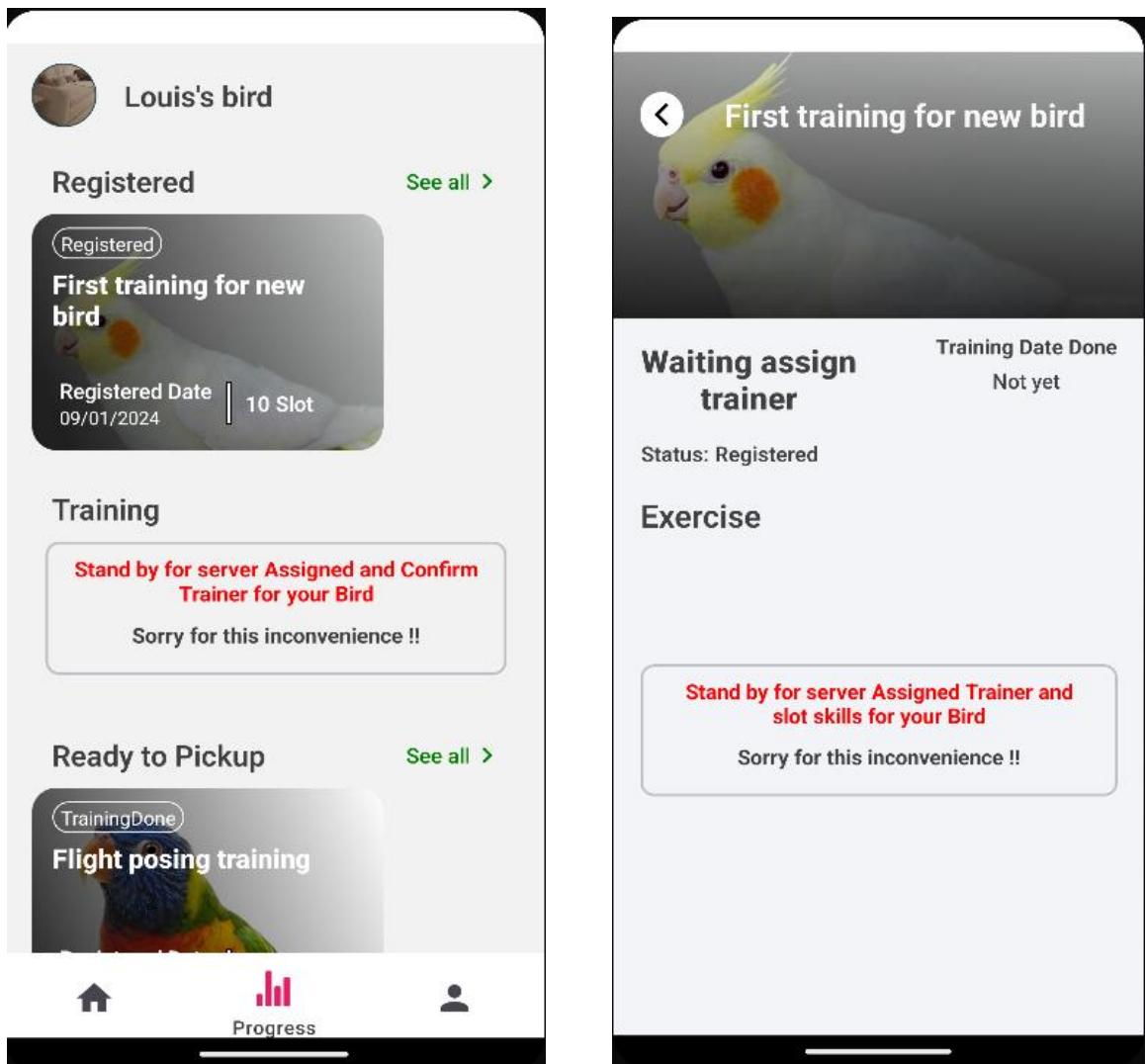
Agility  
training

✅ Registered

3.4.1.6 <Customer> can view list of status that system training bird update

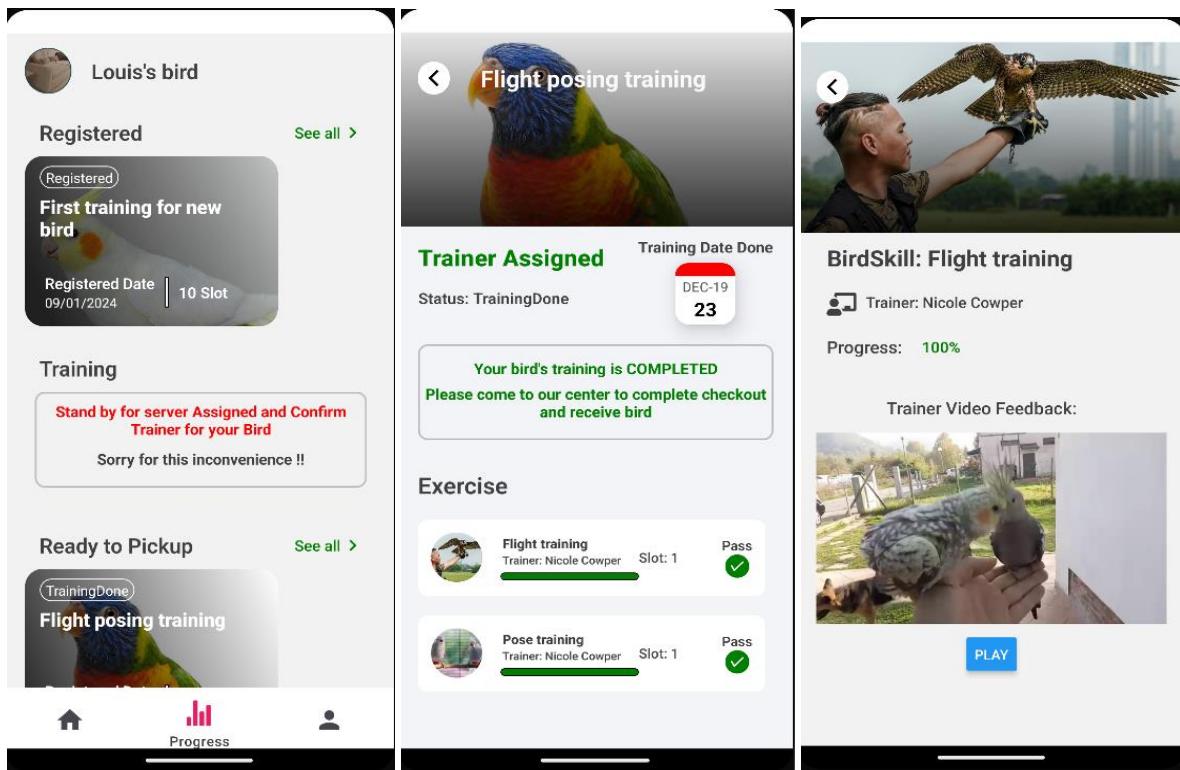
3.4.1.6.1 View Course Status Registered and View Details

- Function Trigger: Customer after registered Course successfully and tab on the “Progress” button on the bottom navigation bar
- Function description: Customer can view Course with status Registered and wait for the system to assign trainer and slot time for training course
- Function details:
  - After successfully registered a training course in 3.4.1.5.1
  - Customer tab on the “Progress” button on the bottom navigation to navigate to the course that have status “Registered”
  - Customer tab on the Course in the “Registered” field and view inside
- Screen Layout:



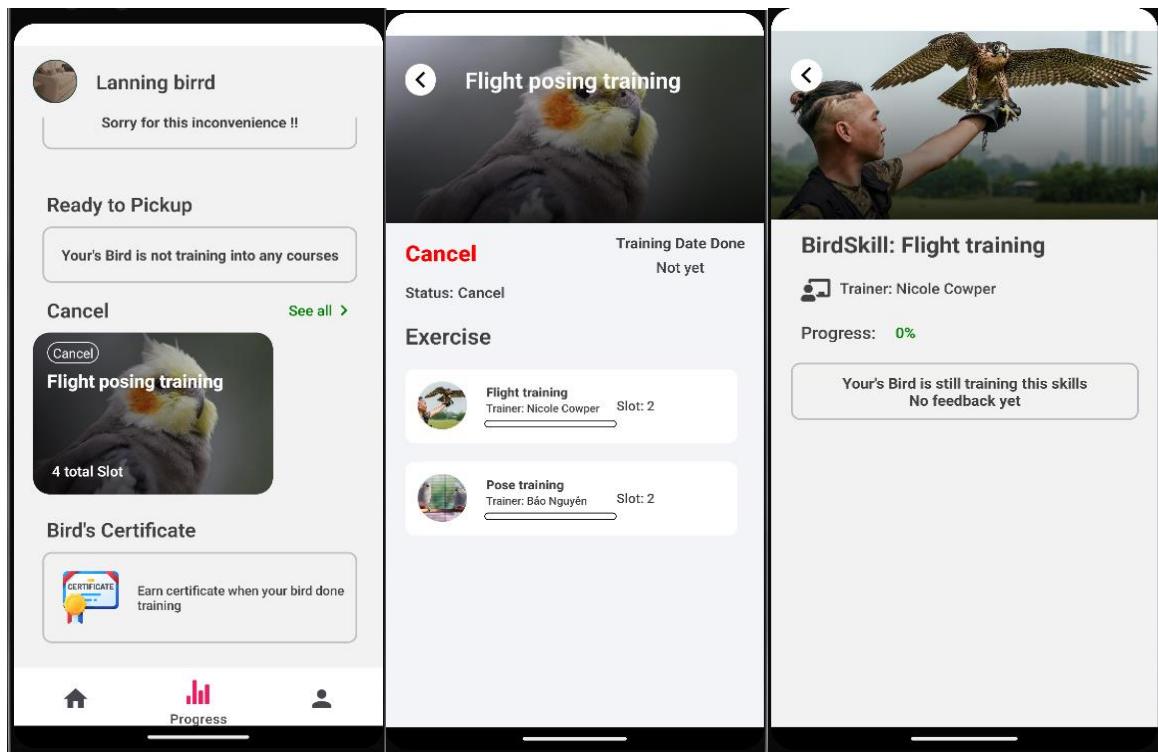
#### 3.4.1.6.2 View Course Status Ready to Pickup, view details and feedback

- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Progress” button on the bottom navigation bar and can only view to check the status of the bird
- Function description: Customer can view Course with status “Ready to Pickup” and start to collect bird from the center and start payment
- Function details:
  - Bird training at the center complete and system update the status to “Ready to Pickup” so as to notify the customer to come to the center to collect the bird and complete payment
  - Customer can view the status “Ready to Pickup” by tab on the “Progress” button on the bottom navigation to navigate to the course that have status “Ready to Pickup”
  - Customer tab on the Course in the “Ready to Pickup” field and view inside
  - Customer tab on the exercise section to view the feedback about the Training with there bird
- Screen Layout:



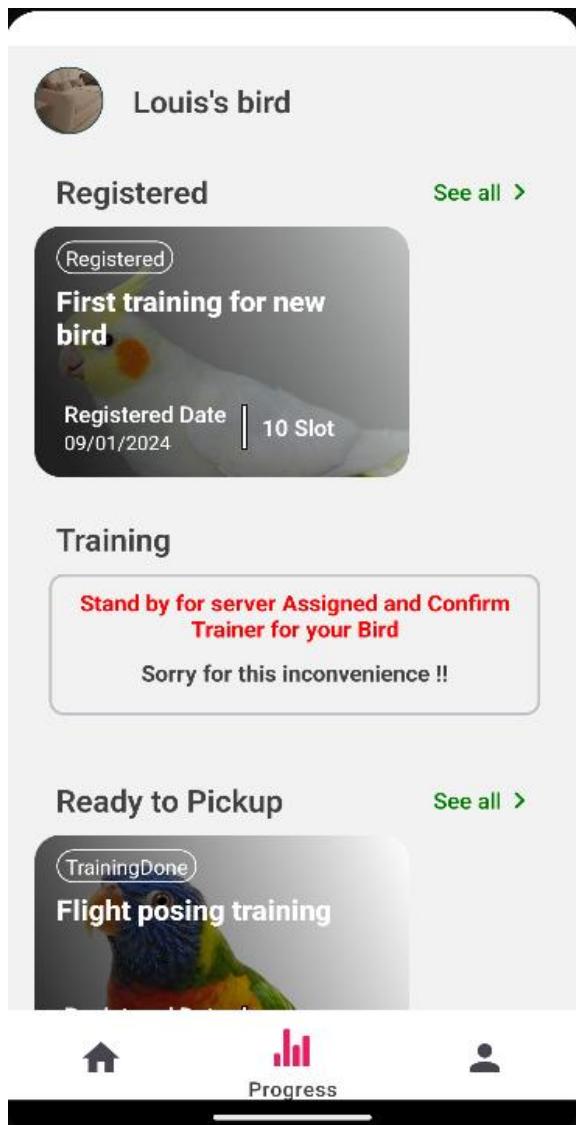
### 3.4.1.6.3 View Course Status Cancel, view details and feedback

- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Progress” button on the bottom navigation bar and can only view to check the status of the bird
- Function description: Customer can view Course with status “**Cancel**” and view details
- Function details:
  - If Customer Cancel training bird or unexpected event happen make the bird can not training, System will update status “**Cancel**”
  - Customer can view the status “**Cancel**” by tab on the “Progress” button on the bottom navigation to navigate to the course that have status “Ready to Pickup”
  - Customer tab on the Course in the “**Cancel**” field and view inside
  - Customer tab on the exercise section to view the feedback about the Training with there bird
- Screen Layout:



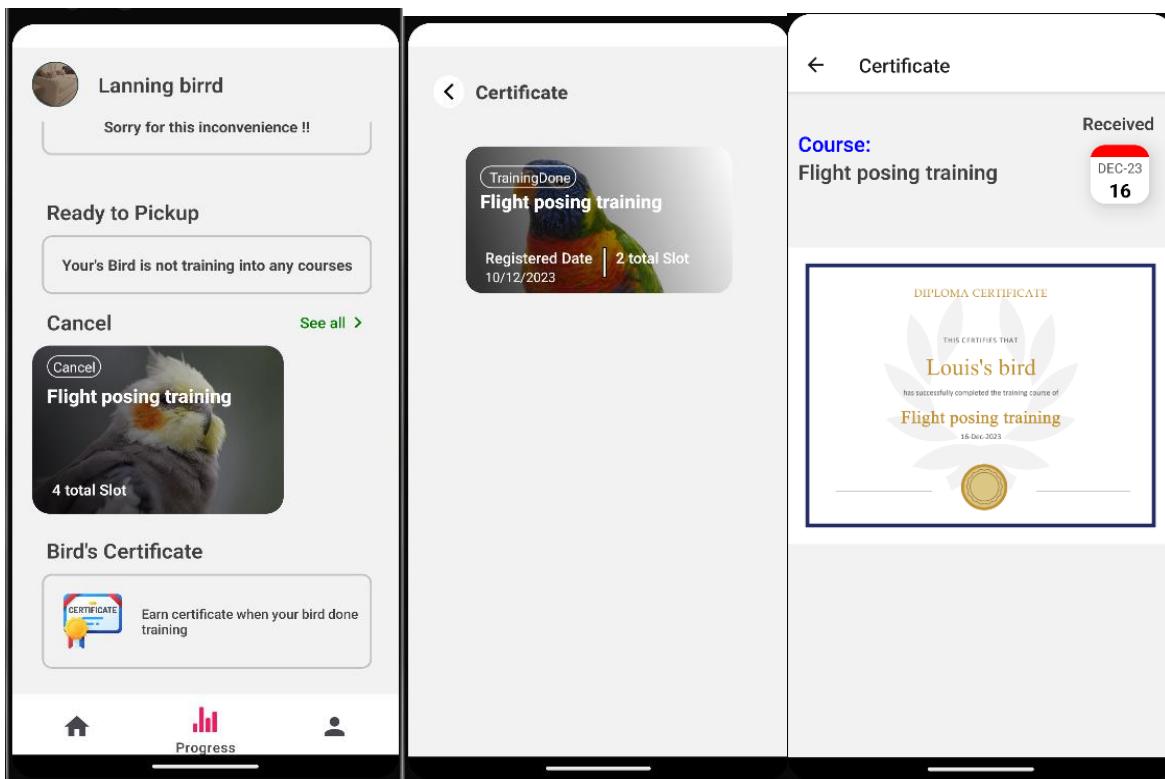
#### 3.4.1.6.4 View Course Status Training

- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Progress” button on the bottom navigation bar and can only view to check the status of the bird
- Function description: Customer can view Course with status “**Training**”
- Function details:
  - After system receive bird Check in from Customer system will update the status of Training Course to Customer to “**Training**”
  - Customer can view the status “**Training**” by tab on the “Progress” button on the bottom navigation to navigate to the course that have status “Ready to Pickup”
  - Customer tab on the Course in the “**Training**” field and view inside
  - Customer can not click to view section in details screen due to lock by the system, only open when slot status to Pass or Not Pass
- Screen Layout:



3.4.1.6.5 View Certificate of the bird that have completed Training Course and Complete if system return bird to customer successfully

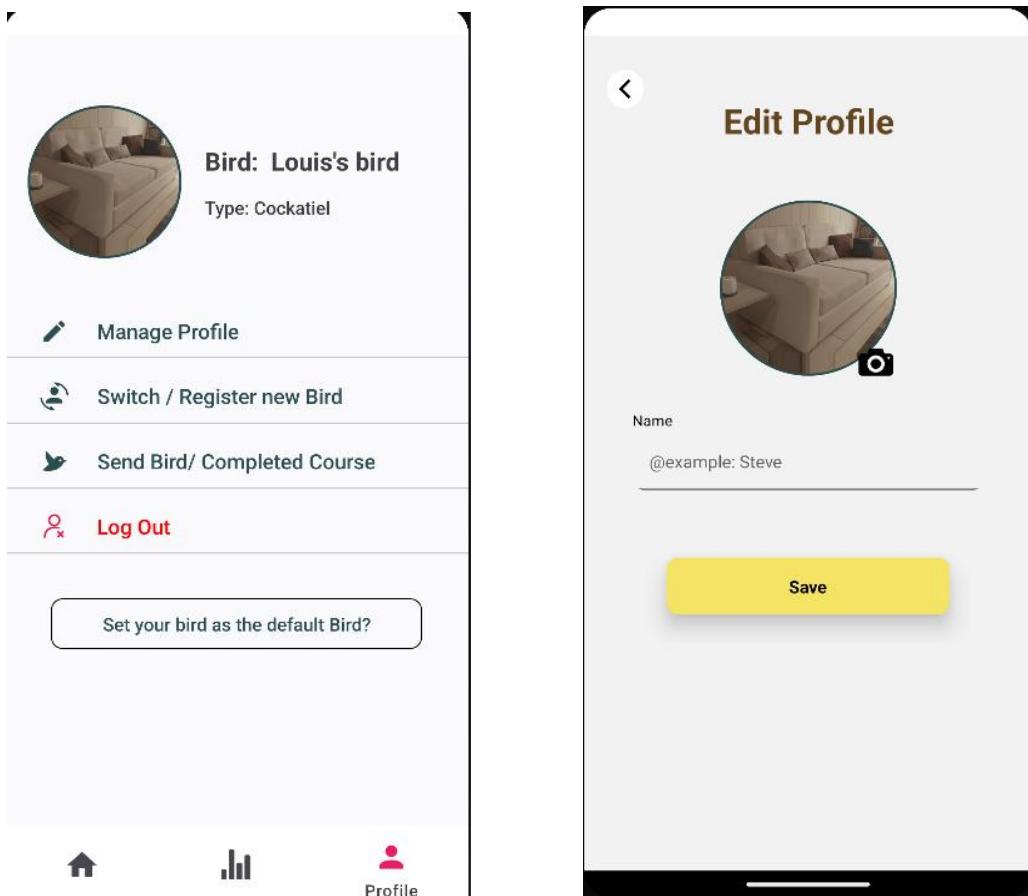
- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Progress” button on the bottom navigation bar and can only view to check the status of the bird
- Function description: Customer can view Certificate with status “**Completed**”
- Function details:
  - After system training bird for Customer complete and return bird to the customer successfully, system will update the status of training Course to Customer to “**Completed**”
  - Customer can view the status “**Completed**” by tab on the “Progress” button on the bottom navigation
  - Customer tab on the “**Certificate**” field and view inside
- Screen Layout:



### 3.4.1.7 <Customer>can Manage bird

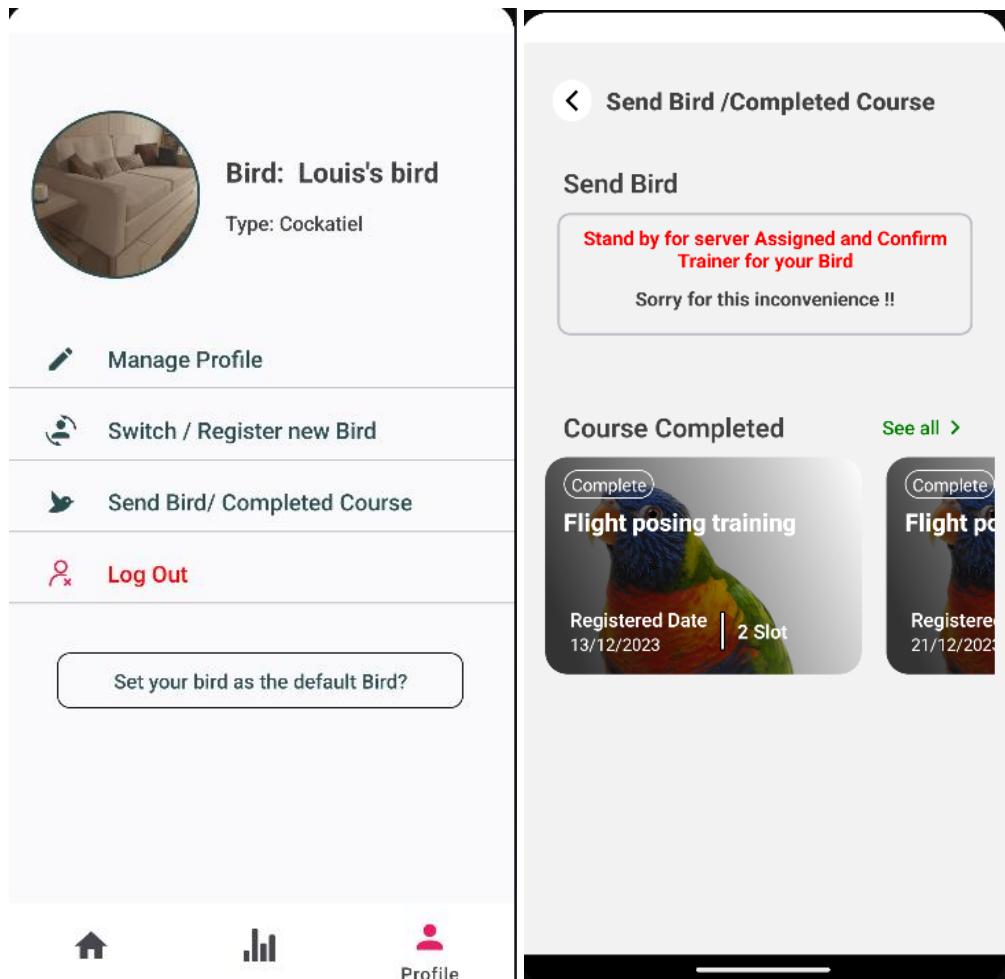
#### 3.4.1.7.1 Can manage profile to change name or image

- Function Trigger: Customer tap on the Manage Profile in the “Profile” tab on the bottom navigation
- Function description: Customer can edit, update bird name and, change image of there bird
- Function details:
  - Customer click on the “Profile” on the bottom of the navigation to view Profile and Settings
  - Customer click on the button “Manage Profile” and open form menu and Image Picker
  - Customer fill the data in the name input field
  - Picture is optional if Customer not change if will keep the old one
  - Customer Click “Save” button and proceed back to the Profile Screen with data update
- Screen Layout:



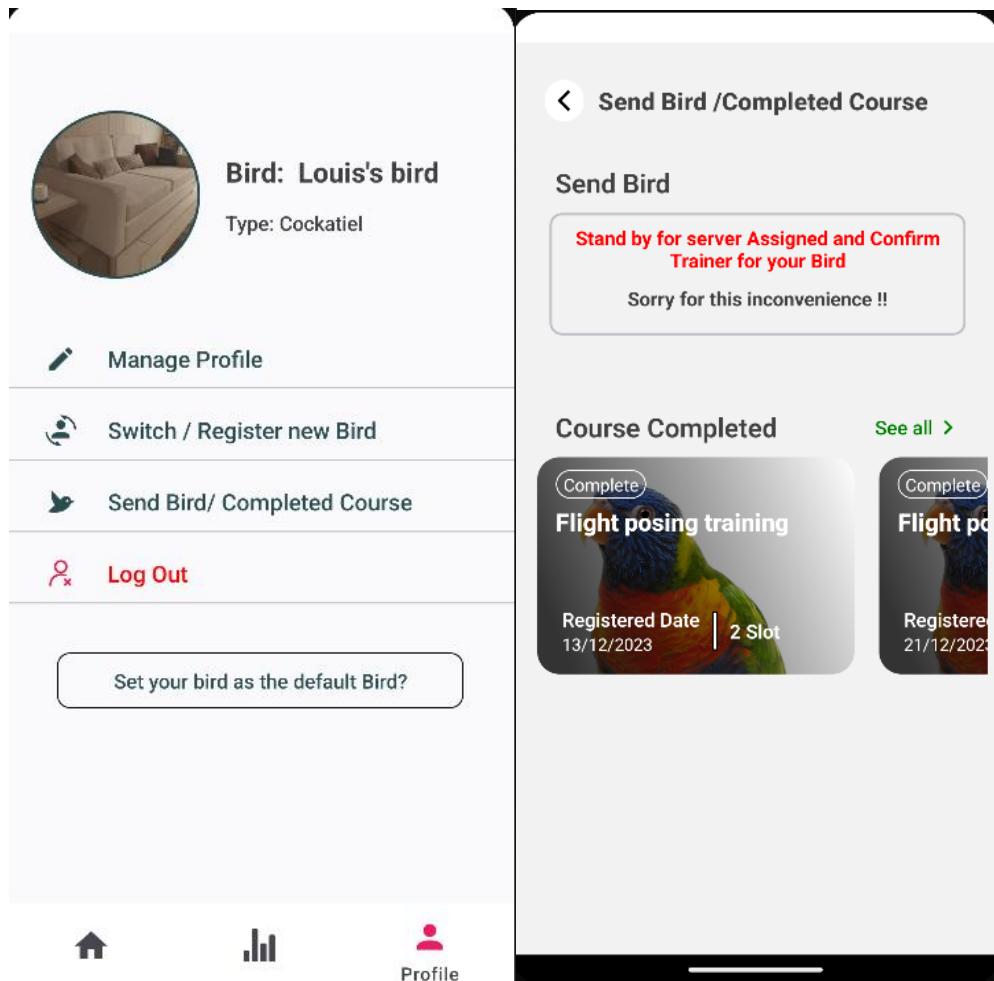
#### 3.4.1.7.2 <Customer> Can View “Send Bird” to check whether your bird ready to send to the Center

- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Profile” button on the bottom navigation bar and can only view to check whether it is ready to start send your bird to the center
- Function description: Customer can view Status and notification about the Course when it ready to send the bird to the customer
- Function details:
  - After system Assign trainer and slot, time to the course complete, it will change the status so as to notify the customer that it ready to bring the bird and start training
  - Customer click in “Send Bird/ Completed Course” to check whether to send your bird to the center or not
  - Customer tab on the Course on the title Send bird to check detail and notification field inside
- Screen Layout:



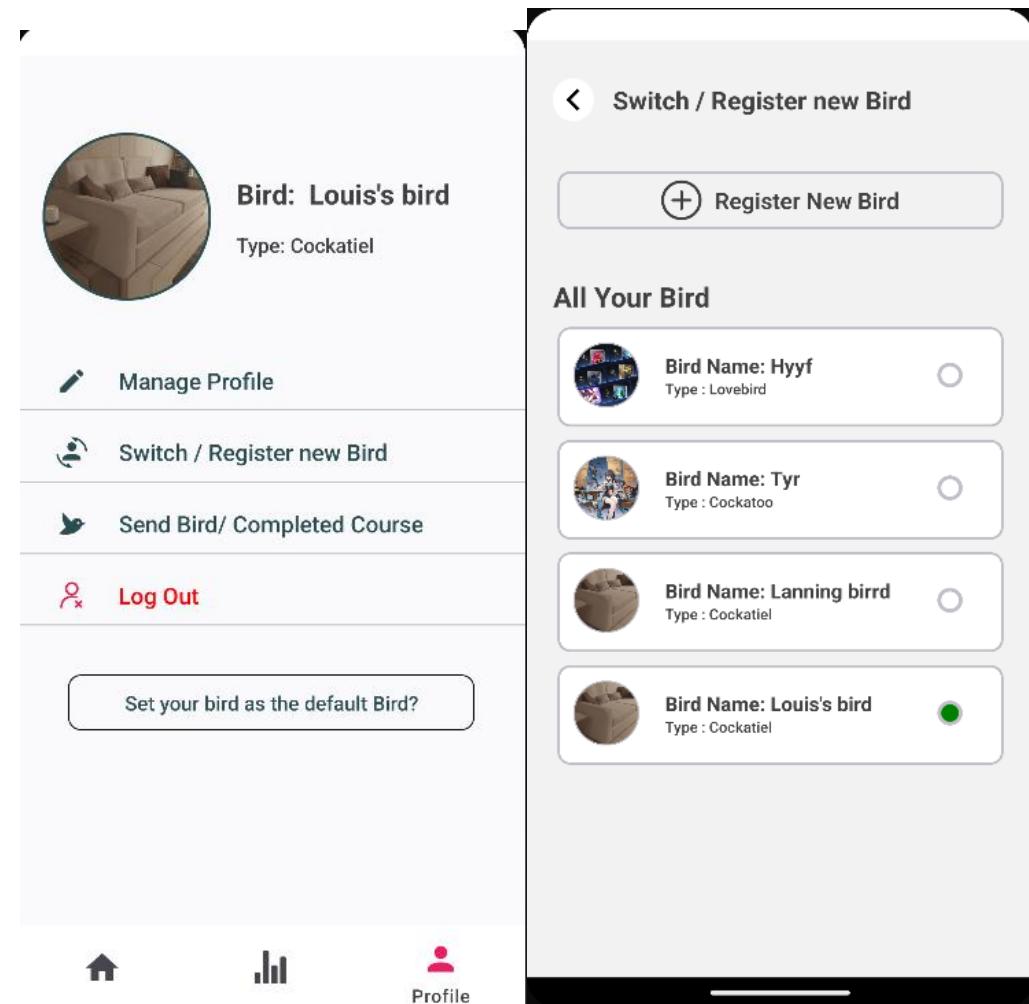
#### 3.4.1.7.3 <Customer> Can View “Completed Course” to check whether your Completed Course and return to the Customer Successfully

- Function Trigger: System update status based on that bird training and return status to the Customer. Customer tab on the “Profile” button on the bottom navigation bar and can only view to check whether your bird Complete all course and return to the Customer Successfully
- Function description: Customer can view Status and notification about the Course when it Complete all course and return to the Customer Successfully
- Function details:
  - Bird training at the center complete and return to the customer Successful so that System will update status to Completed
  - Customer can view the status “Completed” by tab on the “Profile” button on the bottom navigation then click the button “SendBird/ Completed Course” to view list of the Completed Course
  - Customer tab on the Course in the “Completed Course” field and view inside
- Screen Layout:



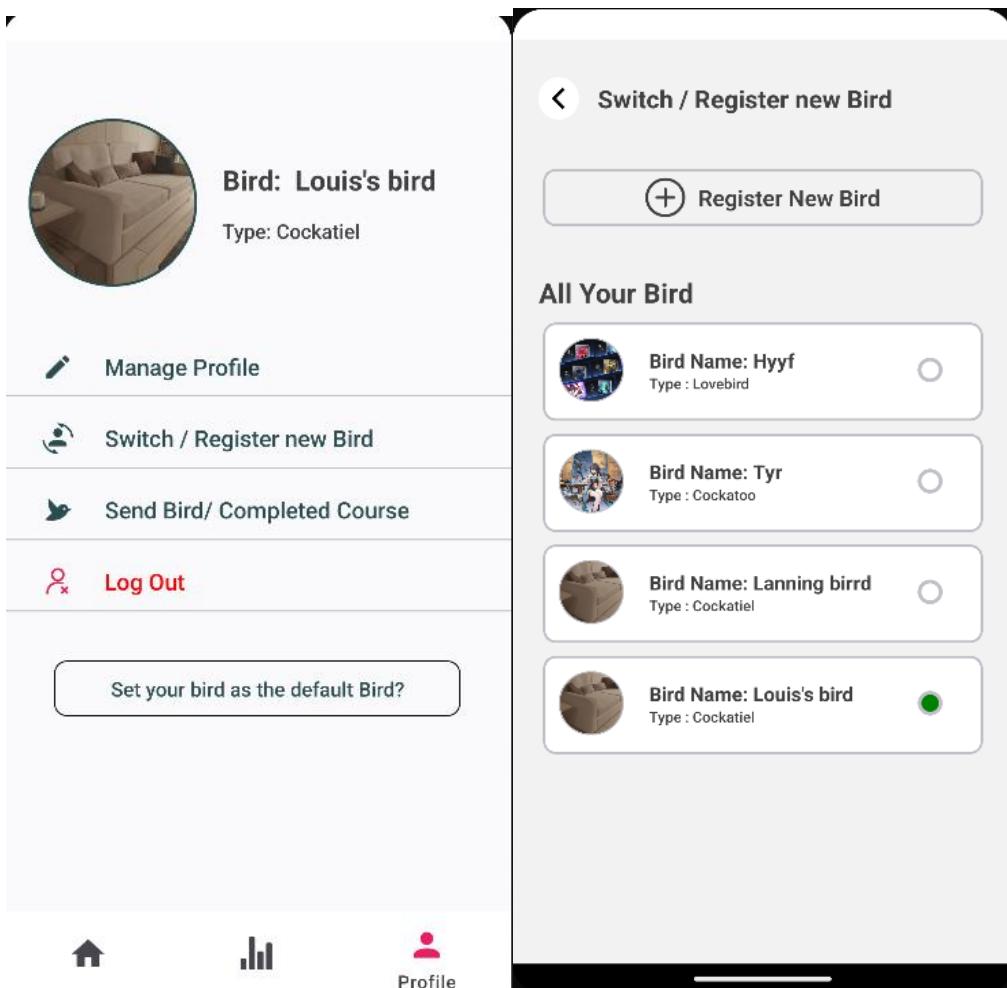
#### 3.4.1.7.4 <Customer> Can Register new bird

- Function Trigger: Customer tap on the “Switch/ Register new Bird” in the “Profile” tab on the bottom navigation
- Function description: Customer can create a new Bird to system with different or the same bird type
- Function details:
  - Customer click on the “Profile” on the bottom of the navigation to view Profile and Settings
  - Customer click on the button “Switch/ Register new Bird” and start register new bird
  - Customer input “Name” field
  - Customer Choose the field Bird Species
  - The system validates the resolving content
    - Success: Navigate to set Image for your Bird
    - Fail: Show Error that name input is invalid or empty
  - Customer Choose an image from mobile storage or take a picture to set Image for Customer’s bird
  - The system validates the resolving content
    - Success: Process and navigate to the Home Screen
    - Fail: Show Error that image input is empty
- Screen Layout:



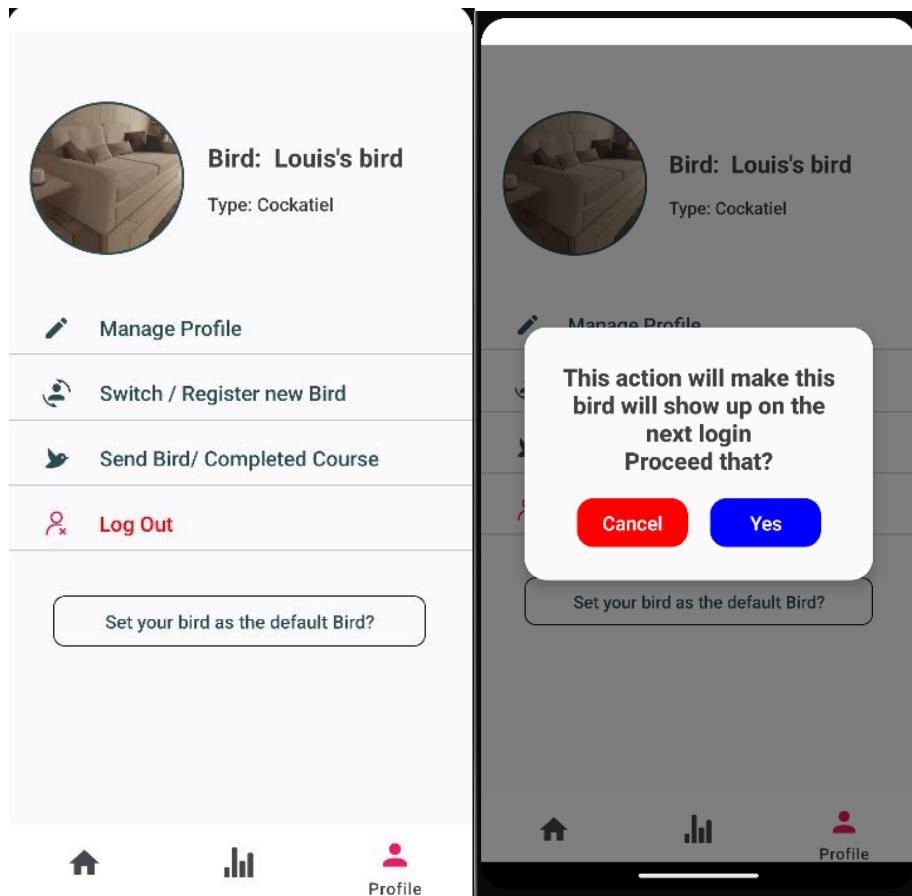
### 3.4.1.7.5 <Customer> Can Switch an already exist bird in the system

- Function Trigger: Customer tap on the “Switch Bird” in the “Profile” tab on the bottom navigation
- Function description: Customer can Switch an already exist bird that they create to the system
- Function details:
  - Customer click on the “Profile” on the bottom of the navigation to view Profile and Settings
  - Customer click on the button “Switch/ Register new Bird”
  - Customer choose bird they want to switch
  - It will navigate back to the home screen with the bird they choose to switch
- Screen Layout:



3.4.1.7.6 <Customer> can set bird as default so as the next login into account the bird will exist as main screen

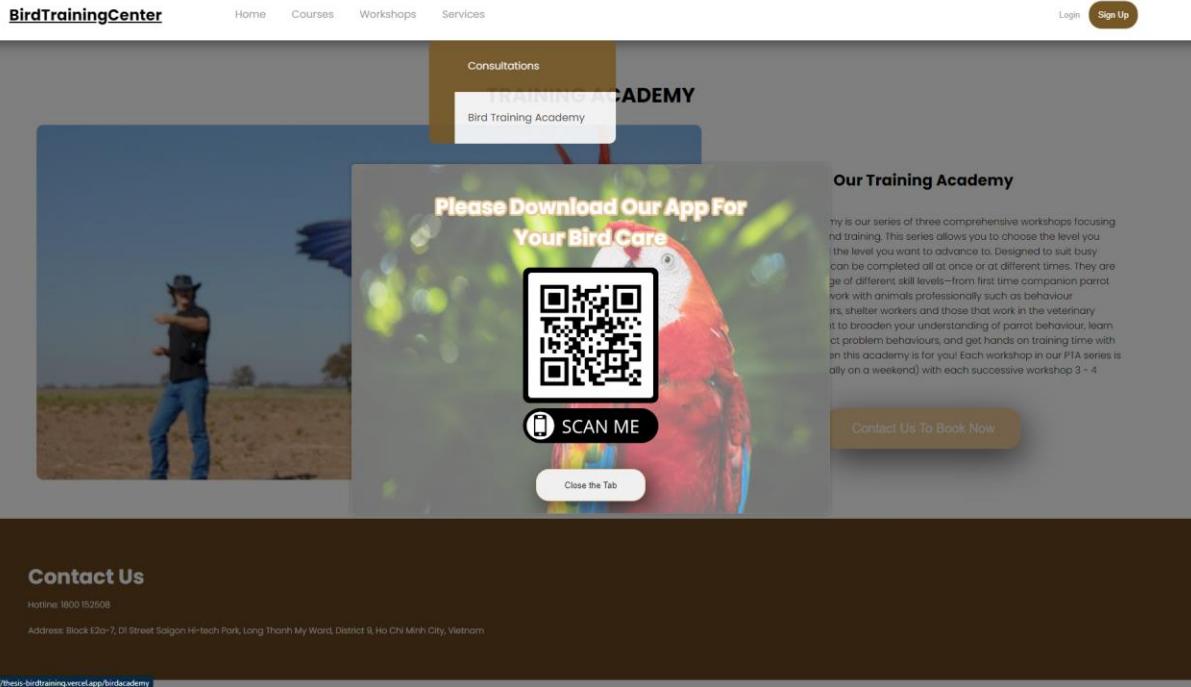
- Function Trigger: Customer tap on the “**set your Bird as the default Bird**” in the “Profile” tab on the bottom navigation
- Function description: Customer can set the bird to the next login into account that bird will present in the main home screen
- Function details:
  - Customer click on the “Profile” on the bottom of the navigation to view Profile and Settings
  - The system validates the resolving content
    - **Show Button “set your Bird as the default Bird”:** Your bird not a default bird so you can set that to set again
    - **Not Show Button “set your Bird as the default Bird”:** Your bird is a default bird so you do not need to that again
  - Customer click on the button “**set your Bird as the default Bird**”
  - Show a pop-up dialog to ask Customer certain with that action
  - The system validates the resolving content
    - Yes: It will navigate back to the home screen and when click back to the profile button **set your Bird as the default Bird** will disappear
    - No: Close the dialog and back to the normal
  - Log in again and see that your bird set default will present
- Screen Layout:



### 3.4.2 Web Application

#### 3.4.2.1 <Guest>, <Customer> Browse Website and download application to mobile

- Function Trigger: <Customer>, <Guest> hover “Service” tab on the navigation bar on the top to show the dropdown and click on the “Bird training Academy”. Then Click on “Contact us to Book Now” Button
- Function description: <Customer>, <Guest> can access to the page to show information about bird training academy and scan qr code to download our center app to mobile
- Function details:
  - <Customer>, <Guest> hover on the Service on the navigation bar and select “Bird training Academy” on the dropdown list.
  - <Customer>, <Guest> click button “Contact us to Book Now”
  - System show dialog popup QR code to download app to mobile
- Screen Layout:



### 3.4.2.2 <Manager> Manage Bird Training Academy

#### 3.4.2.2.1 <Manager> can create new training courses for bird to training

- Function Trigger: <Manager> click button “Create New Training Course”
- Function description: <Manager> can create new training course for bird
- Function details:
  - <Manager> access Management Web and click tab Academy
  - <Manager> select tab Training Course Management to view list course for bird training
  - <Manager> click button “Create new Training Course”
  - System will show up form to fill data create new course
  - <Manager> fill data input fields and choose image
  - <Manager> click Confirm Create Course
  - System will pop-up create successfully and back to page Training Course Management and update new Course just created
- Screen Layout:

Bird Training Center Management							
ACADEMY MANAGEMENT TRAINING COURSE MANAGEMENT PRICE POLICY MANAGEMENT							
CREATE NEW TRAINING COURSE							
Image	Title	Bird Species	Total Slot	Price (VND)	Status	VIEW DETAIL	
	eetestset12	Cockatoo	0	123,123,123	<input type="checkbox"/>	ACTIVE	<button>VIEW DETAIL</button>
	ting	Finch	0	122,000	<input type="checkbox"/>	ACTIVE	<button>VIEW DETAIL</button>
	Flight posing training	Cockatiel	4	100,000	<input checked="" type="checkbox"/>	DISABLE	<button>VIEW DETAIL</button>
	Flight posing training	Cockatiel	2	100,000	<input checked="" type="checkbox"/>	DISABLE	<button>VIEW DETAIL</button>
	First training for new bird	Cockatiel	10	100,000	<input checked="" type="checkbox"/>	DISABLE	<button>VIEW DETAIL</button>
	Monk bird	Quaker Parrot (Monk Parakeet)	0	50,000	<input type="checkbox"/>	ACTIVE	<button>VIEW DETAIL</button>
	Lovebird flight training	Lovebird	5	100,000	<input checked="" type="checkbox"/>	DISABLE	<button>VIEW DETAIL</button>
	Cockatiel entertainment performance	Cockatiel	7	250,000	<input checked="" type="checkbox"/>	DISABLE	<button>VIEW DETAIL</button>

Bird Training Center Management						
ACADEMY MANAGEMENT TRAINING COURSE MANAGEMENT PRICE POLICY MANAGEMENT						
<b>X Create Training Course</b>						
General information				Pictures		
<input type="button" value="Choose Species"/>				<input type="button" value="UPLOAD IMAGE(S)"/> <span>No file is provided!</span>		
Title * <input type="text"/>				<input type="button" value="CANCEL"/> <input type="button" value="CONFIRM CREATE COURSE"/>		
Price * <input type="text"/>						
Description <div style="border: 1px solid #ccc; padding: 5px; height: 100px;"> <p style="margin: 0;">B I U G E I E E Normal</p> </div>						

### 3.4.2.2.2 <Manager> can Active/Disable course

- Function Trigger: <Manager> click button “Active/Disable”
- Function description: <Manager> can active or disable training course for the bird, active will make course appear in mobile, disabled will make it disappear
- Function details:
  - <Manager> access Management Web and click tab Academy
  - <Manager> select tab Training Course Management to view list course for bird training
  - <Manager> click button “Active/Disable”

- The system validates the resolving content
  - **Success:** System will show up a toast successfully “Active/Disable”
  - **Error:** System will show up toast error required manager to update details before Active Course
- Course active will appear a tick or disabled will make tick disappear
- Screen Layout:

Training Course Management						
Image	Title	Bird Species	Total Slot	Price (VND)	Status	
	eetestset12	Cockatoo	0	123,123,123	<input type="checkbox"/> ACTIVE	<button>VIEW DETAIL</button>
	teting	Finch	0	122,000	<input type="checkbox"/> ACTIVE	<button>VIEW DETAIL</button>
	Flight posing training	Cockatiel	4	100,000	<input checked="" type="checkbox"/> DISABLE	<button>DISABLE</button>
	Flight posing training	Cockatiel	2	100,000	<input checked="" type="checkbox"/> DISABLE	<button>DISABLE</button>
	First training for new bird	Cockatiel	10	100,000	<input checked="" type="checkbox"/> DISABLE	<button>DISABLE</button>
	Monk bird	Quaker Parrot (Monk Parakeet)	0	50,000	<input type="checkbox"/> ACTIVE	<button>VIEW DETAIL</button>
	Lovebird flight training	Lovebird	5	100,000	<input checked="" type="checkbox"/> DISABLE	<button>VIEW DETAIL</button>

#### 3.4.2.2.3 <Manager> can update Details about Course

- Function Trigger: <Manager> click button “View Detail”
- Function description: <Manager> can update details about course and add bird skill
- Function details:
  - <Manager> access Management Web and click tab Academy
  - <Manager> select tab Training Course Management to view list course for bird training
  - <Manager> click button “View Detail”
  - System will show up form to fill data update course
  - <Manager> Fill data in input fields to update data and choose bird skill corresponding to name of the course
  - <Manager> Click Confirm Update Course
  - The system validates the resolving content
    - **Success:** System will show up a toast successfully update Course and
    - **Error:** System will show up toast error required manager to fill Description
- Screen Layout:

**Bird Training Center Management**

**Training Course Detail**

**General information**

- Bird Species Name \* Cockatiel
- Title \* Flight posing training
- Price \* 100000
- Slot \* 4

**Pictures**



**Description**

Flying and posing training for bird

**CONFIRM UPDATE COURSE**

**Bird Skills in Course**

Picture	Name	Description	Train Slot	Action
	Flight training	Flight training for birds is an instinctive and vital process in which young avians learn to master the art of flying	2	<b>REMOVE</b>
	Pose training	Pose training for birds involves teaching them to assume specific positions on command	2	<b>REMOVE</b>

**Bird Skills out Course**

Picture	Name	Description	Action
	Speech training	Develops vocabulary and mimicking	<b>ADD</b>
	Behavior training	Modifies behaviors through positive reinforcement	<b>ADD</b>
	Agility training	Improves coordination through obstacle courses	<b>ADD</b>

### 3.4.2.2.4 <Manager> create new Training Policy in training course

- Function Trigger: <Manager> click button “Create New Training Policy”
- Function description: <Manager> can create new Training Policy for bird training course
- Function details:
  - <Manager> access Management Web and click tab Academy
  - <Manager> select tab “Price Policy Management” to view list price policy
  - <Manager> click button “Create New Training Policy”
  - System will show up form to fill data create new price policy
  - <Manager> Fill data in input fields to create new Training Course
  - <Manager> Click button “Confirm Create Policy”
  - The system validates the resolving content

- **Success:** System will show up a toast successfully create new price policy and navigate back to the list of price policy in “Price Policy Management” page
- **Error:** System will show warning each field need to fill data

- Screen Layout:

The screenshot shows the 'PRICE POLICY MANAGEMENT' section of the Bird Training Center Management system. It displays a table with four rows of training policies. Each row includes a 'Name' column, a 'Charge rate' column, and a 'Status' column. The 'Status' column contains buttons labeled 'ACTIVE', 'DISABLE', and 'VIEW DETAIL'. Above the table, two green success messages are displayed: 'Disable successfully!' and 'Active successfully!'. The left sidebar shows navigation links for Manager, Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy.

Name	Charge rate	Status
Bird's health not good	0.25	<input type="checkbox"/> ACTIVE <button>VIEW DETAIL</button>
Success Requested	1	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>
Center Requested	0	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>
Customer Requested	1	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>

The screenshot shows the 'Create Training Price Policy' dialog box. It has a yellow header bar with the title 'Create Training Price Policy'. Below the header, there is a section titled 'General information' containing two input fields: 'Name \*' and 'Charge rate \*'. Both fields have their respective values filled in. At the bottom right of the dialog are two buttons: 'CANCEL' and 'CONFIRM CREATE POLICY'.

### 3.4.2.2.5 <Manager> create update Training Policy in training course

- Function Trigger: <Manager> click button “View Detail”
- Function description: <Manager> can update details about price policy of training course
- Function details:
  - <Manager> access Management Web and click tab Academy
  - <Manager> select tab “Price Policy Management” to view list price policy
  - <Manager> click button “View Detail”

- System will show up form to fill data update price policy
- <Manager> Fill data in input fields to create new Training Course
- <Manager> Click button “Confirm Update Policy”
- The system validates the resolving content
  - **Success:** System will show up a toast successfully update new price policy
  - **Error:** System will show warning each field need to fill data
- Screen Layout:

The screenshots illustrate the 'PRICE POLICY MANAGEMENT' feature of the Bird Training Center Management system. The top screenshot shows a list of existing policies, while the bottom screenshot shows a detailed view and edit form for a specific policy.

**PRICE POLICY MANAGEMENT (Top Screenshot):**

Name	Charge rate	Status	Action
Bird's health not good	0.25	ACTIVE	VIEW DETAIL
Success Requested	1	DISABLE	VIEW DETAIL
Center Requested	0	DISABLE	VIEW DETAIL
Customer Requested	1	DISABLE	VIEW DETAIL

**Training Price Policy Detail (Bottom Screenshot):**

General information

Name *	Bird's health not good
Charge rate *	0.25

CANCEL CONFIRM UPDATE POLICY

### 3.4.2.2.6 <Manager> can show Bird List that customer register there's bird to our system

- Function Trigger: <Manager> click button “Show Bird List”
- Function description: <Manager> can view list bird that customer registered
- Function details:
  - <Manager> access Management Web and click tab Academy

- <Manager> select tab “Academy Management” to view list customer access to system
- <Manager> click button “Show Bird List” of one of customer
- System will show up list bird including customer name, name of bird, type of bird, picture, and button received skill if that bird has completed training course
- Screen Layout:

**Customers**

Customer Name	Email	Phone Number	Avatar	VIEW REQUEST	SHOW BIRD LIST
thg55	hoangthong432001@gmail.com	921888765		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
tg	lastTime@gmail.com	921786665		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Trinh Nam Anh	anhntse151264@fpt.edu.vn	346186122		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Louis Lane	customer4@gmail.com	906941770		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Nguyễn Thọ Thái Bảo	nguyenthothaibao@gmail.com	909290138		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Pham Nhat An	customer1@gmail.com	908941767		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>

**Customer Bird**

Bird Name	Customer Name	Bird Species Name	Bird Color	Bird Description	Bird Picture	
tao de test	thg55	Cockatoo				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
tao de test	thg55	Cockatoo				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
Âh	thg55	Quaker Parrot (Monk Parakeet)				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
the hell	thg55	Cockatiel				<a href="#">VIEW RECEIVED BIRD SKILLS</a>

[BACK](#)

### 3.4.2.2.7 <Manager> can check bird’s skill of each bird after it’s completed Training

- Function Trigger: <Manager> click button “VIEW RECEIVED BIRD SKILL”
- Function description: <Manager> can view list skill that bird acquired when training done
- Function details:

- <Manager> access Management Web and click tab Academy
- <Manager> select tab “Academy Management” to view list customer access to system
- <Manager> click button “Show Bird List” of one of customer
- <Manager> click button “VIEW RECEIVED BIRD SKILL” of one of customer that have bird skill completed
- The system validates the resolving content
  - **Complete Training:** System will show a list of skill that customer’s bird have completed training and acquired
  - **Not have Complete Training:** Nothing happen as the bird still in training
- Screen Layout:

**Customer Bird**

Bird Name	Customer Name	Bird Species Name	Bird Color	Bird Description	Bird Picture	
tao de test	thg55	Cockatoo				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
tao de test	thg55	Cockatoo				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
Åh	thg55	Quaker Parrot (Monk Parakeet)				<a href="#">VIEW RECEIVED BIRD SKILLS</a>
the hell	thg55	Cockatiel				<a href="#">VIEW RECEIVED BIRD SKILLS</a>

[BACK](#)

#### Bird Skill Received For Bird 2

Bird Name	Bird Skill Name	Bird Skill Description	Received Date	Bird Skill Picture
dol lame nek	Flight training	Flight training for birds is an instinctive and vital process in which young avians learn to master the art of flying	11/12/2023	
dol lame nek	Speech training	Develops vocabulary and mimicking	11/12/2023	

[BACK](#)

#### 3.4.2.3 <Staff> Manage Request Training

##### 3.4.2.3.1 <Staff> Can View Bird Training Request

- Function Trigger: <Staff> click “VIEW REQUEST” button next to Customer Avatar column

- Function description: <Staff> can view BirdName, registered CourseTitle, RegisteredDate, Training start/done date and status
- Function details:
  - <Staff> access Management Web and click tab Academy.
  - <Staff> select tab “Academy” at the side bar to view list of customers.
  - <Staff> click “VIEW REQUEST” button and scroll down to see their bird training request.
- Screen layout:

Customer Name	Email	Phone Number	Avatar	VIEW REQUEST	SHOW BIRD LIST
thg55	hoangthong432001@gmail.com	921888765		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
tg	lastTime@gmail.com	921786665		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Trinh Nam Anh	anhtnse151264@fpt.edu.vn	346186122		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Louis Lane	customer4@gmail.com	908941770		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Nguyễn Thủ Thái Bảo	nguyenthothaibao@gmail.com	909290138		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>
Pham Nhat An	customer1@gmail.com	908941767		<button>VIEW REQUEST</button>	<button>SHOW BIRD LIST</button>

### 3.4.2.3.2 <Staff> Can Confirm/Cancel/ViewDetails Bird Training Request

- Function Trigger: <Staff> click “VIEW DETAILS” button to view training request detail, <Staff> can “Confirm” or “Cancel” training request
- Function description:
  - <Staff> can view training request detail to assign/re-assign trainer, send notify mail to <Customer>.
  - <Staff> can “Confirm” training request in order to assign trainer to training.
  - <Staff> can “Cancel” training request in some specific situation.
- Function details:
  - <Staff> access Management Web and click tab Academy.
  - <Staff> select tab “Academy” at the side bar to view list of customers.
  - <Staff> click “VIEW DETAILS” button to view details of training request.
  - <Staff> click “Confirm” button to “Confirm” training request.
  - <Staff> click “Cancel” button “Cancel” training request.

- Screen layout:

**Registered TrainingCourse**

Bird Name	Owner Name	Course Title	Registered	Training start date	Training done date	Status	
Lanning birrd	Louis Lane	Flight posing training	21-12-2023	18-12-2023	19-12-2023	Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Imitating human speech	19-12-2023	10-01-2024		Checkin	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Greeting people	19-12-2023	27-12-2023		Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Cockatiel entertainment performance	19-12-2023	19-12-2023		Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Flight posing training	13-12-2023	11-01-2024		Cancel	<a href="#">CONFIRM</a>
Lanning birrd	Louis Lane	Flight posing training	13-12-2023	20-12-2023	19-12-2023	Complete	<a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	Flight posing training	10-12-2023	19-12-2023	19-12-2023	TrainingDone	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	Flight posing training	10-12-2023	20-12-2023		Checkin	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	First training for new bird	09-01-2024	15-01-2024		Confirmed	<a href="#">CHECK IN CANCEL</a> <a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	First training for new bird	09-01-2024			Registered	<a href="#">CONFIRM CANCEL</a>

#### 3.4.2.3.3 <Staff> Can Assign/Re-assign <Trainer> to Training Request

- Function Trigger: <Staff> click “ASSIGN TRAINER” button or “RE-ASSIGN TRAINER” button to view list of Trainers can train the skill, then click “ASSIGN” button next to trainer’s email to assign.
- Function description:
  - <Staff> assign or re-assign a new trainer to training
- Function details:
  - <Staff> access Management Web and click tab Academy.
  - <Staff> select tab “Academy” at the side bar to view list of customers.
  - <Staff> click “VIEW DETAILS” button to view details of training request.
  - <Staff> click “Confirm” button to view training request details.
  - <Staff> click “ASSIGN TRAINER” button or “RE-ASSIGN TRAINER” button to view list of Trainers can train the skill.
  - <Staff> click “ASSIGN” button to assign <Trainer>.

- Screen layout:

No	Bird Skill Name	Trainer Name	Trainer Email	Training Progression	Total Training Slot	Status		
0	Flight training	Bảo Nguyễn	consultingtrainer1@gmail.com	<div style="width: 0%;">[redacted]</div> 0% Complete	2	Assigned	<a href="#">RE-ASSIGN TRAINER</a>	<a href="#">VIEW TRAINING DETAILS</a>
1	Pose training			<div style="width: 0%;">[redacted]</div> 0% Complete	2	WaitingForAssign	<a href="#">ASSIGN TRAINER</a>	<a href="#">VIEW TRAINING DETAILS</a>

[CANCEL](#)

#### Trainer

Name	Email	
Rachel Riley	trainer3@gmail.com	<a href="#">ASSIGN</a>
Nicole Cowper	trainer2@gmail.com	<a href="#">ASSIGN</a>
Bảo Nguyễn	consultingtrainer1@gmail.com	<a href="#">ASSIGN</a>

#### 3.4.2.3.4 <Staff> Can CheckIn/CheckOut Bird Training Request

- Function Trigger: <Staff> click “Check In” or “Check Out” button on view training request.
- Function description:
  - <Staff> can “Check In” training request in order to receive bird then trainer can start training.
  - <Staff> can “Check Out” training request to complete a request.
- Function details:
  - <Staff> access Management Web and click tab Academy.
  - <Staff> select tab “Academy” at the side bar to view list of customers.
  - <Staff> click “Check In” button to “Check In” training request.
  - <Staff> click “Check Out” button “Check Out” training request.

- Screen layout:

**Receive Note**

Pictures

UPLOAD IMAGE(S)

No file is provided!

CANCEL CONFIRM CHECK IN

**Return Note**

Billing Details

Requested Id	54
Customer	Louis Lane
Course	Imitating human speech
Status	Checkin
Actual Price	50,000
Membership Rank	Standard
Discounted Price	50,000
Total Price	0

Choose Policy

Pictures

UPLOAD IMAGE(S)

No file is provided!

CANCEL CONFIRM CHECK OUT

**Return Note**

### 3.4.2.4 <Trainer> Manage Bird Training Academy

#### 3.4.2.4.1 <Trainer> can view training details

- Function Trigger: <Trainer> click button “VIEW TRAINING DETAIL”
- Function description: <Trainer> can view list status training, slot, and date
- Function details:
  - <Trainer> access Management Web and click tab Academy
  - <Trainer> select tab “Assigned Training Skill” to view list bird skill assigned to this trainer
  - <Trainer> click button “VIEW TRAINING DETAIL” of one of bird’ skill
  - <Trainer> view data inside view training detail page
- Screen Layout:

ASSIGNED TRAINING SKILL					
Bird Skill Name	Trainer Name	Training Progression	Total Training Slot	Status	
Flight training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass	<a href="#">VIEW TRAINING DETAILS</a>
Pose training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass	<a href="#">VIEW TRAINING DETAILS</a>
Target Training	Rachel Riley	<div style="width: 50%; background-color: #ccc;">50% Complete</div>	2	Cancel	
Flight training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Cancel	
Pose training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Cancel	
Speech training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	3	Pass	<a href="#">UPLOAD EVIDENCES</a> <a href="#">VIEW TRAINING DETAILS</a>
Pose training	Rachel Riley	<div style="width: 0%; background-color: #ccc;">0% Complete</div>	2	Assigned	<a href="#">VIEW TRAINING DETAILS</a>
Flight training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	2	Pass	<a href="#">VIEW TRAINING DETAILS</a>

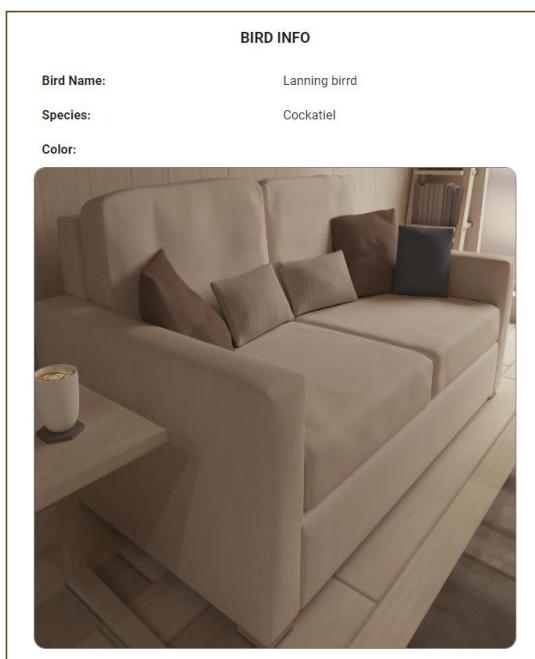
ASSIGNED TRAINING SKILL				
Slot	Date	Trainer Name	Trainer Email	Status
3	18/12/2023	Rachel Riley	trainer3@gmail.com	Done

### 3.4.2.4.2 <Trainer> Can view assigned training slot details then mark done

- Function Trigger: <Trainer> click on the slot in timetable then click “Mark Training Done” button to mark training slot done.
- Function description:
  - <Trainer> can view BirdName, BirdSpecies, BirdSkill Name and Training Description
  - <Trainer> mark training slot as done.
- Function details:
  - <Trainer> access Management Web and click tab Academy.
  - <Trainer> select tab “Timetable” at the side bar to view assigned work.
  - <Trainer> click on green slot to view training details.
  - <Trainer> click “Mark Training Done” button to mark training slot done.
- Screen layout:

## TRAINER SCHEDULE

	<a href="#">Today</a>	<a href="#">Back</a>	<a href="#">Next</a>	January 07 – 13					<a href="#">Agenda</a>	<a href="#">Day</a>	<a href="#">Week</a>	<a href="#">Month</a>
	07 Sun	08 Mon	09 Tue	10 Wed	11 Thu	12 Fri	13 Sat					
8:00 AM		Training bird at the center.				Training bird at the center.	Training bird at the center.					
9:00 AM				Training bird at the center.								
10:00 AM												
11:00 AM												
12:00 PM												
1:00 PM												
2:00 PM												
3:00 PM												
4:00 PM												
5:00 PM												



BIRD INFO	
Bird Name:	Lanning birrd
Species:	Cockatiel
Color:	
	

## 4. Non-Functional Requirements

### 4.1 External Interfaces

*[This section provides information to ensure that the system will communicate properly with users and with external hardware or software/system elements.]*

### 4.2 Quality Attributes

#### 4.2.1 Usability

- Manager, staff, trainer and administrator should need less than one week of training to be productive in using the system.

#### 4.2.2 Reliability

- Manager, staff, trainer and administrator should need training to be productive in using the system.

#### 4.2.3 Performance

- All common response time is under 5 seconds (except for uploading file relating features).

#### 4.2.4 Design Constraints

- The system uses English

- The system only supports Vietnamese dong as currency for any transaction.

#### **4.2.5 Security**

- Each role of user has a specific permission to access the server API (except for API endpoints designed for mobile).

## **5. Requirement Appendix**

### **5.1 Business Rules**

<b>ID</b>	<b>Rule Definition</b>
BR-01	Workshop class slot start time must be 1 day after registration day
BR-02	Workshop class slot modification must be done 3 days before the execution day
BR-03	Workshop class slot modification must be set before the execution of the following workshop class slot
BR-04	Trainer and Staff can check attendance for workshop class
BR-05	Attendance check form must be submitted after the slot start time and within 24 hours after the slot end time
BR-06	On check attendance late, automatically set all attendees as "Attended"
BR-07	Online Course must only be cancelled only if no customer is participating in the course
BR-08	Online Course must be fulfilling sections and lessons to be activated
BR-09	Manager must only log absent for trainer within month of execution day
BR-10	Manager can log absent for trainer in anytime within month
BR-11	Manager must not log absent in the last 3 days of execution month
BR-12	Manager/Staff must assign slot for trainer before at least 1 day
BR-13	A bird cannot register a training course multiple times if it is not cancelled
BR-14	Trainer has to mark training slot done within 1 day
BR-15	Customer must send bird before start training date
BR-16	Manager/Staff cannot cancel training request after check in
BR-17	Manager/Staff must confirm training request before check in
BR-18	Customer when send consultation ticket must fill "Address" if the selected service is In-home

BR-19	Staff must contact the customer (by email or phone number) before assign trainer, confirm or cancel ticket
BR-20	If the service was offline, staff must calculate the distance themselves (Ex: Using Google Map to Calculate the distance from the Bird Training Center to Customer's Address)
BR-21	Trainer must submit the evidence (image) for in-home consultation or record (of Google Meet) for online consultation after finishing the appointment.
BR-22	Staff, Manager and Trainer can only View Bill if the ticket is finished
BR-23	Customer cannot cancel the ticket if staff have confirmed it
BR-24	Trainer must show Customer the Bill after the appointment finish
BR-25	After the appointment finish, Customer can pay in cash immediately, OR pay in cash at the Bird Training Center if Customer does not want to pay immediately

## 5.2 Application Messages List

#	Message code	Message Type	Context	Content
1	MSG01	In red, under the text box	Wrong email or Password	Cannot Login
2	MSG02	In red, under the text box	Input-required fields are empty	The * field is required.
3	MSG03	Toast message	Updating asset(s) information successfully	Update asset(s) successfully.
4	MSG04	Toast message	Adding new asset successfully	Add assets successfully.
5	MSG05	Toast message	Confirming email of asset hand-over is sent successfully	A confirmation email has been sent to {email_address}.
6	MSG06	Toast message	Resetting asset information successfully	Return asset(s) successfully.
7	MSG07	Toast message	Deleting asset information successfully	Delete asset(s) successfully.
8	MSG08	In red, under the text box	Input value length > max length	Exceed max length of {max_length}.
9	MSG09	In line	Username or password is not correct when clicking sign-in	Incorrect user name or password. Please check again.
10	MSG10	In red, under the booking online form	Cannot Send Consulting Ticket because Customer doesn't fill the Address	*Please select a service.

11	MSG11	In red, under the booking online form	Customer doesn't select Trainer	*Please Choose A Trainer Or Other
12	MSG12	In red, under the booking online form	Customer doesn't select Time	*Please select date and time.
13	MSG13	In red, under the booking online form	Cannot Send Consulting Ticket because Customer doesn't fill the Address	Address must be fill if consulting at home
14	MSG14	In red, under the booking online form	Notify that Customer have successfully sent consulting ticket	You Have Successfully Booked
15	MSG15	Toast message	Staff doesn't assign Trainer when click "Assign"	Please choose a Trainer!
16	MSG16	Toast message	Notify that Staff have successfully approved a consulting ticket	Success Approve Ticket
17	MSG17	Toast message	Notify that Staff have successfully canceled a consulting ticket	Success Cancel Ticket
18	MSG18	Toast message	Notify that Trainer have successfully submitted evidence	Update Evidence Successfully
19	MSG19	Toast message	Notify that Trainer have successfully finished the consulting ticket	Success Finish Ticket
20	MSG20	Toast message error	Notify that workshop must have image	Please provide workshop image
21	MSG21	Toast message error	Notify that workshop must have description	Please provide workshop description
22	MSG22	Toast message error	Notify that workshop is created	Create successfully!
23	MSG23	Toast message error	Notify that workshop is modified	Modify successfully!
24	MSG24	Toast message error	Notify that the request has encountered an error	An error has occurred!
25	MSG25	Toast message error	Notify that current content has updated	Update successfully!
26	MSG26	Toast message error	Notify that workshop attendance is submitted	Submitted!

27	MSG27	Toast message error	Any error message response from server	{error?.response?.data?.message}
28	MSG28	alert	Sign-up Successfully	Successfully Registered
29	MSG29	In red, under the text box	Input-required fields are empty in Sign Up	- Invalid email address - Invalid phone number - Invalid username Usernames can only contain letters, numbers, and underscores, and must be between 3-20 characters
30	MSG30	In red, under the text box	Input wrong or empty email or password	WRONG EMAIL OR PASSWORD
31	MSG31	Toast message	Login successfully	Successfully Logged In
32	MSG32	Toast message	Cancel Bird Training Course	Cancel success!
33	MSG33	Toast message	Image input empty	Please provide receive image
34	MSG34	Toast message	Description Note is empty	Please provide receive note
35	MSG35	Toast message	Check-in bird that customer sends to Training Center	Check-in successfully!
36	MSG36	Toast message	Error Check-in bird	An error has occurred!
37	MSG37	Toast message	Assign Bird Successfully	Assign successfully!
38	MSG38	Toast message	Error Assign Bird	An error has occurred!
39	MSG39	Toast message	return image empty	Please provide return image
40	MSG40	Toast message	Image input empty	Please provide return note
41	MSG41	Toast message	Submit Return bird form Successfully	Submit return form success!
42	MSG42	Toast message	Error occurred when return bird	Error return form!
43	MSG43	Toast message	Error submits not provide video evidences	Please provide evidences video
44	MSG44	Toast message	Error submits not select status	Please select training status
45	MSG45	Toast message	Error upload	An error has occurred!
46	MSG46	Toast message	Upload successfully!	Upload successfully!

47	MSG47	Toast message	Training bird skill already have been mark done	Training skill have done!
48	MSG48	Toast message	Slot have been mark as done	Mark slot done!
49	MSG49	Toast message	Assign Trainer	Assign trainer success
50	MSG50	Toast message	Assign trainer failed due to slot busy	Trainer is not free to assign!
51	MSG51	Toast message	Send mail confirm successfully	Send notify email success!
52	MSG52	Toast message	Error occurred while sending mail	An error has occur!
53	MSG53	Toast message	Empty training course image	Please provide course image
54	MSG54	Toast message	Empty training course title	Please provide course title
55	MSG55	Toast message	Empty training course description	Please provide course description
56	MSG56	Toast message	Create training course successfully	Create successfully!
57	MSG57	Toast message	Error while create training course	An error has occured!
58	MSG58	Toast message	Empty field slot when training	Please provide train slot
59	MSG59	Toast message	Add skill successfully	Add skill success!
60	MSG60	Toast message	Remove skill successfully	Remove skill success!
61	MSG61	Toast message	Add skill already in training progress	Skill already training!
62	MSG62	Toast message	Training Course Active Success	Active successfully!
63	MSG63	Toast message	Training Course Active Error	An error has occured!
64	MSG64	Toast message	Training Course Disable Success	Disable successfully!
65	MSG65	Toast message	Training Course Disable Error	An error has occured!
66	MSG66	Toast message	Update Training Course Success	Update successfully!
67	MSG67	Toast message	Update type of consultant success	Update Consultant Type Success
68	MSG68	Toast message	Update Price of consultant success	Update Consultant Price Successfully

69	MSG69	Toast message	Update Distance Price of consultant success	Update Distance Price Successfully
70	MSG70	Toast message	Finish ticket consultant success	Success Finish Ticket
71	MSG71	Toast message	Approve ticket consultant success	Success Approve Ticket
72	MSG72	Toast message	Not select trainer	Please choose a Trainer!
73	MSG73	Toast message	Error assign trainer	Fail Assign Trainer!
74	MSG74	Toast message	Cancel consultant ticket success	Success Cancel Ticket
75	MSG75	Toast message	Approve consultant ticket success	Success Approve Ticket
76	MSG76	Toast message	Distance input empty	Please fill the distance
77	MSG77	Toast message	Approve ticket consultant failed	Fail Approve Ticket
78	MSG78	Toast message	Update Evidence consultant ticket success	Update Evidence Successfully
79	MSG79	Toast message	Fail Evidence consultant ticket success	Fail Update Evidence
80	MSG80	Toast message	Add skill to bird skill	Add successfully!
81	MSG81	Toast message	error Add skill to bird skill	An error has occur
82	MSG82	Toast message	Remove skill to bird skill	Remove successfully!
83	MSG83	Toast message	Error remove skill due to training in-progress	This species skill is currently in customer operation!
84	MSG84	Toast message	Bird species detail is too long	Detail is too long!
85	MSG85	Toast message	Delete trainer failed due to in-progress training skill assigned	Trainer has working experience in this skill!
86	MSG86	Toast message	Submit section in online course successfully	Submit Successfully
87	MSG87	Toast message	Fail to Submit section in online course	Fail To Submit
88	MSG88	Toast message	Switch Course Status success	Successfully changed status
89	MSG89	Toast message	Switch Course Status fail	Cannot change status
90	MSG90	Toast message	Delete Section in Online Course	Delete Successfully

91	MSG91	Toast message	Delete all lesson inside to proceed delete section	Please delete all lessons in this section
92	MSG92	Toast message	Error can not delete	Cannot Delete
93	MSG93	Toast message	Error update section	Fail To Update
94	MSG94	Toast message	Trainer is busy, can not assigned	Cannot Submit, Trainer is already busy on this day
95	MSG95	Toast message	Submit Trainer success	Submitted Successfully
96	MSG96	Toast message	Error not choose trainer to submit	Please Select Trainer
97	MSG97	Toast message	Create new user	Create user successfully
98	MSG98	Toast message	Error create new user	Cannot Create User:
99	MSG99	Toast message	update membership success	Success Update Membership
100	MSG100	Toast message	update membership Fail	Fail Update Membership
101	MSG101	Toast message	Change role of user	Successfully Change Role
102	MSG102	Toast message	Fail Change role of user	Cannot Change Role
103	MSG103	Toast message	Update slot time for trainer	Successfully Update Slot Time
104	MSG104	Toast message	input field time Empty	Please Insert Time
105	MSG105	Toast message	Fail Update slot time for trainer	Fail Update Slot Time
106	MSG106	Toast message	Error created class due to not assign trainer	Please assign trainer to created class!
107	MSG107	Toast message	empty picture in Modify Workshop	No picture provided!
108	MSG108	Toast message	Modify Workshop success	Modify successfully!
109	MSG109	Toast message	Modify Workshop success description	Please provide workshop description
110	MSG110	Toast message	wrong role authorise action	Manager is authorised to do this function
111	MSG111	Toast message	Change status success	Apply change!
112	MSG112	Toast message	Can not change workshop status	Cannot change this workshop's status
113	MSG113	Toast message	Attendance success	Submitted!
114	MSG114	Toast message	Error empty image in create workshop	Please provide workshop image

115	MSG115	Toast message	Error empty description in create workshop	Please provide workshop description
116	MSG116	Toast message	Error change avatar image for customer	Fail Change Avatar:

## IV. Software Design Description

### 1. System Design

#### 1.1 System Architecture

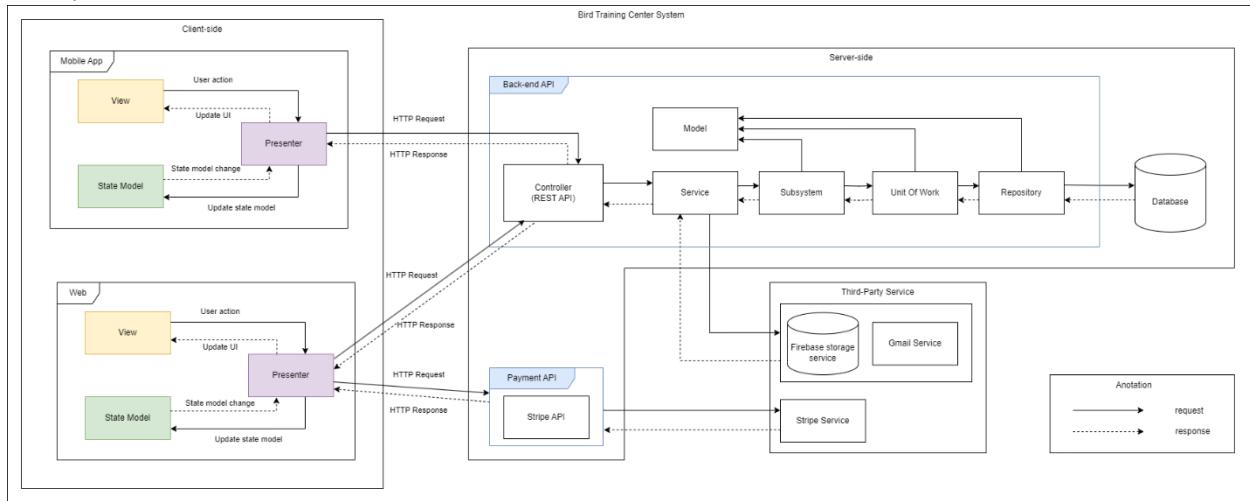


Figure 1.1.1. Overall Architecture Diagram

No.	Name	Description
01	View	Client view that user interacts on.
02	Presenter	Component that renders data to client, fetch data and send request to server-side API.
03	State Model	Component that keeps the states of data to re-render to client view.
04	Controller (REST API)	Handles the requests sent by clients and the responses of the system.
05	Service	Contains all services required with business logic implementation for Controller.
06	Subsystem	Contains all subsystems required for Service.
07	Model	Contains entities and business models for Service's and Subsystem's usage.
08	Unit Of Work	Applies Unit Of Work pattern for data accessing.
09	Repository	Implements repositories for database context.
10	Database	Database storage of the system.
11	Stripe API	Handles the payment requests sent by clients and responds data.
12	Stripe Service	Third party service from Stripe payment.
13	Firebase Storage Service	Third party service from Firebase. Is responsible for storing files such as images, videos, etc.
14	Gmail Service	Third party service from Google. Is responsible for sending mails.

## 1.2 Package Diagram

### 1.2.1 API Package Diagram

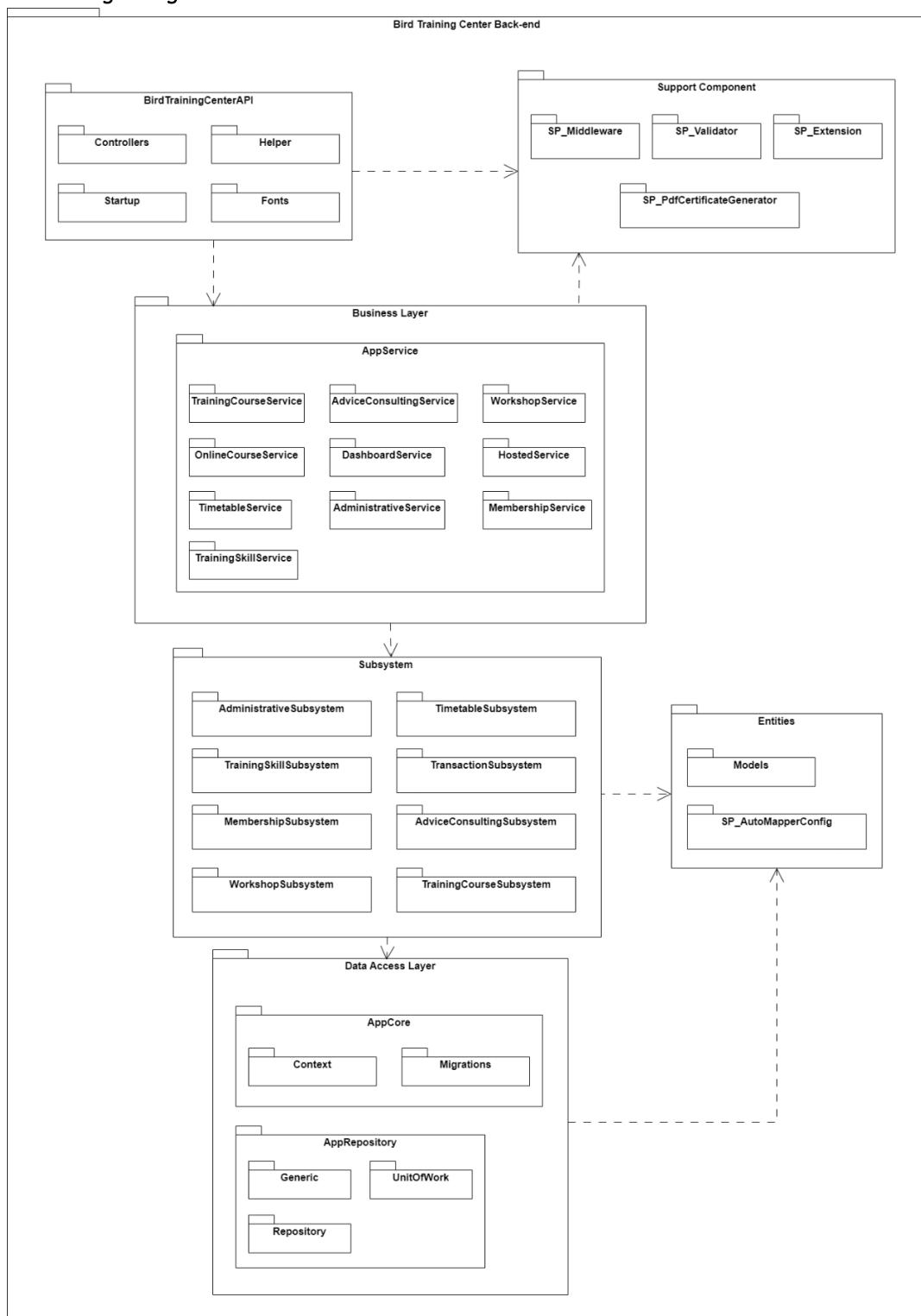


Figure 1.2.1.1 API Package Diagram

### **Package Descriptions**

No	Package	Description
01	BirdTrainingCenterAPI	It serves as the core API layer for the Bird Training Center application. It encapsulates all the API controllers responsible for handling incoming requests and orchestrating interactions with the underlying services. Additionally, this package imports services from other related packages to fulfill its functionality.
02	Controllers	It encapsulates a set of API controllers that define the external endpoints and handle incoming requests for the Bird Training Center application. Each controller is designed to manage a specific aspect of the application's functionality, contributing to the overall API interface.
03	Helper	It provides utility functions and helper classes that augment the functionality of the API controllers for handling incoming requests.
04	Startup	It provides the necessary configurations and initializing essential services. It serves as the entry point for configuring dependency injection, setting up the database context, and orchestrating the startup sequence.
05	Fonts	It manages and provides fonts essential for the PDF certificate generation process within the Bird Training Center application.
06	SP_Middleware	It is dedicated to housing middleware components that provide utility functions for enhanced handling of incoming requests and outgoing responses in the Bird Training Center API.
07	SP_Validator	It offers a set of validation tools and mechanisms to ensure that incoming data models conform to predefined standards, promoting data integrity and API reliability.
08	SP_PdfCertificateGenerator	Its primary function is to facilitate the seamless generation of PDF certificates.
09	SP_Extension	It offers additional utility methods, extensions and enhancements that contribute to the overall efficiency and versatility of the application services in AppService package.
10	AppService	It acts as the central hub for organizing and managing all application services within the entire project.
11	TrainingCourseService	It encapsulates all features related to training course management and customer interactions. This service plays a central role in orchestrating the lifecycle of training courses, catering to both administrative tasks and actions initiated by customers.
12	AdviceConsultingService	It encapsulates all features related to manage consultation sessions, facilitate administrative tasks and enable customer interactions with the consulting services provided by the Center.

13	Workshop Service	It encapsulates all features related to manage the lifecycle of workshops, handle classes, and facilitate customer interactions with the educational classes provided by the Center.
14	OnlineCourseService	It encapsulates all features related to manage the lifecycle of online courses, track customer study progress, and handle course registrations.
15	DashboardService	It encapsulates all features related to the dashboard which is used for aggregating and presenting statistical information, offering insights into key metrics.
16	HostedService	It encapsulates all features for background service that runs continuously in the application.
17	TimetableService	It encapsulates all features for supporting another service, providing the necessary functionalities to streamline the management of trainer schedules.
18	AdministrativeService	It provides functionality for administering critical functionalities, including user management and account management.
19	MembershipService	It provides functionality for managing customer memberships.
20	TrainingSkillService	It provides functionality for managing trainer skills as for adaptability in training bird progression.
21	AdministrativeSubsystem	It encapsulates the actual and specified features and functionalities mainly to AdministrativeService and other service usages.
22	TimetableSubsystem	It encapsulates the specified features and functionalities mainly to TimetableService and other service usages.
23	TrainingSkillSubsystem	It encapsulates the specified features and functionalities mainly to TrainingSkillService and other service usages.
24	TransactionSubsystem	It encapsulates the specified features and functionalities mainly to transaction feature provided in other services.
25	MembershipSubsystem	It encapsulates the specified features and functionalities mainly to membership management service.
26	AdviceConsultingSubsystem	It encapsulates the specified features and functionalities to advice consultation service.
27	WorkshopSubsystem	It encapsulates the specified features and functionalities to workshop and workshop class service.
28	TrainingCourseSubsystem	It encapsulates the specified features and functionalities to bird training course service.
29	Models	It encapsulates the context entities and business models.
30	SP_AutoMapperConfig	It encapsulates the mapping feature of all the models and entities in Models package

31	AppCore	It encapsulates the database context and migrations.
32	Context	It encapsulates the database context in AppCore package.
33	Migrations	It encapsulates the migrations of database in AppCore package.
34	AppRepository	It encapsulates the repositories of all the entities in database context as Unit Of Work pattern.
35	Repository	It encapsulates the repositories in database context.
36	Generic	It encapsulates the generic pattern of repositories in repository.
37	UnitOfWork	It encapsulates the unit of work pattern of repositories in database context.

### 1.2.2 Web Application Package Diagram

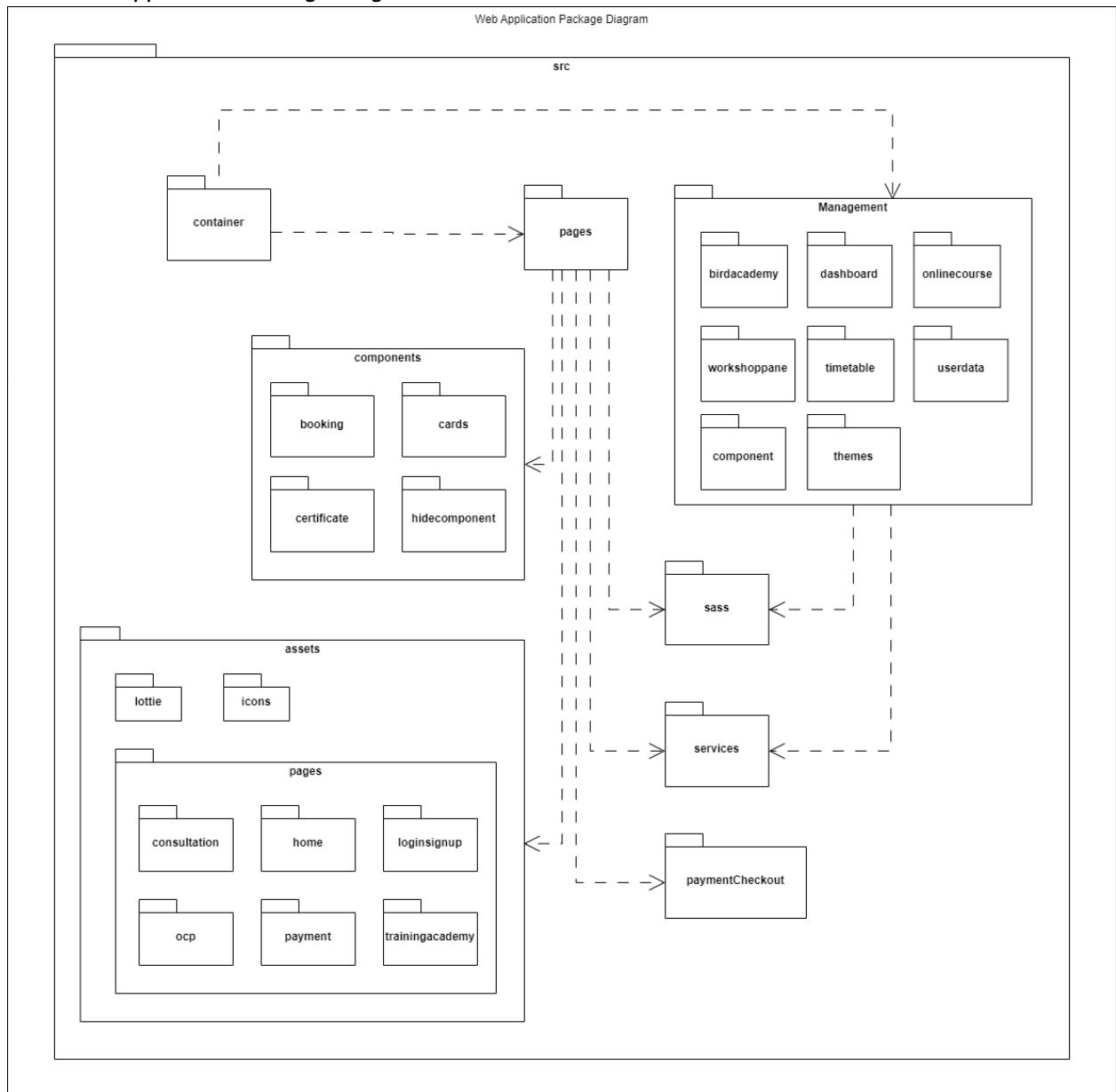


Figure 1.2.2.1 Web Application Package Diagram

**Package Description**

No	Package	Description
01	src	Main package of the application, any components must be placed within this package.
02	container	It contains major components acting as main activities for the application such as layout, navigation, footer.
03	pages	It contains various components that supports customer view and provides general view for important features such as authentication, payment, etc.
04	components	It contains components supporting those within pages package as external usage.
05	booking	It contains the components for booking advice consultation service.
06	cards	It contains the components supporting card view for workshop and online course as utilities.
07	certificate	It contains the components supporting customer's certificates view for workshop and online course as utilities.
08	hidecomponent	It contains the component supporting hidden feature for its children components.
09	assets	It contains the various assets for customer view.
10	lottie	It contains the loading component for user experience.
11	icons	It contains various client-side icons for user experience.
12	pages	It contains multiple packages as external extension for main features.
13	consultation	It contains various assets for consultation service.
14	home	It contains various assets for home page.
15	loginsignup	It contains various assets for login and signup pages.
16	ocp	It contains assets for general usage.
17	payment	It contains assets for payment page.
18	trainingacademy	It contains assets for training course landing page.
19	Management	It contains all the related components for management features.
20	birdacademy	It contains components for training course management.
21	dashboard	It contains components for viewing statistics, overview and general management features such as bird skills, bird species, trainer skills, etc.
22	onlinecourse	It contains components for online course management.
23	workshoppane	It contains components for workshop management and workshop class management.

24	timetable	It contains components for timetable features for trainers.
25	userdata	It contains components for user management for administrator.
26	component	It contains components as utilities for other packages.
27	themes	It contains provided themes for other packages.
28	sass	It contains scss files for beautify the view within the application.
29	services	It contains multiple service files for fetching data, handling requests and responses within the application.
30	paymentCheckout	It contains various components used for Stripe payment.

### 1.2.3 Mobile Application Package Diagram

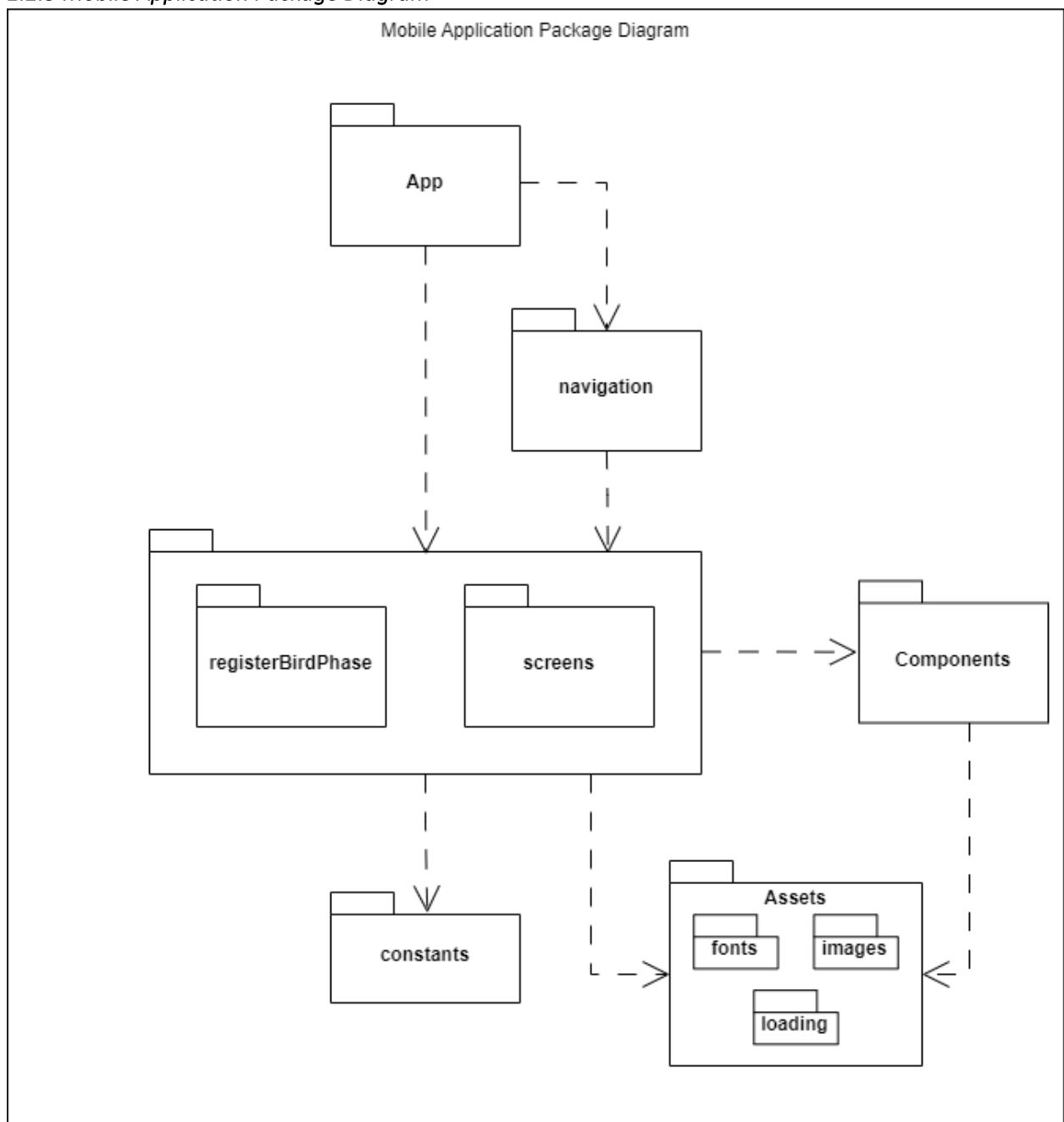


Figure 1.2.3.1 Mobile Application Package Diagram

## *Package Description*

No	Package	Description
01	App	It encapsulates the initialized component for the application.
02	navigation	It provides the navigating feature for the application.
03	registerBirdPhase	It encapsulates the components for registering the bird as its complexity.
04	screens	It encapsulates the components for various screens.
05	Components	It encapsulates the components for external usage such as loading screen, etc.
06	constants	It encapsulates the constants for the application.
07	Assets	It encapsulates the assets for the application.
08	fonts	It encapsulates the provided fonts in the application.
09	images	It encapsulates the provided images used in the application.
10	loading	It encapsulates the related add-in for loading components.

## 2. Database Design

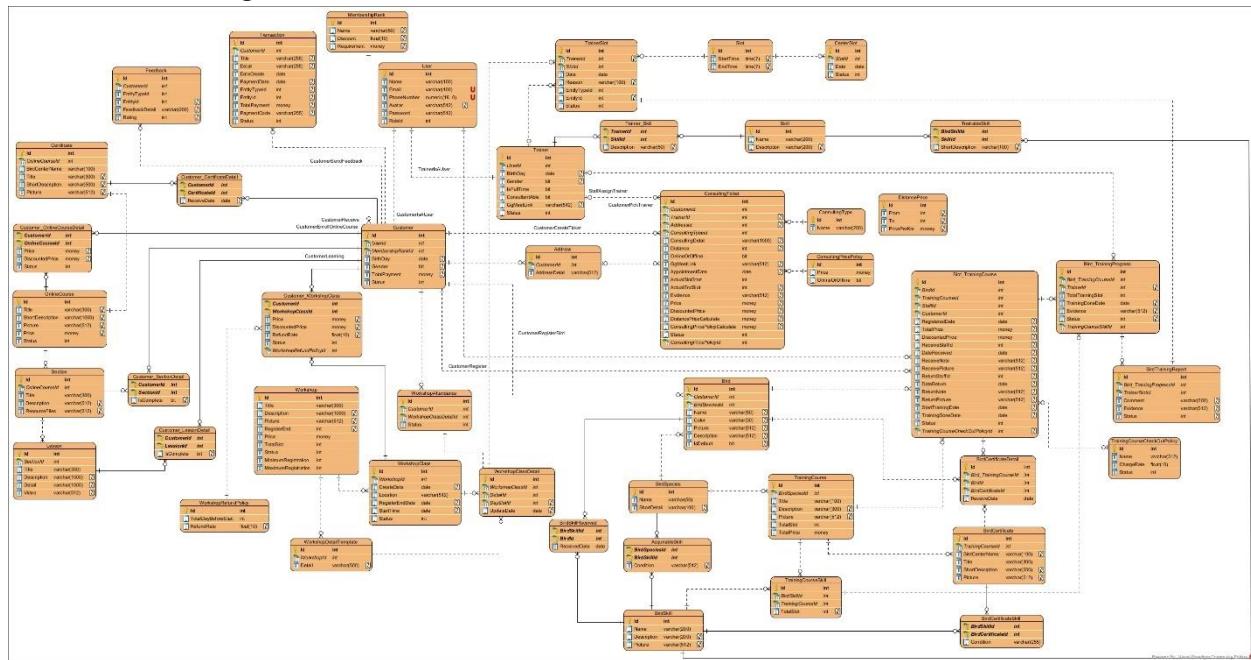


Figure 2.1 Database Relationship Diagram

## 2.1 Table Descriptions

No	Table	Description
01	User	<i>Describe user account in the system</i>
02	Customer	<i>Describe customer identity and additional information inherited from User.</i>
03	Address	<i>Store customer's addresses.</i>
04	MembershipRank	<i>Identify customer's membership for discount charge.</i>
05	Transaction	<i>Store customer's transactions.</i>
06	Trainer	<i>Describe trainer identity and additional information inherited from User.</i>
07	TrainerSlot	<i>Define slot that trainer is occupied with task or logged absent.</i>

08	<i>Slot</i>	Describe a period of time as working hours in slot.
09	<i>CenterSlot</i>	Define slot that center is absent from working.
10	<i>Trainer_Skill</i>	Describe trainer skills
12	<i>Skill</i>	Describe skills that trainer can be acquired.
13	<i>TrainableSkill</i>	Describe bird skills that can be trained by trainer skill.
14	<i>BirdSkill</i>	Describe bird skills that can be trained.
15	<i>AcquirableSkill</i>	Describe bird skills that must be trained within adaptable species.
16	<i>BirdSpecies</i>	Describe bird species that the system supports in center.
17	<i>Bird</i>	Describe customer's bird identity in the system.
18	<i>TrainingCourse</i>	Describe training course in the system.
19	<i>TrainingCourseSkill</i>	Describe bird skills and taken slots that the training course trains.
20	<i>Bird_TrainingCourse</i>	Define registered bird with a training course.
21	<i>BirdCertificateDetail</i>	Define the certificate of registered bird that completes the training course.
22	<i>BirdCertificate</i>	Describe the training course's certificate template.
23	<i>BirdCertificateSkill</i>	Describe the bird skills that the certificate stands for.
24	<i>Bird_TrainingProgress</i>	Define the progress of the bird in training.
25	<i>BirdTrainingReport</i>	Define the result of the training progress.
26	<i>TrainingCourseCheckOutPolicy</i>	Define the policy in check-out training bird.
27	<i>ConsultingTicket</i>	Define the consultation information of customer request.
28	<i>ConsultingType</i>	Describe the type of the consultation chosen by the customer.
29	<i>ConsultingPricePolicy</i>	Describe the type of policy which is applied to consulting ticket.
30	<i>DistancePrice</i>	Define the price policy of distance traveled from the center to customer location.
31	<i>Workshop</i>	Describe the workshop's information in the system.
32	<i>WorkshopDetailTemplate</i>	Define the template detail description of each slot in workshop.
33	<i>WorkshopClass</i>	Define the classes within the workshop.
34	<i>WorkshopClassDetail</i>	Define the slot within the class including detail template description.
35	<i>WorkshopRefundPolicy</i>	Define the policy of workshop refund requested by customer.
36	<i>Customer_WorkshopClass</i>	Define the customer registration of the workshop.
37	<i>WorkshopAttendance</i>	Define the attendance status of customer with the workshop class detail.
38	<i>OnlineCourse</i>	Describe the online course in the system.
39	<i>Section</i>	Describe the section in the online course.
40	<i>Lesson</i>	Describe the lesson in the section.
41	<i>Certificate</i>	Define the base template of the online course's certificate.
42	<i>Customer_CertificateDetail</i>	Define the customer's certificate based on the online course's certificate template.
43	<i>Customer_OnlineCourseDetail</i>	Define the customer registration of the online course.
44	<i>Customer_SectionDetail</i>	Define the customer's tracking of the study progress within the section.
45	<i>Customer_LessonDetail</i>	Define the customer's tracking of the study progress within the lesson.
46	<i>Feedback</i>	Define the customer's feedback of the services.

## 2.2 Attribute Data Dictionary

Entity Name	Attributes	Description	Domain	Null
User	<i>Id {PK}</i>	Unique identifier of user	<i>int</i>	No
	<i>Name</i>	User's name	<i>varchar(100)</i>	Yes
	<i>Email</i>	Unique, user's email	<i>varchar(100)</i>	No
	<i>PhoneNumber</i>	Unique, user's phone number	<i>numeric(18,0)</i>	No
	<i>Avatar</i>	User's avatar	<i>varchar(512)</i>	Yes

	<i>Password</i>	<i>User's password</i>	<i>varchar(512)</i>	Yes
	<i>RoleId</i>	<i>User's role, defined as enum:</i> + Customer = 0 + Trainer = 1 + Staff = 2 + Manager = 3 + Administrator = 4	<i>int</i>	No
<i>Customer</i>	<i>Id {PK}</i>	<i>Unique identifier of customer</i>	<i>int</i>	No
	<i>UserId {FK}</i>	<i>Foreign key, user identifier</i>	<i>int</i>	No
	<i>MembershipRankId {FK}</i>	<i>Foreign key, membership identifier</i>	<i>int</i>	No
	<i>BirthDay</i>	<i>Customer's birthdate</i>	<i>date</i>	Yes
	<i>Gender</i>	<i>Customer's gender</i>	<i>bit</i>	Yes
	<i>TotalPayment</i>	<i>Customer's total payment from all the transactions</i>	<i>money</i>	Yes
	<i>Status</i>	<i>Customer's account status, defined as enum:</i> + Active = 0 + Inactive = 1 + Charged = 2	<i>int</i>	No
<i>Address</i>	<i>Id {PK}</i>	<i>Unique identifier of user's address</i>	<i>int</i>	No
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	No
	<i>AddressDetail</i>	<i>Customer's address</i>	<i>varchar(512)</i>	No
<i>MembershipRank</i>	<i>Id {PK}</i>	<i>Unique identifier of membership rank</i>	<i>int</i>	No
	<i>Name</i>	<i>Membership rank's name</i>	<i>varchar(50)</i>	No
	<i>Discount</i>	<i>Membership's discount</i>	<i>float(10)</i>	Yes
	<i>Requirement</i>	<i>Membership's requirement to apply this rank</i>	<i>money</i>	Yes
<i>Transaction</i>	<i>Id {PK}</i>	<i>Unique identifier of transaction</i>	<i>int</i>	No
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	No
	<i>Title</i>	<i>Transaction's title</i>	<i>varchar(255)</i>	Yes
	<i>Detail</i>	<i>Transaction's detail</i>	<i>varchar(255)</i>	Yes
	<i>DateCreate</i>	<i>Transaction's created date</i>	<i>date</i>	No
	<i>PaymentDate</i>	<i>Transaction's date of payment</i>	<i>date</i>	Yes
	<i>EntityTypeld</i>	<i>Identifier of payment target, defined as enum:</i>	<i>int</i>	Yes

		+ Others = 1 + AdviceConsulting = 2 + WorkshopClass = 3 + OnlineCourse = 4 + TrainingCourse = 5 + Topup = 6 + AbsentRequest = 7		
	EntityId	<i>Identifier of target entity</i>	<i>Int</i>	Yes
	TotalPayment	<i>Total cost amount of transaction</i>	<i>money</i>	Yes
	PaymentCode	<i>Transaction's payment code</i>	<i>varchar(255)</i>	Yes
	Status	<i>Transaction's status, defined as enum: + Unpaid = 0 + Paid = 1</i>	<i>int</i>	No
Trainer	<i>Id {PK}</i>	<i>Unique identifier of trainer</i>	<i>int</i>	No
	<i>UserId {FK}</i>	<i>Foreign key, user identifier</i>	<i>int</i>	No
	<i>BirthDay</i>	<i>Trainer's birthdate</i>	<i>date</i>	Yes
	<i>Gender</i>	<i>Trainer's gender</i>	<i>bit</i>	Yes
	<i>IsFullTime</i>	<i>Trainer's status of being available for fulltime working</i>	<i>bit</i>	No
	<i>ConsultantAble</i>	<i>Trainer's status of being available for consultation service</i>	<i>bit</i>	No
	<i>GgMeetLink</i>	<i>Trainer's google meet link for consultation online service</i>	<i>varchar(512)</i>	Yes
	<i>Status</i>	<i>Trainer's account status, defined by enum: + Disabled = 0 + Working = 1</i>	<i>int</i>	No
TrainerSlot	<i>Id {PK}</i>	<i>Unique identifier of trainer slot</i>	<i>int</i>	No
	<i>TrainerId {FK}</i>	<i>Foreign key, trainer identifier</i>	<i>int</i>	Yes
	<i>SlotId {FK}</i>	<i>Foreign key, slot identifier</i>	<i>int</i>	No
	<i>Date</i>	<i>Trainer slot's date</i>	<i>date</i>	No
	<i>Reason</i>	<i>Trainer slot's reason</i>	<i>varchar(100)</i>	Yes
	<i>EntityType</i>	<i>Identifier of trainer slot target, defined as enum: + Others = 1 + AdviceConsulting = 2 + WorkshopClass = 3</i>	<i>int</i>	No

		+ OnlineCourse = 4 + TrainingCourse = 5 + Topup = 6 + AbsentRequest = 7		
	EntityId	Identifier of target entity	int	Yes
	Status	Trainer slot's status, defined as enum: + Disabled = 0 + Enabled = 1	int	No
Slot	<i>Id {PK}</i>	Unique identifier of slot	int	No
	<i>StartTime</i>	Slot's defined start time	time(7)	Yes
	<i>EndTime</i>	Slot's defined end time	time(7)	Yes
CenterSlot	<i>Id {PK}</i>	Unique identifier for center off slot	int	No
	<i>SlotId {FK}</i>	Foreign key, slot identifier	int	No
	<i>Date</i>	Center off slot's date	date	No
	<i>Status</i>	Center's slot status, defined as enum: + Disabled = 0 + Enabled = 1	int	No
Trainer_Skill	<i>TrainerId {PK, FK}</i>	Foreign key and composite key, trainer's identifier	int	No
	<i>SkillId {PK, FK}</i>	Foreign key and composite key, skill identifier	int	No
	<i>Description</i>	Trainer skill's description	varchar(50)	Yes
Skill	<i>Id {PK}</i>	Unique identifier for skill	int	No
	<i>Name</i>	Skill's name	varchar(200)	No
	<i>Description</i>	Skill's description	varchar(200)	Yes
TrainableSkill	<i>BirdSkillId {PK, FK}</i>	Foreign key and composite key, bird skill identifier	int	No
	<i>SkillId {PK, FK}</i>	Foreign key and composite key, skill identifier	int	No
	<i>ShortDescription</i>	Trainable skill's description	varchar(100)	Yes
BirdSkill	<i>Id {PK}</i>	Unique identifier for bird skill	int	No
	<i>Name</i>	Bird skill's name	varchar(200)	No
	<i>Description</i>	Bird skill's description	varchar(200)	Yes
	<i>Picture</i>	Bird skill's picture	varchar(512)	Yes
AcquirableSkill	<i>BirdSpeciesId {PK, FK}</i>	Foreign key and composite key, bird species identifier	int	No

	<i>BirdSkillId {PK, FK}</i>	<i>Foreign key and composite key, bird skill identifier</i>	<i>int</i>	<i>No</i>
	<i>Condition</i>	<i>Acquirable skill's condition between bird species and bird skill</i>	<i>varchar(512)</i>	<i>Yes</i>
<i>BirdSpecies</i>	<i>Id {PK}</i>	<i>Unique identifier for bird species</i>	<i>int</i>	<i>No</i>
	<i>Name</i>	<i>Bird species's name</i>	<i>varchar(50)</i>	<i>No</i>
	<i>ShortDetail</i>	<i>Bird species's short detail</i>	<i>varchar(100)</i>	<i>Yes</i>
<i>Bird</i>	<i>Id {PK}</i>	<i>Unique identifier for customer's bird</i>	<i>int</i>	<i>No</i>
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>BirdSpeciesId {FK}</i>	<i>Foreign key, bird species identifier</i>	<i>int</i>	<i>No</i>
	<i>Name</i>	<i>Bird's name</i>	<i>varchar(50)</i>	<i>Yes</i>
	<i>Color</i>	<i>Bird's color</i>	<i>varchar(30)</i>	<i>Yes</i>
	<i>Picture</i>	<i>Bird's picture</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>Description</i>	<i>Bird's description</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>IsDefault</i>	<i>Status for first interactive bird appears on customer home page</i>	<i>bit</i>	<i>Yes</i>
<i>TrainingCourse</i>	<i>Id {PK}</i>	<i>Unique identifier for training course</i>	<i>int</i>	<i>No</i>
	<i>BirdSpeciesId {FK}</i>	<i>Foreign key, bird species identifier</i>	<i>int</i>	<i>No</i>
	<i>Title</i>	<i>Training course's title</i>	<i>varchar(100)</i>	<i>No</i>
	<i>Description</i>	<i>Training course's description</i>	<i>varchar(300)</i>	<i>Yes</i>
	<i>Picture</i>	<i>Training course's landing picture</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>TotalSlot</i>	<i>Training course's total slot of training. Calculated by sum of total slot in each training skill</i>	<i>int</i>	<i>No</i>
	<i>TotalPrice</i>	<i>Training course's total price</i>	<i>money</i>	<i>No</i>
<i>TrainingCourseSkill</i>	<i>Id {PK}</i>	<i>Unique identifier for training course skill</i>	<i>int</i>	<i>No</i>
	<i>BirdSkillId {FK}</i>	<i>Foreign key, bird skill identifier</i>	<i>int</i>	<i>No</i>
	<i>TrainingCourseId {FK}</i>	<i>Foreign key, training course identifier</i>	<i>int</i>	<i>No</i>
	<i>TotalSlot</i>	<i>Training course skill's total slot of training.</i>	<i>int</i>	<i>Yes</i>

<i>Bird_TrainingCourse</i>	<i>Id {PK}</i>	<i>Unique identifier for registered bird in training course</i>	<i>int</i>	<i>No</i>
	<i>BirdId {FK}</i>	<i>Foreign key, registered bird's identifier</i>	<i>int</i>	<i>No</i>
	<i>TrainingCourseld {FK}</i>	<i>Foreign key, training course identifier</i>	<i>int</i>	<i>No</i>
	<i>StaffId {FK}</i>	<i>Foreign key with Id in table User, staff identifier, to log the staff confirming the registration</i>	<i>int</i>	<i>No</i>
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>RegisteredDate</i>	<i>Customer's bird's date of registering training course</i>	<i>date</i>	<i>Yes</i>
	<i>TotalPrice</i>	<i>Total price of the training</i>	<i>money</i>	<i>Yes</i>
	<i>DiscountedPrice</i>	<i>Discounted price of the training</i>	<i>money</i>	<i>Yes</i>
	<i>ReceiveStaffId {FK}</i>	<i>Foreign key with Id in table User, staff identifier, to log the staff receiving the customer's bird</i>	<i>int</i>	<i>Yes</i>
	<i>DateReceived</i>	<i>Date of receiving customer's bird at the center</i>	<i>date</i>	<i>Yes</i>
	<i>ReceiveNote</i>	<i>Note of receiving customer's bird at the center</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>ReceivePicture</i>	<i>Picture of receiving customer's bird at the center</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>ReturnStaffId {FK}</i>	<i>Foreign key with Id in table User, staff identifier, to log the staff returning the customer's bird</i>	<i>int</i>	<i>Yes</i>
	<i>DateReturn</i>	<i>Date of returning customer's bird at the center</i>	<i>date</i>	<i>Yes</i>
	<i>ReturnNote</i>	<i>Note of returning customer's bird at the center</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>ReturnPicture</i>	<i>Picture of returning customer's bird at the center</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>StartTrainingDate</i>	<i>Date of training first happening</i>	<i>date</i>	<i>Yes</i>

	<i>TrainingDoneDate</i>	<i>Date of training done</i>	<i>date</i>	Yes
	<i>Status</i>	<i>Status of training, defined as enum: + Disable = 0 + Active = 1 + Modifying = 2</i>	<i>int</i>	No
	<i>TrainingCourseCheckOutPolicyId {FK}</i>	<i>Foreign key, training course check out policy identifier</i>	<i>int</i>	Yes
<i>BirdCertificateDetail</i>	<i>Id {PK}</i>	<i>Unique identifier for customer's bird certificate detail</i>	<i>int</i>	No
	<i>Bird_TrainingCourseId {FK}</i>	<i>Foreign key, registered bird in training course identifier</i>	<i>int</i>	No
	<i>BirdId {FK}</i>	<i>Foreign key, registered bird identifier</i>	<i>int</i>	No
	<i>BirdCertificateId {FK}</i>	<i>Foreign key, training course certificate identifier</i>	<i>int</i>	No
	<i>ReceiveDate</i>	<i>Received date of registered bird's certificate</i>	<i>date</i>	No
<i>BirdCertificate</i>	<i>Id {PK}</i>	<i>Unique identifier for training course certificate</i>	<i>int</i>	No
	<i>TrainingCourseId {FK}</i>	<i>Foreign key, training course identifier</i>	<i>int</i>	No
	<i>BirdCenterName</i>	<i>Center's name for Bird Certificate</i>	<i>varchar(100)</i>	Yes
	<i>Title</i>	<i>Certificate's title</i>	<i>varchar(300)</i>	No
	<i>ShortDescription</i>	<i>Certificate's short description</i>	<i>varchar(500)</i>	Yes
	<i>Picture</i>	<i>Certificate's picture</i>	<i>varchar(512)</i>	Yes
<i>BirdCertificateSkill</i>	<i>BirdSkillId {PK, FK}</i>	<i>Foreign key and composite key, bird skill identifier</i>	<i>int</i>	No
	<i>BirdCertificateId {PK, FK}</i>	<i>Foreign key and composite key, bird training course certificate identifier</i>	<i>int</i>	No
	<i>Condition</i>	<i>Condition description for bird skill and certificate</i>	<i>varchar(255)</i>	No
<i>Bird_TrainingProgress</i>	<i>Id {PK}</i>	<i>Unique identifier for training progress</i>	<i>int</i>	No
	<i>Bird_TrainingCourseId {FK}</i>	<i>Foreign key, bird training course identifier</i>	<i>int</i>	No
	<i>TrainerId</i>	<i>Foreign key, trainer identifier</i>	<i>int</i>	Yes

	<i>TotalTrainingSlot</i>	<i>Total slot to spent of the progress</i>	<i>int</i>	No
	<i>TrainingDoneDate</i>	<i>Date of completing the progress</i>	<i>date</i>	Yes
	<i>Evidence</i>	<i>Provided evidence as link to file of the progress</i>	<i>varchar(512)</i>	Yes
	<i>Status</i>	<i>Progress's status, defined as enum: + WaitingForTimetable = 0 + WaitingForAssign = 1 + Assigned = 2 + Training = 3 + NotPass = 4 + Pass = 5 + Cancel = 6</i>	<i>int</i>	Yes
	<i>TrainingCourseSkillId</i>	<i>Foreign key, training course skill identifier</i>	<i>int</i>	No
<i>BirdTrainingReport</i>	<i>Id {PK}</i>	<i>Unique identifier for training report</i>	<i>int</i>	No
	<i>Bird_TrainingProgressId {FK}</i>	<i>Foreign key, training progress identifier</i>	<i>int</i>	No
	<i>TrainerSlotId {FK}</i>	<i>Foreign key, trainer slot identifier</i>	<i>int</i>	No
	<i>Comment</i>	<i>Trainer's comment for reporting</i>	<i>varchar(200)</i>	Yes
	<i>Evidence</i>	<i>Trainer's evidence included in report</i>	<i>varchar(512)</i>	Yes
	<i>Status</i>	<i>Report's status, defined as enum: + NotYet = 0 + Done = 1</i>	<i>int</i>	Yes
<i>TrainingCourseCheckOutPolicy</i>	<i>Id {PK}</i>	<i>Unique identifier of training course check out policy</i>	<i>int</i>	No
	<i>Name</i>	<i>Policy's name</i>	<i>varchar(512)</i>	No
	<i>ChargeRate</i>	<i>Policy's defined charged rate</i>	<i>float(10)</i>	No
	<i>Status</i>	<i>Policy's status, defined as enum: + Disabled = 0 + Active = 1</i>	<i>int</i>	No
<i>ConsultingTicket</i>	<i>Id {PK}</i>	<i>Unique identifier for consulting ticket</i>	<i>int</i>	No
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	No
	<i>TrainerId {FK}</i>	<i>Foreign key, trainer identifier</i>	<i>int</i>	Yes
	<i>AddressId {FK}</i>	<i>Foreign key, customer's address identifier</i>	<i>int</i>	Yes

	<i>ConsultingTypeId {FK}</i>	<i>Foreign key, consulting type identifier</i>	<i>int</i>	<i>No</i>
	<i>ConsultingDetail</i>	<i>Consulting ticket's detail defined by customer</i>	<i>varchar(1000)</i>	<i>Yes</i>
	<i>Distance</i>	<i>Distance traveled from customer's address (Offline Consultation)</i>	<i>int</i>	<i>Yes</i>
	<i>OnlineOrOffline</i>	<i>Defining online or offline ticket. Default value: false is Offline</i>	<i>bit</i>	<i>No</i>
	<i>GgMeetLink</i>	<i>Defining google meet link attached in trainer's profile (Online Consultation)</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>AppointmentDate</i>	<i>Desired date for the ticket selected by customer</i>	<i>date</i>	<i>Yes</i>
	<i>ActualSlotStart</i>	<i>Desired slot time start stated by customer</i>	<i>Int</i>	<i>No</i>
	<i>ActualEndSlot</i>	<i>Actual slot end time stated by executing trainer</i>	<i>Int</i>	<i>Yes</i>
	<i>Evidence</i>	<i>Links to evidences provided by executing trainer</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>Price</i>	<i>Ticket's price</i>	<i>money</i>	<i>Yes</i>
	<i>DiscountedPrice</i>	<i>Ticket's discounted price</i>	<i>money</i>	<i>Yes</i>
	<i>DistancePriceCalculate</i>	<i>Ticket's distance price</i>	<i>money</i>	<i>Yes</i>
	<i>ConsultingPricePolicyCalculate</i>	<i>Ticket's price based on policy</i>	<i>money</i>	<i>Yes</i>
	<i>Status</i>	<i>Ticket's status, defined by enum: + Cancelled = 0 + WaittingForApprove = 1 + Approved = 2 + Finished = 3</i>	<i>int</i>	<i>No</i>
	<i>ConsultingPricePolicyId</i>	<i>Foreign key, consulting price policy identifier</i>	<i>int</i>	<i>No</i>
<i>ConsultingType</i>	<i>Id {PK}</i>	<i>Unique identifier for consulting type</i>	<i>int</i>	<i>No</i>
	<i>Name</i>	<i>Consulting type's name</i>	<i>varchar(200)</i>	<i>No</i>
<i>ConsultingPricePolicy</i>	<i>Id {PK}</i>	<i>Unique identifier for consulting price policy</i>	<i>int</i>	<i>No</i>
	<i>Price</i>	<i>Price stated in policy</i>	<i>money</i>	<i>No</i>
	<i>OnlineOrOffline</i>	<i>Defining which policy should apply to ticket.</i>	<i>bit</i>	<i>No</i>

		<i>Default value: false is Offline</i>		
<i>DistancePrice</i>	<i>Id {PK}</i>	<i>Unique identifier for distance price policy</i>	<i>int</i>	<i>No</i>
	<i>From</i>	<i>Stating the starting kilometers to apply</i>	<i>int</i>	<i>Yes</i>
	<i>To</i>	<i>Stating the limit kilometers to apply</i>	<i>int</i>	<i>Yes</i>
	<i>PricePerKm</i>	<i>Distance policy's price</i>	<i>money</i>	<i>Yes</i>
<i>Workshop</i>	<i>Id {PK}</i>	<i>Unique identifier for workshop</i>	<i>int</i>	<i>No</i>
	<i>Title</i>	<i>Workshop's title</i>	<i>varchar(300)</i>	<i>No</i>
	<i>Description</i>	<i>Workshop's description</i>	<i>varchar(1000)</i>	<i>Yes</i>
	<i>Picture</i>	<i>Workshop's picture</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>RegisterEnd</i>	<i>Defined maximum duration in days to close registration, apply to classes within workshop</i>	<i>int</i>	<i>Yes</i>
	<i>Price</i>	<i>Workshop's price, apply to classes within workshop</i>	<i>money</i>	<i>No</i>
	<i>TotalSlot</i>	<i>Workshop's total slot spent to complete, apply to classes within workshop</i>	<i>int</i>	<i>No</i>
	<i>Status</i>	<i>Workshop's status, defined as enum: + Inactive = 0 + Active = 1</i>	<i>int</i>	<i>No</i>
	<i>MinimumRegistration</i>	<i>Workshop's minimum registration, apply to classes within workshop</i>	<i>int</i>	<i>No</i>
	<i>MaximumRegistration</i>	<i>Workshop's maximum registration, apply to classes within workshop</i>	<i>int</i>	<i>No</i>
<i>WorkshopDetailTemplate</i>	<i>Id {PK}</i>	<i>Unique identifier for workshop detail template</i>	<i>int</i>	<i>No</i>
	<i>WorkshopId {PK, FK}</i>	<i>Foreign key, workshop identifier, number of records affected by workshop's total slot</i>	<i>int</i>	<i>No</i>
	<i>Detail</i>	<i>Detail of the template</i>	<i>varchar(500)</i>	<i>Yes</i>
<i>WorkshopClass</i>	<i>Id {PK}</i>	<i>Unique identifier for class</i>	<i>int</i>	<i>No</i>
	<i>WorkshopId {FK}</i>	<i>Foreign key, workshop identifier</i>	<i>int</i>	<i>No</i>

	<i>CreateDate</i>	<i>Created date of the class</i>	<i>date</i>	Yes
	<i>Location</i>	<i>Class's hosted location</i>	<i>varchar(512)</i>	No
	<i>RegisterEndDate</i>	<i>Stated by class's created date plus the register end date of the workshop. Affecting to class's status as time passed by this value</i>	<i>date</i>	Yes
	<i>StartTime</i>	<i>Desired open registration time for class, stated by class's created date and business rule applied above. Affecting to class's status as time passed by this value</i>	<i>date</i>	Yes
	<i>Status</i>	<i>Class's status, defined as enum: + Pending = 0 + OpenRegistration = 1 + ClosedRegistration = 2 + OnGoing = 3 + Completed = 4 + Cancelled = 5</i>	<i>int</i>	No
<i>WorkshopClassDetail</i>	<i>Id {PK}</i>	<i>Unique identifier for class detail</i>	<i>int</i>	No
	<i>WorkshopClassId {FK}</i>	<i>Foreign key, class's identifier</i>	<i>int</i>	No
	<i>DetailId {FK}</i>	<i>Foreign key, detail template identifier</i>	<i>int</i>	No
	<i>DaySlotId {FK}</i>	<i>Foreign key, trainer slot identifier</i>	<i>int</i>	Yes
	<i>UpdateDate</i>	<i>Tracked date of modifying</i>	<i>date</i>	Yes
<i>WorkshopRefundPolicy</i>	<i>Id {PK}</i>	<i>Unique identifier for workshop refund policy</i>	<i>int</i>	No
	<i>TotalDayBeforeStart</i>	<i>Stated date refunded before class's execution time</i>	<i>int</i>	No
	<i>RefundRate</i>	<i>Policy's refund rate</i>	<i>float(10)</i>	Yes
<i>Customer_WorkshopClass</i>	<i>CustomerId {PK, FK}</i>	<i>Foreign key and composite key, customer identifier</i>	<i>int</i>	No
	<i>WorkshopClassId {PK, FK}</i>	<i>Foreign key and composite key, workshop class identifier</i>	<i>int</i>	No
	<i>Price</i>	<i>Price paid as customer pay</i>	<i>money</i>	Yes

	<i>DiscountedPrice</i>	<i>Discounted price as customer pay</i>	<i>money</i>	<i>Yes</i>
	<i>RefundRate</i>	<i>Refund rate defined in workshop refund policy, only if refund requested</i>	<i>float(10)</i>	<i>Yes</i>
	<i>Status</i>	<i>Customer registration to class's status, defined as enum: + Unpaid = 0 + Paid = 1 + DemandRefund = 2 + Refunded = 3</i>	<i>int</i>	<i>No</i>
	<i>WorkshopRefundPolicyId</i>	<i>Foreign key, workshop refund policy identifier</i>	<i>int</i>	<i>Yes</i>
<i>WorkshopAttendance</i>	<i>Id {PK}</i>	<i>Unique identifier for workshop class detail's customer attendance</i>	<i>int</i>	<i>No</i>
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>WorkshopClassDetailId {FK}</i>	<i>Foreign key, class detail identifier</i>	<i>int</i>	<i>No</i>
	<i>Status</i>	<i>Attendance status, defined as enum: + NotYet = 0 + Attended = 1 + Absent = 2</i>	<i>int</i>	<i>No</i>
<i>OnlineCourse</i>	<i>Id {PK}</i>	<i>Unique identifier for online course</i>	<i>int</i>	<i>No</i>
	<i>Title</i>	<i>Online course's title</i>	<i>varchar(300)</i>	<i>Yes</i>
	<i>ShortDescription</i>	<i>Online course's description</i>	<i>varchar(1000)</i>	<i>Yes</i>
	<i>Picture</i>	<i>Online course's picture</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>Price</i>	<i>Online course's price</i>	<i>money</i>	<i>Yes</i>
	<i>Status</i>	<i>Online course's status, defined as enum: + INACTIVE = 0 + ACTIVE = 1 + CANCELLED = 2</i>	<i>int</i>	<i>No</i>
<i>Section</i>	<i>Id {PK}</i>	<i>Unique identifier for section in online course</i>	<i>int</i>	<i>No</i>
	<i>OnlineCourseId {FK}</i>	<i>Foreign key, online course identifier</i>	<i>int</i>	<i>No</i>
	<i>Title</i>	<i>Section's title</i>	<i>varchar(300)</i>	<i>No</i>
	<i>Description</i>	<i>Section description</i>	<i>varchar(512)</i>	<i>Yes</i>
	<i>ResourceFiles</i>	<i>Section's resource files' links</i>	<i>varchar(512)</i>	<i>Yes</i>
<i>Lesson</i>	<i>Id {PK}</i>	<i>Unique identifier for lesson in section</i>	<i>int</i>	<i>No</i>
	<i>SectionId {FK}</i>	<i>Foreign key, section identifier</i>	<i>int</i>	<i>No</i>

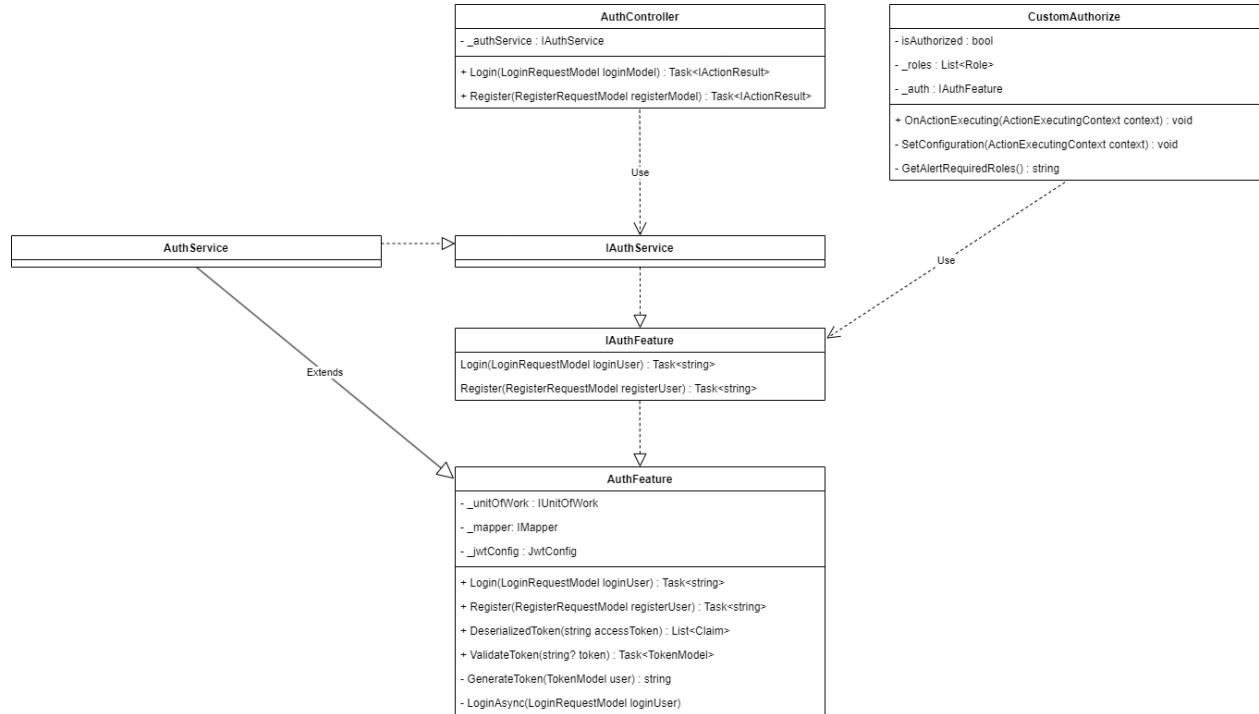
	<i>Title</i>	<i>Lesson's title</i>	<i>varchar(300)</i>	<i>No</i>
	<i>Description</i>	<i>Lesson's description</i>	<i>varchar(1000)</i>	<i>Yes</i>
	<i>Detail</i>	<i>Lesson's detail</i>	<i>varchar(1000)</i>	<i>Yes</i>
	<i>Video</i>	<i>Lesson's link to video</i>	<i>varchar(512)</i>	<i>Yes</i>
<i>Certificate</i>	<i>Id {PK}</i>	<i>Unique identifier for online course certificate template</i>	<i>int</i>	<i>No</i>
	<i>OnlineCourseId {FK}</i>	<i>Foreign key, online course identifier</i>	<i>int</i>	<i>No</i>
	<i>BirdCenterName</i>	<i>Certificate's center name</i>	<i>varchar(100)</i>	<i>No</i>
	<i>Title</i>	<i>Certificate's title</i>	<i>varchar(300)</i>	<i>Yes</i>
	<i>ShortDescription</i>	<i>Certificate's short description</i>	<i>varchar(500)</i>	<i>Yes</i>
	<i>Picture</i>	<i>Certificate's provided picture</i>	<i>varchar(512)</i>	<i>Yes</i>
<i>Customer_CertificateDetail</i>	<i>CustomerId {PK, FK}</i>	<i>Foreign key and composite key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>Certifteicateld {PK, FK}</i>	<i>Foreign key and composite key, certificate template identifier</i>	<i>int</i>	<i>No</i>
	<i>ReceiveDate</i>	<i>Date of publication of customer's certificate</i>	<i>date</i>	<i>Yes</i>
<i>Customer_OnlineCourseDetail</i>	<i>CustomerId {PK, FK}</i>	<i>Foreign key and composite key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>OnlineCourseId {PK, FK}</i>	<i>Foreign key and composite key, online course identifier</i>	<i>int</i>	<i>No</i>
	<i>Price</i>	<i>Price as customer's purchase</i>	<i>money</i>	<i>Yes</i>
	<i>DiscountedPrice</i>	<i>Discounted price as customer's purchase</i>	<i>money</i>	<i>Yes</i>
	<i>Status</i>	<i>Tracked customer progress in online course, defined as enum: + Enrolled = 0 + Completed = 1</i>	<i>int</i>	<i>No</i>
<i>Customer_SectionDetail</i>	<i>CustomerId {PK, FK}</i>	<i>Foreign key and composite key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>SectionId {PK, FK}</i>	<i>Foreign key and composite key, section identifier</i>	<i>int</i>	<i>No</i>
	<i>IsComplete</i>	<i>Tracked customer progress in section, default value: false is not complete</i>	<i>bit</i>	<i>Yes</i>

<i>Customer_LessonDetail</i>	<i>CustomerId {PK, FK}</i>	<i>Foreign key and composite key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>LessonId {PK, FK}</i>	<i>Foreign key and composite key, lesson identifier</i>	<i>int</i>	<i>No</i>
	<i>IsComplete</i>	<i>Tracked customer progress in lesson, default value: false is not complete</i>	<i>bit</i>	<i>Yes</i>
<i>Feedback</i>	<i>Id {PK}</i>	<i>Unique identifier for feedback</i>	<i>int</i>	<i>No</i>
	<i>CustomerId {FK}</i>	<i>Foreign key, customer identifier</i>	<i>int</i>	<i>No</i>
	<i>EntityTypeId</i>	<i>Identifier of feedback target, defined as enum: + Others = 1 + AdviceConsulting = 2 + WorkshopClass = 3 + OnlineCourse = 4 + TrainingCourse = 5 + Topup = 6 + AbsentRequest = 7</i>	<i>int</i>	<i>No</i>
	<i>EntityId</i>	<i>Identifier of feedback target</i>	<i>int</i>	<i>Yes</i>
	<i>FeedbackDetail</i>	<i>Feedback's detail</i>	<i>int</i>	<i>Yes</i>
	<i>Rating</i>	<i>Feedback's rating</i>	<i>int</i>	<i>Yes</i>

### 3. Detailed Design

#### 3.1 Authentication/Authorization

##### 3.1.1 Class Diagram



##### 3.1.2 Class Specification

###### 3.1.2.1 AuthController

No	Attribute	Data Type	Visibility	Description
01	_authService	IAuthService	Private	

No	Method	Return Type	Visibility	Description
01	Login	Task<ActionResult>	Public	Parameters: + loginModel: LoginRequestModel (3.1.2.2)
02	Register	Task<ActionResult>	Public	Parameters: + registerModel: RegisterRequestModel (3.1.2.3)

###### 3.1.2.2 LoginRequestModel

No	Attribute	Data Type	Visibility	Description
01	Email	string	Public	
02	Password	string	Public	

No	Method	Return Type	Visibility	Description

###### 3.1.2.3 RegisterRequestModel

No	Attribute	Data Type	Visibility	Description

01	Email	string	Public	Apply regex: ^[\w-\.]+@[{\w-}]+\.{\w-}{2,4}\$
02	Password	string	Public	
03	Name	string	Public	Apply regex: ^[a-zA-Z0-9_]{3,20}\$
04	PhoneNumber	string	Public	Apply regex: ^(?:\+?84 0)(?:\d{9} (?:\d{2}-\d{3}-\d{4}) (?:\d{3}-\d{2}-\d{2}-\d{2}))\$

No	Method	Return Type	Visibility	Description

### 3.1.2.4 CustomAuthorize

- This class is inherited from ActionFilterAttribute class using Microsoft.AspNetCore.Mvc.Filters library.

- This class is implementing IActionFilter interface using Microsoft.AspNetCore.Mvc.Filters library.

No	Attribute	Data Type	Visibility	Description
01	isAuthorized	bool	Private	
02	_roles	List<Role>	Private	Role: enum values referred to table User, RoleId field in <a href="#">2.2</a>
03	_auth	IAuthFeature	Private	

No	Method	Return Type	Visibility	Description
01	OnActionExecuting	void	Public	Parameters: + context: ActionExecutingContext  This function is overridden one in ActionFilterAttribute
02	SetConfiguration	void	Private	Parameters: + context: ActionExecutingContext
03	GetAlertRequiredRoles	string	Private	Append error message on unauthorized cases

### 3.1.2.5 IAuthService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.1.2.6 AuthService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

--	--	--	--	--

### 3.1.2.7 IAuthFeature

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	Login	Task<string>	Public	Parameters: + loginUser: LoginRequestModel (3.1.2.2)
02	Register	Task<string>	Public	Parameters: + registerUser: RegisterRequestModel (3.1.2.3)

### 3.1.2.8 AuthFeature

No	Attribute	Data Type	Visibility	Description
01	_unitOfWork	IUnitOfWork	Private	
02	_mapper	IMapper	Private	
03	_jwtConfig	JwtConfig	Private	

No	Method	Return Type	Visibility	Description
01	Login	Task<string>	Public	Parameters: + loginUser: LoginRequestModel (3.1.2.2)
02	Register	Task<string>	Public	Parameters: + registerUser: RegisterRequestModel (3.1.2.3)
03	DeserializedToken	List<Claim>	Public	Parameters: + accessToken: string
04	ValidateToken	Task<TokenModel>	Public	Parameters: + token: string? Return: + TokenModel (3.1.2.9)
05	GenerateToken	string	Private	Parameters: + user: TokenModel (3.1.2.9)
06	LoginAsync	Task<TokenModel>	Private	Parameters: + loginUser: LoginRequestModel (3.1.2.2) Return: + TokenModel (3.1.2.9)

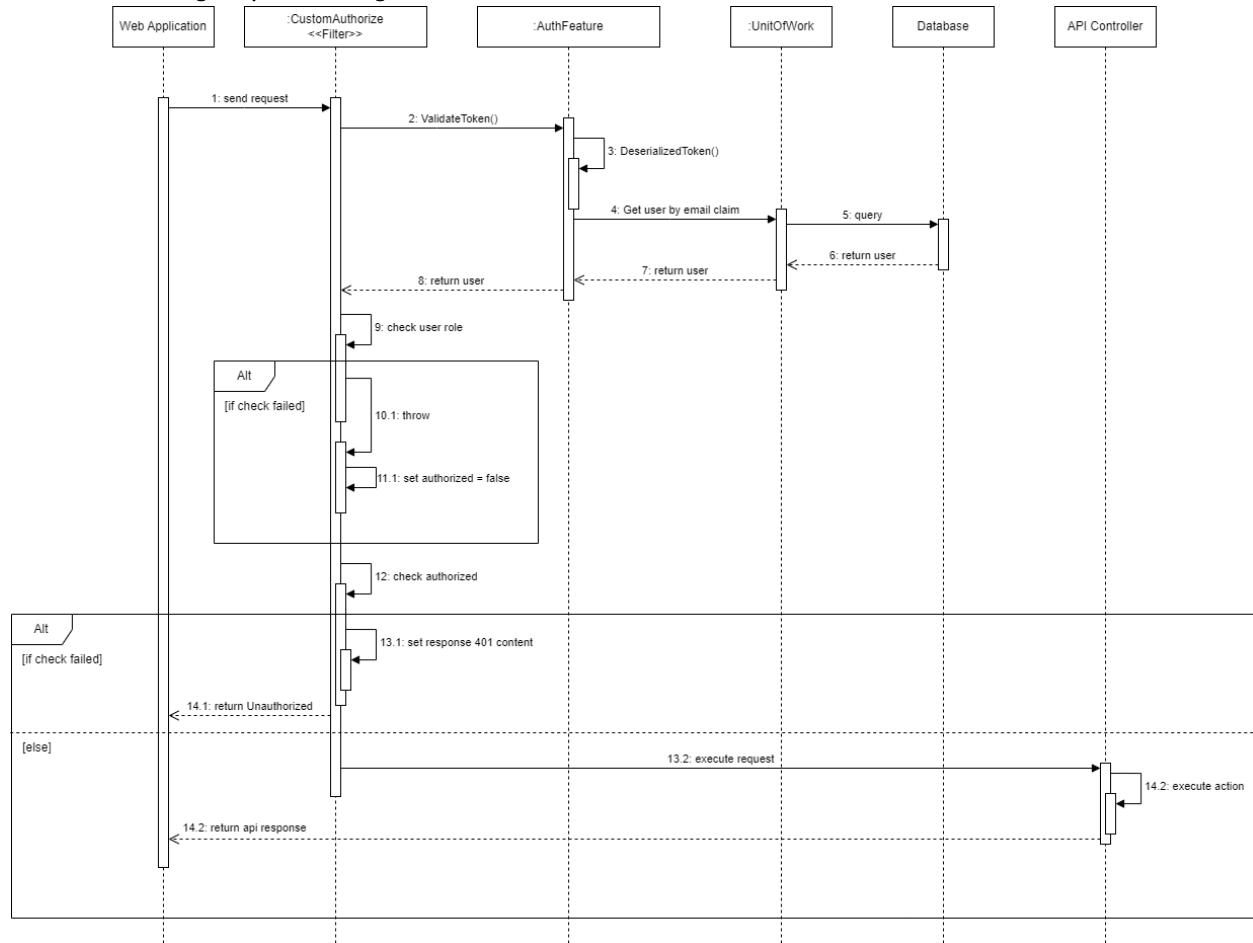
### 3.1.2.9 TokenModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	Name	string	Public	

03	Email	string	Public	
04	Avatar	string	Public	
05	Role	Role	Public	Enum values referring to table User, RoleId field in <a href="#">2.2</a>

No	Method	Return Type	Visibility	Description

### 3.1.3 Authorizing Sequence Diagram



## 3.2 Training Course Management Feature

### 3.2.1 Class Diagram

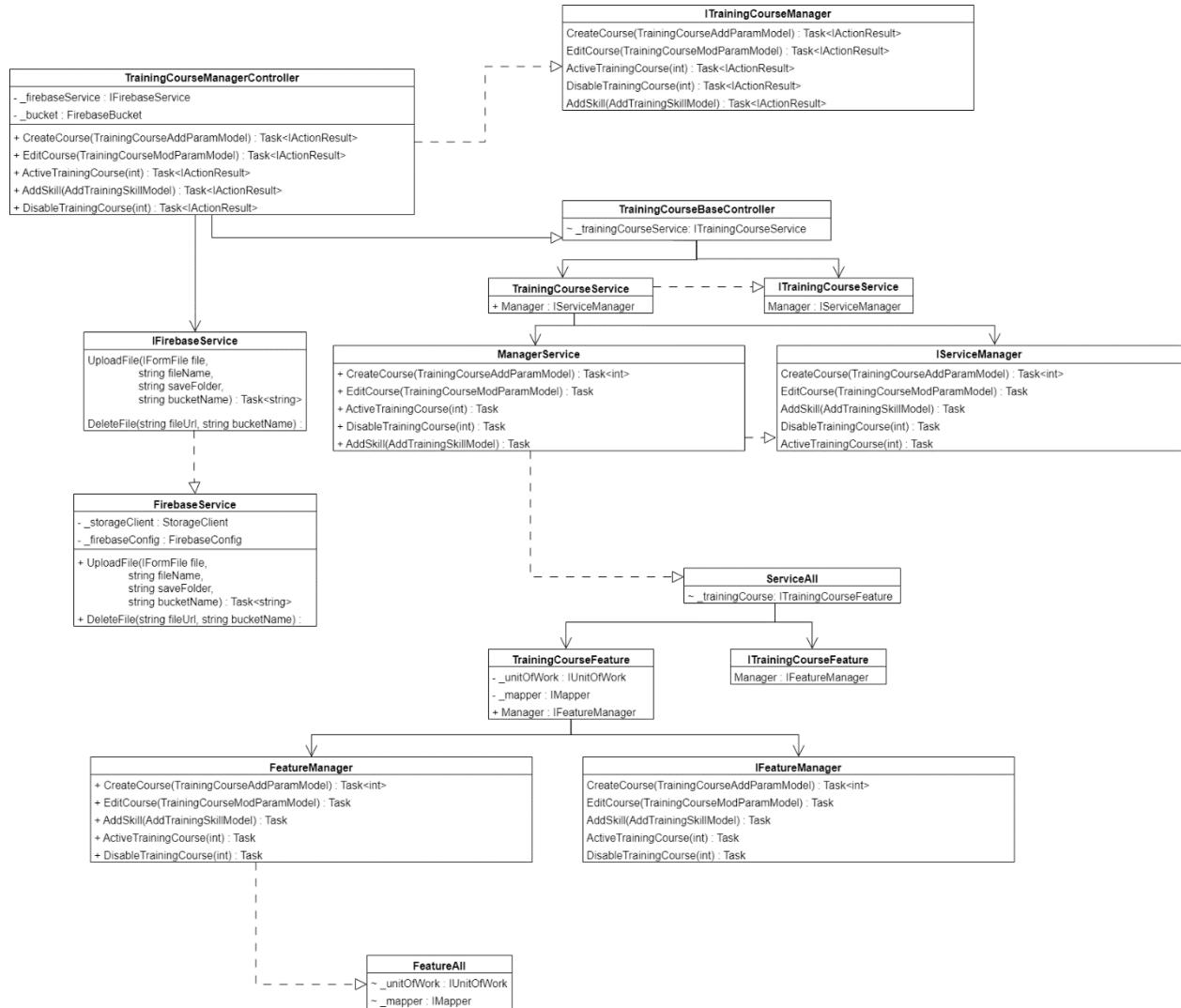


Figure 3.2.1.1

### 3.2.2 Class Specification

#### 3.2.2.1 TrainingCourseManagerController

No	Attribute	Data Type	Visibility	Description
01	<code>_firebaseService</code>	<code>IFirebaseService</code>	Private	Dependency Injection
02	<code>_bucket</code>	<code>FirebaseBucket</code>	Private	Firebase bucket's configuration model

No	Method	Return Type	Visibility	Description
01	<code>CreateCourse</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for creating training course
02	<code>EditCourse</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for modifying training course
03	<code>ActiveTrainingCourse</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for activating training course

04	<i>DisableTrainingCourse</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for deactivating training course</i>
05	<i>AddSkill</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API add bird skill to training course</i>

### 3.2.2.2 FirebaseService

No	Attribute	Data Type	Visibility	Description
01	<i>_storageClient</i>	<i>StorageClient</i>	<i>Private</i>	
02	<i>firebaseConfig</i>	<i>FirebaseConfig</i>	<i>Private</i>	

No	Method	Return Type	Visibility	Description
01	<i>UploadFile</i>	<i>Task&lt;string&gt;</i>	<i>Public</i>	<i>Params:</i> <i>+file: IFormFile</i> <i>+fileName: string</i> <i>+saveFolder: string</i> <i>+bucketName: string</i>
02	<i>DeleteFile</i>	<i>Task&lt;bool&gt;</i>	<i>Public</i>	<i>Params:</i> <i>+filePath: string</i> <i>+bucketName: string</i>

### 3.2.2.3 TrainingCourseAddParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>BirdSpeciesId</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string</i>	<i>Public</i>	
03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>Pictures</i>	<i>List&lt;IFormFile&gt;?</i>	<i>Public</i>	<i>Uploaded images for training course</i>
05	<i>TotalPrice</i>	<i>decimal</i>	<i>Public</i>	<i>Must be positive</i>

No	Method	Return Type	Visibility	Description
01	<i>ToTrainingCourseModel</i>	<i>TrainingCourseAddModel</i>	<i>Public</i>	<i>Converting function</i>

### 3.2.2.4 TrainingCourseModParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>BirdSpeciesId</i>	<i>int?</i>	<i>Public</i>	
03	<i>Title</i>	<i>string?</i>	<i>Public</i>	
04	<i>Description</i>	<i>string?</i>	<i>Public</i>	
05	<i>Pictures</i>	<i>List&lt;IFormFile&gt;?</i>	<i>Public</i>	<i>Uploaded images for training course</i>
06	<i>TotalPrice</i>	<i>decimal?</i>	<i>Public</i>	<i>Must be positive</i>

No	Method	Return Type	Visibility	Description
01	<i>ToTrainingCourseModel</i>	<i>TrainingCourseModifyModel</i>	<i>Public</i>	<i>Converting function</i>

### 3.2.2.5 AddTrainingSkillModel

No	Attribute	Data Type	Visibility	Description
01	<i>BirdSkillId</i>	<i>int</i>	<i>Public</i>	
02	<i>TrainingCourseId</i>	<i>int</i>	<i>Public</i>	

03	TotalSlot	<i>int</i>	Public	Must be positive
----	-----------	------------	--------	------------------

### 3.2.2.6 ManagerService

No	Method	Return Type	Visibility	Description
01	CreateCourse	<i>Task&lt;int&gt;</i>	Public	Params: + trainingCourse: <i>TrainingCourseAddModel</i>
02	EditCourse	<i>Task</i>	Public	Params: + trainingCourse: <i>TrainingCourseModifyModel</i>
03	ActiveTrainingCourse	<i>Task</i>	Public	Params: + trainingCourseId: int
04	DisableTrainingCourse	<i>Task</i>	Public	Params: + trainingCourseId: int
05	AddSkill	<i>Task</i>	Public	Params: + trainingCourseSkill: <i>AddTrainingSkillModel</i>

### 3.2.2.7 ServiceAll

No	Attribute	Data Type	Visibility	Description
01	_trainingcourse	<i>ITrainingCourseFeature</i>	Internal	

### 3.2.2.8 FeatureManager

No	Method	Return Type	Visibility	Description
01	CreateCourse	<i>Task&lt;int&gt;</i>	Public	Params: + trainingCourse: <i>TrainingCourseAddModel</i>
02	EditCourse	<i>Task</i>	Public	Params: + trainingCourse: <i>TrainingCourseModifyModel</i>
03	ActiveTrainingCourse	<i>Task</i>	Public	Params: + trainingCourseId: int
04	DisableTrainingCourse	<i>Task</i>	Public	Params: + trainingCourseId: int
05	AddSkill	<i>Task</i>	Public	Params: + trainingCourseSkill: <i>AddTrainingSkillModel</i>

### 3.2.2.9 FeatureAll

No	Attribute	Data Type	Visibility	Description
01	_trainingcourse	<i>ITrainingCourseFeature</i>	Internal	

### 3.2.3 Create Training Course Sequence Diagram

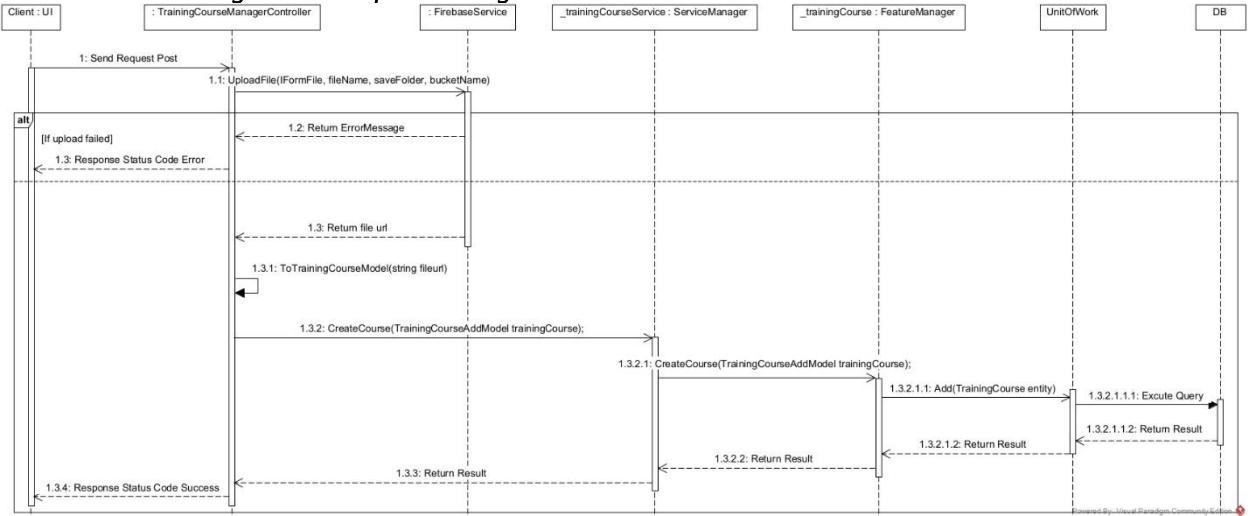


Figure 3.2.3.1

### 3.2.4 Add Bird Skill to Training Course Sequence Diagram

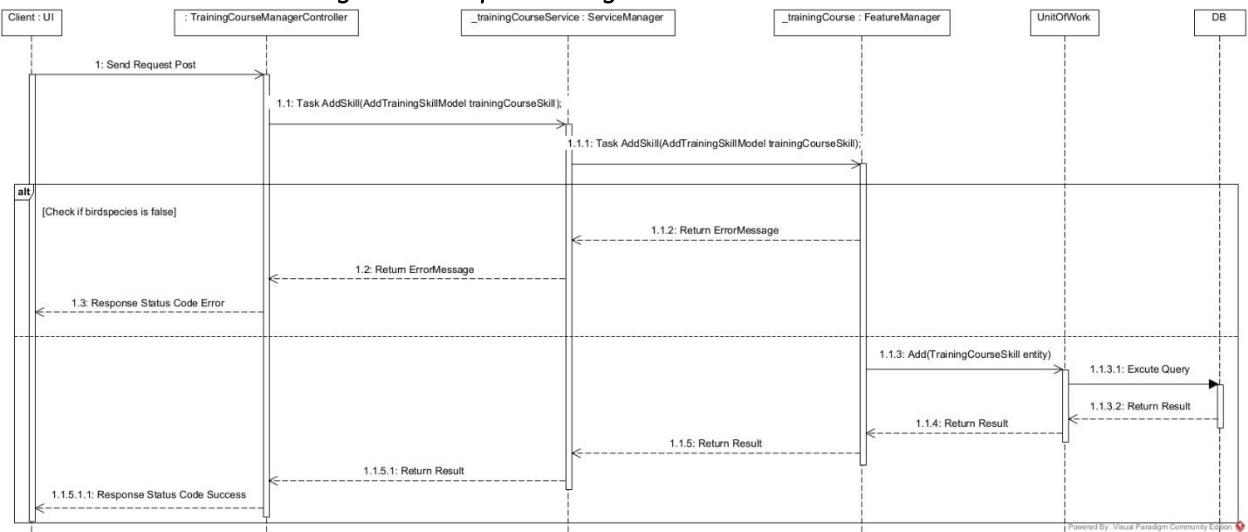


Figure 3.2.4.1

### 3.3 Training Management Feature

#### 3.3.1 ClassDiagram

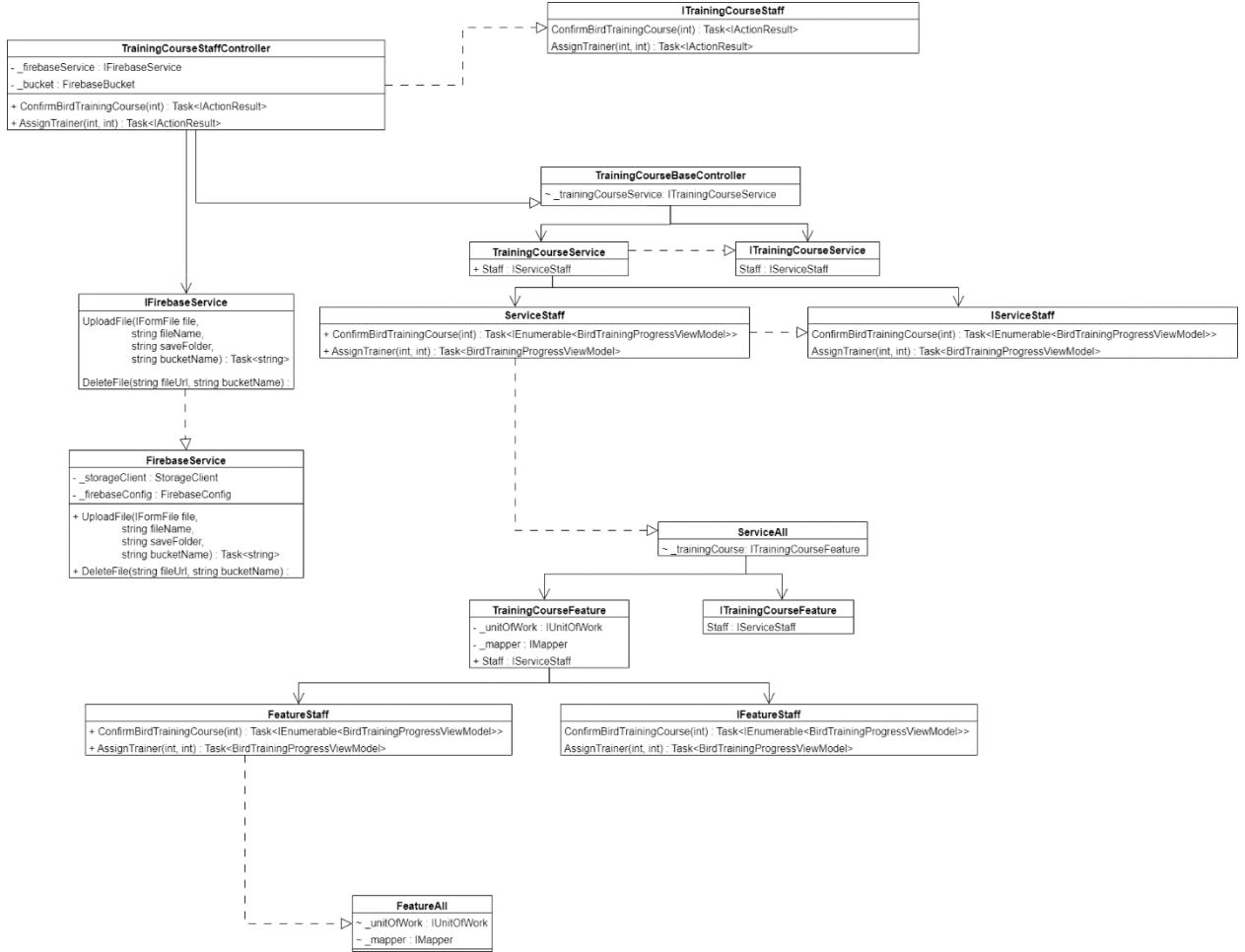


Figure 3.3.1.1

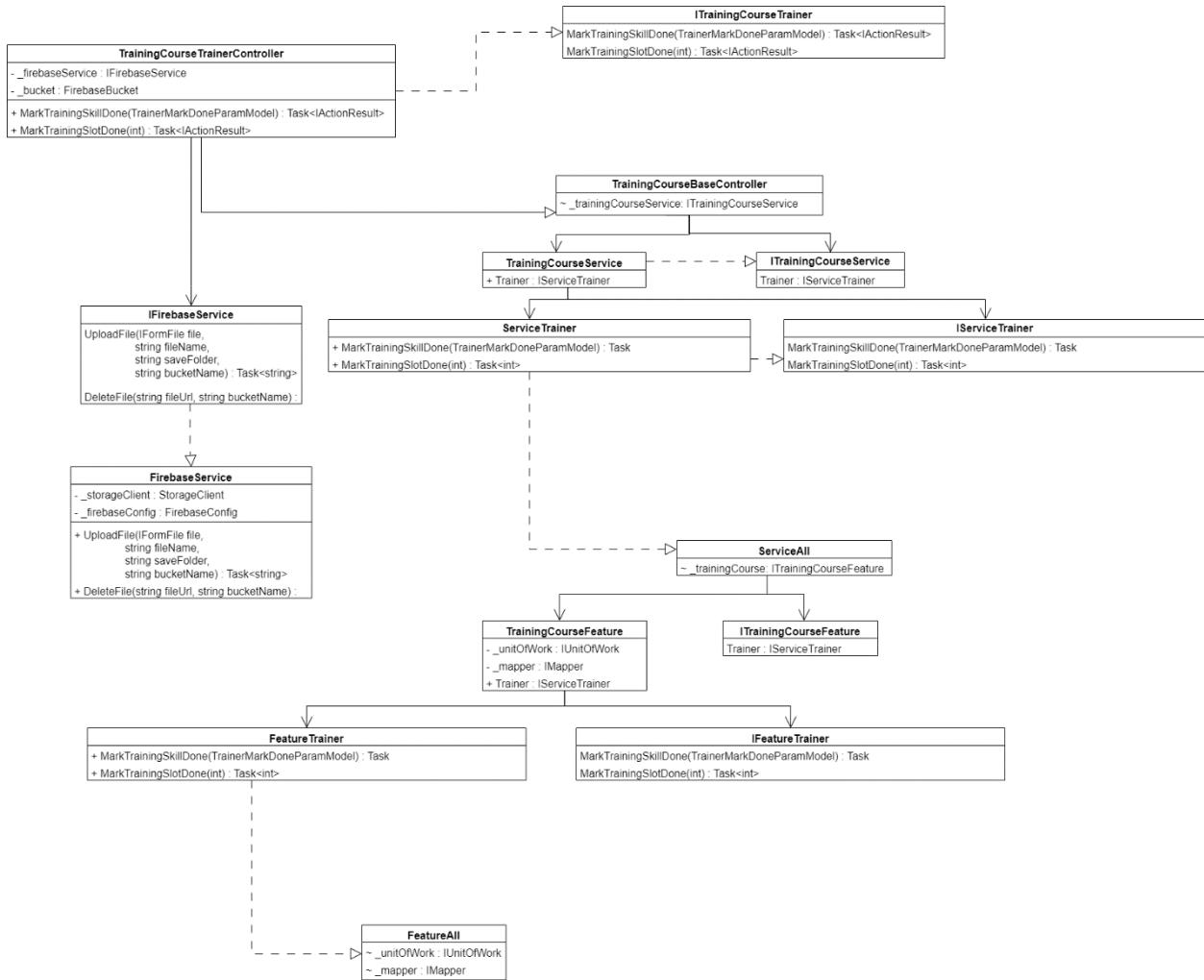


Figure 3.3.1.2

### 3.3.2 Class Specification

#### 3.3.2.1 TrainingCourseStaffController

No	Attribute	Data Type	Visibility	Description
01	<code>_firebaseService</code>	<code>IFirebaseService</code>	Private	Dependency Injection
02	<code>_bucket</code>	<code>FirebaseBucket</code>	Private	Firebase bucket's configuration model

No	Method	Return Type	Visibility	Description
01	<code>ConfirmBirdTrainingCourse</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for confirm training request
02	<code>AssignTrainer</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for assign trainer to training bird

#### 3.3.2.2 TrainingCourseTrainerController

No	Attribute	Data Type	Visibility	Description
01	<code>_firebaseService</code>	<code>IFirebaseService</code>	Private	Dependency Injection
02	<code>_bucket</code>	<code>FirebaseBucket</code>	Private	Firebase bucket's configuration model

No	Method	Return Type	Visibility	Description
01	<code>MarkTrainingSkillDone</code>	<code>Task&lt;ActionResult&gt;</code>	<code>Public</code>	<i>API endpoint for marking training a skill has done</i>
02	<code>MarkTrainingSlotDone</code>	<code>Task&lt;ActionResult&gt;</code>	<code>Public</code>	<i>API endpoint for marking training slot has done</i>

### 3.3.2.3 TrainerMarkDoneParamModel

No	Attribute	Data Type	Visibility	Description
01	<code>Id</code>	<code>int</code>	<code>Public</code>	
02	<code>Evidences</code>	<code>List&lt;IFormFile&gt;?</code>	<code>Public</code>	<i>Uploaded evidence videos for bird training progress</i>
03	<code>Status</code>	<code>string</code>	<code>Public</code>	

No	Method	Return Type	Visibility	Description
01	<code>ToMarkSkill</code>	<code>MarkSkillDone</code>	<code>Public</code>	<i>Converting function</i>

### 3.3.2.4 StaffService

No	Method	Return Type	Visibility	Description
01	<code>ConfirmBirdTrainingCourse</code>	<code>Task&lt;IEnumerable&lt;BirdTrainingProgressViewModel&gt;&gt;</code>	<code>Public</code>	<i>Params: + birdTrainingCourseId: int</i>
02	<code>AssignTrainer</code>	<code>Task&lt;BirdTrainingProgressViewModel&gt;</code>	<code>Public</code>	<i>Params: + progressId: int + trainerId: int</i>

### 3.3.2.5 FeatureStaff

No	Method	Return Type	Visibility	Description
01	<code>ConfirmBirdTrainingCourse</code>	<code>Task&lt;IEnumerable&lt;BirdTrainingProgressViewModel&gt;&gt;</code>	<code>Public</code>	<i>Params: + birdTrainingCourseId: int</i>
02	<code>AssignTrainer</code>	<code>Task&lt;BirdTrainingProgressViewModel&gt;</code>	<code>Public</code>	<i>Params: + progressId: int + trainerId: int</i>

### 3.3.2.5 TrainerService

No	Method	Return Type	Visibility	Description
01	<code>MarkTrainingSkillDone</code>	<code>Task</code>	<code>Public</code>	<i>Params: + markDone: MarkSkillDone</i>
02	<code>MarkTrainingSlotDone</code>	<code>Task&lt;int&gt;</code>	<code>Public</code>	<i>Params: + birdTrainingReportId: int</i>

### 3.3.2.6 FeatureTrainer

No	Method	Return Type	Visibility	Description
01	<code>MarkTrainingSkillDone</code>	<code>Task</code>	<code>Public</code>	<i>Params: + markDone: MarkSkillDone</i>
02	<code>MarkTrainingSlotDone</code>	<code>Task&lt;int&gt;</code>	<code>Public</code>	<i>Params: + birdTrainingReportId: int</i>

### 3.3.3 Staff Confirm Request Sequence Diagram

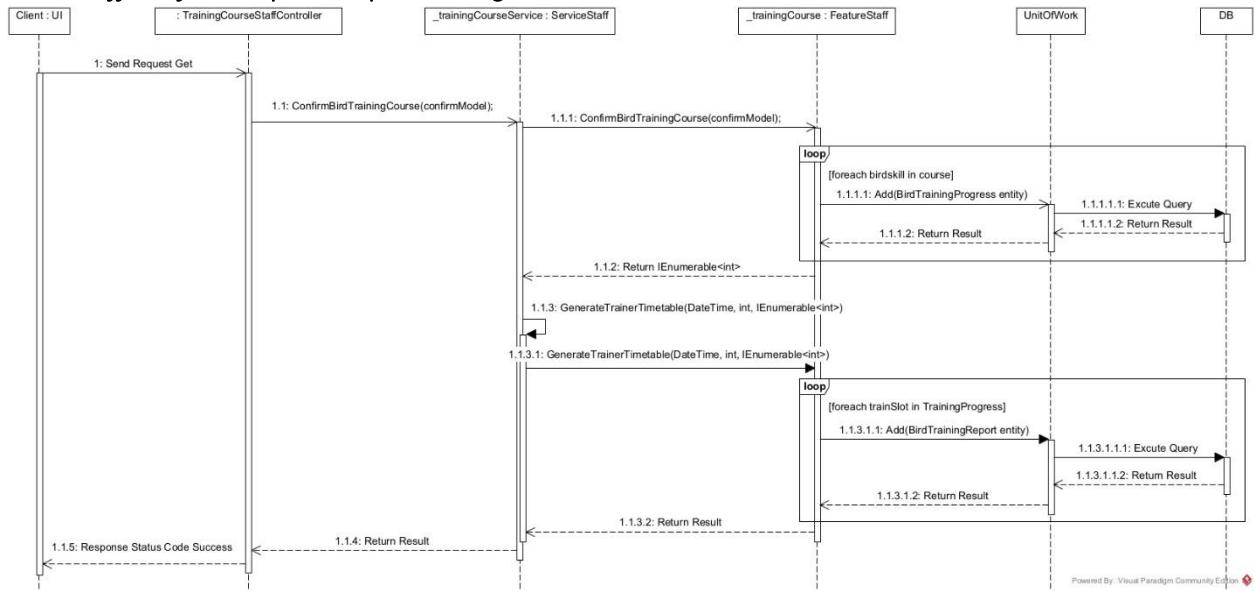


Figure 3.3.3.1

### 3.3.4 Staff Assign Trainer To Course Sequence Diagram

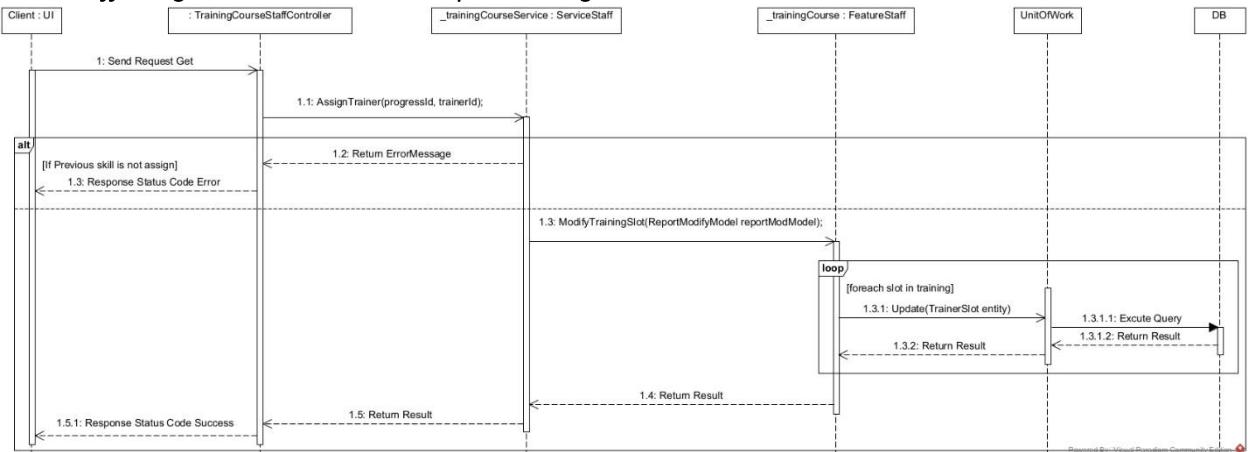


Figure 3.3.4.1

### 3.3.5 Trainer Mark Slot Done Sequence Diagram

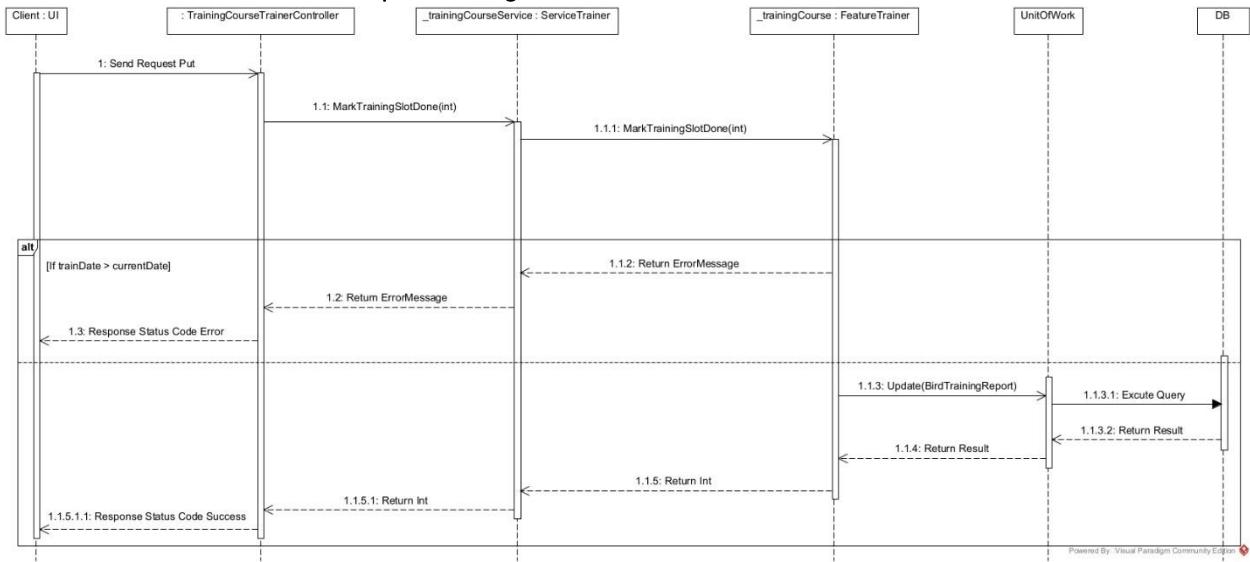


Figure 3.3.5.1

### 3.3.6 Trainer Mark Skill Done Sequence Diagram

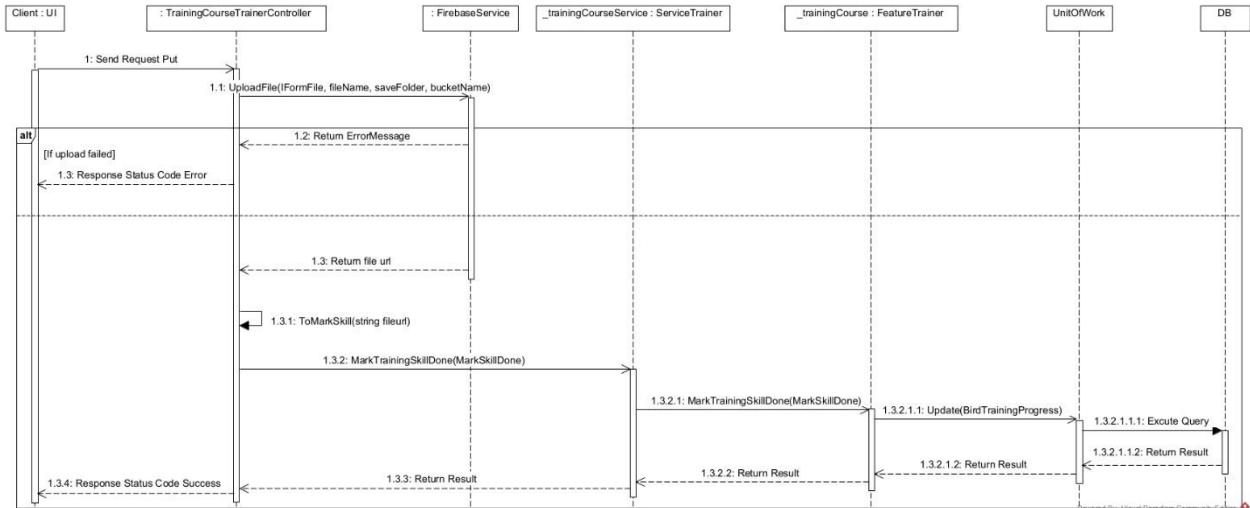


Figure 3.3.6.1

## 3.4 Customer Feature In Training Course

### 3.4.1 ClassDiagram

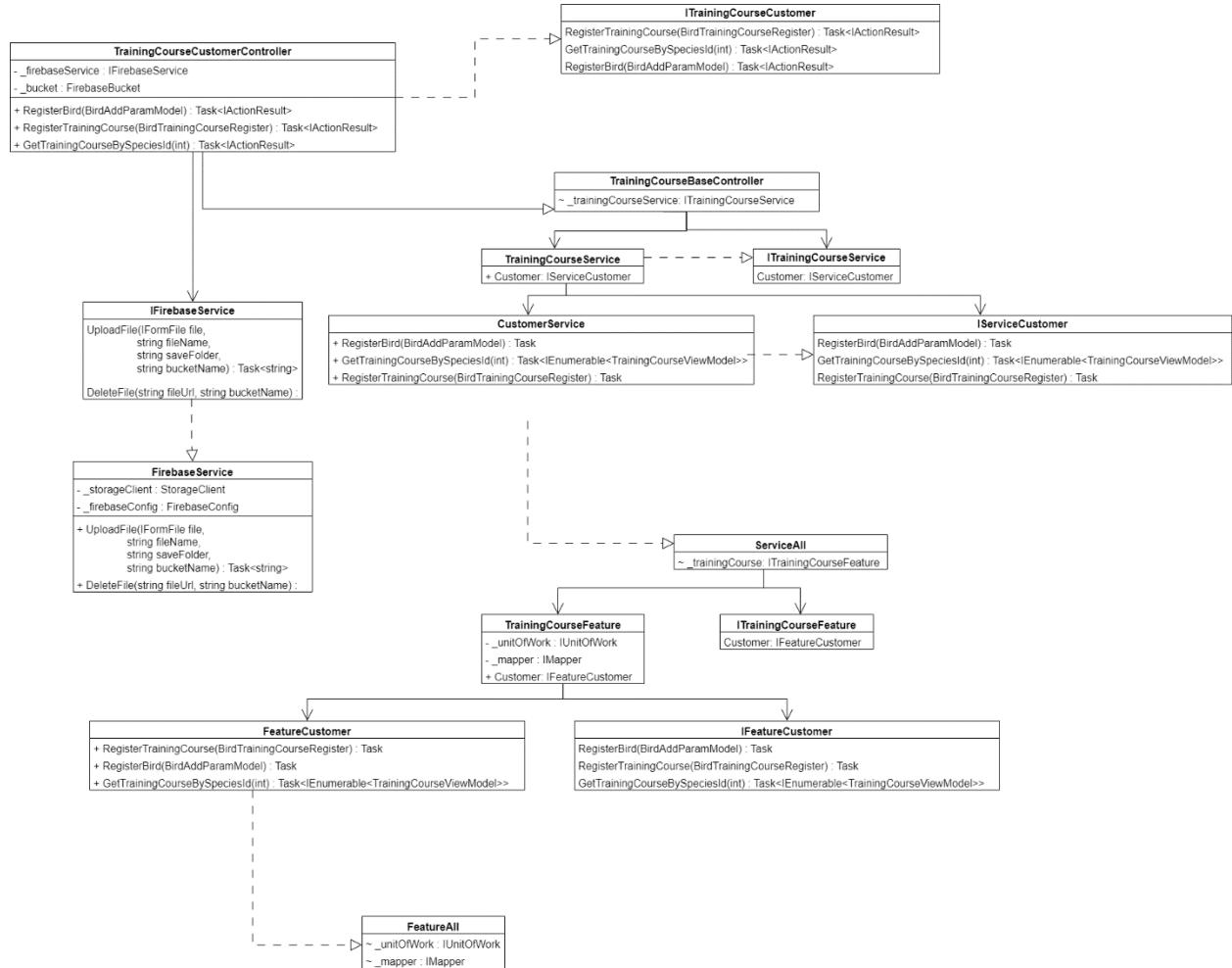


Figure 3.4.1.1

### 3.4.2 Class Specification

#### 3.4.2.1 TrainingCourseCustomerController

No	Attribute	Data Type	Visibility	Description
01	<code>_firebaseService</code>	<code>IFirebaseService</code>	Private	Dependency Injection
02	<code>_bucket</code>	<code>FirebaseBucket</code>	Private	Firebase bucket's configuration model

No	Method	Return Type	Visibility	Description
01	<code>RegisterBird</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for register bird
02	<code>RegisterTrainingCourse</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for register training course for bird
03	<code>GetTrainingCourseBySpeciesId</code>	<code>Task&lt;ActionResult&gt;</code>	Public	API endpoint for getting course suitable for bird

#### 3.4.2.2 BirdAddParamModel

No	Attribute	Data Type	Visibility	Description
01	<code>CustomerId</code>	<code>int</code>	Public	
02	<code>BirdSpeciesId</code>	<code>int</code>	Public	

03	Name	string?	Public	
04	Color	string?	Public	
05	Picture	IFormFile?	Public	Uploaded image for bird
06	Description	string?	Public	
07	IsDefault	Bool?	Public	

No	Method	Return Type	Visibility	Description
01	ToBirdAddModel	BirdAddModel	Public	Converting function

#### 3.4.2.3 BirdTrainingCourseRegister

No	Attribute	Data Type	Visibility	Description
01	BirdId	int	Public	
02	TrainingCourseId	int	Public	
03	CustomerId	int	Public	
04	TotalPrice	decimal?	Public	Must be positive

#### 3.4.2.4 CustomerService

No	Method	Return Type	Visibility	Description
01	RegisterBird	Task<BirdViewModel>	Public	Params: + bird: BirdAddModel
02	RegisterTrainingCourse	Task	Public	Params: +birdTrainingCourseRegister: BirdTrainingCourseRegister
03	GetTrainingCourseBySpeciesId	Task<IEnumerable<TrainingCourseView Model>>	Public	Params: + birdSpeciesId: int

#### 3.4.2.5 FeatureCustomer

No	Method	Return Type	Visibility	Description
01	RegisterBird	Task<BirdViewModel>	Public	Params: + bird: BirdAddModel
02	RegisterTrainingCourse	Task	Public	Params: +birdTrainingCourseRegister: BirdTrainingCourseRegister
03	GetTrainingCourseBySpeciesId	Task<IEnumerable<TrainingCourseView Model>>	Public	Params: + birdSpeciesId: int

### 3.4.3 Customer Register Bird Sequence Diagram

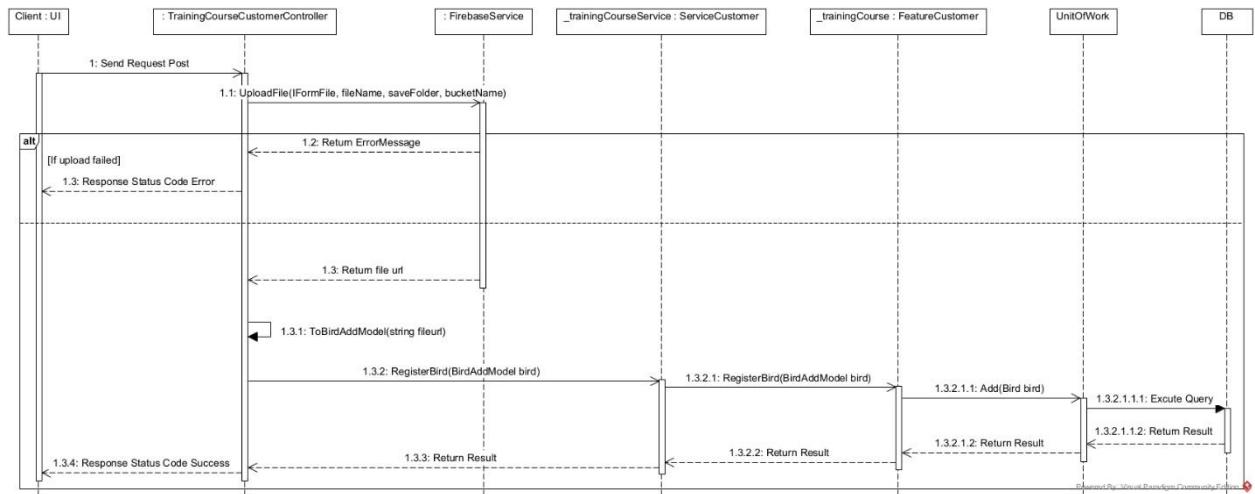


Figure 3.4.3.1

### 3.4.4 Customer View Training Course Sequence Diagram

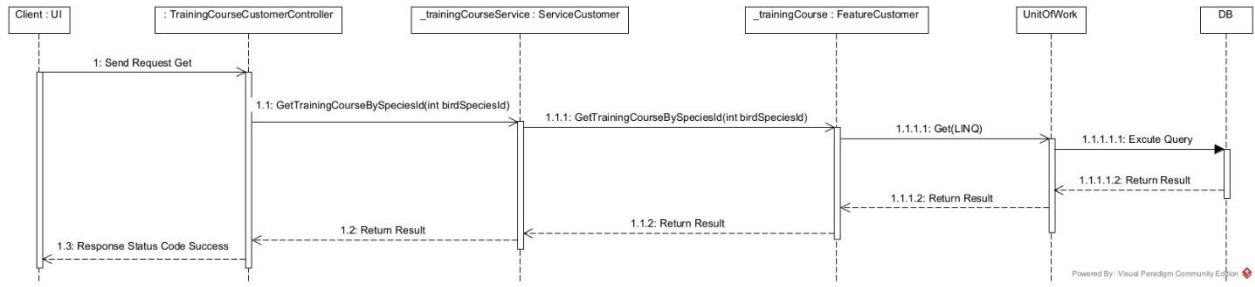


Figure 3.4.4.1

### 3.4.5 Customer Register Training Course for Bird Sequence Diagram

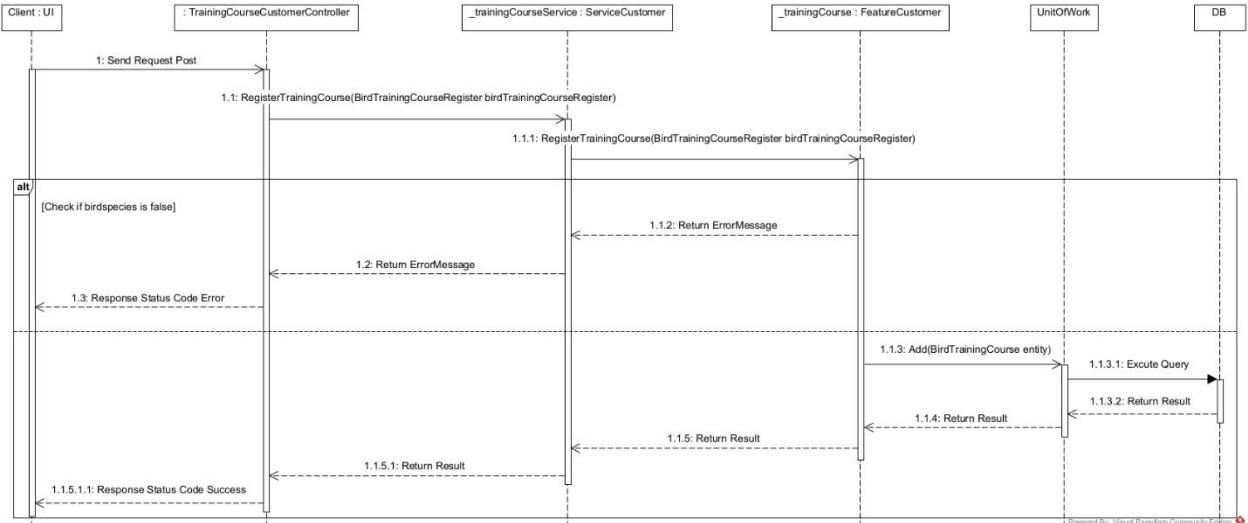
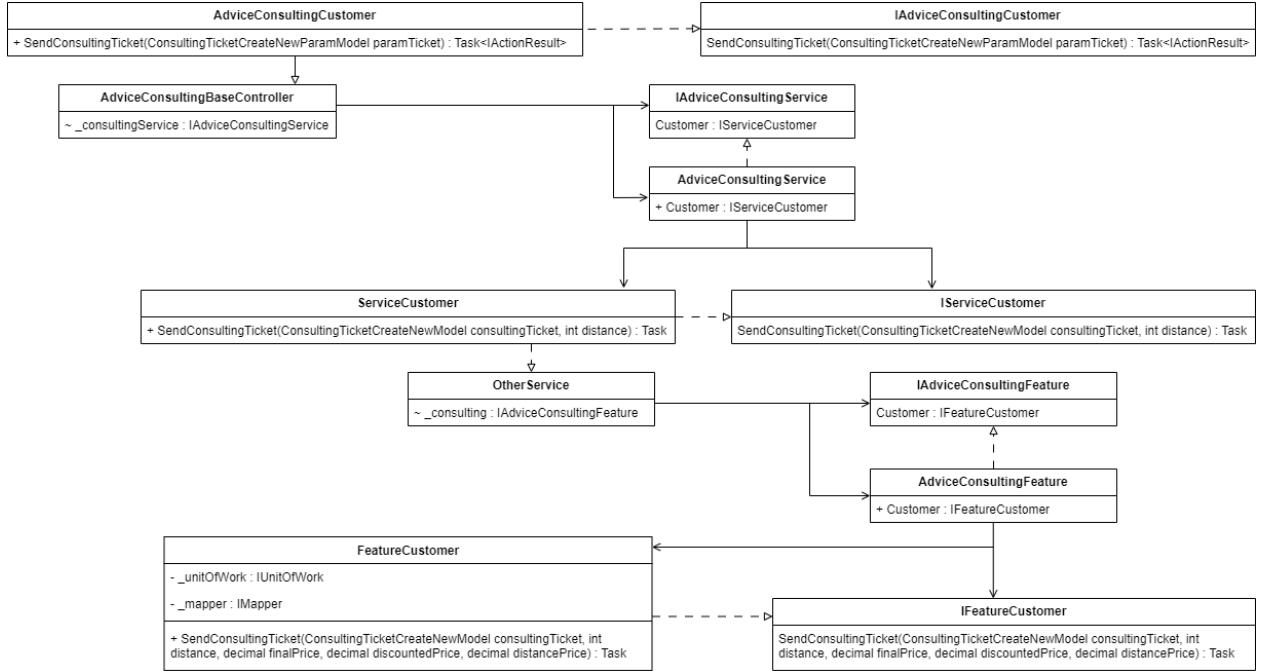


Figure 3.4.5.1

## 3.5 Send Consulting Ticket Feature

### 3.5.1 Class Diagram



### 3.5.2 Class Specification

#### 3.5.2.1 AdviceConsultingCustomer

No	Method	Return Type	Visibility	Description
01	SendConsultingTicket	Task<ActionResult>	Public	API endpoint for creating consulting ticket

##### 3.5.2.1.1 ConsultingTicketCreateNewParamModel

No	Attribute	Datatype	Visibility	Description
01	TrainerId	int	Public	
02	Address	string?	Public	
03	ConsultingTypeld	int	Public	
04	ConsultingDetail	string?	Public	
05	OnlineOrOffline	bool	Public	Service of the ticket. True is Online and False is Offline
06	AppointmentDate	DateTime	Public	
07	ActualSlotStart	int	Public	Id of the started slot

#### 3.5.2.2 ServiceCustomer

No	Method	Return Type	Visibility	Description
01	SendConsultingTicket	Task	Public	

##### 3.5.2.2.1 ConsultingTicketCreateNewParamModel

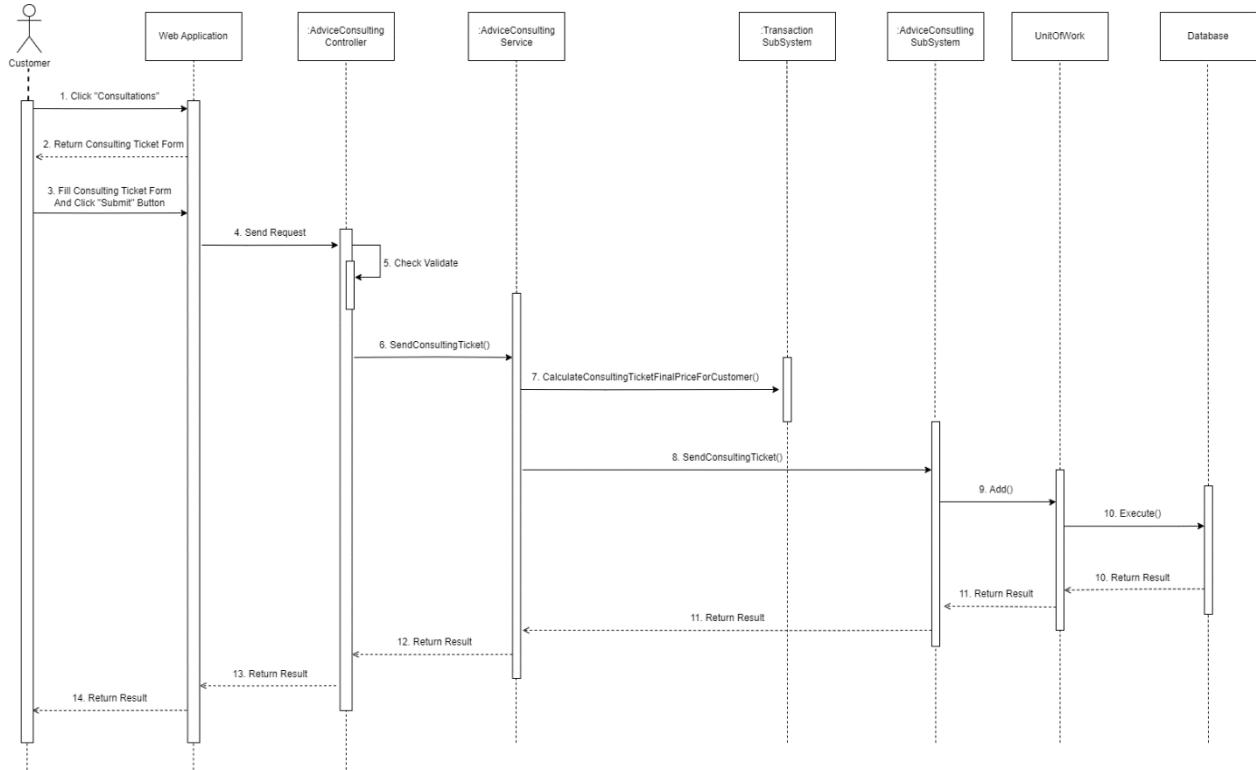
No	Attribute	Datatype	Visibility	Description
01	CustomerId	int	Public	
02	TrainerId	int	Public	
03	AddressDetail	string?	Public	
04	ConsultingTypeld	int	Public	
05	ConsultingDetail	string?	Public	

06	OnlineOrOffline	bool	Public	<i>Service of the ticket. True is Online and False is Offline</i>
07	AppointmentDate	DateTime	Public	
08	ActualSlotStart	int	Public	<i>Id of the started slot</i>

### 3.5.2.3 FeatureCustomer

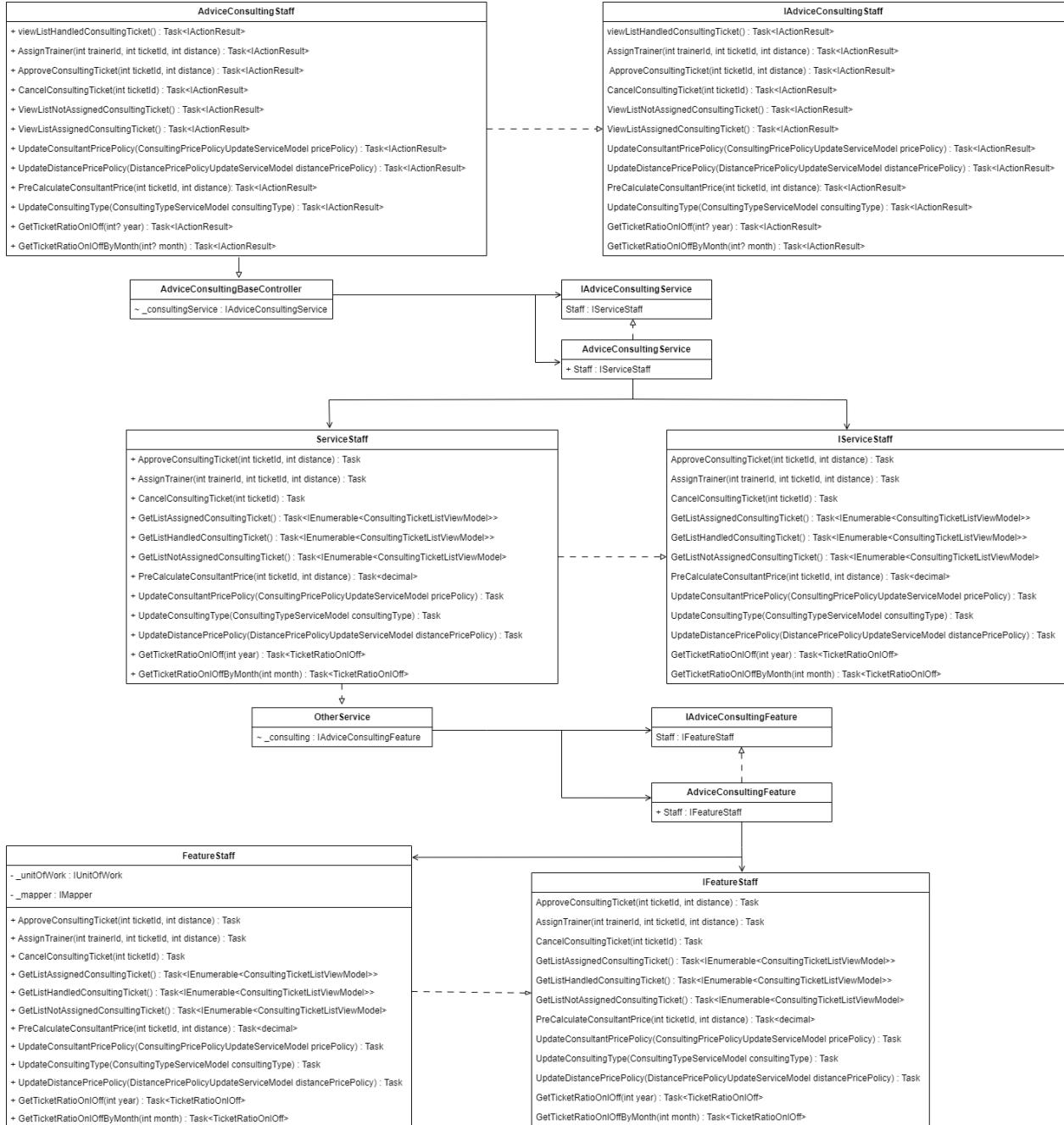
No	Method	Return Type	Visibility	Description
01	SendConsultingTicket	Task	Public	

### 3.5.3 Sequence Diagram



## 3.6 Confirm Consulting Ticket Feature

### 3.6.1 Class Diagram



### 3.6.2 Class Specification

#### 3.6.2.1 AdviceConsultingStaff

No	Method	Return Type	Visibility	Description
01	viewListHandledConsultingTicket	Task<ActionResult>	Public	API endpoint for listing handled consulting ticket
02	AssignTrainer	Task<ActionResult>	Public	API endpoint for assign trainer into ticket that still not have trainer
03	ApproveConsultingTicket	Task<ActionResult>	Public	API endpoint for approve the consulting ticket

04	CancelConsultingTicket	Task<ActionResult>	Public	<i>API endpoint for cancel the consulting ticket</i>
05	ViewListNotAssignedConsultingTicket	Task<ActionResult>	Public	<i>API endpoint for listing not assigned consulting ticket</i>
06	ViewListAssignedConsultingTicket	Task<ActionResult>	Public	<i>API endpoint for listing assigned consulting ticket</i>
07	UpdateConsultantPricePolicy	Task<ActionResult>	Public	<i>API endpoint for update consultant price policy</i>
08	UpdateDistancePricePolicy	Task<ActionResult>	Public	<i>API endpoint for update distance price policy</i>
09	PreCalculateConsultantPrice	Task<ActionResult>	Public	<i>API endpoint for calculate the ticket price after staff input the distance</i>
10	UpdateConsultingType	Task<ActionResult>	Public	<i>API endpoint for update consulting type</i>
11	GetTicketRatioOnOff	Task<ActionResult>	Public	<i>API endpoint for calculate the rate between online ticket and offline ticket in year</i>
12	GetTicketRatioOnOffByMonth	Task<ActionResult>	Public	<i>API endpoint for calculate the rate between online ticket and offline ticket by month</i>

#### 3.6.2.1.1 ConsultingPricePolicyUpdateServiceModel

No	Attribute	Datatype	Visibility	Description
01	Id	int	Public	
02	Price	decimal?	Public	

#### 3.6.2.1.2 DistancePricePolicyUpdateServiceModel

No	Attribute	Datatype	Visibility	Description
01	Id	int	Public	
02	PricePerKm	decimal	Public	

#### 3.6.2.1.3 ConsultingTypeServiceModel

No	Attribute	Datatype	Visibility	Description
01	Id	int	Public	
02	Name	String?	Public	

#### 3.6.2.2 ServiceStaff

No	Method	Return Type	Visibility	Description
01	ApproveConsultingTicket	Task	Public	
02	AssignTrainer	Task	Public	
03	CancelConsultingTicket	Task	Public	

04	GetListAssignedConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
05	GetListHandledConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
06	GetListNotAssignedConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
07	PreCalculateConsultantPrice	Task<decimal>	Public	
08	UpdateConsultantPricePolicy	Task	Public	
09	UpdateConsultingType	Task	Public	
10	UpdateDistancePricePolicy	Task	Public	
11	GetTicketRatioOnOff	Task<TicketRatioOnOff>	Public	
12	GetTicketRatioOnOffByMonth	Task<TicketRatioOnOff>	Public	

### 3.6.2.2.1 ConsultingTicketListViewModel

No	Attribute	Datatype	Visibility	Description
01	Id	int	Public	
02	OnlineOrOffline	bool	Public	
03	AppointmentDate	DateTime?	Public	
04	ActualSlotStart	string	Public	
05	ActualEndSlot	string?	Public	
06	Status	Models.Enum.ConsultingTicket.Status	Public	

### 3.6.2.2.2 TicketRatioOnOff

No	Attribute	Datatype	Visibility	Description
01	Online	List<Data>	Public	
02	Offline	List<Data>	Public	

### 3.6.2.2.3 Data

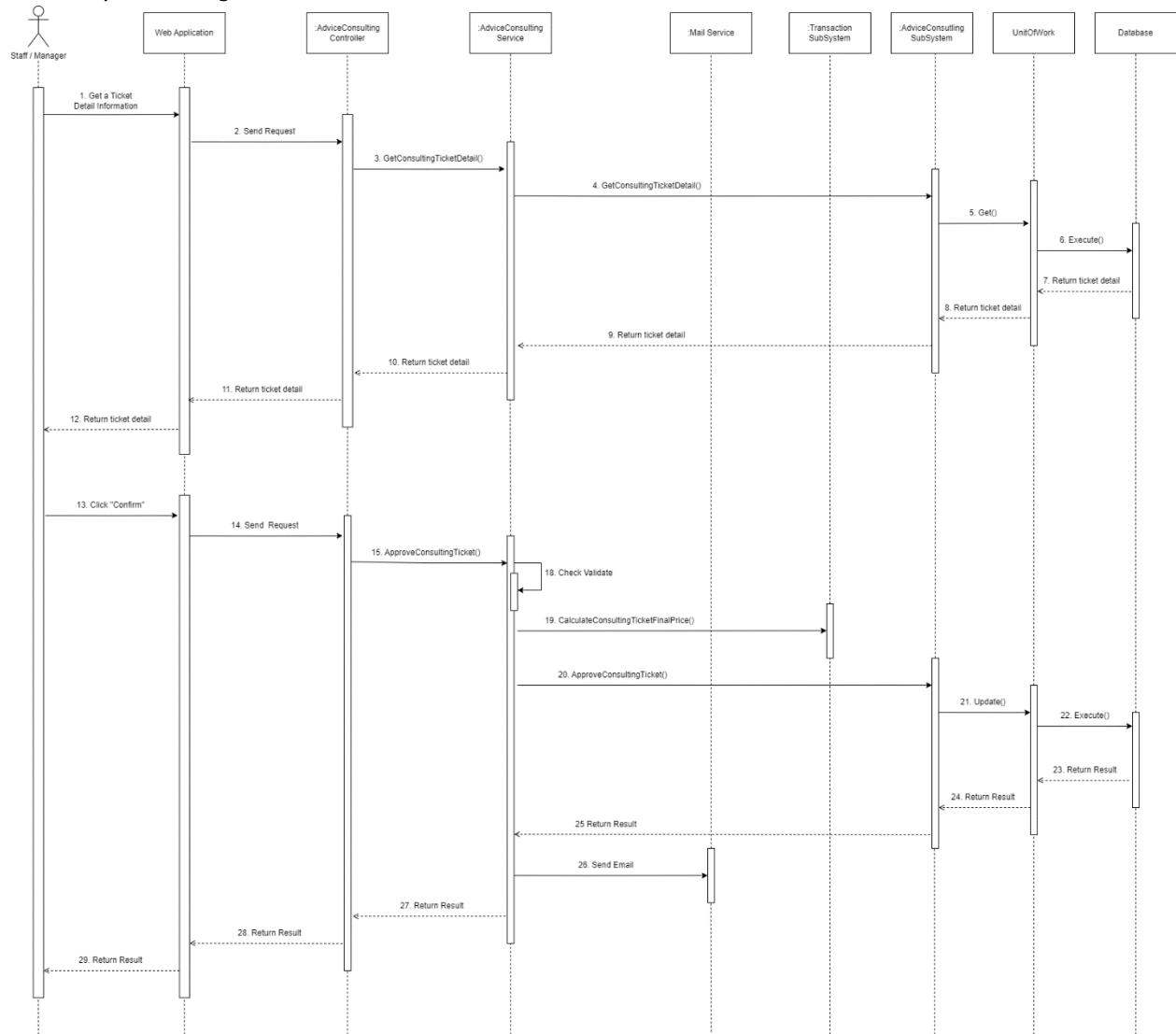
No	Attribute	Datatype	Visibility	Description
01	Label	Models.Enum.Month	Public	
02	Y	double	Public	

### 3.6.2.3 FeatureStaff

No	Method	Return Type	Visibility	Description
01	ApproveConsultingTicket	Task	Public	
02	AssignTrainer	Task	Public	
03	CancelConsultingTicket	Task	Public	
04	GetListAssignedConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	

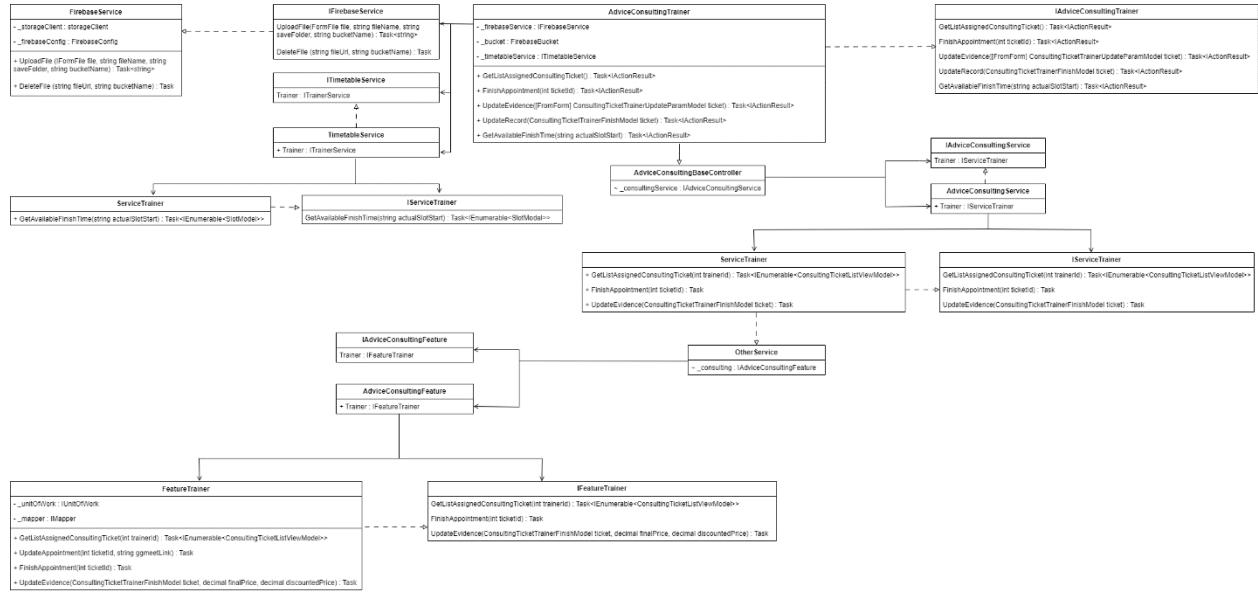
05	GetListHandledConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
06	GetListNotAssignedConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
07	PreCalculateConsultantPrice	Task<decimal>	Public	
08	UpdateConsultantPricePolicy	Task	Public	
09	UpdateConsultingType	Task	Public	
10	UpdateDistancePricePolicy	Task	Public	
11	GetTicketRatioOnOff	Task<TicketRatioOnOff>	Public	
12	GetTicketRatioOnOffByMonth	Task<TicketRatioOnOff>	Public	

### 3.6.3 Sequence Diagram



## 3.7 Submit Consulting Evidence Feature

### 3.7.1 Class Diagram



### 3.7.2 Class Specification

#### 3.7.2.1 AdviceConsultingTrainer

No	Method	Return Type	Visibility	Description
01	GetListAssignedConsultingTicket	Task<IActionResult>	Public	API endpoint for listing assigned consulting ticket
02	FinishAppointment	Task<IActionResult>	Public	API endpoint for finishing consulting ticket
03	UpdateEvidence	Task<IActionResult>	Public	API endpoint for updating evidence for offline consulting ticket
04	UpdateRecord	Task<IActionResult>	Public	API endpoint for updating record for online consulting ticket
05	GetAvailableFinishTime	Task<IActionResult>	Public	API endpoint for listing not assigned consulting ticket

#### 3.7.2.2 ServiceTrainer

No	Method	Return Type	Visibility	Description
01	GetListAssignedConsultingTicket	Task<Enumerable<ConsultingTicketListViewModel>>	Public	
02	FinishAppointment	Task	Public	
03	UpdateEvidence	Task	Public	

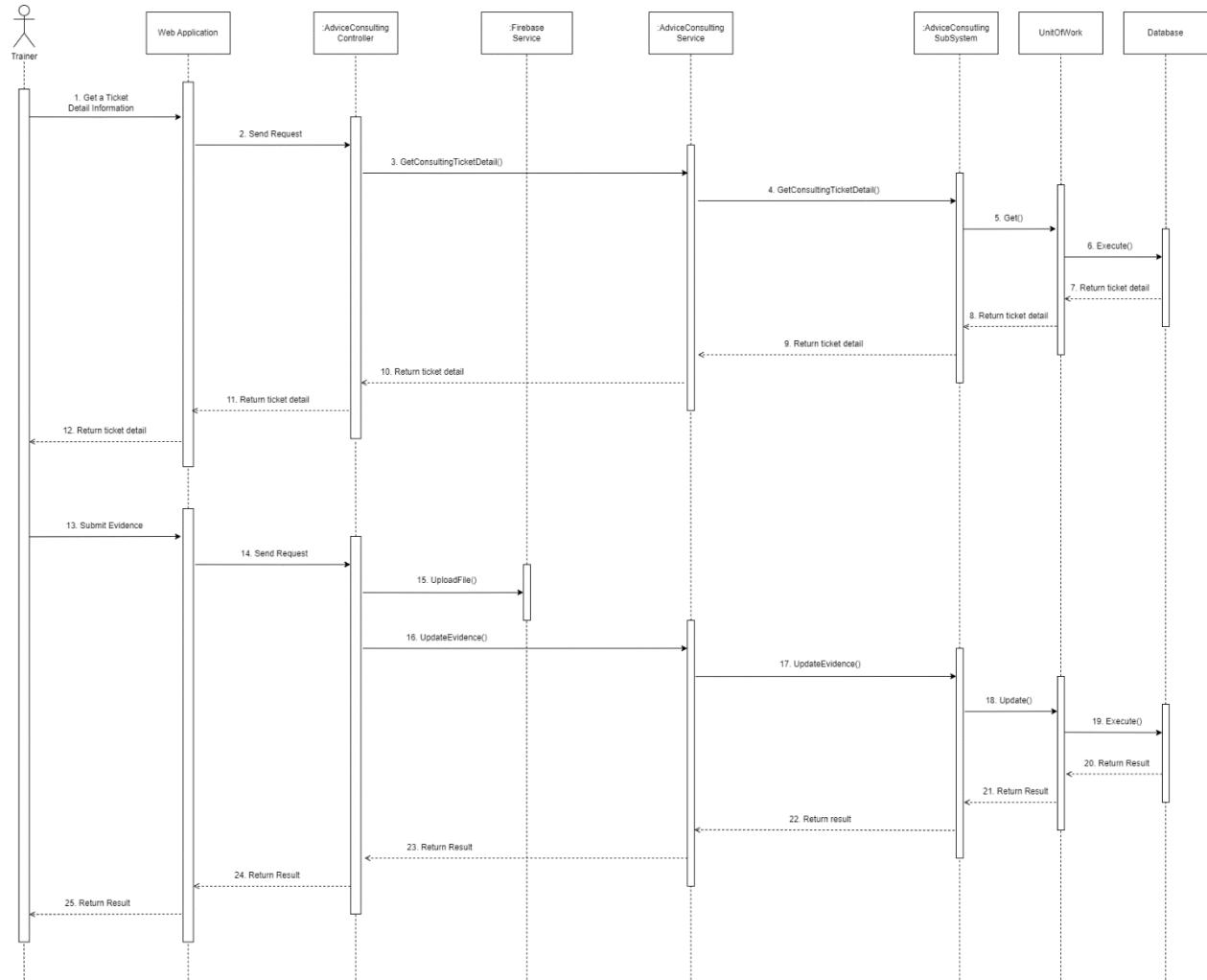
##### 3.7.2.2.1 ConsultingTicketTrainerFinishModel

No	Attribute	Datatype	Visibility	Description
01	Id	int	Public	
02	ActualSlotStart	string	Public	
03	ActualEndSlot	int	Public	
04	Evidence	string?	Public	

### 3.7.2.3 FeatureTrainer

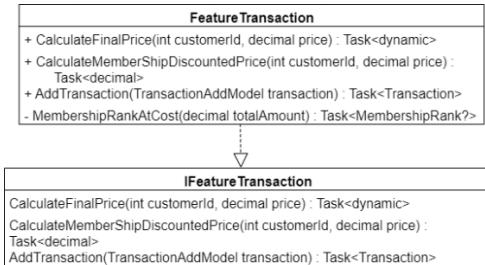
No	Method	Return Type	Visibility	Description
01	GetListAssignedConsultingTicket	Task<IEnumerable<ConsultingTicketListViewModel>>	Public	
02	FinishAppointment	Task	Public	
03	UpdateEvidence	Task	Public	

### 3.7.3 Sequence Diagram



## 3.8 Payment Feature

### 3.8.1 Class Diagram



### 3.8.2 Class Specification

#### 3.8.2.1 IFeatureTransaction

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CalculateFinalPrice	Task<dynamic>	Public	Parameters: + customerId: int + price: decimal
02	CalculateMemberShipDiscountedPrice	Task<decimal>	Public	Parameters: + customerId: int + price: decimal
03	AddTransaction	Task<Transaction>	Public	Parameters: + transaction: TransactionAddModel (3.8.2.2) Return: + Transaction (3.8.2.3)

#### 3.8.2.2 TransactionAddModel

No	Attribute	Data Type	Visibility	Description
01	CustomerId	int	Public	
02	Title	string?	Public	
03	Detail	string?	Public	
04	EntityTypeid	int?	Public	Enum values referred to table Transaction, EntityTypeId field in <a href="#">2.2</a>
05	EntityId	int?	Public	Enum values referred to table Transaction, EntityId field in <a href="#">2.2</a>
06	TotalPayment	decimal?	Public	
07	PaymentCode	string?	Public	
08	Status	int	Public	Enum values referred to table Transaction, status field in <a href="#">2.2</a>

No	Method	Return Type	Visibility	Description

### 3.8.2.3 Transaction

- This is entity model referring to table Transaction in [2.2](#).

### 3.8.2.4 FeatureTransaction

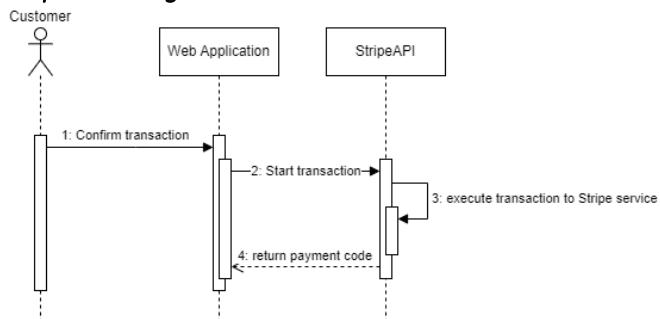
No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CalculateFinalPrice</i>	<i>Task&lt;dynamic&gt;</i>	<i>Public</i>	<i>Parameters:</i> + customerId: int + price: decimal
02	<i>CalculateMembershipDiscountedPrice</i>	<i>Task&lt;decimal&gt;</i>	<i>Public</i>	<i>Parameters:</i> + customerId: int + price: decimal
03	<i>AddTransaction</i>	<i>Task&lt;Transaction&gt;</i>	<i>Public</i>	<i>Parameters:</i> + transaction: <i>TransactionAddModel</i> (3.8.2.2) <i>Return:</i> + Transaction (3.8.2.3)
04	<i>MembershipRankAtCost</i>	<i>Task&lt;MembershipRank ?&gt;</i>	<i>Private</i>	<i>Parameters:</i> + totalAmount: decimal <i>Return:</i> + MembershipRank (3.8.2.5)

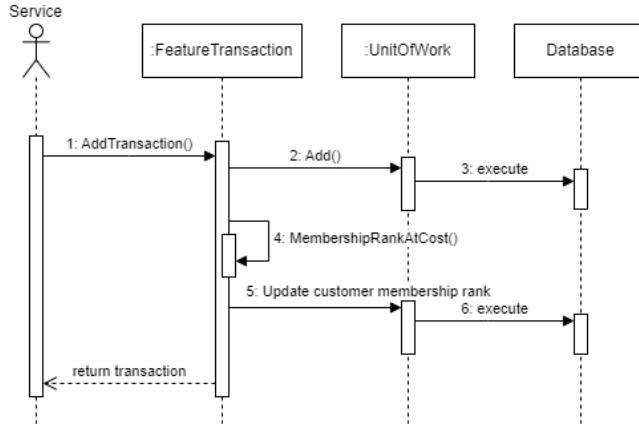
### 3.8.2.5 MembershipRank

- This is entity model referred to table MembershipRank in [in 2.2](#).

### 3.8.3 Stripe Integrating Sequence Diagram

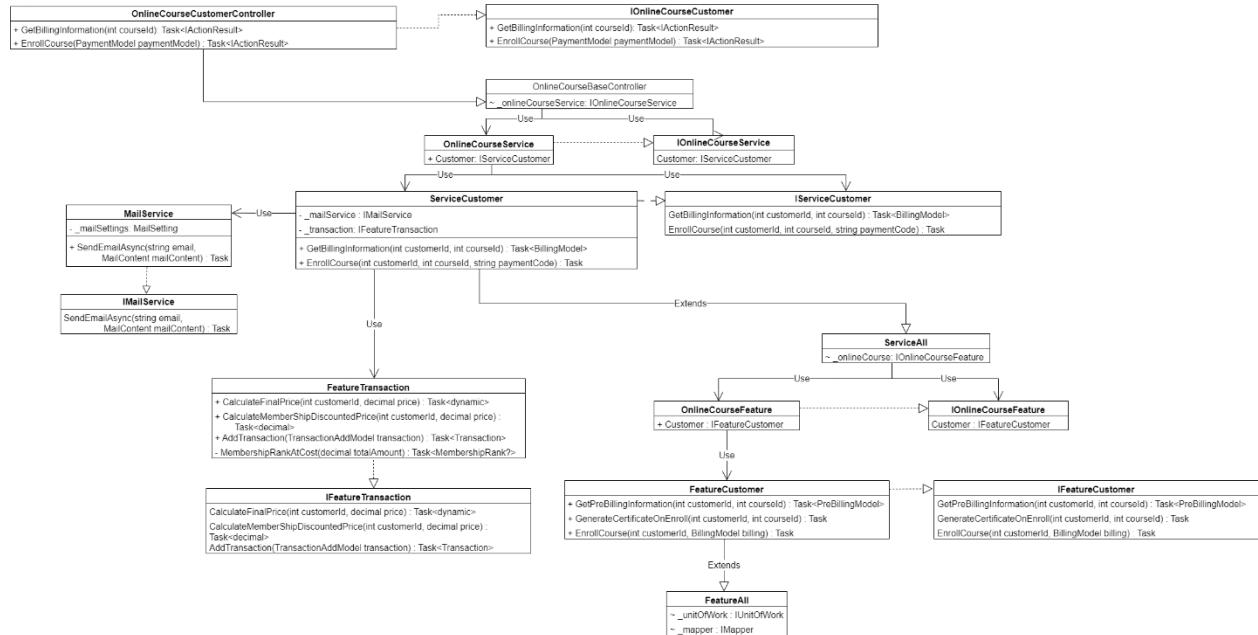


### 3.8.4 Add Transaction Sequence Diagram



## 3.9 Enroll Online Course Feature

### 3.9.1 Class Diagram



### 3.9.2 Class Specification

#### 3.9.2.1 IOnlineCourseCustomer

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetBillingInformation	Task<IActionResult>	Public	Parameters: + courseId: int
02	EnrollCourse	Task<IActionResult>	Public	Parameters: + paymentModel: PaymentModel (3.9.2.2)

#### 3.9.2.2 PaymentModel

No	Attribute	Data Type	Visibility	Description
01	CourseId	int	Public	
02	PaymentCode	string	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.3 OnlineCourseCustomerController

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetBillingInformation	Task<ActionResult>	Public	Parameters: + courseId: int
02	EnrollCourse	Task<ActionResult>	Public	Parameters: + paymentModel: PaymentModel (3.9.2.2)

### 3.9.2.4 OnlineCourseBaseController

No	Attribute	Data Type	Visibility	Description
01	_onlineCourseService	IOnlineCourseService	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.5 IOnlineCourseService

No	Attribute	Data Type	Visibility	Description
01	Customer	IServiceCustomer	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.6 OnlineCourseService

No	Attribute	Data Type	Visibility	Description
01	Customer	IServiceCustomer	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.7 IServiceCustomer

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetBillingInformation	Task<BillingModel>	Public	Parameters: + customerId: int + courseId: int Return: + BillingModel (3.9.2.8)
02	EnrollCourse	Task	Public	Parameters: + customerId: int + courseId: int + paymentCode: string

### 3.9.2.8 BillingModel

No	Attribute	Data Type	Visibility	Description
01	Courseld	int	Public	
02	Email	string	Public	
03	CourseTitle	string	Public	
04	CoursePrice	decimal	Public	
05	DiscountedPrice	decimal	Public	
06	TotalPrice	decimal	Public	
07	DiscountRate	float	Public	
08	MembershipName	string	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.9 ServiceCustomer

No	Attribute	Data Type	Visibility	Description
01	_mailService	IMailService	Public	
02	_transaction	IFeatureTransaction	Public	

No	Method	Return Type	Visibility	Description
01	GetBillingInformation	Task<BillingModel>	Public	Parameters: + customerId: int + courseld: int Return: + BillingModel (3.9.2.8)
02	EnrollCourse	Task	Public	Parameters: + customerId: int + courseld: int + paymentCode: string
01	GetBillingInformation	Task<BillingModel>	Public	Parameters: + customerId: int + courseld: int Return: + BillingModel (3.9.2.8)
02	EnrollCourse	Task	Public	Parameters: + customerId: int + courseld: int + paymentCode: string

### 3.9.2.10 IMailService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	SendEmailAsync	Task	Public	Parameters: + email: string + mailContent: MailContent (3.9.2.11)

### 3.9.2.11 MailContent

No	Attribute	Data Type	Visibility	Description

01	<i>Subject</i>	<i>string</i>	<i>Public</i>	
02	<i>HtmlMessage</i>	<i>string</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.9.2.12 MailService

No	Attribute	Data Type	Visibility	Description
01	<i>_mailSettings</i>	<i>MailSetting</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description
01	<i>SendEmailAsync</i>	<i>Task</i>	<i>Public</i>	Parameters: + email: <i>string</i> + mailContent: <i>MailContent</i> (3.9.2.11)

### 3.9.2.13 MailSetting

No	Attribute	Data Type	Visibility	Description
01	<i>Mail</i>	<i>string</i>	<i>Public</i>	
02	<i>DisplayName</i>	<i>string</i>	<i>Public</i>	
03	<i>Password</i>	<i>string</i>	<i>Public</i>	
04	<i>Host</i>	<i>string</i>	<i>Public</i>	
05	<i>Port</i>	<i>int</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description
01	<i>SendEmailAsync</i>	<i>Task</i>	<i>Public</i>	Parameters: + email: <i>string</i> + mailContent: <i>MailContent</i> (3.9.2.11)

### 3.9.2.14 IFeatureTransaction

- Referring to [3.8.2.1](#)

### 3.9.2.15 FeatureTransaction

- Referred to [3.8.2.4](#)

### 3.9.2.16 ServiceAll

No	Attribute	Data Type	Visibility	Description
01	<i>_onlineCourse</i>	<i>IOnlineCourseFeature</i>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.9.2.17 IOnlineCourseFeature

No	Attribute	Data Type	Visibility	Description
01	<i>Customer</i>	<i>IFeatureCustomer</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.9.2.18 OnlineCourseFeature

No	Attribute	Data Type	Visibility	Description
01	<i>Customer</i>	<i>IFeatureCustomer</i>	Public	

No	Method	Return Type	Visibility	Description

### 3.9.2.19 IFeatureCustomer

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetPreBillingInformation	Task<PreBillingModel>	Public	Parameters: + customerId: int + courseId: int Return: + PreBillingModel (3.9.2.20)
02	GenerateCertificateOnEnroll	Task	Public	Parameters: + customerId: int + courseId: int
03	EnrollCourse	Task	Public	Parameters: + customerId: int + billing: BillingModel (3.9.2.8)

### 3.9.2.20 PreBillingModel

No	Attribute	Data Type	Visibility	Description
01	<i>Email</i>	<i>string</i>	<i>Public</i>	
02	<i>CourseTitle</i>	<i>string</i>	<i>Public</i>	
03	<i>MembershipName</i>	<i>string</i>	<i>Public</i>	
04	<i>DiscountPercent</i>	<i>float</i>	<i>Public</i>	
05	<i>CoursePrice</i>	<i>decimal</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.9.2.21 FeatureCustomer

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetPreBillingInformation	Task<PreBillingModel>	Public	Parameters: + customerId: int + courseId: int Return: + PreBillingModel (3.9.2.20)
02	GenerateCertificateOnEnroll	Task	Public	Parameters: + customerId: int

				+ courseId: int
03	EnrollCourse	Task	Public	Parameters: + customerId: int + billing: BillingModel (3.9.2.8)

### 3.9.2.22 FeatureAll

No	Attribute	Data Type	Visibility	Description
01	<code>_unitOfWork</code>	<code>IUnitOfWork</code>	<i>Internal</i>	
02	<code>_mapper</code>	<code>IMapper</code>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.9.3 Sequence Diagram

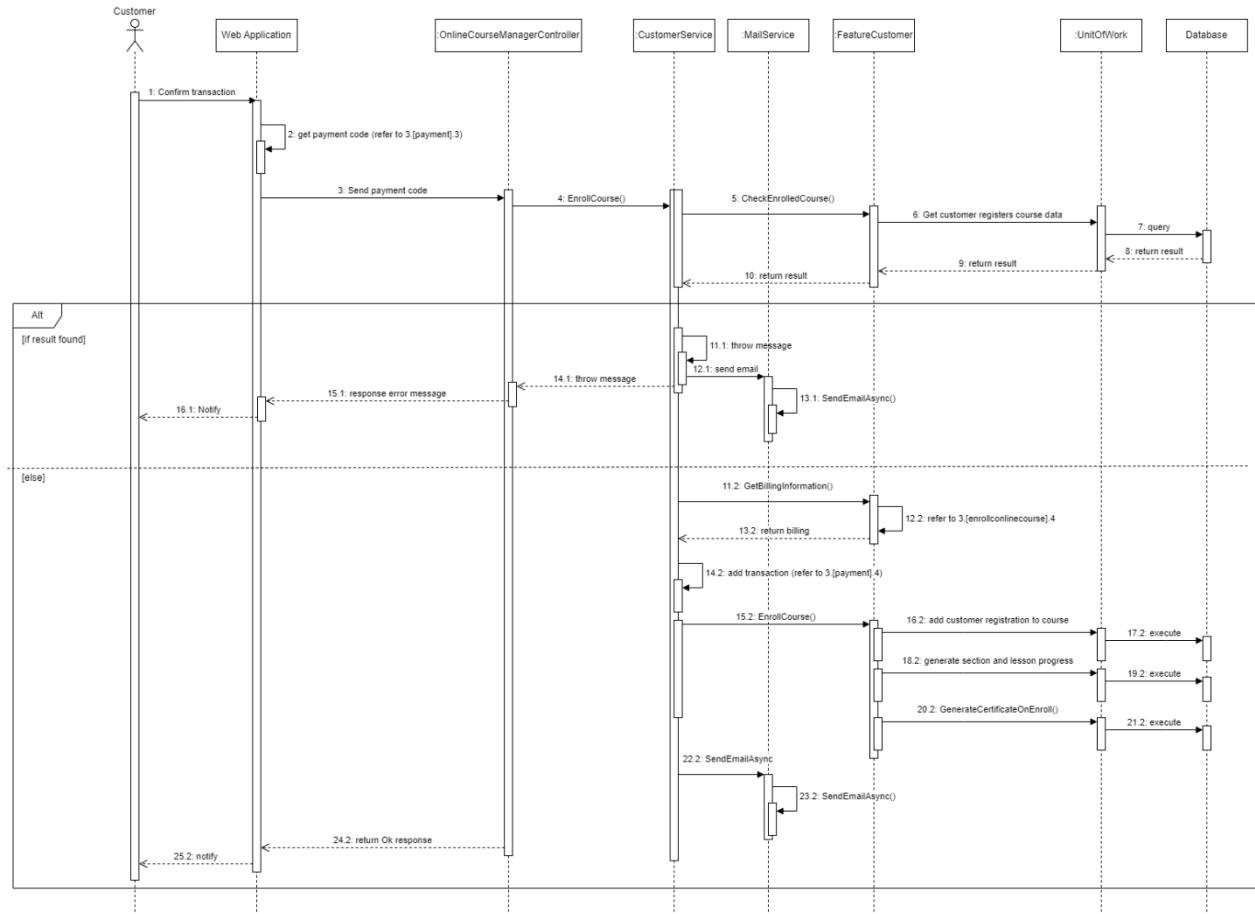
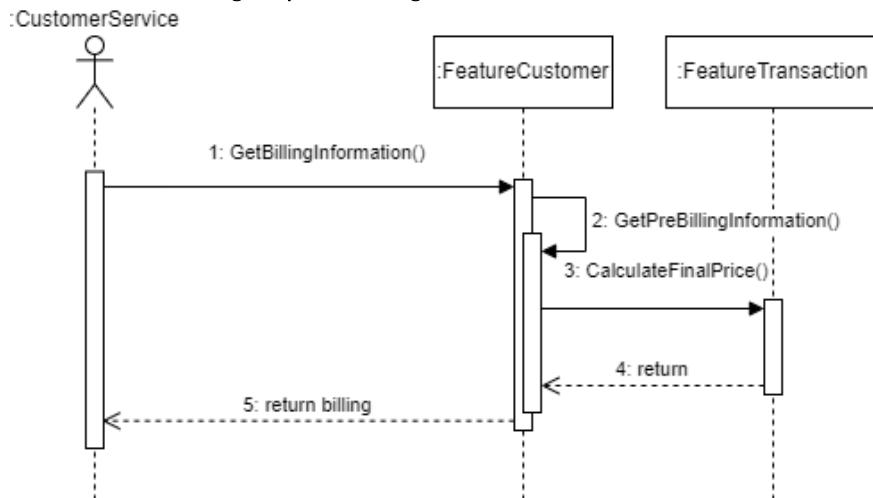


Figure 3.9.3 Create Online Course Sequence Diagram

### 3.9.4 Get Online Course Get Billing Sequence Diagram



## 3.10 Workshop Management Feature

### 3.10.1 Class Diagram

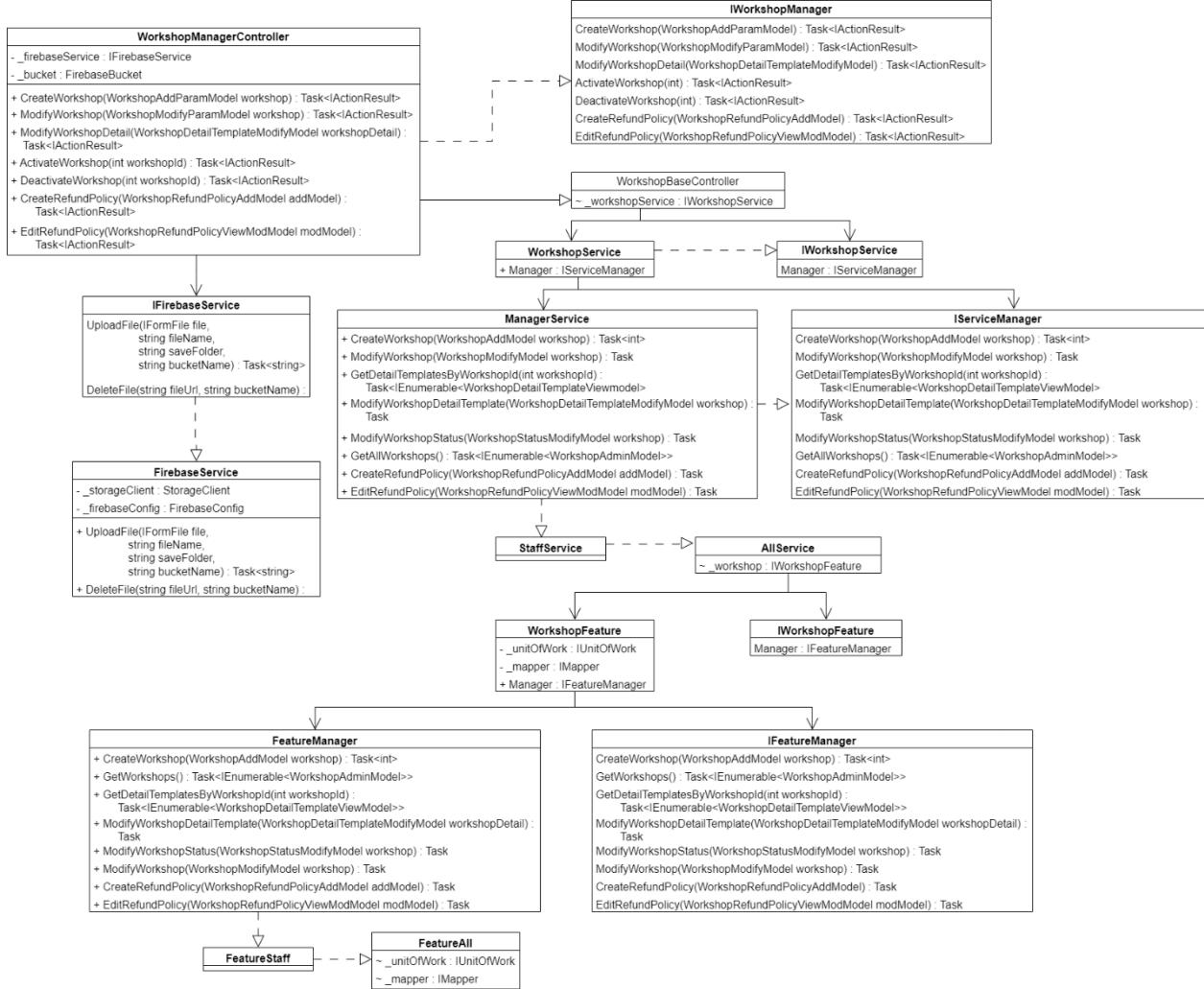


Figure 3.10.1 Workshop Management Feature Class Diagram

### 3.10.2 Class Specification

#### 3.10.2.1 WorkshopManagementController

No	Attribute	Data Type	Visibility	Description
01	_firebaseService	IFirebaseService	Private	Dependency Injection
02	_bucket	FirebaseBucket	Private	Firebase bucket's configuration model

No	Method	Return Type	Visibility	Description
01	CreateWorkshop	Task<ActionResult>	Public	API endpoint for creating workshop
02	ModifyWorkshop	Task<ActionResult>	Public	API endpoint for modifying workshop
03	ModifyWorkshopDetailTemplate	Task<ActionResult>	Public	API endpoint for modifying workshop template detail
04	ActivateWorkshop	Task<ActionResult>	Public	API endpoint for activating workshop

05	<i>DeactivateWorkshop</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for deactivating workshop</i>
06	<i>CreateRefundPolicy</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for creating new refund policy in workshop service</i>
07	<i>EditRefundPolicy</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for modifying refund policy in workshop service</i>

### 3.10.2.2 FirebaseBucket

No	Attribute	Data Type	Visibility	Description
01	<i>General</i>	<i>string</i>	<i>Public</i>	<i>Predefined bucket in firebase storage</i>

No	Method	Return Type	Visibility	Description

### 3.10.2.3 WorkshopAddParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>Title</i>	<i>string</i>	<i>Public</i>	
02	<i>Description</i>	<i>string</i>	<i>Public</i>	
03	<i>Pictures</i>	<i>List&lt;IFormFile&gt;</i>	<i>Public</i>	<i>Uploaded images for workshop</i>
04	<i>RegisterEnd</i>	<i>int</i>	<i>Public</i>	<i>Must be positive</i>
05	<i>Price</i>	<i>decimal</i>	<i>Public</i>	<i>Must be positive</i>
06	<i>TotalSlot</i>	<i>int</i>	<i>Public</i>	<i>Must be positive</i>
07	<i>MinimumRegistration</i>	<i>int</i>	<i>Public</i>	<i>Must be positive</i>
08	<i>MaximumRegistration</i>	<i>int</i>	<i>Public</i>	<i>Must be positive</i>

No	Method	Return Type	Visibility	Description
01	<i>ToWorkshopAddModel</i>	<i>WorkshopAddModel</i>	<i>Public</i>	<i>Converting function</i>

### 3.10.2.4 WorkshopModifyParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	<i>Workshop's id</i>
02	<i>Title</i>	<i>string?</i>	<i>Public</i>	
03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>Pictures</i>	<i>List&lt;IFormFile&gt;?</i>	<i>Public</i>	<i>Uploaded images for workshop</i>
05	<i>RegisterEnd</i>	<i>int?</i>	<i>Public</i>	<i>Must be positive if exists</i>
06	<i>Price</i>	<i>decimal?</i>	<i>Public</i>	<i>Must be positive if exists</i>
07	<i>TotalSlot</i>	<i>int?</i>	<i>Public</i>	<i>Must be positive if exists</i>
08	<i>MinimumRegistration</i>	<i>int?</i>	<i>Public</i>	<i>Must be positive if exists</i>
09	<i>MaximumRegistration</i>	<i>int?</i>	<i>Public</i>	<i>Must be positive if exists</i>

No	Method	Return Type	Visibility	Description
01	<i>ToWorkshopModifyModel</i>	<i>WorkshopModifyModel</i>	<i>Public</i>	<i>Converting function</i>

### 3.10.2.5 WorkshopDetailTemplateModifyModel

No	Attribute	Data Type	Visibility	Description

01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Detail</i>	<i>string</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

#### 3.10.2.6 WorkshopRefundPolicyAddModel

No	Attribute	Data Type	Visibility	Description
01	TotalDayBeforeStart	<i>int</i>	<i>Public</i>	
02	RefundRate	<i>float?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

#### 3.10.2.7 WorkshopRefundPolicyViewModModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	TotalDayBeforeStart	<i>int</i>	<i>Public</i>	
03	RefundRate	<i>float?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

#### 3.10.2.8 IWorkshopManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshop</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for creating workshop</i>
02	<i>ModifyWorkshop</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for modifying workshop</i>
03	<i>ModifyWorkshopDetailTemplate</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for modifying workshop template detail</i>
04	<i>ActivateWorkshop</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for activating workshop</i>
05	<i>DeactivateWorkshop</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for deactivating workshop</i>
06	<i>CreateRefundPolicy</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for creating new refund policy in workshop service</i>
07	<i>EditRefundPolicy</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	<i>API endpoint for modifying refund policy in workshop service</i>

#### 3.10.2.9 WorkshopBaseController

No	Attribute	Data Type	Visibility	Description
01	<i>_workshopService</i>	<i>IWorkshopService</i>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.10.2.10 IWorkshopService

No	Attribute	Data Type	Visibility	Description
01	Manager	<i>IServiceManager</i>	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.11 IWorkshopService

No	Attribute	Data Type	Visibility	Description
01	Manager	<i>IServiceManager</i>	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.12 IServiceManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshop</i>	<i>Task&lt;int&gt;</i>	Public	Parameters: + workshop : <i>WorkshopAddModel</i> (3.10.2.13)
02	<i>ModifyWorkshop</i>	<i>Task</i>	Public	Parameters: + workshop : <i>WorkshopModifyModel</i> (3.10.2.14)
03	<i>GetDetailTemplatesByWorkshopId</i>	<i>Task&lt;IEnumerable&lt;WorkshopDetailTemplateViewModel&gt;&gt;</i>	Public	Parameters: + workshopId : int Return: + <i>WorkshopDetailTemplateViewModel</i> (3.10.2.15)
04	<i>ModifyWorkshopDetailTemplate</i>	<i>Task</i>	Public	Parameters: + workshop : <i>WorkshopDetailTemplateModifyModel</i> (3.10.2.5)
05	<i>ModifyWorkshopStatus</i>	<i>Task</i>	Public	Parameters: + workshop : <i>WorkshopStatusModifyModel</i> (3.10.2.16)
06	<i> GetAllWorkshops</i>	<i>Task&lt;IEnumerable&lt;WorkshopAdminModel&gt;&gt;</i>	Public	Return: + <i>WorkshopAdminModel</i> (3.10.17)
07	<i>CreateRefundPolicy</i>	<i>Task</i>	Public	Parameters:

				+ addModel : WorkshopRefundPolicyAdd Model (3.10.2.6)
08	EditRefundPolicy	Task	Public	Parameters: + modModel : WorkshopRefundPolicyVie wModel (3.10.2.7)

### 3.10.2.13 WorkshopAddModel

No	Attribute	Data Type	Visibility	Description
01	Title	string	Public	
02	Description	string	Public	
03	Picture	string	Public	
04	MinimumRegistration	int	Public	
05	MaximumRegistration	int	Public	
06	RegisterEnd	int?	Public	
07	Price	decimal	Public	
08	TotalSlot	int	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.14 WorkshopModifyModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	MinimumRegistration	int?	Public	
03	MaximumRegistration	int?	Public	
04	Title	string?	Public	
05	Description	string?	Public	
06	Picture	string?	Public	
07	RegisterEnd	int?	Public	
08	Price	decimal?	Public	
09	TotalSlot	int?	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.15 WorkshopDetailTemplateViewModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	Detail	string	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.16 WorkshopStatusModifyModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	Status	int	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.17 WorkshopAdminModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	MinimumRegistration	int	Public	
03	MaximumRegistration	int	Public	
04	Title	string	Public	
05	Description	string	Public	
06	Picture	string?	Public	
07	RegisterEnd	int?	Public	
08	Price	decimal	Public	
09	TotalSlot	int	Public	
10	Status	Status	Public	Enum: + Active = 1 + Inactive = 0

No	Method	Return Type	Visibility	Description

### 3.10.2.18 ManagerService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CreateWorkshop	Task<int>	Public	Parameters: + workshop: WorkshopAddModel (3.10.2.13)
02	ModifyWorkshop	Task	Public	Parameters: + workshop: WorkshopModifyModel (3.10.2.14)
03	GetDetailTemplatesByWorkshopId	Task<IEnumerable<WorkshopDetailTemplateViewModel>>	Public	Parameters: + workshopId: int Return: + WorkshopDetailTemplateViewModel (3.10.2.15)
04	ModifyWorkshopDetailTemplate	Task	Public	Parameters: + workshop: WorkshopDetailTemplateModifyModel (3.10.2.5)
05	ModifyWorkshopStatus	Task	Public	Parameters: + workshop: WorkshopStatusModifyModel (3.10.2.16)
06	GetAllWorkshops	Task<IEnumerable<WorkshopAdminModel>>	Public	Return: + WorkshopAdminModel (3.10.2.17)

07	<i>CreateRefundPolicy</i>	Task	Public	Parameters: + addModel: <i>WorkshopRefundPolicyAddModel</i> (3.10.2.6)
08	<i>EditRefundPolicy</i>	Task	Public	Parameters: + modModel: <i>WorkshopRefundPolicyViewModModel</i> (3.10.2.7)

### 3.10.2.19 StaffService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.10.2.20 AllService

No	Attribute	Data Type	Visibility	Description
01	_workshop	<i>IWorkshopFeature</i>	Internal	

No	Method	Return Type	Visibility	Description

### 3.10.2.21 IWorkshopFeature

No	Attribute	Data Type	Visibility	Description
01	Manager	<i>IFeatureManager</i>	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.22 WorkshopFeature

No	Attribute	Data Type	Visibility	Description
01	_unitOfWork	<i>IUnitOfWork</i>	Private	
02	_mapper	<i>IMapper</i>	Private	
03	Manager	<i>IFeatureManager</i>	Public	

No	Method	Return Type	Visibility	Description

### 3.10.2.23 FeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshop</i>	<i>Task&lt;int&gt;</i>	Public	Parameters: + workshop: <i>WorkshopAddModel</i> (3.10.2.13)
02	<i>GetWorkshops</i>	<i>Task&lt;IEnumerable&lt;WorkshopAdminModel&gt;&gt;</i>	Public	Return: <i>WorkshopAdminModel</i> (3.10.2.17)

03	<i>GetDetailTemplatesByWorkshopId</i>	<i>Task&lt;IEnumerable&lt;WorkshopDetailTemplateViewModel&gt;&gt;</i>	<i>Public</i>	<i>Parameters:</i> + workshopId: int <i>Return:</i> + <i>WorkshopDetailTemplateViewModel (3.10.2.15)</i>
04	<i>ModifyWorkshopDetailTemplate</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + workshopDetail: <i>WorkshopDetailTemplateModifyModel (3.10.2.5)</i>
05	<i>ModifyWorkshopStatus</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + workshop: <i>WorkshopStatusModifyModel (3.10.2.16)</i>
06	<i>ModifyWorkshop</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + workshop: <i>WorkshopModifyModel (3.10.2.14)</i>
07	<i>CreateRefundPolicy</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + addModel: <i>WorkshopRefundPolicyAddModel (3.10.2.6)</i>
08	<i>EditRefundPolicy</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + modModel: <i>WorkshopRefundPolicyViewModModel (3.10.2.7)</i>

#### 3.10.2.24 IFeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshop</i>	<i>Task&lt;int&gt;</i>	<i>Public</i>	<i>Parameters:</i> + workshop: <i>WorkshopAddModel (3.10.2.13)</i>
02	<i>GetWorkshops</i>	<i>Task&lt;IEnumerable&lt;WorkshopAdminModel&gt;&gt;</i>	<i>Public</i>	<i>Return:</i> <i>WorkshopAdminModel (3.10.2.17)</i>
03	<i>GetDetailTemplatesByWorkshopId</i>	<i>Task&lt;IEnumerable&lt;WorkshopDetailTemplateViewModel&gt;&gt;</i>	<i>Public</i>	<i>Parameters:</i> + workshopId: int <i>Return:</i> + <i>WorkshopDetailTemplateViewModel (3.10.2.15)</i>
04	<i>ModifyWorkshopDetailTemplate</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + workshopDetail: <i>WorkshopDetailTemplateModifyModel (3.10.2.5)</i>
05	<i>ModifyWorkshopStatus</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i>

				+ workshop: WorkshopStatusModifyModel (3.10.2.16)
06	ModifyWorkshop	Task	Public	Parameters: + workshop: WorkshopModifyModel (3.10.2.14)
07	CreateRefundPolicy	Task	Public	Parameters: + addModel: WorkshopRefundPolicyAddModel (3.10.2.6)
08	EditRefundPolicy	Task	Public	Parameters: + modModel: WorkshopRefundPolicyViewModModel (3.10.2.7)

### 3.10.2.25 FeatureStaff

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.10.2.25 FeatureAll

No	Attribute	Data Type	Visibility	Description
01	_unitOfWork	IUnitOfWork	Internal	
02	_mapper	IMapper	Internal	

No	Method	Return Type	Visibility	Description

### 3.10.2.26 IFirebaseService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	UploadFile	Task<string>	Public	Parameters: + file: IFormFile + filename: string + saveFolder: string + bucketName: string
02	DeleteFile	Task<bool>	Public	Parameters: + urlString: string + bucketName: string

### 3.10.2.27 FirebaseService

No	Attribute	Data Type	Visibility	Description
01	_firebaseConfig	FirebaseConfig	Private	FirebaseConfig (3.10.29)
02	_storageClient	StorageClient	Private	StorageClient(3.10.28)

No	Method	Return Type	Visibility	Description

01	<i>UploadFile</i>	<i>Task&lt;string&gt;</i>	<i>Public</i>	<b>Parameters:</b> + file: <i>IFormFile</i> + filename: <i>string</i> + saveFolder: <i>string</i> + bucketName: <i>string</i>
----	-------------------	---------------------------	---------------	---

### 3.10.2.28 StorageClient

- This model is imported using Google.Cloud.Storage.V1 library.

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.10.2.29 FirebaseConfig

No	Attribute	Data Type	Visibility	Description
01	<i>Storage</i>	<i>string</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.10.3 Create Workshop Sequence Diagram

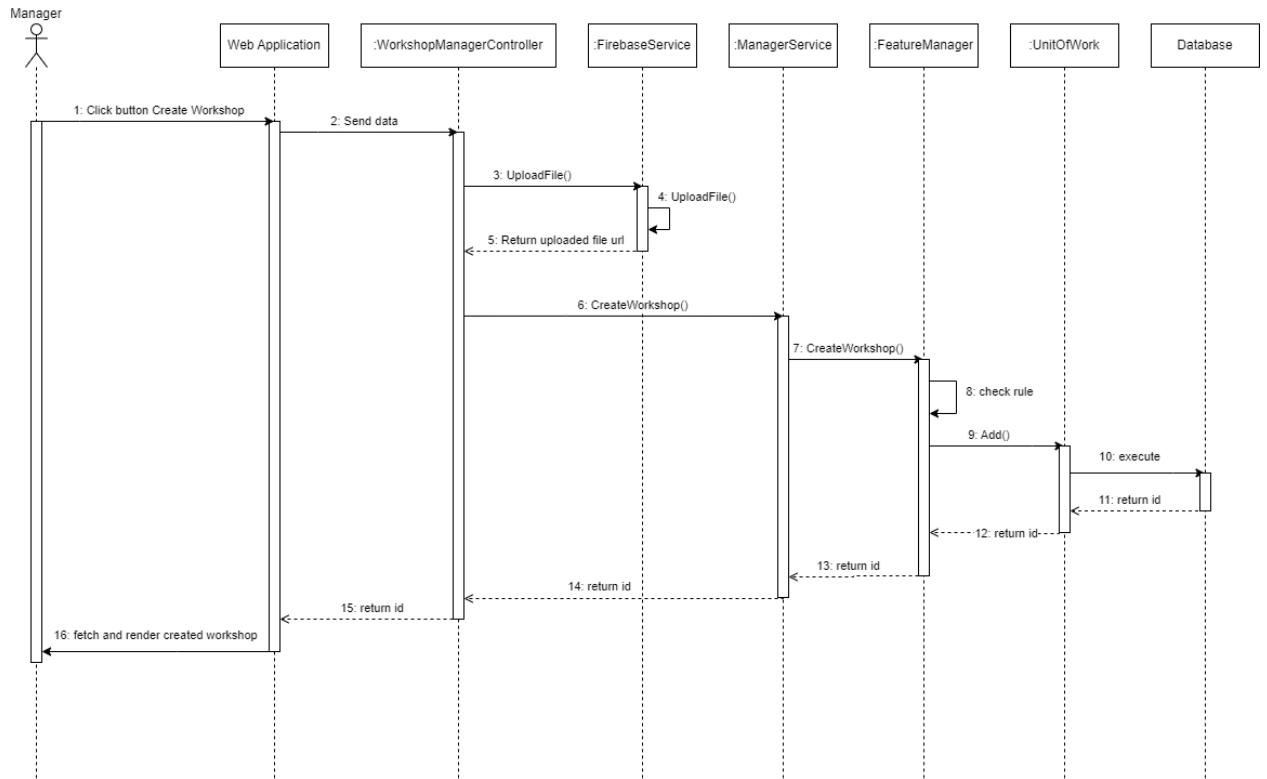


Figure 3.10.3 Create Workshop Sequence Diagram

### 3.10.4 Modify Workshop Sequence Diagram

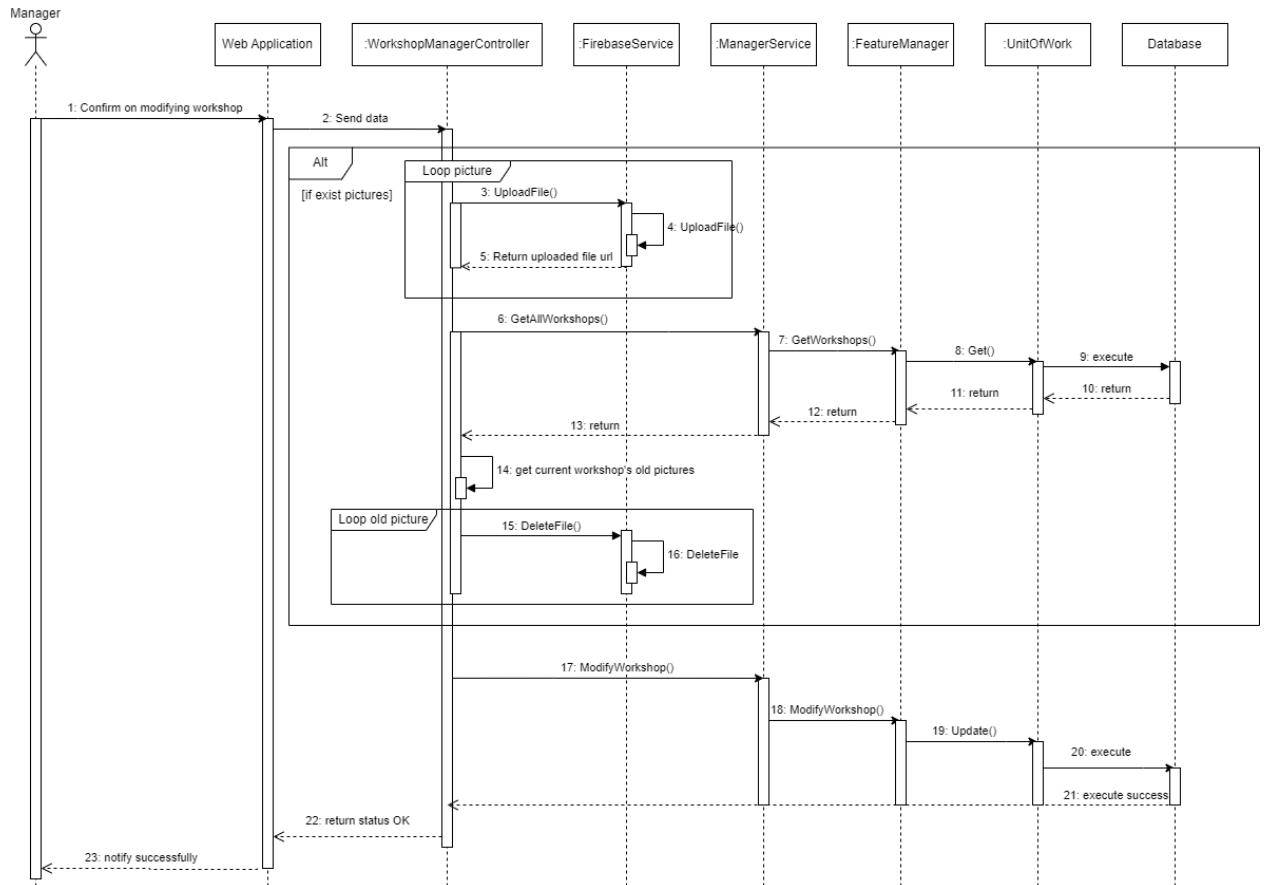


Figure 3.10.4 Modify Workshop Sequence Diagram

## 3.11 Workshop Class Management Feature

### 3.11.1 Class Diagram

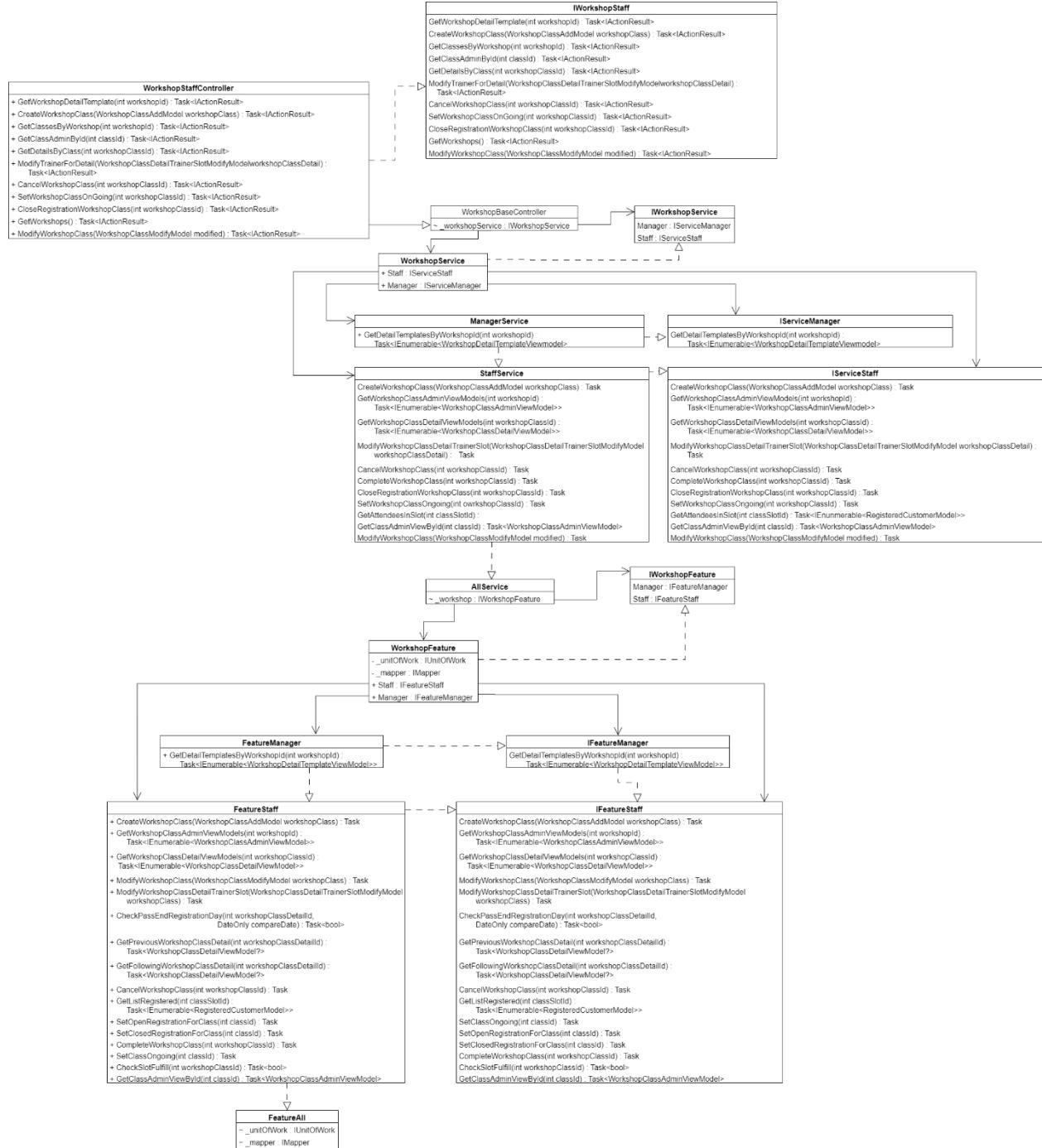


Figure 3.11.1 Workshop Class Management Feature Class Diagram

### 3.11.2 Class Specification

#### 3. [classmanagement].2.1 IWorkshopStaff

No	Attribute	Data Type	Visibility	Description
----	-----------	-----------	------------	-------------

No	Method	Return Type	Visibility	Description
01	GetWorkshopDetailTemplate	Task<ActionResult>	Public	Parameters: + workshopId: int

02	<i>CreateWorkshopClass</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClass:</i> <i>WorkshopClassAddModel</i> (3.11.2.2)
03	<i>GetClassesByWorkshop</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopId: int</i>
04	<i>GetClassAdminById</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>classId: int</i>
05	<i>GetDetailsByClass</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>
06	<i>ModifyTrainerForDetail</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassDetail:</i> <i>WorkshopClassDetailTrainerSlotModifyModel</i> (3.11.2.14)
07	<i>CancelWorkshopClass</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>
08	<i>SetWorkshopClassOnGoing</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>
09	<i>CloseRegistrationWorkshopClass</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>
10	<i>GetWorkshops</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	
11	<i>ModifyWorkshopClass</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>modified:</i> <i>WorkshopClassModifyModel</i> (3.11.2.4)

### 3.11.2.2 WorkshopClassAddModel

No	Attribute	Data Type	Visibility	Description
01	<i>WorkshopId</i>	<i>int</i>	<i>Public</i>	
02	<i>StartTime</i>	<i>DateOnly</i>	<i>Public</i>	
03	<i>Location</i>	<i>string</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.3 WorkshopClassDetailTrainerSlotOnlyModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>ClassId</i>	<i>int</i>	<i>Public</i>	
02	<i>TrainerId</i>	<i>int</i>	<i>Public</i>	
03	<i>SlotId</i>	<i>int</i>	<i>Public</i>	
04	<i>Date</i>	<i>DateOnly</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.4 WorkshopClassModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>StartTime</i>	<i>DateOnly?</i>	<i>Public</i>	
03	<i>Location</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.5 WorkshopStaffController

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetWorkshopDetailTemplate	Task<IActionResult>	Public	Parameters: + workshopId: int
02	CreateWorkshopClass	Task<IActionResult>	Public	Parameters: + workshopClass: WorkshopClassAddModel (3.11.2.2)
03	GetClassesByWorkshop	Task<IActionResult>	Public	Parameters: + workshopId: int
04	GetClassAdminById	Task<IActionResult>	Public	Parameters: + classId: int
05	GetDetailsByClass	Task<IActionResult>	Public	Parameters: + workshopClassId: int
06	ModifyTrainerForDetail	Task<IActionResult>	Public	Parameters: + workshopClassDetail: WorkshopClassDetailTraineeSlotModifyModel (3.11.2.14)
07	CancelWorkshopClass	Task<IActionResult>	Public	Parameters: + workshopClassId: int
08	SetWorkshopClassOnGoing	Task<IActionResult>	Public	Parameters: + workshopClassId: int
09	CloseRegistrationWorkshopClass	Task<IActionResult>	Public	Parameters: + workshopClassId: int
10	GetWorkshops	Task<IActionResult>	Public	
11	ModifyWorkshopClass	Task<IActionResult>	Public	Parameters: + modified: WorkshopClassModifyModel (3.11.2.4)

### 3.11.2.6 WorkshopBaseController

- [Refer to 3.10.2.9](#)

### 3.11.2.7 IWorkshopService

No	Attribute	Data Type	Visibility	Description
01	Manager	IServiceManager	Public	
02	Staff	IServiceStaff	Public	

No	Method	Return Type	Visibility	Description

### 3.11.2.8 WorkshopService

No	Attribute	Data Type	Visibility	Description
01	Manager	IServiceManager	Public	
02	Staff	IServiceStaff	Public	

No	Method	Return Type	Visibility	Description

### 3.11.2.9 IServiceManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetDetailTemplatesByWorkshopId	Task<IEnumerable<WorkshopDetailTemplateViewModel>>	Public	Parameters: + workshopId: int Return: + WorkshopDetailTemplateViewModel (3.10.2.15)

### 3.11.2.10 ManagerService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	GetDetailTemplatesByWorkshopId	Task<IEnumerable<WorkshopDetailTemplateViewModel>>	Public	Parameters: + workshopId: int Return: + WorkshopDetailTemplateViewModel (3.10.2.15)

### 3.11.2.11 IStaffService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CreateWorkshopClass	Task	Public	Parameters: + workshopClass: WorkshopClassAddModel (3.11.2.2)
02	GetWorkshopClassAdminViewMode	Task<IEnumerable<WorkshopClassAdminViewModel>>	Public	Parameters: + workshopId: int Return: + WorkshopClassAdminViewModel (3.11.2.12)
03	GetWorkshopClassDetailViewModels	Task<IEnumerable<WorkshopClassDetailViewModel>>	Public	Parameters: + workshopsClassId: int Return:

				+ WorkshopClassDetailView Model (3.11.2.13)
04	ModifyWorkshopClassD etailTrainerSlot	Task	Public	Parameters: + workshopClassDetail: WorkshopClassDetailTraine rSlotModifyModel (3.11.2.14)
05	CancelWorkshopClass	Task	Public	Parameters: + workshopClassId: int
06	CompleteWorkshopClas s	Task	Public	Parameters: + workshopClassId: int
07	SetWorkshopClassOnGo ing	Task	Public	Parameters: + workshopClassId: int
08	GetAttendeesInSlot	Task<IEnumerable<Regi steredCustomerModel> >	Public	Parameters: + classSlotId: int Return: + RegisteredCustomerModel (3.11.2.15)
09	GetClassAdminViewById	Task<WorkshopClassAd minViewModel>>	Public	Parameters: + classId: int Return: + WorkshopClassAdminView Model (3.11.2.12)
10	ModifyWorkshopClass	Task	Public	Parameters: + modified: WorkshopClassModifiyMod el (3.11.2.16)

### 3.11.2.12 WorkshopClassAdminViewModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	Public	
02	<i>WorkshopId</i>	<i>int</i>	Public	
03	<i>RegisterPeriod</i>	<i>int</i>	Public	
04	<i>StartTime</i>	<i>DateTime</i>	Public	
05	<i>RegisterEndDate</i>	<i>DateTime</i>	Public	
06	<i>RegistrationAmount</i>	<i>RegistrationAmountMo del?</i>	Public	<i>RegistrationAmountModel (3.11.2.17)</i>
07	<i>MinimumRegistration</i>	<i>int</i>	Public	
08	<i>MaximumRegistration</i>	<i>int</i>	Public	
09	<i>Location</i>	<i>string</i>	Public	
10	<i>Status</i>	<i>Status</i>	Public	Refer to enum values in <i>WorkshopClass</i> , <i>status</i> field at <a href="#">2.2</a>

No	Method	Return Type	Visibility	Description

### 3.11.2.13 WorkshopClassDetailViewModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>SlotId</i>	<i>int?</i>	<i>Public</i>	
03	<i>Detail</i>	<i>string</i>	<i>Public</i>	
04	<i>StartTime</i>	<i>TimeSpan?</i>	<i>Public</i>	
05	<i>EndTime</i>	<i>TimeSpan?</i>	<i>Public</i>	
06	<i>Date</i>	<i>DateTime?</i>	<i>Public</i>	
07	<i>Trainer</i>	<i>TrainerWorkshopModeI?</i>	<i>Public</i>	<i>TrainerWorkshopModel (3.11.2.18)</i>
08	<i>Status</i>	<i>Status?</i>	<i>Public</i>	<i>Refer to enum values in WorkshopAttendance, status field at <a href="#">2.2</a></i>

No	Method	Return Type	Visibility	Description

### 3.11.2.14 WorkshopClassDetailTrainerSlotModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>ClassId</i>	<i>int</i>	<i>Public</i>	
02	<i>TrainerId</i>	<i>int</i>	<i>Public</i>	
03	<i>SlotId</i>	<i>int</i>	<i>Public</i>	
04	<i>Date</i>	<i>DateOnly</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.15 RegisteredCustomerModel

No	Attribute	Data Type	Visibility	Description
01	<i>Avatar</i>	<i>string</i>	<i>Public</i>	
02	<i>CustomerId</i>	<i>int</i>	<i>Public</i>	
03	<i>CustomerName</i>	<i>string</i>	<i>Public</i>	
04	<i>PhoneNumber</i>	<i>string</i>	<i>Public</i>	
05	<i>Email</i>	<i>string</i>	<i>Public</i>	
06	<i>Status</i>	<i>Status</i>	<i>Public</i>	<i>Refer to enum values in WorkshopAttendance, status field at <a href="#">2.2</a></i>
07	<i>WorkshopClassDetailId</i>	<i>int</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.16 WorkshopClassModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>StartTime</i>	<i>DateOnly?</i>	<i>Public</i>	
03	<i>Location</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.17 RegistrationAmountModel

No	Attribute	Data Type	Visibility	Description
01	<i>ClassId</i>	<i>int?</i>	<i>Public</i>	
02	<i>Registered</i>	<i>int</i>	<i>Public</i>	
03	<i>Maximum</i>	<i>int</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.18 TrainerWorkshopModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Name</i>	<i>string</i>	<i>Public</i>	
03	<i>Email</i>	<i>string</i>	<i>Public</i>	
04	<i>Avatar</i>	<i>string</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.11.2.19 StaffService

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshopClass</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClass:</i> <i>WorkshopClassAddModel</i> (3.11.2.2)
02	<i>GetWorkshopClassAdminViewMode</i>	<i>Task&lt;IEnumerable&lt;WorkshopClassAdminViewModel&gt;&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopId: int</i> <i>Return:</i> + <i>WorkshopClassAdminViewModel</i> (3.11.2.12)
03	<i>GetWorkshopClassDetailViewModels</i>	<i>Task&lt;IEnumerable&lt;WorkshopClassDetailViewModel&gt;&gt;</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopsClassId: int</i> <i>Return:</i> + <i>WorkshopClassDetailViewModel</i> (3.11.2.13)
04	<i>ModifyWorkshopClassDetailTrainerSlot</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassDetail:</i> <i>WorkshopClassDetailTrainerSlotModifyModel</i> (3.11.2.14)
05	<i>CancelWorkshopClass</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>
06	<i>CompleteWorkshopClass</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + <i>workshopClassId: int</i>

07	<i>SetWorkshopClassOnGoing</i>	Task	Public	Parameters: + workshopClassId: int
08	<i>GetAttendeesInSlot</i>	Task<IEnumerable<RegisteredCustomerModel>>	Public	Parameters: + classSlotId: int Return: + RegisteredCustomerModel (3.11.2.15)
09	<i>GetClassAdminViewById</i>	Task<WorkshopClassAdminViewModel>>	Public	Parameters: + classId: int Return: + WorkshopClassAdminView Model (3.11.2.12)
10	<i>ModifyWorkshopClass</i>	Task	Public	Parameters: + modified: WorkshopClassModifiyModel (3.11.2.16)
11	<i>CompareCurrentDate</i>	Task	Private	Parameters: + slotDetail: WorkshopClassDetailView Model (3.11.2.13)
12	<i>ComparePreviousSlotDetail</i>	Task	Private	Parameters: + previousSlotDetail: WorkshopClassDetailView Model (3.11.2.13] + modifiedDate: DateTime + modifiedSlotStartTime: TimeSpan
13	<i>CompareFollowingSlotDetail</i>	Task	Private	Parameters: + followingSlotDetail: WorkshopClassDetailView Model (3.11.2.13] + modifiedDate: DateTime + modifiedSlotStartTime: TimeSpan

### 3.11.2.20 AllService

No	Attribute	Data Type	Visibility	Description
01	_workshop	IWorkshopFeature	Internal	

No	Method	Return Type	Visibility	Description

### 3.11.2.21 IWorkshopFeature

No	Attribute	Data Type	Visibility	Description
01	Manager	IFeatureManager	Public	
02	Staff	IFeatureStaff	Public	

No	Method	Return Type	Visibility	Description

### 3.11.2.22 IFeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<code>GetDetailTemplatesByWorkshopId</code>	<code>Task&lt;IEnumerable&lt;WorkshopDetailTemplateViewModel&gt;&gt;</code>	Public	<i>Parameters:</i> + workshopId: int <i>Return:</i> + <code>WorkshopDetailTemplateViewModel (3.10.2.15)</code>

### 3.11.2.23 FeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<code>GetDetailTemplatesByWorkshopId</code>	<code>Task&lt;IEnumerable&lt;WorkshopDetailTemplateViewModel&gt;&gt;</code>	Public	<i>Parameters:</i> + workshopId: int <i>Return:</i> + <code>WorkshopDetailTemplateViewModel (3.10.2.15)</code>

### 3.11.2.24 IFeatureStaff

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<code>CreateWorkshopClass</code>	<code>Task</code>	Public	<i>Parameters:</i> + workshopClass: <code>WorkshopClassAddModel (3.11.2.2)</code>
02	<code>GetWorkshopClassAdminViewModels</code>	<code>Task&lt;IEnumerable&lt;WorkshopClassAdminViewModel&gt;&gt;</code>	Public	<i>Parameters:</i> + workshopId: int <i>Return:</i> + <code>WorkshopClassAdminViewModel (3.11.2.12)</code>
03	<code>GetWorkshopClassDetailViewModels</code>	<code>Task&lt;IEnumerable&lt;WorkshopClassDetailViewModel&gt;&gt;</code>	Public	<i>Parameters:</i> + workshopClassId: int <i>Return:</i> + <code>WorkshopClassDetailViewModel (3.11.2.13)</code>
04	<code>ModifyWorkshopClass</code>	<code>Task</code>	Public	<i>Parameters:</i> + workshopClass: <code>WorkshopClassModifyModel (3.11.2.4)</code>

05	<i>ModifyWorkshopClassDetailTrainerSlot</i>	Task	Public	<i>Parameters:</i> + workshopClass: <i>WorkshopClassDetailTraineeSlotModifyModel</i> (3.11.2.14)
06	<i>CheckPassEndRegistration</i>	Task<bool>	Public	<i>Parameters:</i> + workshopClassDetailId: int + compareDate: DateOnly
07	<i>GetPreviousWorkshopClassDetail</i>	Task<WorkshopClassDetailViewModel?>	Public	<i>Parameters:</i> + workshopClassDetailId: int <i>Return:</i> + <i>WorkshopClassDetailView Model</i> (3.11.13)
08	<i>GetFollowingWorkshopClassDetail</i>	Task<WorkshopClassDetailViewModel?>	Public	<i>Parameters:</i> + workshopClassDetailId: int <i>Return:</i> + <i>WorkshopClassDetailView Model</i> (3.11.13)
09	<i>CancelWorkshopClass</i>	Task	Public	<i>Parameters:</i> + workshopClassId: int
10	<i>GetListRegistered</i>	Task<IEnumerable<RegisteredCustomerModel>>	Public	<i>Parameters:</i> + classSlotId: int
11	<i>SetClassOnGoing</i>	Task	Public	<i>Parameters:</i> + classId: int
12	<i>SetOpenRegistrationFor Class</i>	Task	Public	<i>Parameters:</i> + classId: int
13	<i>SetClosedRegistrationFo rClass</i>	Task	Public	<i>Parameters:</i> + classId: int
14	<i>CompleteWorkshopClas s</i>	Task	Public	<i>Parameters:</i> + workshopClassId: int
15	<i>CheckSlotFulfill</i>	Task<bool>	Public	<i>Parameters:</i> + workshopClassId: int
16	<i>GetClassAdminViewById</i>	Task<WorkshopClassAdminViewModel>	Public	<i>Parameters:</i> + classId: int

### 3.11.2.25 FeatureStaff

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateWorkshopClass</i>	Task	Public	<i>Parameters:</i> + workshopClass: <i>WorkshopClassAddModel</i> (3.11.2.2)

02	<i>GetWorkshopClassAdminViewModels</i>	<i>Task&lt;IEnumerable&lt;WorkshopClassAdminViewModel&gt;&gt;</i>	Public	<i>Parameters:</i> + workshopId: int <i>Return:</i> + WorkshopClassAdminViewModel (3.11.2.12)
03	<i>GetWorkshopClassDetailViewModels</i>	<i>Task&lt;IEnumerable&lt;WorkshopClassDetailViewModel&gt;&gt;</i>	Public	<i>Parameters:</i> + workshopClassId: int <i>Return:</i> + WorkshopClassDetailViewModel (3.11.2.13)
04	<i>ModifyWorkshopClass</i>	<i>Task</i>	Public	<i>Parameters:</i> + workshopClass: WorkshopClassModifyModel (3.11.2.4)
05	<i>ModifyWorkshopClassDetailTrainerSlot</i>	<i>Task</i>	Public	<i>Parameters:</i> + workshopClass: WorkshopClassDetailTrainerSlotModifyModel (3.11.2.14)
06	<i>CheckPassEndRegistration</i>	<i>Task&lt;bool&gt;</i>	Public	<i>Parameters:</i> + workshopClassDetailId: int + compareDate: DateOnly
07	<i>GetPreviousWorkshopClassDetail</i>	<i>Task&lt;WorkshopClassDetailViewModel?&gt;</i>	Public	<i>Parameters:</i> + workshopClassDetailId: int <i>Return:</i> + WorkshopClassDetailViewModel (3.11.13)
08	<i>GetFollowingWorkshopClassDetail</i>	<i>Task&lt;WorkshopClassDetailViewModel?&gt;</i>	Public	<i>Parameters:</i> + workshopClassDetailId: int <i>Return:</i> + WorkshopClassDetailViewModel (3.11.13)
09	<i>CancelWorkshopClass</i>	<i>Task</i>	Public	<i>Parameters:</i> + workshopClassId: int
10	<i>GetListRegistered</i>	<i>Task&lt;IEnumerable&lt;RegisteredCustomerModel&gt;&gt;</i>	Public	<i>Parameters:</i> + classSlotId: int
11	<i>SetClassOnGoing</i>	<i>Task</i>	Public	<i>Parameters:</i> + classId: int
12	<i>SetOpenRegistrationForClass</i>	<i>Task</i>	Public	<i>Parameters:</i> + classId: int
13	<i>SetClosedRegistrationForClass</i>	<i>Task</i>	Public	<i>Parameters:</i> + classId: int

14	<i>CompleteWorkshopClass</i>	<i>Task</i>	Public	Parameters: + workshopClassId: int
15	<i>CheckSlotFulfill</i>	<i>Task&lt;bool&gt;</i>	Public	Parameters: + workshopClassId: int
16	<i>GetClassAdminViewById</i>	<i>Task&lt;WorkshopClassAdminViewModel&gt;</i>	Public	Parameters: + classId: int
17	<i>IsAbleToCheckAttendance</i>	<i>Task&lt;bool&gt;</i>	Public	Parameters: + classSlotId: int

### 3.11.2.26 FeatureAll

No	Attribute	Data Type	Visibility	Description
01	<i>_unitOfWork</i>	<i>IUnitOfWork</i>	<i>Internal</i>	
02	<i>_mapper</i>	<i>IMapper</i>	<i>Internal</i>	

### 3.11.3 State Transition Diagram

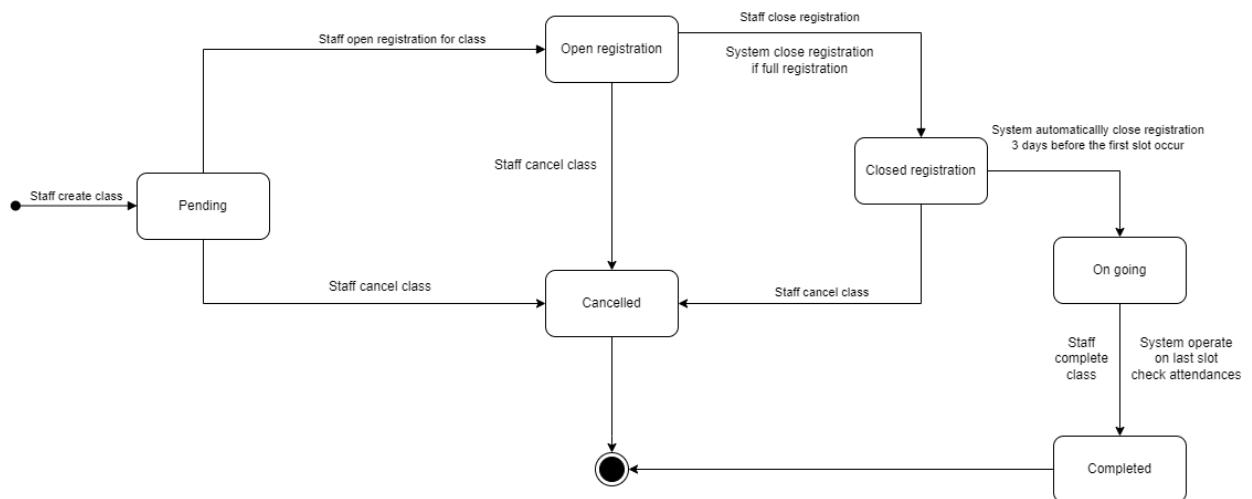


Figure 3.11.3 State Transition Diagram for Workshop Class

### 3.11.4 Create Workshop Class Sequence Diagram

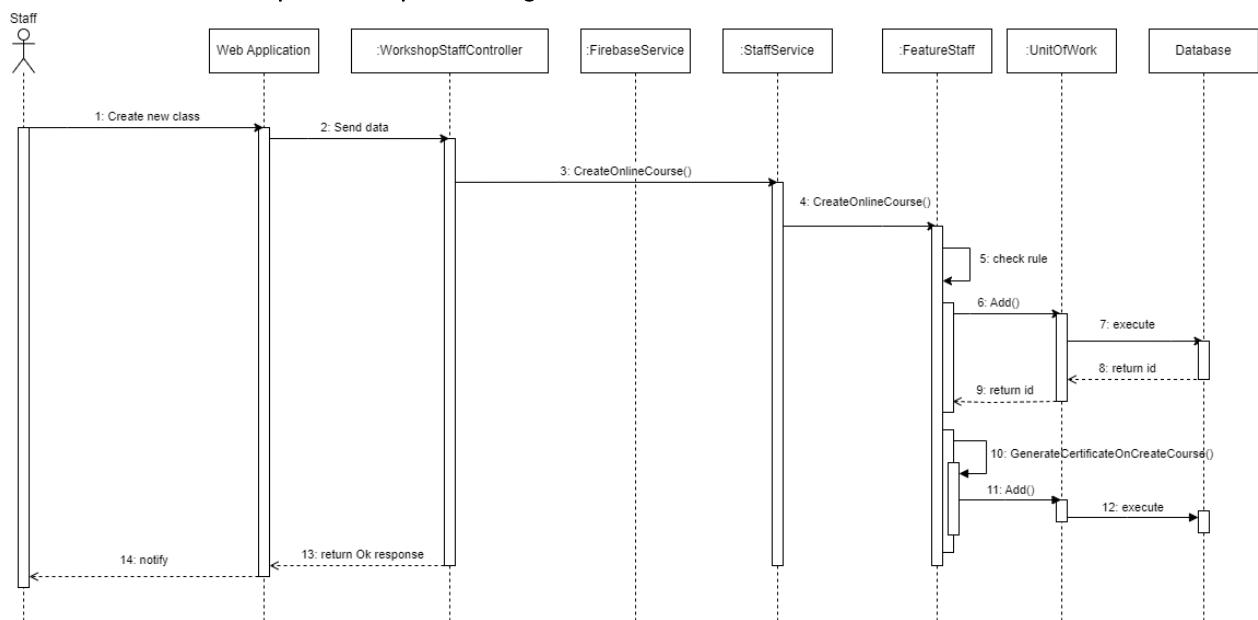


Figure 3.11.4 Create Workshop Class Sequence Diagram

## 3.12 Online Course Management Feature

### 3.12.1 Class Diagram

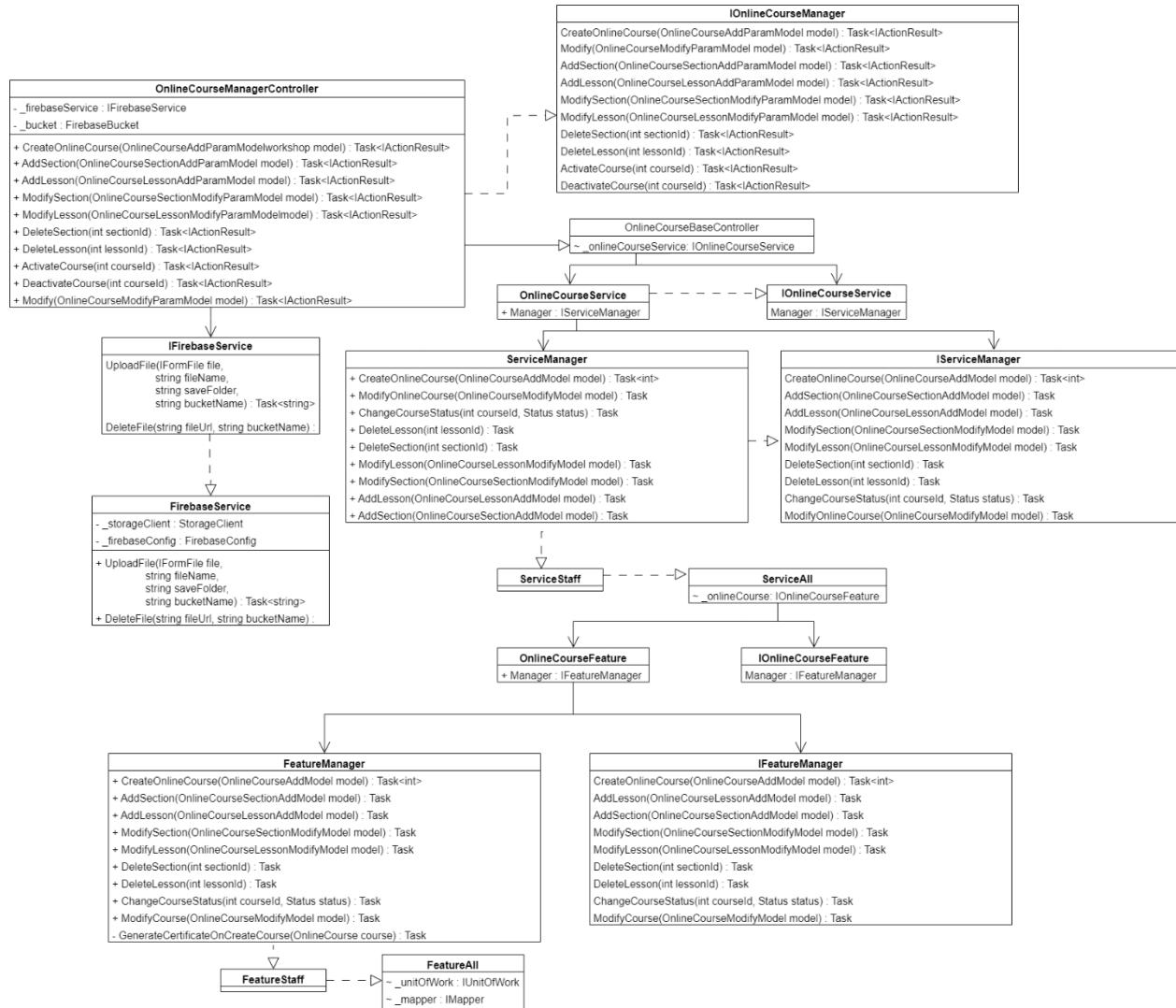


Figure 3.12.1

### 3.12.2 Class Specification

#### 3.12.2.1 IOnlineCourseManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CreateOnlineCourse	Task<ActionResult>	Public	Parameters: + model: <i>OnlineCourseAddParamModel (3.12.2.2)</i>
02	Modify	Task<ActionResult>	Public	Parameters: + model: <i>OnlineCourseModifyParamModel (3.12.2.3)</i>
03	AddSection	Task<ActionResult>	Public	Parameters:

				+ model: <i>OnlineCourseSectionAddParamModel</i> (3.12.2.4)
04	<i>AddLesson</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseLessonAddParamModel</i> ((3.12.2.5))
05	<i>ModifySection</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseSectionModifyParamModel</i> (3.12.2.6)
06	<i>ModifyLesson</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseLessonModifyParamModel</i> (3.12.2.7)
07	<i>DeleteSection</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + sectionId: int
08	<i>DeleteLesson</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + lessonId: int
09	<i>DeactivateCourse</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + courseid: int

### 3.12.2.2 OnlineCourseAddParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>Title</i>	<i>string</i>	<i>Public</i>	
02	<i>ShortDescription</i>	<i>string</i>	<i>Public</i>	
03	<i>Picture</i>	<i>IFormFile</i>	<i>Public</i>	
04	<i>Price</i>	<i>decimal</i>	<i>Public</i>	<i>Must be positive number</i>

No	Method	Return Type	Visibility	Description
01	<i>ToOnlineCourseAddModel</i>	<i>OnlineCourseAddModel</i>	<i>Public</i>	Parameters: + picture: string Return: + <i>OnlineCourseAddModel</i> (3.12.2.8)

### 3.12.2.3 OnlineCourseModifyParamModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
01	<i>Title</i>	<i>string?</i>	<i>Public</i>	
02	<i>ShortDescription</i>	<i>string?</i>	<i>Public</i>	
03	<i>Picture</i>	<i>IFormFile?</i>	<i>Public</i>	
04	<i>Price</i>	<i>decimal?</i>	<i>Public</i>	<i>Must be positive number</i>

No	Method	Return Type	Visibility	Description
01	<i>ToOnlineCourseModifyModel</i>	<i>OnlineCourseModifyModel</i>	<i>Public</i>	Parameters: + picture: string? Return: + <i>OnlineCourseModifyModel</i> (3.12.2.9)

### 3.12.2.4 OnlineCourseSectionAddParamModel

No	Attribute	Data Type	Visibility	Description
01	CourseId	int	Public	
02	Title	string	Public	
03	Description	string	Public	
04	ResourceFiles	List<IFormFile>?	Public	

No	Method	Return Type	Visibility	Description
01	ToOnlineCourseLessonModifyModel	OnlineCourseSectionAddModel	Public	<p>Parameters:</p> <p>+ files: string? – default value: null</p> <p>Return:</p> <p>+</p> <p>OnlineCourseSectionAddModel (3.12.2.10)</p>

### 3.12.2.5 OnlineCourseLessonAddParamModel

No	Attribute	Data Type	Visibility	Description
01	SectionId	int	Public	
02	Title	string	Public	
03	Description	string?	Public	
04	Detail	string?	Public	
06	Video	IFormFile?	Public	

No	Method	Return Type	Visibility	Description
01	ToOnlineCourseLessonAddModel	OnlineCourseLessonAddModel	Public	<p>Parameters:</p> <p>+ video: string? – default value: null</p> <p>Return:</p> <p>+</p> <p>OnlineCourseLessonAddModel (3.12.2.11)</p>

### 3.12.2.6 OnlineCourseSectionModifyParamModel

No	Attribute	Data Type	Visibility	Description
01	Id	int	Public	
02	Title	string?	Public	
03	Description	string?	Public	
04	ResourceFiles	List<IFormFile>?	Public	

No	Method	Return Type	Visibility	Description
01	ToOnlineCourseSectionModifyModel	OnlineCourseSectionModifyModel	Public	<p>Parameters:</p> <p>+ files: string? – default value: null</p> <p>Return:</p> <p>+</p> <p>OnlineCourseSectionModifyModel (3.12.2.12)</p>

### 3.12.2.7 OnlineCourseLessonModifyParamModel

No	Attribute	Data Type	Visibility	Description
----	-----------	-----------	------------	-------------

01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string?</i>	<i>Public</i>	
03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>Detail</i>	<i>string?</i>	<i>Public</i>	
06	<i>Video</i>	<i>IFormFile?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description
01	<i>ToOnlineCourseLessonModifyModel</i>	<i>OnlineCourseLessonModifyModel</i>	<i>Public</i>	<p>Parameters:</p> <p>+ video: <i>string?</i> – default value: null</p> <p>Return:</p> <p>+</p> <i>OnlineCourseLessonModifyModel</i> (3.12.2.13)

### 3.12.2.8 OnlineCourseAddModel

No	Attribute	Data Type	Visibility	Description
01	<i>Title</i>	<i>string</i>	<i>Public</i>	
02	<i>ShortDescription</i>	<i>string</i>	<i>Public</i>	
03	<i>Picture</i>	<i>string</i>	<i>Public</i>	
04	<i>Price</i>	<i>decimal</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.9 OnlineCourseModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string?</i>	<i>Public</i>	
03	<i>ShortDescription</i>	<i>string?</i>	<i>Public</i>	
04	<i>Picture</i>	<i>string?</i>	<i>Public</i>	
05	<i>Price</i>	<i>decimal?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.10 OnlineCourseSectionAddModel

No	Attribute	Data Type	Visibility	Description
01	<i>CourseId</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string</i>	<i>Public</i>	
03	<i>Description</i>	<i>string</i>	<i>Public</i>	
04	<i>ResourceFiles</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.11 OnlineCourseLessonAddModel

No	Attribute	Data Type	Visibility	Description
01	<i>SectionId</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string</i>	<i>Public</i>	

03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>Detail</i>	<i>string?</i>	<i>Public</i>	
05	<i>Video</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.12 OnlineCourseSectionModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string?</i>	<i>Public</i>	
03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>ResourceFiles</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.13 OnlineCourseLessonModifyModel

No	Attribute	Data Type	Visibility	Description
01	<i>Id</i>	<i>int</i>	<i>Public</i>	
02	<i>Title</i>	<i>string?</i>	<i>Public</i>	
03	<i>Description</i>	<i>string?</i>	<i>Public</i>	
04	<i>Detail</i>	<i>string?</i>	<i>Public</i>	
05	<i>Video</i>	<i>string?</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.14 OnlineCourseManagerController

No	Attribute	Data Type	Visibility	Description
01	<i>_firebaseService</i>	<i>IFirebaseService</i>	<i>Private</i>	<i>IFirebaseService</i> referring to <a href="#">3.10.2.26</a>
02	<i>_bucket</i>	<i>FirebaseBucket</i>	<i>Private</i>	<i>FirebaseBucket</i> referring to <a href="#">3.10.2.2</a>

No	Method	Return Type	Visibility	Description
01	<i>CreateOnlineCourse</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseAddParamModel</i> (3.12.2.2)
02	<i>Modify</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseModifyParamModel</i> (3.12.2.3)
03	<i>AddSection</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseSectionAddParamModel</i> (3.12.2.4)
04	<i>AddLesson</i>	<i>Task&lt;IActionResult&gt;</i>	<i>Public</i>	<i>Parameters:</i>

				+ model: <i>OnlineCourseLessonAddParamModel</i> ((3.12.2.5))
05	<i>ModifySection</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseSectionModifyParamModel</i> (3.12.2.6)
06	<i>ModifyLesson</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseLessonModifyParamModel</i> (3.12.2.7)
07	<i>DeleteSection</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + sectionId: int
08	<i>DeleteLesson</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + lessonId: int
09	<i>DeactivateCourse</i>	<i>Task&lt;ActionResult&gt;</i>	<i>Public</i>	Parameters: + courseid: int

### 3.12.2.15 FirebaseService

- This model is referred to [3.10.27.](#)

### 3.12.2.16 OnlineCourseBaseController

No	Attribute	Data Type	Visibility	Description
01	<i>_onlineCourseService</i>	<i>IOnlineCourseService</i>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.17 IOnlineCourseService

No	Attribute	Data Type	Visibility	Description
01	<i>Manager</i>	<i>IServiceManager</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.18 OnlineCourseService

No	Attribute	Data Type	Visibility	Description
01	<i>Manager</i>	<i>IServiceManager</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.19 IServiceManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateOnlineCourse</i>	<i>Task&lt;int&gt;</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseAddModel</i> (3.12.2.8)

02	<i>AddSection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseSectionAddModel</i> (3.12.2.10)
03	<i>AddLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseLessonAddModel</i> (3.12.2.11)
04	<i>ModifySection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseSectionModifyModel</i> (3.12.2.12)
05	<i>ModifyLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseLessonModifyModel</i> (3.12.2.13)
06	<i>DeleteSection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + sectionId: int
07	<i>DeleteLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + lessonId: int
08	<i>ChangeCourseStatus</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + courseId: int + status: Status (enum values referring to table <i>OnlineCourse</i> , status field in <a href="#">2.2</a> )
09	<i>ModifyOnlineCourse</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseModifyModel</i> (3.12.2.9)

### 3.12.2.20 ServiceManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	<i>CreateOnlineCourse</i>	<i>Task&lt;int&gt;</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseAddModel</i> (3.12.2.8)
02	<i>AddSection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseSectionAddModel</i> (3.12.2.10)
03	<i>AddLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseLessonAddModel</i> (3.12.2.11)
04	<i>ModifySection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i>

				+ model: <i>OnlineCourseSectionModifyModel</i> (3.12.2.12)
05	<i>ModifyLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseLessonModifyModel</i> (3.12.2.13)
06	<i>DeleteSection</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + sectionId: int
07	<i>DeleteLesson</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + lessonId: int
08	<i>ChangeCourseStatus</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + courseId: int + status: Status (enum values referring to table <i>OnlineCourse</i> , status field in 2.2)
09	<i>ModifyOnlineCourse</i>	<i>Task</i>	<i>Public</i>	<i>Parameters:</i> + model: <i>OnlineCourseModifyModel</i> (3.12.2.9)

### 3.12.2.21 ServiceStaff

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.12.2.22 ServiceAll

No	Attribute	Data Type	Visibility	Description
01	<i>_onlineCourse</i>	<i>IOnlineCourseFeature</i>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.23 IOnlineCourseFeature

No	Attribute	Data Type	Visibility	Description
01	<i>Manager</i>	<i>IFeatureManager</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.24 OnlineCourseFeature

No	Attribute	Data Type	Visibility	Description
01	<i>Manager</i>	<i>IFeatureManager</i>	<i>Public</i>	

No	Method	Return Type	Visibility	Description

### 3.12.2.25 IFeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CreateOnlineCourse	Task<int>	Public	Parameters: + model: <i>OnlineCourseAddModel</i> (3.12.2.8)
02	AddLesson	Task	Public	Parameters: + model: <i>OnlineCourseLessonAddModel</i> (3.12.2.11)
03	AddSection	Task	Public	Parameters: + model: <i>OnlineCourseSectionAddModel</i> (3.12.2.10)
04	ModifySection	Task	Public	Parameters: + model: <i>OnlineCourseSectionModifyModel</i> (3.12.2.12)
05	ModifyLesson	Task	Public	Parameters: + model: <i>OnlineCourseLessonModifyModel</i> (3.12.2.13)
06	DeleteSection	Task	Public	Parameters: + sectionId: int
07	DeleteLesson	Task	Public	Parameters: + lessonId: int
08	ChangeCourseStatus	Task	Public	Parameters: + courseId: int + status: Status (enum values referring to table <i>OnlineCourse</i> , status field in 2.2)
09	ModifyCourse	Task	Public	Parameters: + model: <i>OnlineCourseModifyModel</i> (3.12.2.9)

### 3.12.2.26 FeatureManager

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description
01	CreateOnlineCourse	Task<int>	Public	Parameters: + model: <i>OnlineCourseAddModel</i> (3.12.2.8)
02	AddLesson	Task	Public	Parameters:

				+ model: <i>OnlineCourseLessonAddModel</i> (3.12.2.11)
03	<i>AddSection</i>	<i>Task</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseSectionAddModel</i> (3.12.2.10)
04	<i>ModifySection</i>	<i>Task</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseSectionModifyModel</i> (3.12.2.12)
05	<i>ModifyLesson</i>	<i>Task</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseLessonModifyModel</i> (3.12.2.13)
06	<i>DeleteSection</i>	<i>Task</i>	<i>Public</i>	Parameters: + sectionId: int
07	<i>DeleteLesson</i>	<i>Task</i>	<i>Public</i>	Parameters: + lessonId: int
08	<i>ChangeCourseStatus</i>	<i>Task</i>	<i>Public</i>	Parameters: + courseId: int + status: Status (enum values referring to table <i>OnlineCourse</i> , status field in 2.2)
09	<i>ModifyCourse</i>	<i>Task</i>	<i>Public</i>	Parameters: + model: <i>OnlineCourseModifyModel</i> (3.12.2.9)

### 3.12.2.27 FeatureStaff

No	Attribute	Data Type	Visibility	Description

No	Method	Return Type	Visibility	Description

### 3.12.2.28 FeatureAll

No	Attribute	Data Type	Visibility	Description
01	<i>_unitOfWork</i>	<i>IUnitOfWork</i>	<i>Internal</i>	
02	<i>_mapper</i>	<i>IMapper</i>	<i>Internal</i>	

No	Method	Return Type	Visibility	Description

### 3.12.3 Create Online Course Sequence Diagram

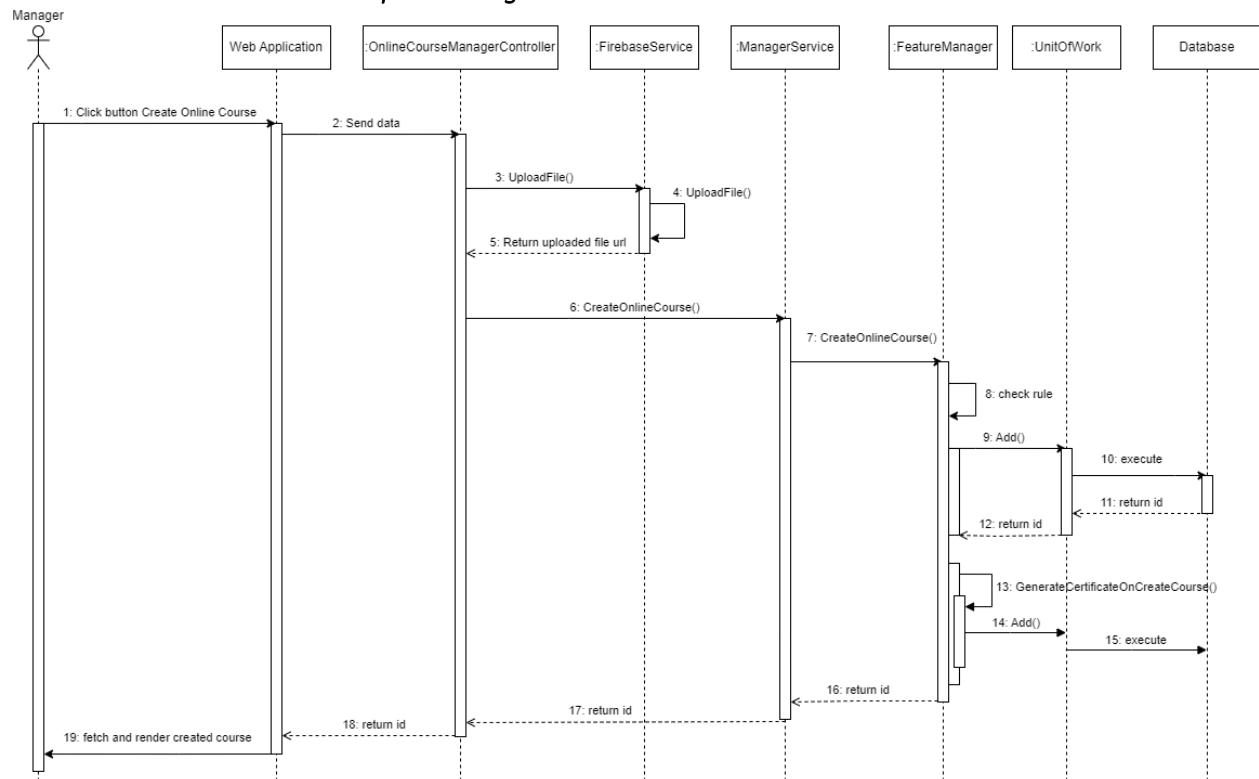


Figure 3.12.3 Create Online Course Sequence Diagram

## V. Software Testing Documentation

### 1. Scope of Testing

#### 1.1. Test model

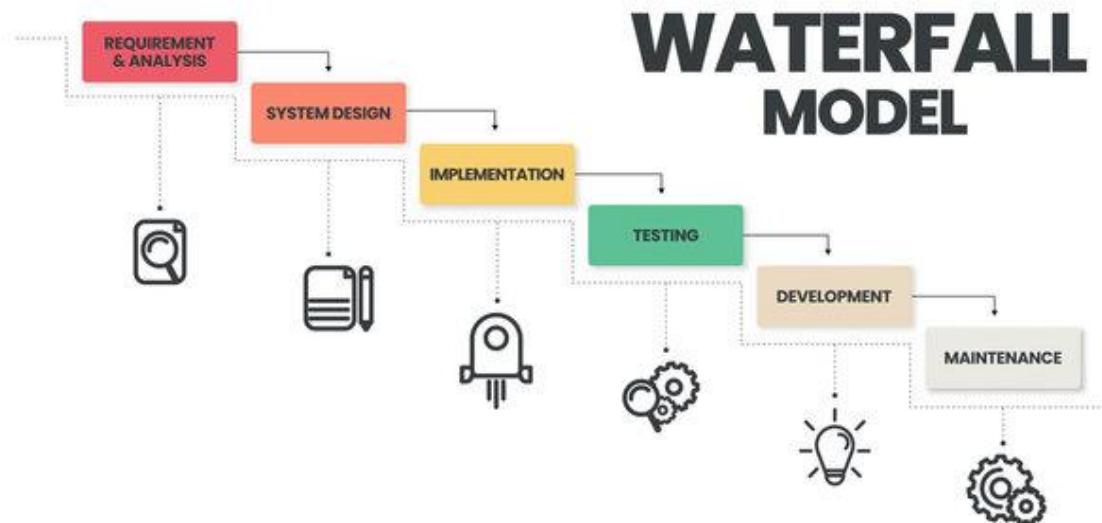


Figure 1.1: Test model

Our team decided to choose Waterfall Model to implement the testing process for this project due to its advantages. Requirements are completed early within the project, enabling the team to define the complete project scope, create an entire schedule, and style the general application. It improves resource utilisation because tasks may be split to be worked in parallel or grouped to leverage resource skills. it's a much better application design because there's a more complete understanding of all the wants and deliverables.

#### 1.2. Test levels

ID	Testing phase	Description
1	Integration testing	Integration testing is the level of software testing individual units combined and tested in a group. The purpose of this is to define and find faults in software
2	System testing	System testing is the level of testing software that evaluates how various component of an application interact together into a full complete integrated system

### 1.3. In scope

Module Name	Application Role	Description
Sign up for Customer	Unauthorised User	<b>Unauthorised user:</b> when an unauthorised user that does not have an account and wants to join in the system with the default role "Customer". Unauthorised users must sign up with a Phone number. After verifying the phone number that has registered successfully and using the app.
Login Web app for Customer	Unauthorised User	<b>Unauthorised User:</b> when an unauthorised user that has an account in the system
Login Mobile app for Customer	Customer	After create account in website customer can only use that account to login and create bird in mobile app
Logout	Authenticated User	<b>Authenticated User:</b> when an unauthorised user that has an account in the system and wants to end a session
Get List Online Course	Customer	<b>Customer:</b> Can view list of online to join course
View list detail online Courses	Customer	View list detail of all Course "Courses" when click "View More Courses"
Search Online Course	Customer	Show the course you want based on the character typed in the search bar
View detail Online Course	Customer	Displays detailed information about the selected course
Enrol Online Course	Customer	Start register to study online course
Online payment	Customer	Start payment online to start study
Study Online Course	Customer	Study online after complete payment
View certificate	Customer	Complete study and earn certificate
View registered course	Customer	After study can view registered course
Add new Course	Manager	Manager create new online course
Change status course	Manager	Manager change status online course
Update section	Manager	Manager update sections in online course
Delete sections	Manager	Manager delete sections in online course
Add new sections	Manager	Manager add new sections in online course
Add new lessons	Manager	Manager add new lessons in online course
Get List Workshop	Customer	<b>Customer:</b> Can view list of online to join Workshop
View list detail Workshop	Customer	View list detail of all Workshop when click "View More"
Search Workshop	Customer	Show the course you want based on the character typed in the search bar
View detail Workshop	Customer	Displays detailed information about the selected Workshop
Enrol Workshop	Customer	Start register to Workshop
Online payment	Customer	Start payment online to start join Workshop
View registered Workshop	Customer	After study can view registered Workshop
Create workshop	Manager	Manager create new workshop
Edit workshop	Manager	Manager can edit workshop

Change slot inside workshop details	Manager	Manager can change slot when create new workshop
Change status active/ inactive in workshop detail	Manager	Manager can change status when create new workshop
Create new class in workshop details	Manager	Manager can create new class in workshop details
Upload image in workshop class	Manager	Manager can upload new image in workshop class
Edit workshop information inside workshop classes	Manager	Manager can edit new image in workshop class
Booking online services	Customer	Customer can start booking online
Cancel ticket assigned	Manager+ staff	Manage and staff can cancel ticket been assigned to trainer
Approve ticket assigned	Manager+ staff	Manage and staff can Approve ticket been assigned to trainer
Confirm Finish Appointment	Manager+ staff	Manage and staff can Confirm ticket been assigned to trainer
Update consultant price policy	Manager+ staff	Manage and staff can Update price policy of consultant
Cancel Update consultant price policy	Manager+ staff	Manage and staff can Cancel update price policy of consultant
Update consultant type	Manager+ staff	Manage and staff can Update type of consultant
Cancel Update consultant type	Manager+ staff	Manage and staff can Cancel update type of consultant
Update evidence and end time un view detail ticket	Trainer	Trainer update evidence and press end time
Confirm bill information	Trainer	Trainer confirms bill information
Register new bird in mobile	Customer	Customer new to system can register new bird
Switch bird in mobile	Customer	Customer can switch already create bird in system
Edit bird profile in mobile	Customer	Customer can edit profile of bird in system
Set bird default in mobile	Customer	Customer can set bird as default so as it will appear in main screen as the next login
Change status of registered course of academy	Manger	Manger can change status of registered course in bird academy
Disabled/Active training course in bird academy	Manger	Manger can Disabled/Active training course in bird academy
Cancel Edit course information in the detail view	Manger	Manager can Cancel Edit course information in the detail view of Training Course Management

Add new skill in Edit course information in the detail view	Manger	Manager can Add new skill in the detail view of Training Course Management
Create new course in the bird academy	Manger	View the Create new course page when clicking Create New Training Course
Cancel Create new course in the bird academy	Manger	Cancel Create new course page when clicking Create New Training Course
Disabled/Active price policy in bird academy	Manger	Manger can Disabled/Active price policy in bird academy

#### 1.4 Out of Scope

- All of these features will not be tested because they are NOT included in the Software

Requirement Specification:

- User Interfaces.
- Hardware Interfaces.
- Software Interfaces.
- Database logic.
- Communications Interfaces.
- Website Security and Performance.

### 2. Test Strategy

#### 2.1 Testing Types

- Functional testing
- Black box testing
- System testing

#### 2.2 Test Levels

Type of Tests	Test Level			
	Unit	Integration	System	Acceptance
Functional Testing	X	X	X	
Black box Testing	X	X	X	X
System testing	X		X	

#### 2.3 Supporting Tools

Purpose	Tool	Vendor/In-house	Version
API Testing	Postman	Postman	v10.20
Web page	VS Code	Microsoft	1.85
Mobile	Android Studio	Google	8

### 3. Test Plan

#### 3.1 Human Resources

Worker/Doer	Role	Specific Responsibilities/Comments
Hoàng Đình Thông	Developer	System testing/ Unit test
Nguyễn Thành Trung	Developer	Functional testing / Black box Testing
Nguyễn Thị Thái Bảo	Developer	Functional testing

Phạm Nhật An	Developer	System testing/ Unit test
Trịnh Nam Anh	Developer	Functional testing

### 3.2 Test Environment

Purpose	Tool	Provider	Version
Running web testing	Edge	Google	120.0.2210.91
Running app testing	Xiaomi	Xiaomi	12.5.3.0

### 3.3 Test Milestones

Milestone Task	Start Date	End Date
Sign in account Customer with valid input information	25/12/2023	25/12/2023
Sign in account Customer with invalid input information	25/12/2023	25/12/2023
Sign in account Customer with null value input	25/12/2023	25/12/2023
Sign in when Customer input confirm password correct	25/12/2023	25/12/2023
Sign in when Customer input confirm password incorrect	25/12/2023	25/12/2023
Sign in account Staff with valid input information	25/12/2023	25/12/2023
Sign in account Staff ff with invalid input information	25/12/2023	25/12/2023
Sign in account Staff with null value input	25/12/2023	25/12/2023
Sign in when Staff input confirm password correct	25/12/2023	25/12/2023
Sign in when Staff input confirm password incorrect	25/12/2023	25/12/2023
Sign in account Trainer with valid input information	25/12/2023	25/12/2023
Sign in account Trainer with invalid input information	25/12/2023	25/12/2023
Sign in account Trainer with null value input	25/12/2023	25/12/2023
Sign in when Trainer input confirm password correct	25/12/2023	25/12/2023
Sign in when Trainer input confirm password incorrect	25/12/2023	25/12/2023
Sign in account Manager with valid input information	25/12/2023	25/12/2023
Sign in account Manager with invalid input information	25/12/2023	25/12/2023
Sign in account Manager with null value input	25/12/2023	25/12/2023
Sign in when Manager input confirm password correct	25/12/2023	25/12/2023
Sign in when Manager input confirm password incorrect	25/12/2023	25/12/2023
Logout when Customer/Staff/Manager/Trainer click “logout” button	25/12/2023	25/12/2023
Login when Customer/Staff/Manager/Trainer click “Login” button	25/12/2023	25/12/2023
Login with null input information	25/12/2023	25/12/2023
Login with wrong format email	25/12/2023	25/12/2023
Start create an account when the user clicks “Doesn't have an account? <a href="#">click here</a> ”	25/12/2023	25/12/2023
Back to login when user click on “Already have an account? <a href="#">click here</a> ”	25/12/2023	25/12/2023
Back to home page when user click on “ <a href="#">Back to home page</a> ”	25/12/2023	25/12/2023
Switch to tab Course when click on “Course” in tab navigator	25/12/2023	25/12/2023
Switch to tab Workshops when click on “Workshops” in tab navigator	25/12/2023	25/12/2023
Show tooltip when hover on “Services” in tab navigator	25/12/2023	25/12/2023
Show tooltip when hover in Hi” name Account”	25/12/2023	25/12/2023
Show tooltip when hover in Hi” Settings”	25/12/2023	25/12/2023
Click on courses on “Courses” tab	25/12/2023	25/12/2023
Payment online to start study in each course	25/12/2023	25/12/2023
View course lesson when click “study”	25/12/2023	25/12/2023
View certificate when completed course	25/12/2023	25/12/2023
Search course in “view more” buttons in tab “Course”	25/12/2023	25/12/2023
Search course with keyword	25/12/2023	25/12/2023
Click on workshops on “workshops” tab	25/12/2023	25/12/2023

Payment online to participate in each workshop	25/12/2023	25/12/2023
View workshops class/ slots	25/12/2023	25/12/2023
View workshops based on status	25/12/2023	25/12/2023
Search workshops in "view more" buttons in tab "Course"	25/12/2023	25/12/2023
Search workshops with keyword	25/12/2023	25/12/2023
View consultant when hover "services" tab and click consultants	25/12/2023	25/12/2023
Start create an appointment online or in-home	25/12/2023	25/12/2023
View bird academy when hover "services" tab and click bird academy	25/12/2023	25/12/2023
View detailed list of courses in "Courses" when clicking "View More Courses"	9/1/2023	9/1/2023
Change status in view detail	9/1/2023	9/1/2023
Update Sections inside view detail	9/1/2023	9/1/2023
Delete Sections inside the detail view and display a message saying you want to delete	9/1/2023	9/1/2023
Create new sections when clicking ADD NEW SECTION	9/1/2023	9/1/2023
View lesson inside each created section	9/1/2023	9/1/2023
Create new lessons within each section	9/1/2023	9/1/2023
Cancel creates a new lesson within each section	9/1/2023	9/1/2023
Create a new online course by clicking the "CREATE ONLINE COURSE" button	9/1/2023	9/1/2023
Cancel create new workshop when pressing the < button	9/1/2023	9/1/2023
Upload new Images in the workshop details screen when clicking Confirm	9/1/2023	9/1/2023
Edit workshop information in the workshop details screen when clicking Confirm	9/1/2023	9/1/2023
Save changes slot in the workshop details screen	9/1/2023	9/1/2023
View Create new classes dialog	9/1/2023	9/1/2023
Click Create to create new classes when the create new classes dialog is turned on	9/1/2023	9/1/2023
Press the Cancel button when creating new classes when the create new classes dialog is turned on	9/1/2023	9/1/2023
Reset Modify of Open/Close Registration and Hosted Location in detail status classes Pending/Open Registration/ Closed Registration/ On Going/ Completed/ Cancelled	9/1/2023	9/1/2023
Confirm Modify of Open/Close Registration and Hosted Location in detail status classes Pending/Open Registration/ Closed Registration/ On Going/ Completed/ Cancelled	9/1/2023	9/1/2023
Show evidence in detail ticket in List Finished	9/1/2023	9/1/2023
Click confirm bill information in the View Bill dialog when clicking the View Bill button	9/1/2023	9/1/2023
Update evidence and end time in detail ticket	9/1/2023	9/1/2023
Click Confirm after clicking Update field in Consultant Type	9/1/2023	9/1/2023
Click Cancel after clicking Update field in Consultant Type	9/1/2023	9/1/2023
Click Confirm after clicking Update field in Consultant Price Policy or Distance Price Policy	9/1/2023	9/1/2023
Click Cancel after clicking Update field in Consultant Price Policy or Distance Price Policy	9/1/2023	9/1/2023
Click Approve ticket to assign trainer without filling in distance information	9/1/2023	9/1/2023
Confirm Create Policy to Create a new price policy when clicking Create New Training Policy in Price Policy Management	9/1/2023	9/1/2023
Click Disabled price policy on any price policy in Price Policy Management	9/1/2023	9/1/2023
Click Active of any price policy in Price Policy Management	9/1/2023	9/1/2023
Update price policy in the detail view of Price Policy Management	9/1/2023	9/1/2023
Add skills to the course in the details view of Training Course Management	9/1/2023	9/1/2023
Click any Active course in Training Course Management	9/1/2023	9/1/2023

Click Disabled any course in Training Course Management	9/1/2023	9/1/2023
Send confirmation email after assigning trainer to re-assign when Click on RE-ASSIGN TRAINER in View Details Registered TrainingCourse when Click on View Details with status Cancel	9/1/2023	9/1/2023
Confirm Check-in inside the Check-in page of Registered Training Course with Confirm status	9/1/2023	9/1/2023
Cancel Check-in inside the Check-in page of Registered Training Course with status Confirm	9/1/2023	9/1/2023
View Check-in in Registered TrainingCourse when Click on Check-in with Confirm status	9/1/2023	9/1/2023

#### 4. Test Cases

Milestone Task	File Name
System Test	<a href="#">Report5 Test Report - Google Trang tính</a>

## VI. Release Package & User Guides

### 1. Deliverable Package

No.	Deliverable Item	Description
1	Project Schedule/Tracking	We use Google Sheets in order to track the process of Task in the project Link: <a href="#">FU CP Tasks - Google Sheets</a>
2	Source Codes	BirdTrainingSystem.API.zip BirdTrainingSystem.WebApplication.zip BirdTrainingSystem.MobileApplication.zip BirdTrainingSystem.StripeAPI.zip Thesis-BirdTraining.apk
3	Database Scripts	BirdTrainingCenterSystem.schema.sql BirdTrainingCenterSystem.data.sql
4	Final Report Document	<a href="#">Report7 Final Project Report.docx</a> <a href="#">Report7 Final Project Report.pdf</a>
5	Test Report	<a href="#">Report5 Test Report.xlsx</a>
6	Slides	<a href="#">Bird trainingCenter.slide.pdf</a> <a href="#">Bird trainingCenter.slide.pptx</a>
7	Video demo	Demo videos for each main flow <ul style="list-style-type: none"> <li>• <a href="#">Training course</a></li> <li>• <a href="#">Consulting</a></li> <li>• <a href="#">Workshop</a></li> <li>• <a href="#">Online Course</a></li> </ul>
8	Resources	Resource for demo

### 2. Installation Guides

#### 2.1 System Requirements

##### 2.1.1 Hardware requirements

###### 2.1.1.1 Web Application

	Minimum Requirements	Maximum Requirements
Internet Connection	Wi-Fi (< 12 Mbps)	Ethernet or Wi-Fi (>= 12 Mbps)
Operating System	Windows 7	Windows 10 or Windows 11
Processors	Intel Core i3 8th	Intel Core i9 Latest Version

	<b>Minimum Requirements</b>	<b>Maximum Requirements</b>
Memory	4GB RAM	8GB RAM or more
Browser	Edge (v79), Chrome (v69), etc...	Edge (Latest Version), Chrome (Latest Version), etc...

#### 2.1.1.2 Mobile Application

<b>Device (Android)</b>	<b>Spec requirements</b>
<b>Operating Systems</b>	Android 10 or higher
<b>Processor</b>	Octa-core (1x2.84 GHz Cortex-X1 & 3x2.42 GHz Cortex-A78 & 4x1.80 GHz Cortex-A55 (Xiaomi))
<b>Storage</b>	32GB or higher
<b>RAM</b>	Minimum 2GB

#### 2.1.1.3 Software Requirements

<b>Device (Android)</b>	<b>Spec requirements</b>	<b>Description</b>
<b>Operating Systems</b>	Windows 10 or higher	Operating system and platform for development
<b>Web Browser</b>	Edge (v79 or higher), Chrome (v69 or higher), etc...	For Web app
<b>Mobile</b>	Android 10 or higher	For Mobile

## 2.2 Setup Files

1. BirdTrainingCenterSystem.schema.sql: database scripts for the system.
2. BirdTrainingSystem.StripeAPI.zip: Stripe API for integrating with Stripe Payment Service.
3. BirdTrainingSystem.API.zip: System API.
4. BirdTrainingSystem.WebApplication.zip: Website application for the system.
5. BirdTrainingSystem.MobileApplication.zip: source code for mobile application.

## 2.3 Installation Instruction

### 2.3.1 *Setup environment*

- Install Nodejs and NPM by the following guidelines link: <https://phoenixnap.com/kb/install-node-js-npm-on-windows>
- Install .net 6 runtime in the following link: <https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.26-windows-x64-installer>

### 2.3.2 *Setup Database*

- Install PostgreSQL by the following guidelines:  
<https://www.postgresqltutorial.com/postgresql-getting-started/install-postgresql/>
- Install PostgreSQL management tool by the following guidelines:  
<https://www.postgresqltutorial.com/postgresql-getting-started/connect-to-postgresql-database/>
- Run BirdTrainingCenterSystem.schema.sql with management tool to initialize the database

### 2.3.3 *Setup Stripe API*

- Extract BirdTrainingSystem.StripeAPI.zip, then go to extracted project folder
- Open terminal at current location and use these commands:
  - `npm install`
  - `node app.js`
- The stripe API is running.

### 2.3.4 *Setup API Server*

- Extract BirdTrainingSystem.API.zip, then go to extracted project folder
- Go to BirdTrainingCenterAPI folder, open terminal at current location and use the following command:  
`dotnet publish "BirdTrainingCenterAPI.csproj" -c Release -o ./app/publish  
/p:UseAppHost=false`
- From BirdTrainingCenterAPI folder, go to \app\publish, open “appsettings.json” file and change the data of the attribute “ConnectionData” matching to the connection to the database at step [2.3.2](#) for the following fields:
  - Server
  - Port
  - Database
  - UID
  - Password
- Open terminal at current location and use the following command:  
`dotnet BirdTrainingCenterAPI.dll`
- The API server is running.

### 2.3.5 Setup Front-end Web application

- Extract BirdTrainingSystem.WebApplication.zip, then go to extracted project folder
- In extracted project folder, set the following attribute in .env file:
  - REACT\_APP\_API="<https://localhost:5001>"
- Then open terminal at current location and use the following commands:
  - `npm install --force`
  - `npm start`
- The web application is running

### 2.3.6 Setup Mobile application

- Extract BirdTrainingSystem.MobileApplication.zip, then go to extracted project folder
- At BirdTrainingSystem.MobileApplication.zip, open the terminal and use the following commands:
  - `npm install`
  - `npx expo start --reset-cache`
- The terminal run into this like this image below:



- In Mobile download expo app and scan QR to run mobile in your phone
- if using Virtual Device Emulator of Android Studio, use can press button "a" to open emulator and start using application

### 3. User Manual

#### 3.1 Overview

- Register and login for authentication and authorization.

- Customer service:

+ Workshop service.

+ Advice Consultation booking service.

+ Online Course service.

+ Bird Training Academy download guidance on website.

+ Bird Training Academy mobile application.

- Trainer timetable.

- Service management:

+ Workshop management.

+ Workshop class management.

+ Consultation booking management.

+ Online Course management.

+ Training Course management.

- Administration management:

+ Account management

##### 3.1.1. Register, login for authentication and authorization

##### 3.1.2. Customer Interaction

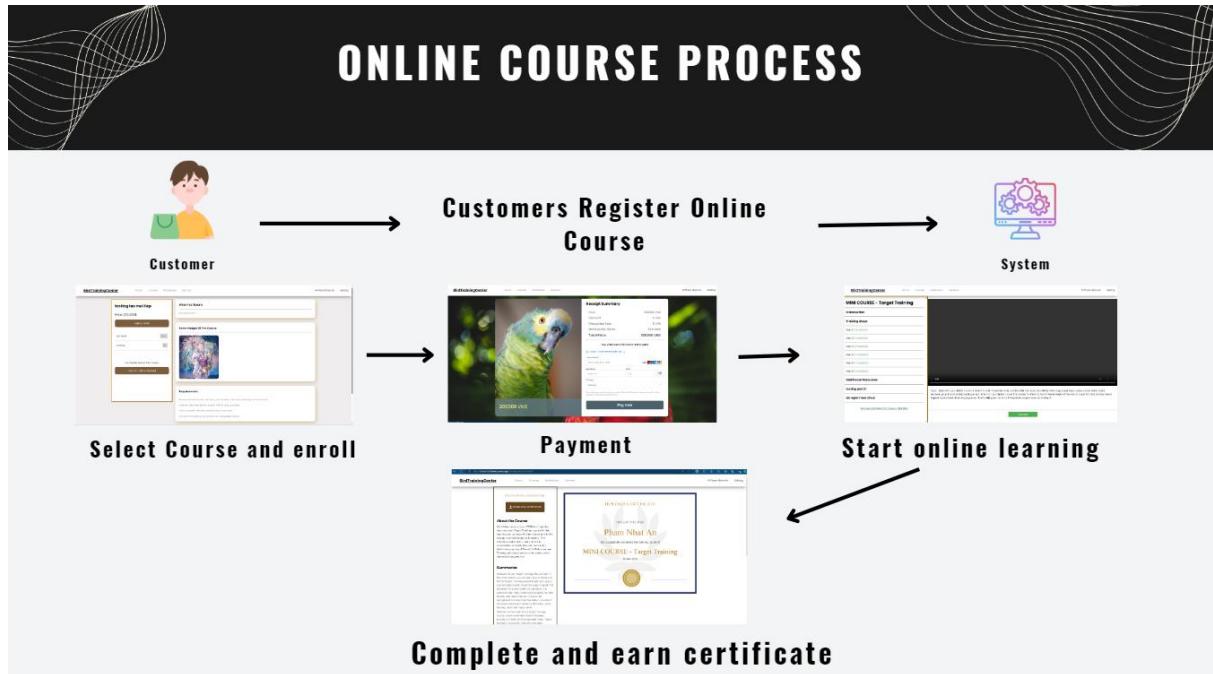
###### 3.1.2.1 Workshop Service



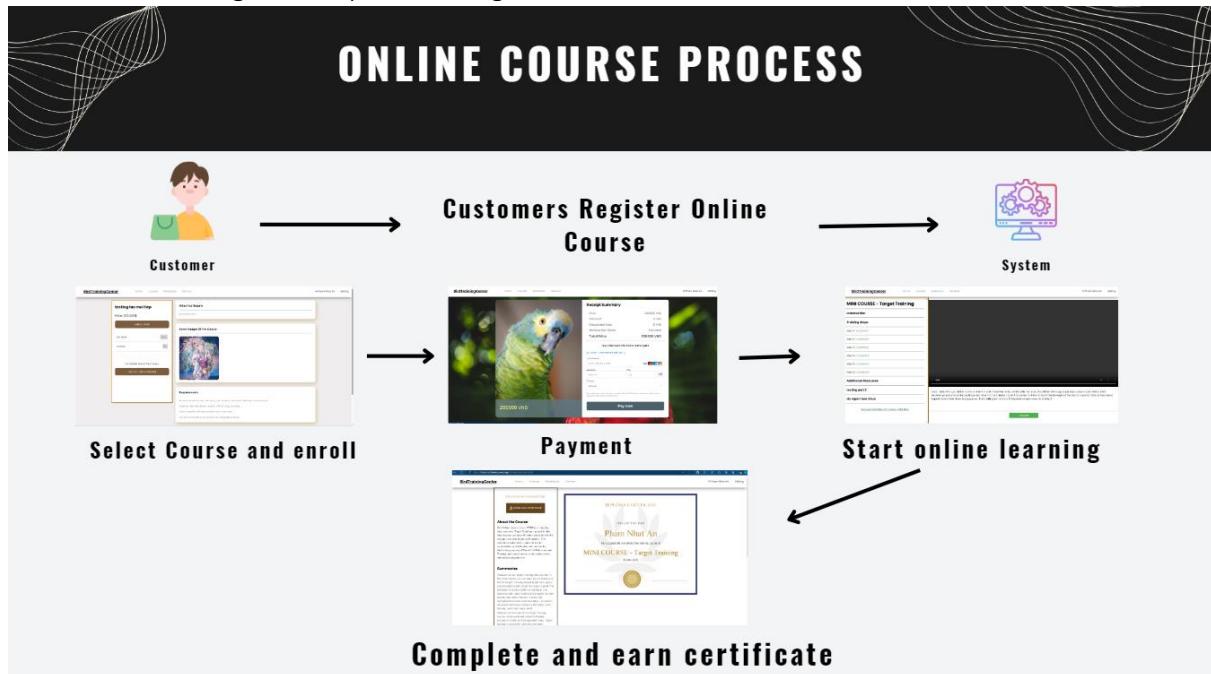
### 3.1.2.2 Advice Consultation Booking Service



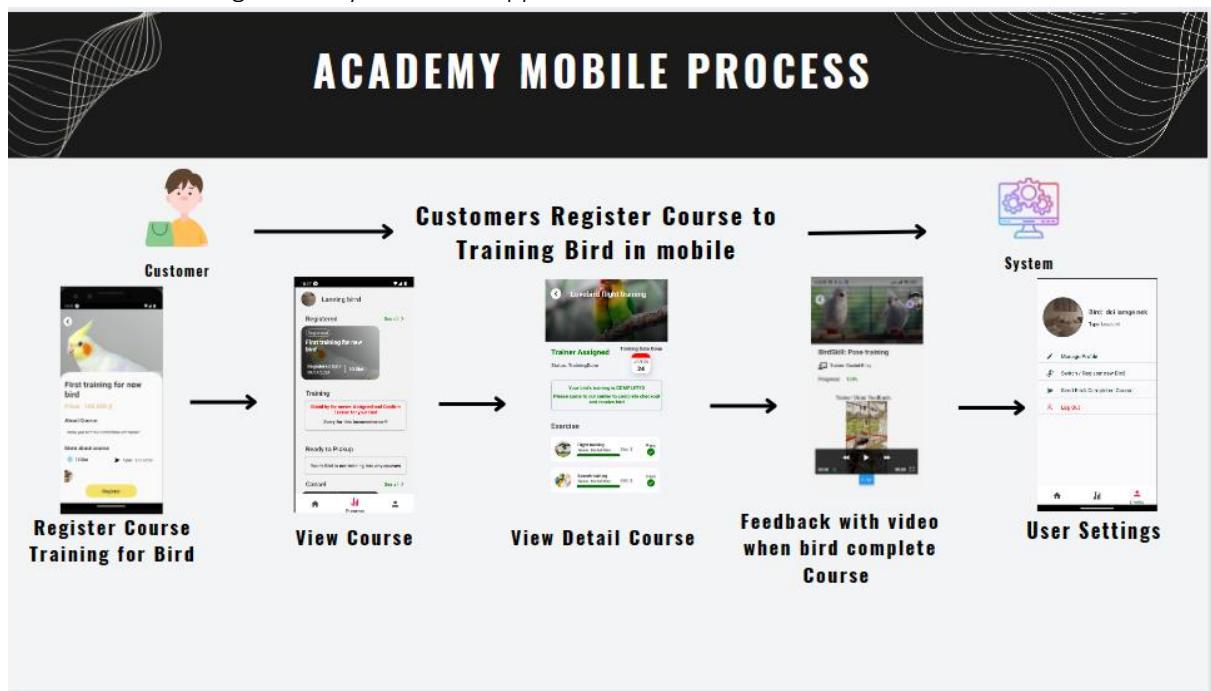
### 3.1.2.3 Online Course Service



3.1.2.4 Bird Training Academy download guidance on website

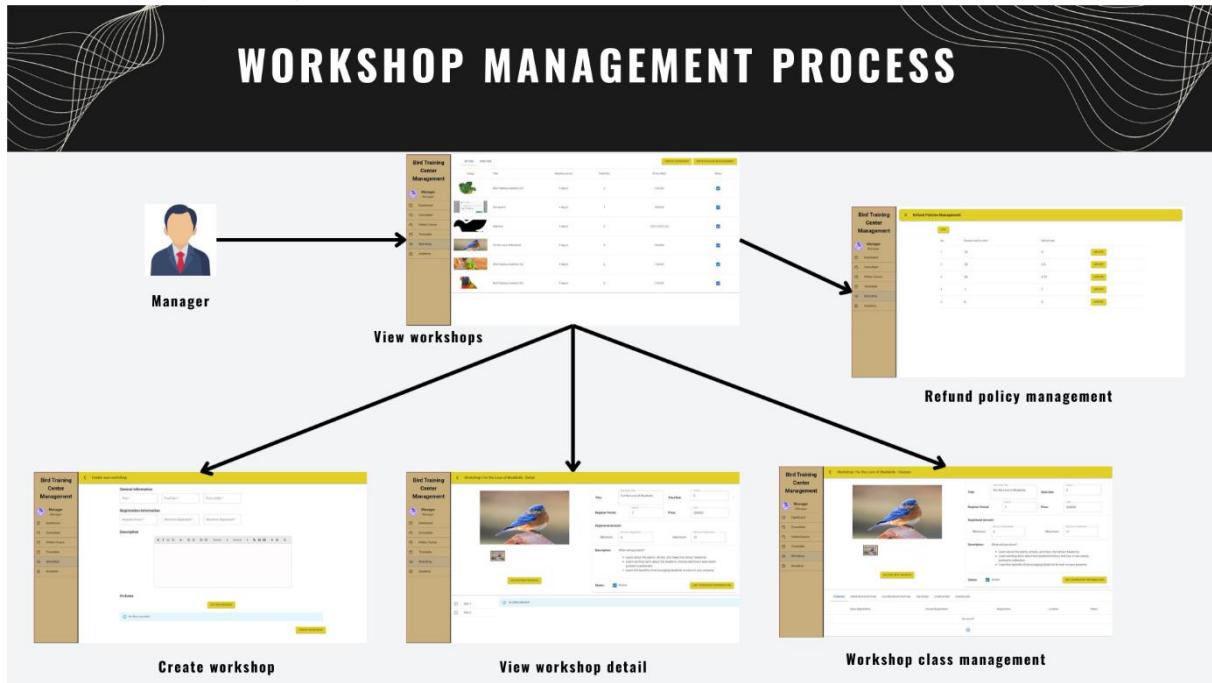


3.1.2.5 Bird Training Academy on mobile application

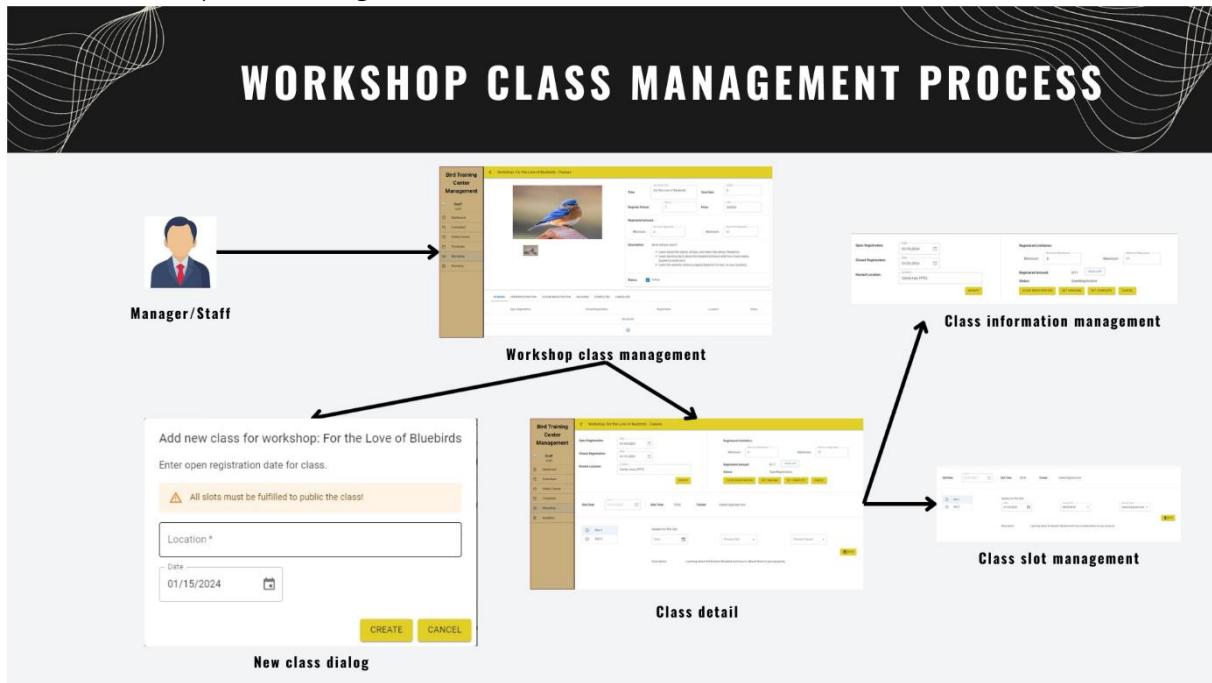


### 3.1.3. Service Management

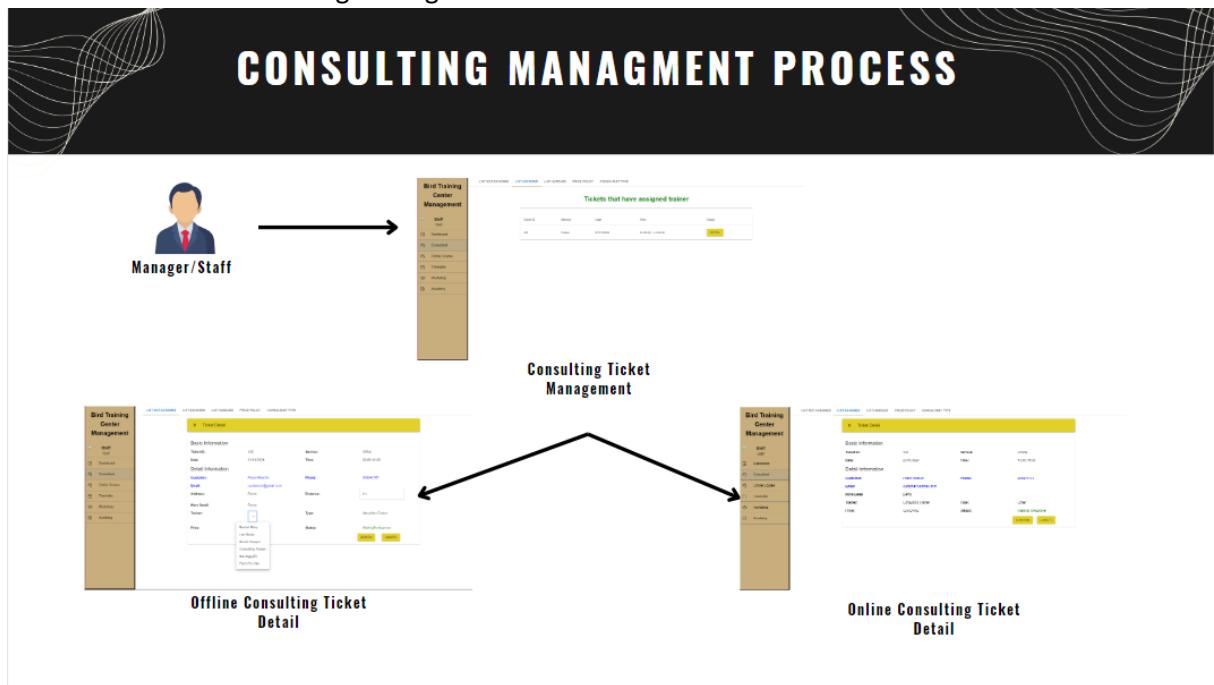
#### 3.1.3.1 Workshop Management



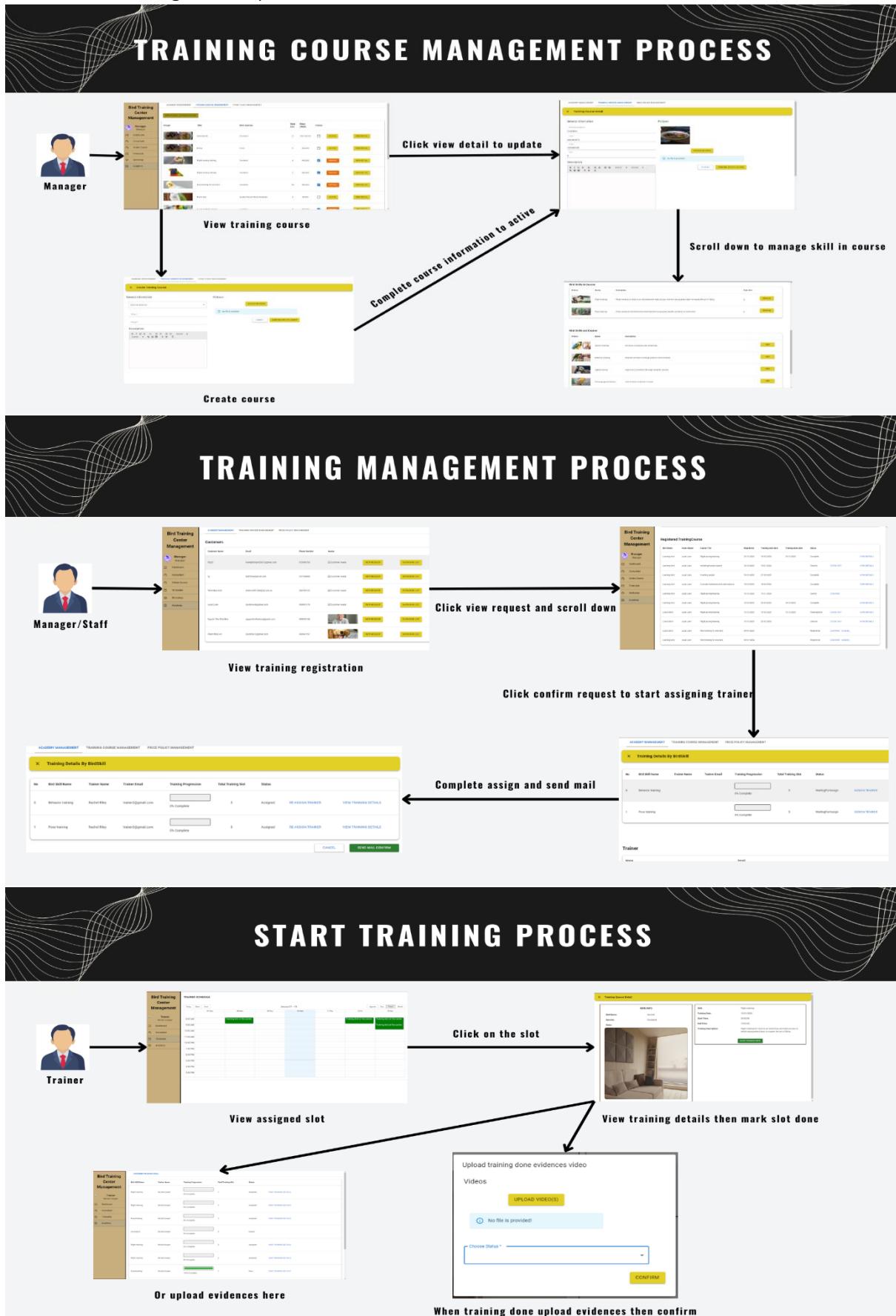
#### 3.1.3.2 Workshop Class Management



### 3.1.3.3 Consultation Booking Management



### 3.1.3.4 Bird Training Academy

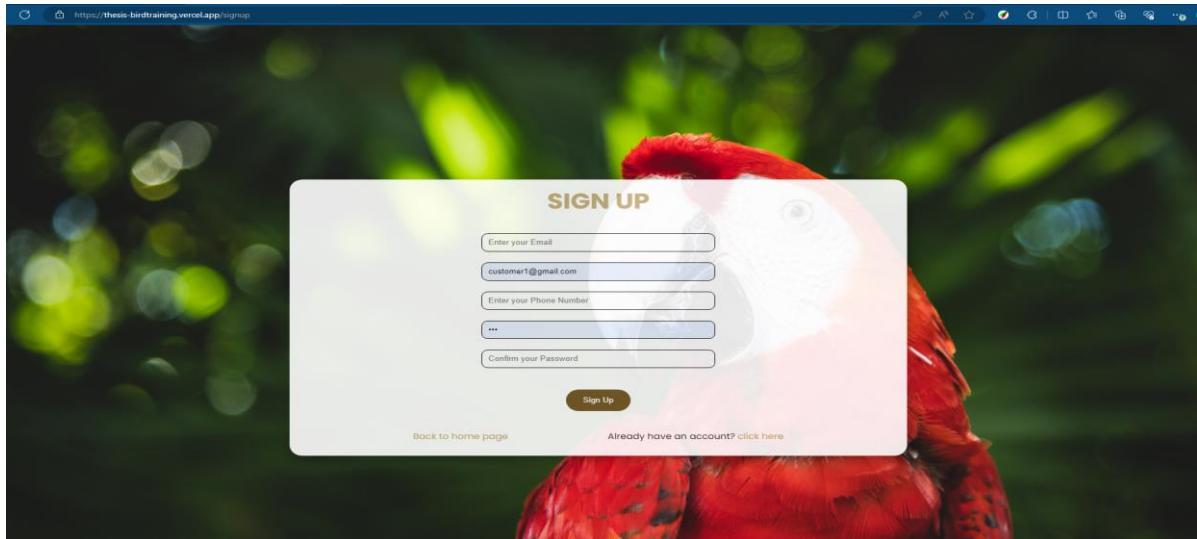


### *3.1.4. Administration management*

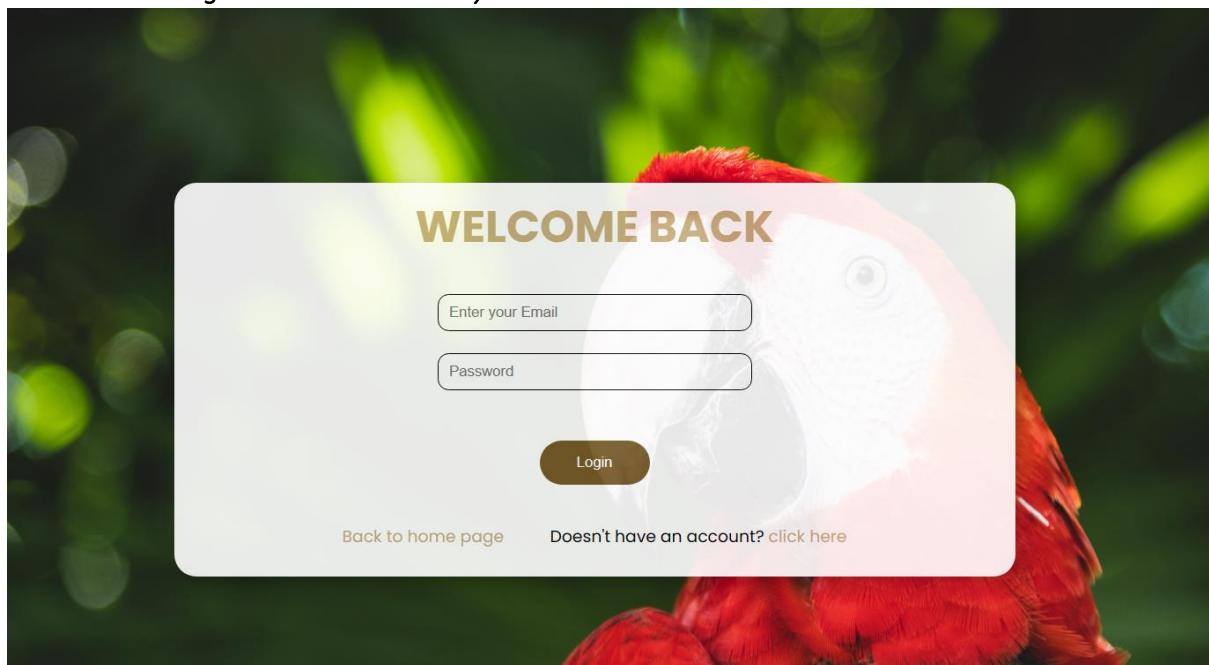
3.2 <Guest> Register and login for authentication and authorization

#### *3.2.1 <Guest> Create Account to Start Login and access to system*

To start access to our website, customer need to create an account from our system



#### *3.2.2 <Guest> Login Account access to system*



1. Enter [customer1@gmail.com](mailto:customer1@gmail.com)
2. Enter password: 123
3. Click Login

### 3.3 <Customer> Customer Service

#### 3.3.1 <Customer> Workshop Service

##### 3.3.1.1 <Customer> Can View Workshop

The screenshot shows a web browser window for 'BirdTrainingCenter'. The URL is https://thesis.birdtraining.vercel.app/workshops. The page features a top navigation bar with links for Home, Courses, Workshops, and Services. On the right, there are user profile icons for 'Hi Pham Nhat An' and 'Setting'. Below the navigation is a large banner image of a colorful parrot. A heading 'Explore our popular Workshops' is centered above four workshop cards. The first card is 'Bird Training Academy' (301 sessions) for 'Advanced Applications and Behaviour Modification' at 100,000 (vnd). The second card is 'tét report 5' (11222 sessions) for 'Advanced Applications and Behaviour Modification' at 200,000 (vnd). The third card is 'ádasdsa' (sadasd sessions) for 'Advanced Applications and Behaviour Modification' at 123,123,123 (vnd). The fourth card is 'For the Love of Bluebirds' for 'Learn about the plants, shrubs, and birds' at 200,000 (vnd).

1. Click Workshops navigation tab
2. View Workshop

##### 3.3.1.2 <Customer> Can View List Workshop and Search Workshop needed

This screenshot shows the same 'Workshops' page as the previous one, but with a search result overlay. The search term 'tét' is highlighted in the search bar. The results show the 'tét report 5' workshop from the previous screenshot, with its details: 11222 sessions, 200,000 (vnd), and a 'Read more' link.

https://thesis-birdtraining.vercelapp.workshoplist

**BirdTrainingCenter** Home Courses Workshops Services Hi Pham Nhat An Setting

bird

Explore our Workshops



**Bird Training Academy 301**

Advanced Applications and Behaviour Modification

100,000 (vnd)

[Read more](#)



**For the Love of Bluebirds**

What will you learn?

- Learn about the plants, shrubs,

200,000 (vnd)

[Read more](#)



**Bird Training Academy 101**

Parrot Behaviour, Body Language and Training Basics

100,000 (vnd)

[Read more](#)



**Bird Training Academy 201**

Foundation Training and Training for Medical Procedures

100,000 (vnd)

[Read more](#)

[Click Here To View Your Workshop](#)

1. Click View More to show list Workshop
2. Search Workshop that you want to join

### 3.3.1.3. <Customer> Can view workshop details, Enroll and start payment to join workshop

https://thesis-birdtraining.vercelapp.workshoplist/1

**BirdTrainingCenter** Home Courses Workshops Services Hi Pham Nhat An Setting

**Bird Training Academy 101**



**Description:**  
An in-depth look at wild and captive parrot behaviour, how it changes in captivity, and how to read it. Setting up for desired behaviour including diet, environment and interactions. Applied Behaviour Analysis, training theory, principles and basic applications.

**Start-Time:** 08:00:00 / **End-Time:** 09:00:00  
**Date:** January 16th, 2024

**Trainer:** Rachel Riley



**Description:**  
Applying basic training with live birds.

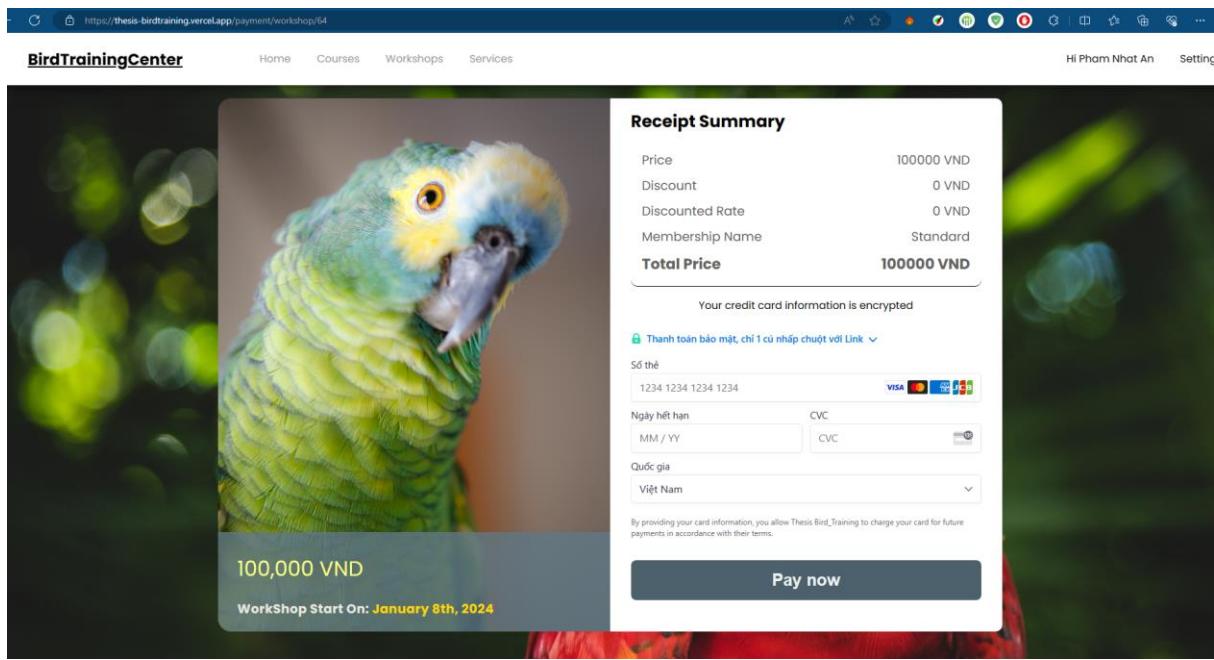
**Start-Time:** 09:00:00 / **End-Time:** 10:00:00

[Close](#) [ENROLL NOW](#)

**Contact Us**

Hotline: 1800 152508

Address: Block E2a-7, D1 Street Saigon Hi-tech Park, Long Thanh My Ward, District 9, Ho Chi Minh City, Vietnam



1. Click on any Workshop
2. Click “Enroll Now”
3. Fill data payment card number 4242 4242 4242 4242
4. expires date, cvc can be input manually
5. Click Pay now

3.3.1.4. <Customer> Can view Workshop registered to check status

The screenshot shows a dashboard titled "Your Registered Workshop Classes" with four sections:

- On going**: Displays the message "No class found!"
- Close Registration**: Displays the message "No class found!"
- Open Registration**: Displays the message "No class found!"
- Completed**: This section is currently empty.

1. Click Workshop
2. Scroll to Your Registered Workshop Classes

### 3.3.2 <Customer> Advice Consultation Booking

3.3.2.1 <Customer> Online Booking Consultant Appointment

- Click Consultants, scroll down the form, fill the form information then click confirm to send the Consulting Ticket

## CONSULTATION SERVICE



### In-Home Consultation

Available for clients within the Perth, South East Queensland and Hawkes Bay (New Zealand) and surrounding regions. If you are unsure if your area is covered please get in touch. Initial 1 hour in-home consultation is \$210\*. You have the option to include a written Parrot Life Training Plan specific to your parrot for an additional \$65.00 (recommended). Any optional follow up or regular consultations for existing clients are discounted to \$165.00 per session, these normally last 1 hour. Parrot Life Training Plans for follow up sessions are optional and are an extra \$65.00. For Western Australia, Queensland and New Zealand, a distance fee of 50c/km applies over 20km from your consultant's location. If the distance from our offices is greater than 100km, you will be given a direct quote from your consultant. \*Prices are in local currency

1. Scroll down to see the consulting ticket form.

## BOOKING ONLINE

- ① Select a service
  - In-home consultation
  - Online consultation
- CONTINUE**   **BACK**
  
- ② Which Trainer would you like to make an appointment with?
  
- ③ Select Date & Time
  
- ④ Please enter contact information
  - Last step

2. Select service.

## BOOKING ONLINE

- Select a service (Online Consultation)
  
- ② Which Trainer would you like to make an appointment with?
  - Bảo Nguyễn
  - Thích Tu Văn
  - Consulting Trainer**
  - Rachel Riley
  - Other
- CONTINUE**   **BACK**
  
- ③ Select Date & Time
  
- ④ Please enter contact information
  - Last step

3. Select trainer.

---

## BOOKING ONLINE

✓ Select a service (Online Consultation)

✓ Which Trainer would you like to make an appointment with?

1 Select Date & Time

Select Date  
01/31/2024

Time  
08:00  
09:00  
10:00  
11:00  
13:00  
14:00  
15:00  
16:00

4. Select Date & Time.

---

## BOOKING ONLINE

✓ Select a service (Online Consultation)

✓ Which Trainer would you like to make an appointment with?

✓ Select Date & Time

Please enter contact information

Last step

Consulting Detail (Optional)  
I need to help about my parrot

Address (Optional)

Consulting Type  
Training bird at home

**CONFIRM**   **BACK**

5. Fill the Consulting Detail (Optional) and Address (Not Optional if selected Service is In-Home Consultation), then click CONFIRM to send the ticket.

### 3.3.3 <Customer> Online Course Service

#### 3.3.3.1 <Customer> View online courses

- Click courses, search Courses on the search bar when click in View More Courses to search needed course

**BirdTrainingCenter**

Home Courses Workshops Services Hi Pham Nhat An Setting

Top of our categories  
Explore our popular courses

**testing report 5**  
aaaaaaa test report dài quá  
20,000 (vnd)  
Read more

**Course Test 1**  
This is Course Test 1  
20,000 (vnd)  
Read more

**testing tao moi tiep**  
thanhthienhoan  
200,000 (vnd)  
Read more

**MINI COURSE - Target Training**  
Get lifetime access to our FREE mini,  
10,000 (vnd)  
Read more

[Click Here To View Your Finished Courses](#)

[VIEW MORE COURSES](#)

<http://thanh-thienhoan.com/courses/test/>

**BirdTrainingCenter**

Home Courses Workshops Services Hi Pham Nhat An Setting

Explore our Courses

**course test 1**

**Course Test 1**  
This is Course Test 1  
20,000 (vnd)  
Read more

[Click Here To View Your Finished Courses](#)

1. Click tab Courses,
2. Click “View More Courses”
3. Start Search Courses

### 3.3.3.2 <Customer> View course detail, enroll course and complete payment online

The screenshot shows two pages from the BirdTrainingCenter website.

**Course Detail Page:**

- Title:** testing tao moi tiep
- Price:** 200,000\$
- Enroll Now:** A button to start the enrollment process.
- Skill level:** Basic
- Enrolled:** 100
- For Details About The Course:** A link to get more information.
- Call Us:** (+84) 90 456 0264
- What You'll Learn:** A section with the text "thonhotnhotnh".
- Some Images Of The Course:** An image of a person in a colorful kimono holding fireworks.
- Requirements:**
  - Become an advanced, confident, and modern JavaScript developer from scratch.
  - Have an intermediate skill level of Python programming.
  - Have a portfolio of various data analysis projects.
  - Use the numpy library to create and manipulate arrays.

**Payment Summary Page:**

- Receipt Summary:**

Price	200000 VND
Discount	0 VND
Discounted Rate	0 VND
Membership Name	Standard
<b>Total Price</b>	<b>200000 VND</b>
- Card Information:**
  - Your credit card information is encrypted.
  - Secure, 1-click checkout with Link
  - Card number: 1234 1234 1234 1234
  - Expiration: MM / YY
  - CVC
  - Country: Vietnam
- Pay now:** A large button to complete the payment.

1. Click course so as to view details
2. Click “Enroll Now”
3. Fill data payment card number 4242 4242 4242 4242
4. expires date, cvc can be input manually
5. Click Pay now

### 3.3.3.3 <Customer> Start Study with course after complete payment

The screenshot shows a course titled "testing tao moi tiep" with a price of 200,000 VND. It has a skill level of "Basic" and 100 enrolled students. A "STUDY" button is present. To the right, under "What You'll Learn", is the text "thonhotnhoithnh". Below that is a section titled "Some Images Of The Course" featuring an image of a person in a traditional kimono. At the bottom, there is a "Requirements" section.

The screenshot shows a "MINI COURSE - Target Training" section. On the left, there is a list of training steps: Step 1 (Completed), Step 2 (Completed), Step 3 (Completed), Step 4 (Completed), Step 5 (Completed), and Step 6 (Completed). On the right, a video player displays "Step 1:" with the following text: "Start with your bird in an environment in which they feel most comfortable. For most this will be their cage or perhaps a play stand." The video player shows a progress bar at 0:00 / 0:26.

1. After Complete Payment Purchase, Click back to the Course you proceed
2. Click Study
3. Start Online Study

### 3.3.3.4 <Customer> Complete studying course and receive course certificate

**BirdTrainingCenter**

Home Courses Workshops Services

Hi Pham Nhat An Setting

## MINI COURSE - Target Training

Completed

Introduction

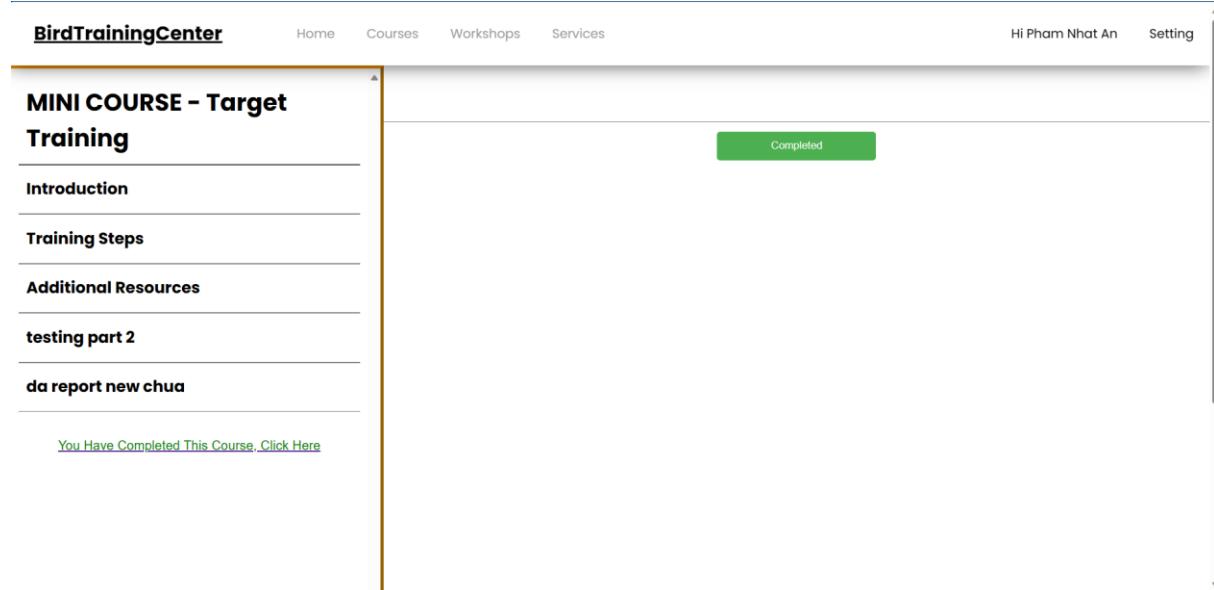
Training Steps

Additional Resources

testing part 2

da report new chua

You Have Completed This Course, Click Here



**BirdTrainingCenter**

Home Courses Workshops Services

Hi Pham Nhat An Setting

[Back to Online Course List Page](#)

[DOWNLOAD CERTIFICATE](#)

**About the Course**

Get lifetime access to our FREE mini, step-by-step course to 'Target Train' your parrot! In this short course, we cover 6 steps to train your bird to engage in a basic targeting behaviour. The content included in this course is not for reproduction or distribution and remains the intellectual property of Parrot Life Behaviour and Training and should under no circumstances be shared with anyone else.

**Summarize**

Welcome to our 'Target Training Mini Course'! In this short course, you will learn how to teach your bird to target in 6 easy steps! Target training is a core foundation skill. Target training is a great first behaviour to work on with our parrots as it is simple to train, easy to learn and forgiving for new trainers and naive learners. It is also the springboard to many other foundation, novel and advanced behaviours including stationing, crate training, recall and many more!

Welcome to the heart of our Target Training course, where we break down the training process into clear and manageable steps. Target training is a powerful method to enhance

**DIPLOMA CERTIFICATE**

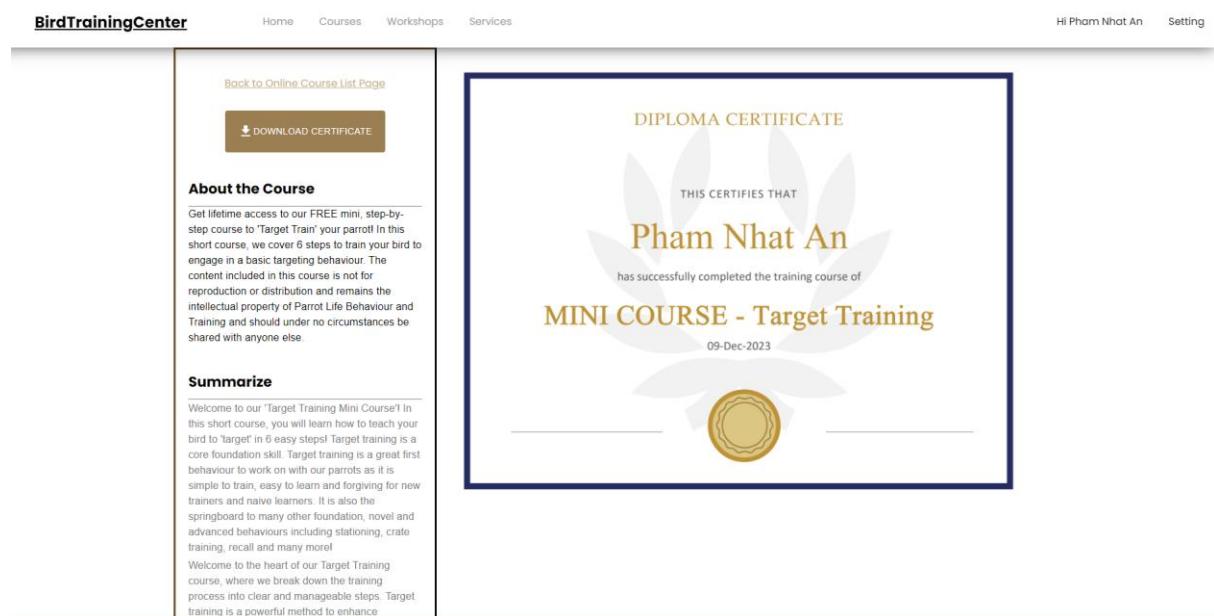
THIS CERTIFIES THAT

**Pham Nhat An**

has successfully completed the training course of

**MINI COURSE - Target Training**

09-Dec-2023



1. After study completely
2. Click "You Have Completed this Course to show Certificate"

### 3.3.4 <Customer> Bird Training Academy Download Guidance on Website

#### 3.3.4.1 <Customer> access to Consultant through navigation bar

The screenshot shows the 'BirdTrainingCenter' website. At the top, there is a navigation bar with links for Home, Courses, Workshops, Services, Hi Pham Nhat An, and Setting. The main content area has a title 'TRAINING ACADEMY' and a large image of a man in a cowboy hat standing next to a large blue and red parrot perched on a branch. To the right of the image is a section titled 'Our Training Academy' with descriptive text about the series of workshops. Below the text is a yellow button labeled 'Contact Us To Book Now'.

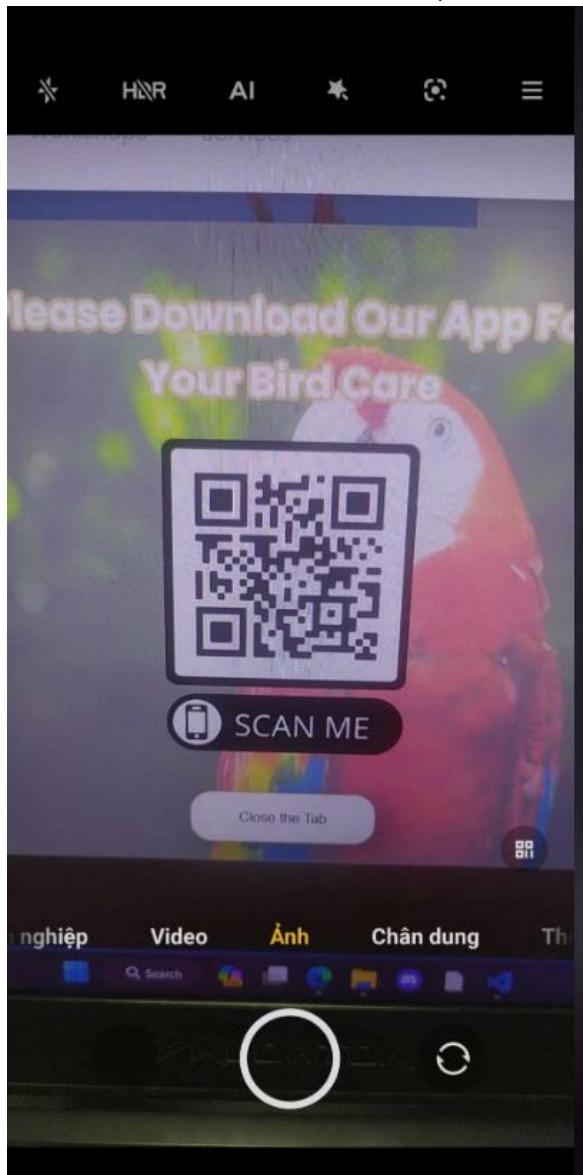
1. Hover in services and Click Bird Training Academy
2. View Bird Training Academy

#### 3.3.4.2 <Customer> Click Contact us to Book Now and scan QR code to start download app

The screenshot shows the same website as above, but with a pop-up dialog box overlaid. The dialog has a background image of a parrot and contains text encouraging users to download the app for bird care, a QR code with a 'SCAN ME' button, and a 'Close the Tab' button. The main content area below the dialog is partially visible, showing the 'Our Training Academy' section and the 'Contact Us To Book Now' button.

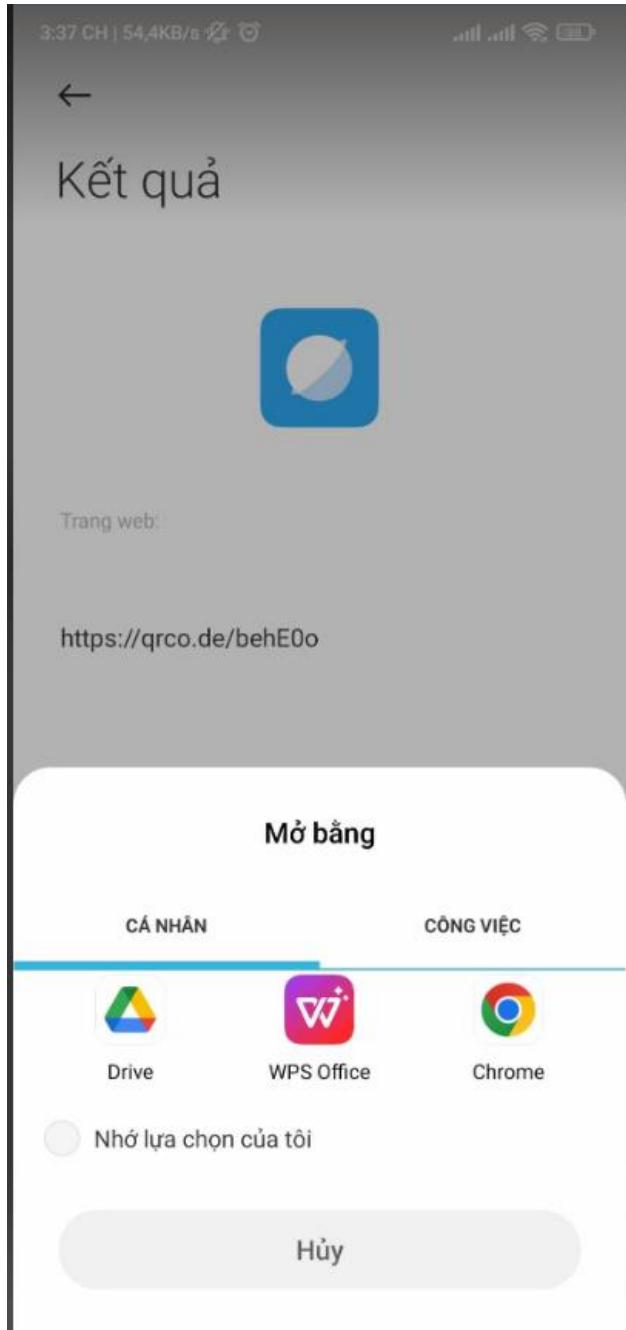
1. Click Contact Us To Book Now
2. Pop up Dialog
3. Use Phone camera and scan QR code

3.3.4.3. <Customer> Scan QR code by camera and click into website



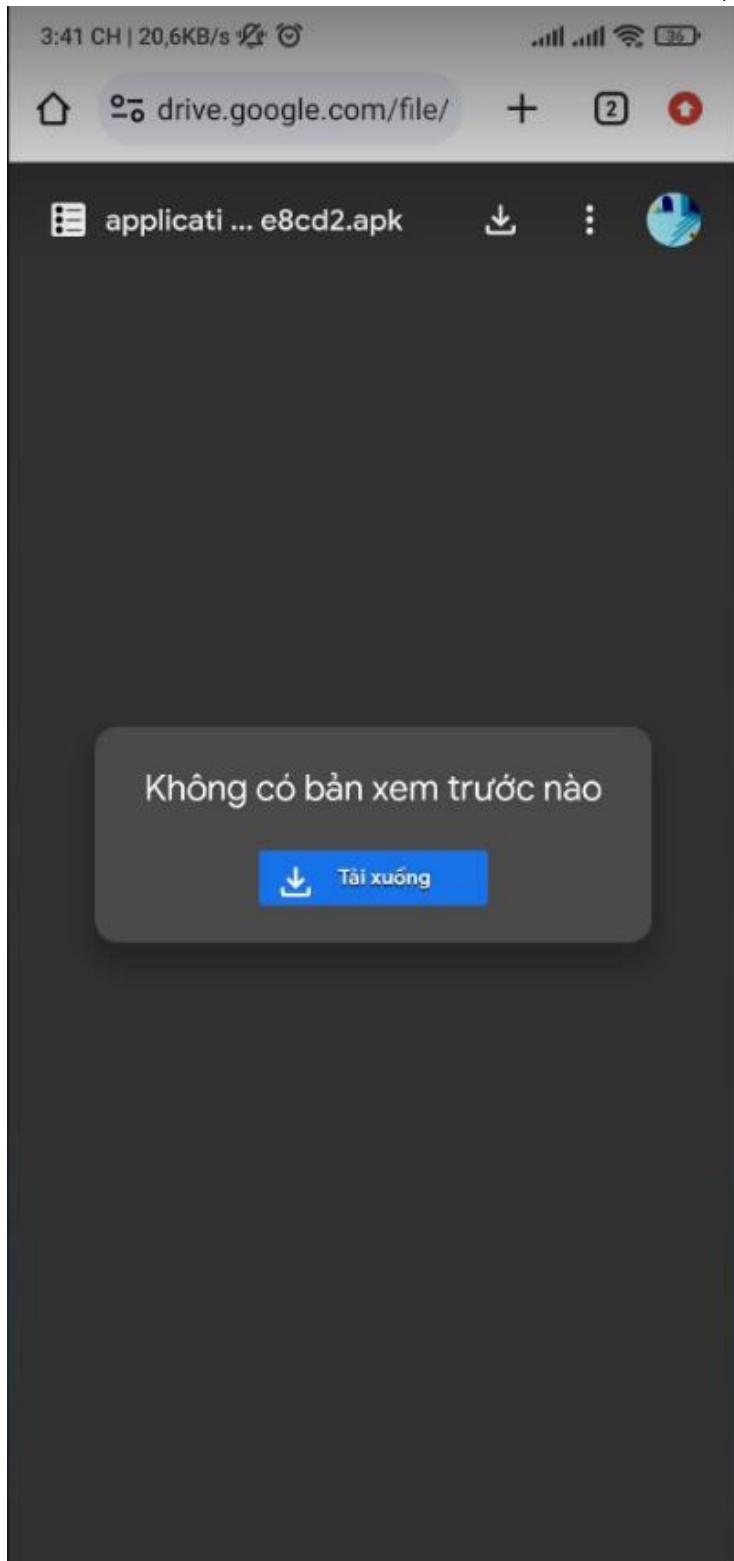
1. Click into icon  to start access QR code website

3.3.4.4 <Customer> Click into website and press Chrome



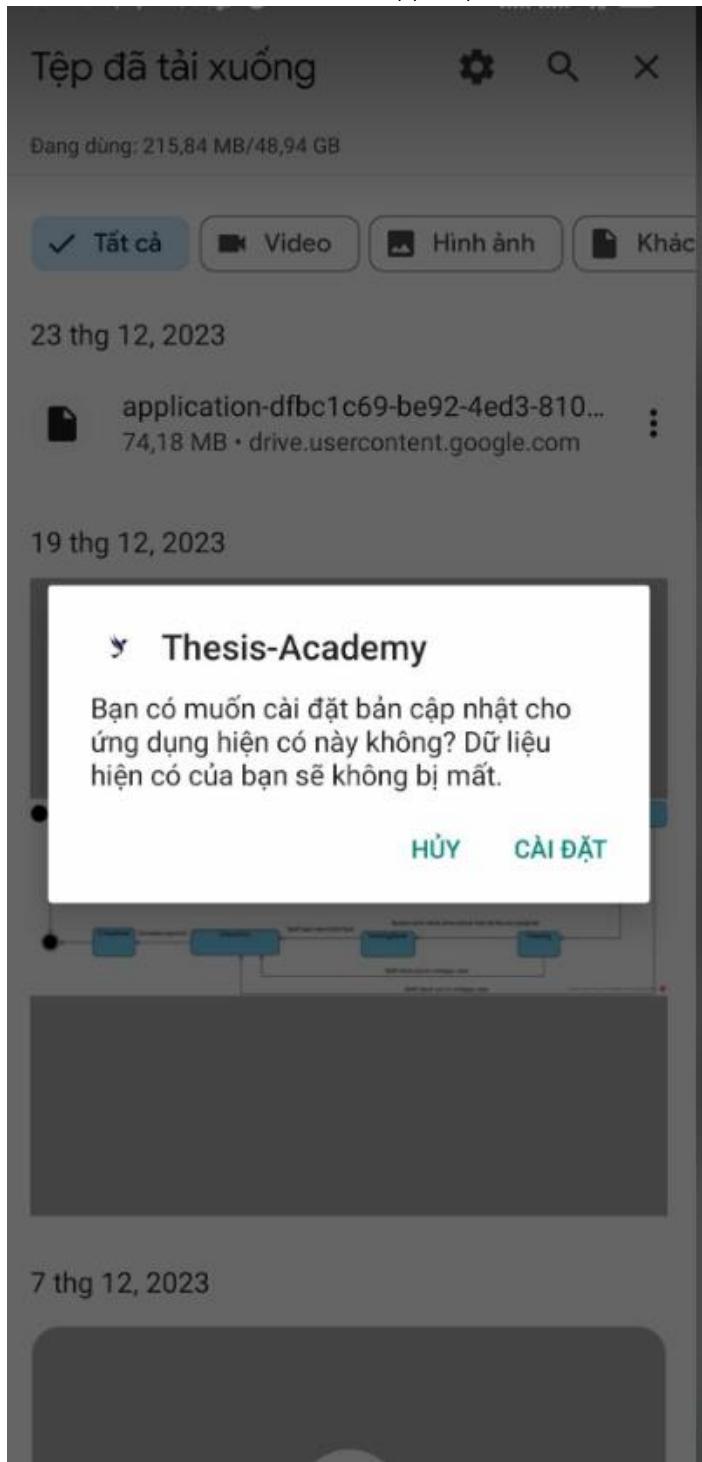
1. After process 3.3.4.3 finish, Pop up screen above, Press open by Chrome

3.3.4.5. <Customer> Click Download and start to download app apk



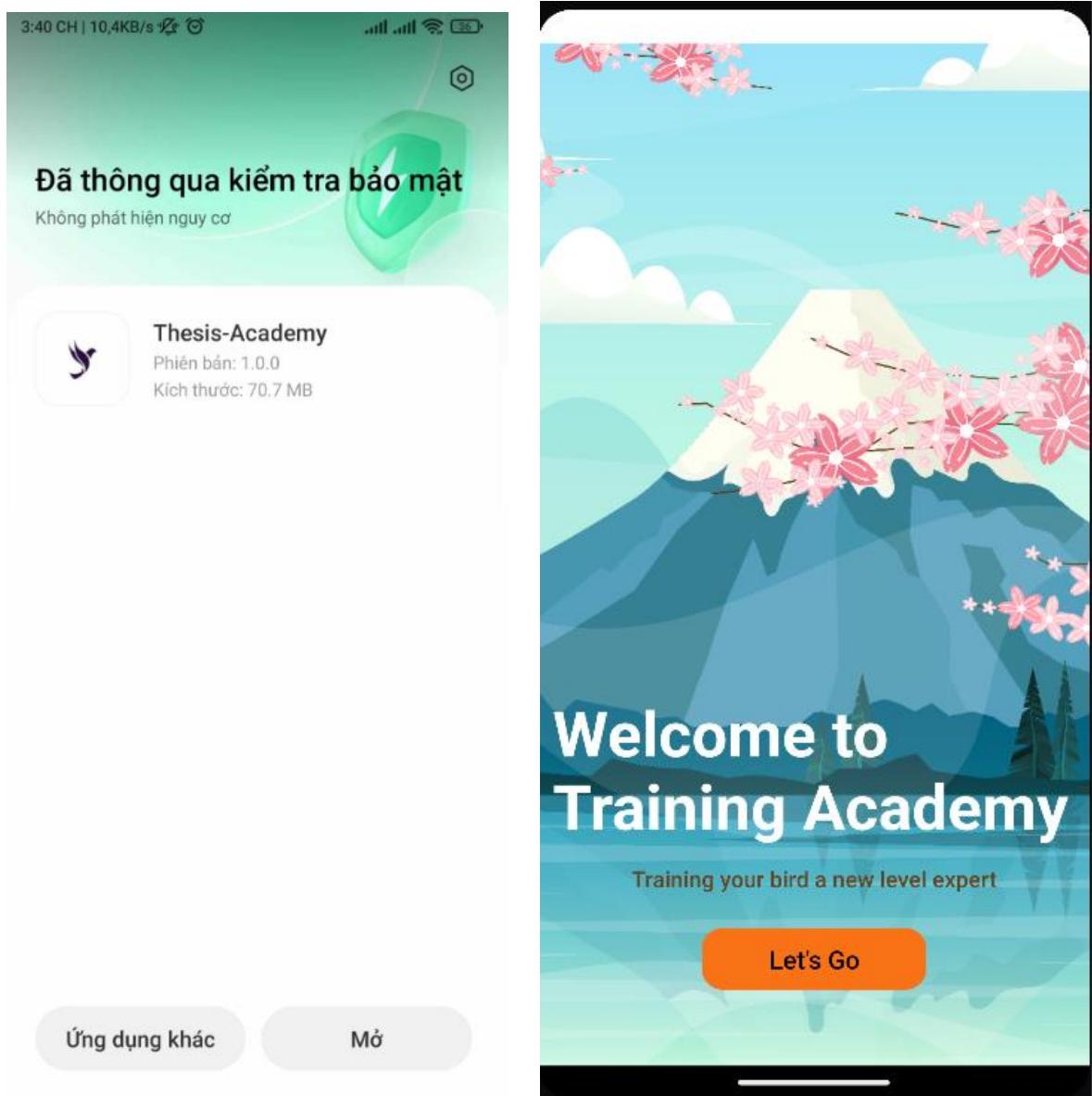
1. After 3.3.4.4 Complete , navigate to above picture
2. Click Download ( Tải xuống) to start download app

3.3.4.6 <Customer> start install app to phone



1. Press install and start to install to phone

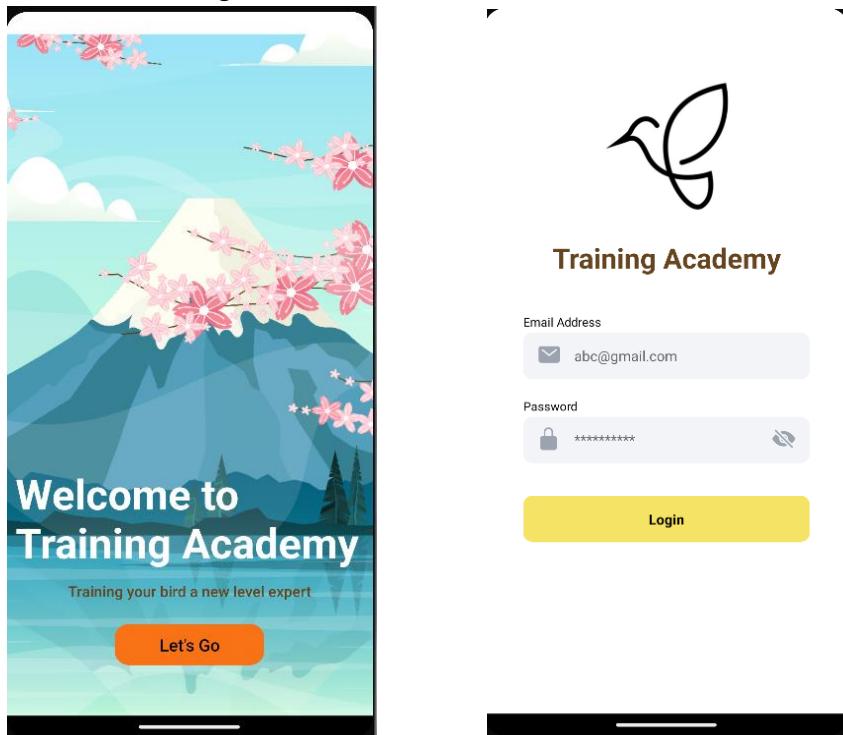
3.3.4.7 <Customer> after download success, start to open the app



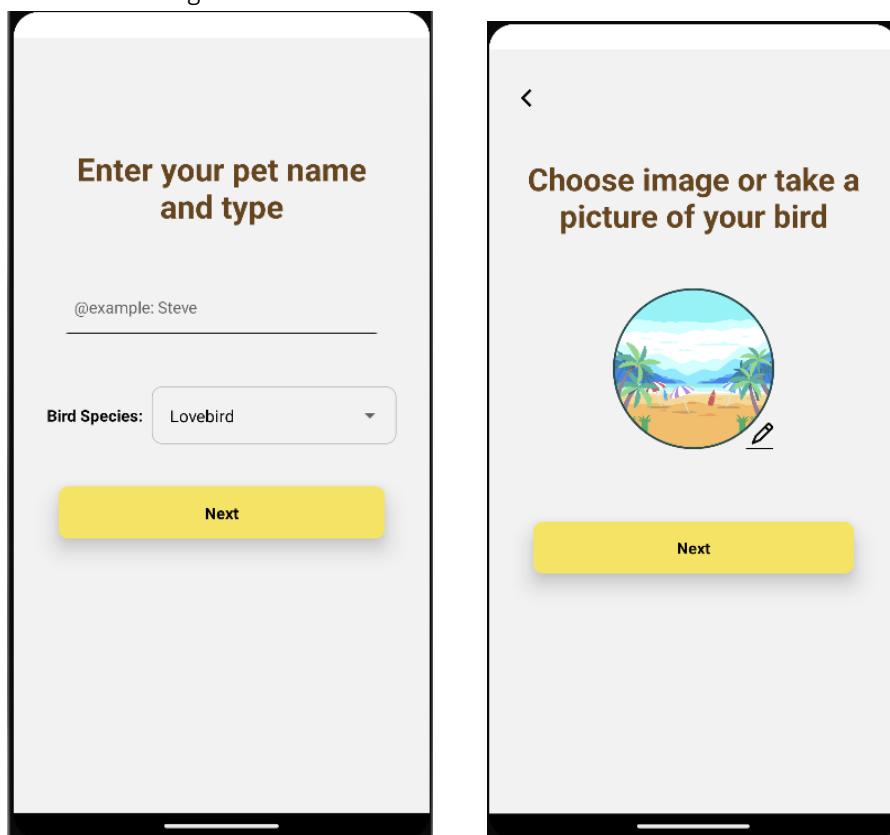
1. Click install successfully
2. Open the app and start to use

### 3.3.5 <Customer> Bird Training Academy Mobile Application

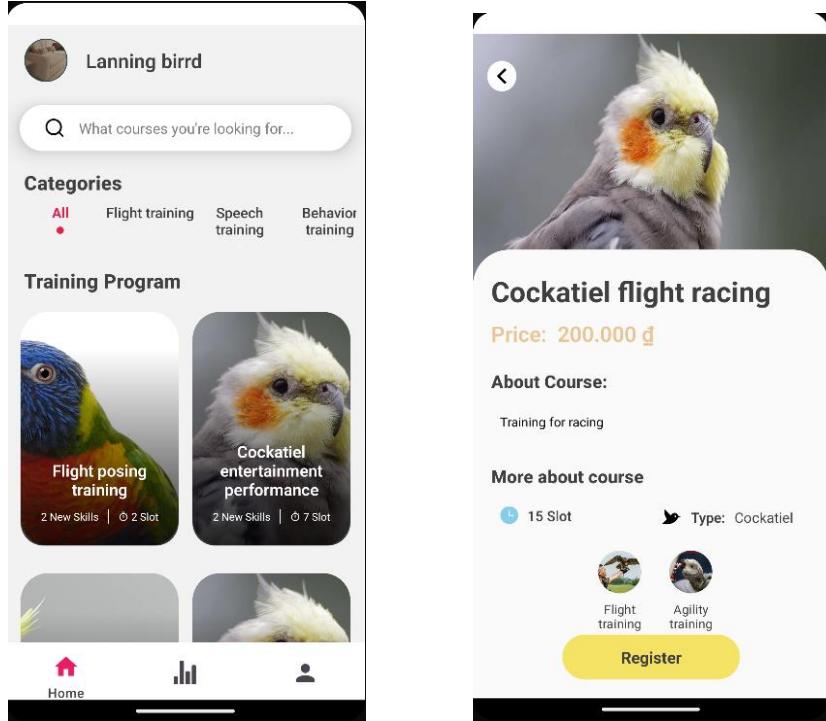
3.3.5.1: <Customer> after create account in our website, can only use that account to login and use our mobile for bird training



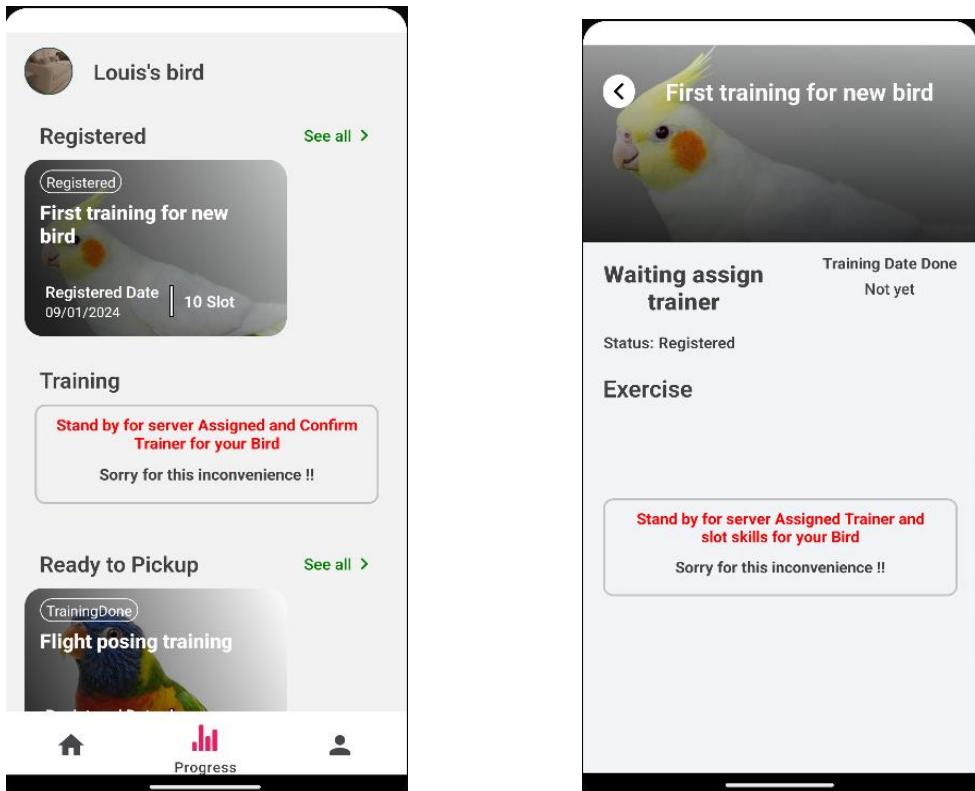
3.3.5.2. <Customer> Register Bird for new account



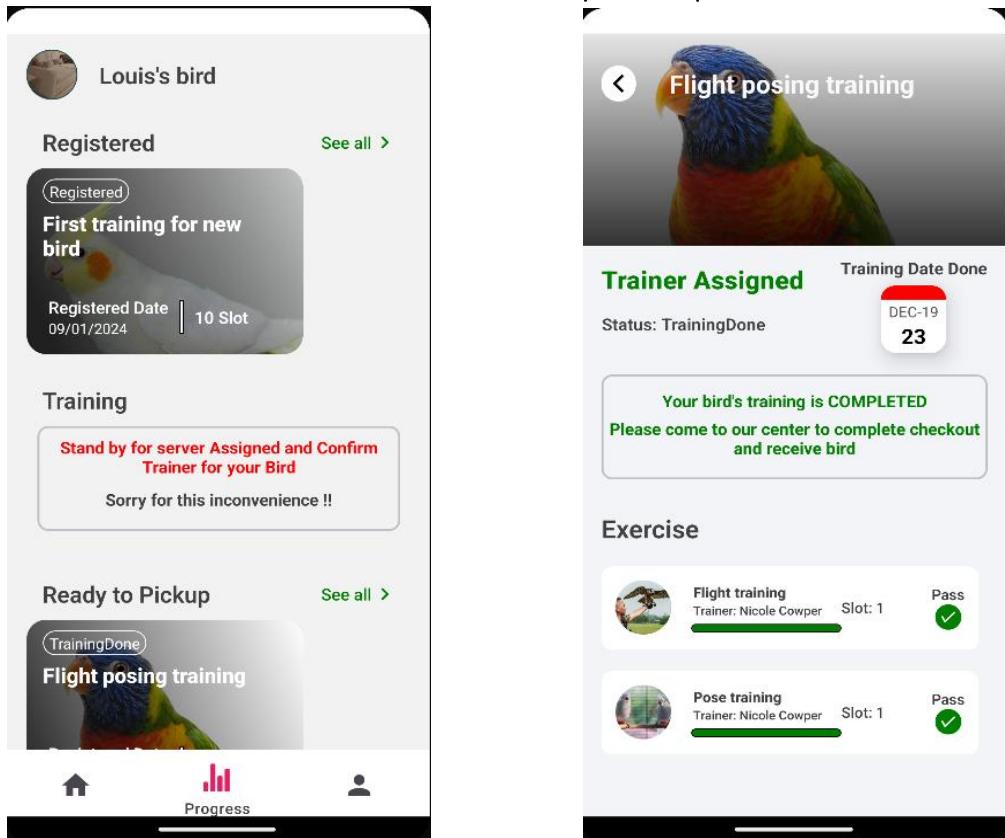
3.3.5.3 <Customer> Register course to start training for Bird



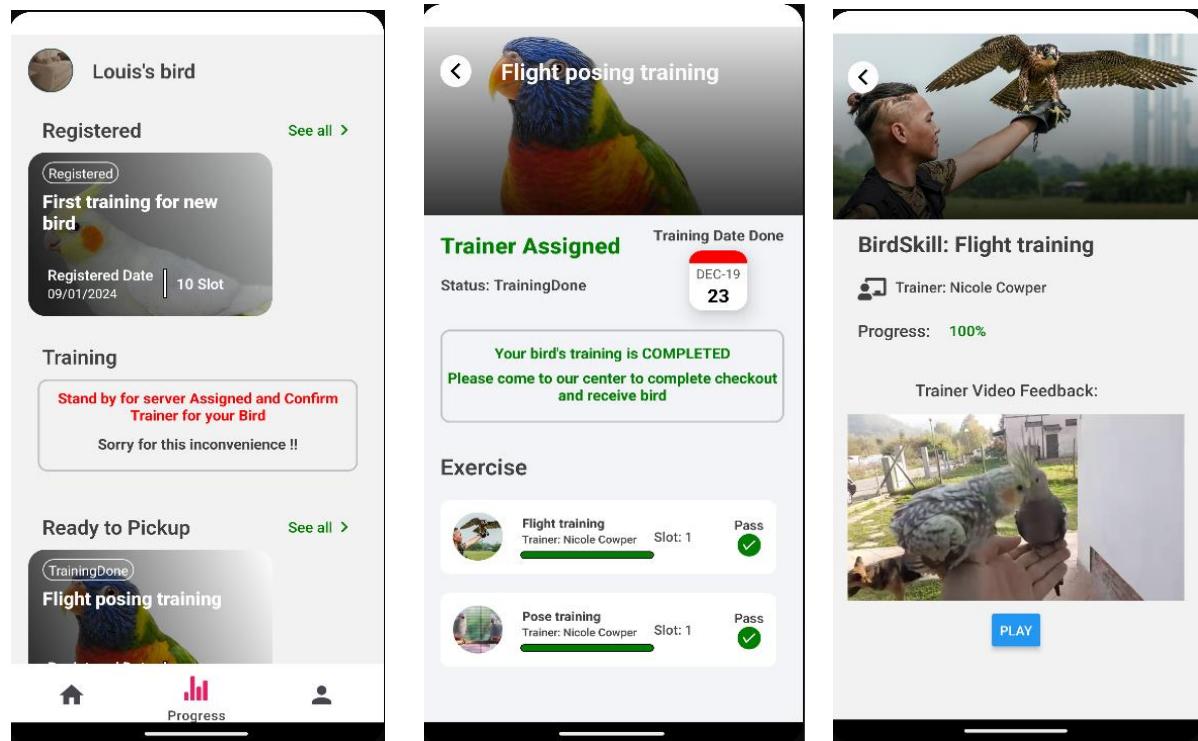
3.3.5.4 <Customer> can view list course with status **Registered** and click to view details of that registered course



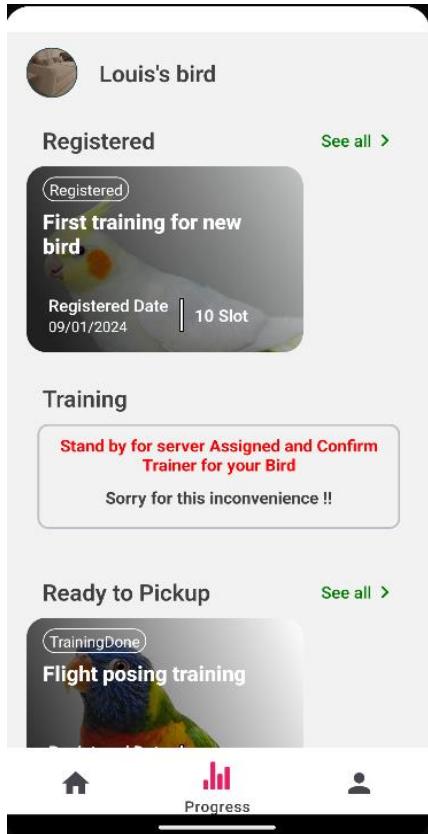
3.3.5.5 <Customer> can view list course with status Ready to Pickup



3.3.5.6. <Customer> can view detail exercise in course status Ready to Pickup



3.3.5.7 <Customer> can view list course with status **Training**



3.3.5.8 <Customer> can view list course with status **Cancel**

Lanning birrd

Sorry for this inconvenience !!

Ready to Pickup

Your's Bird is not training into any courses

Cancel See all >

(Cancel) Flight posing training

4 total Slot

Bird's Certificate

Earn certificate when your bird done training

Flight training Trainer: Nicole Cowper Slot: 2

Pose training Trainer: Bao Nguyen Slot: 2

Progress

Flight posing training

< Cancel Training Date Done Not yet Status: Cancel Exercise

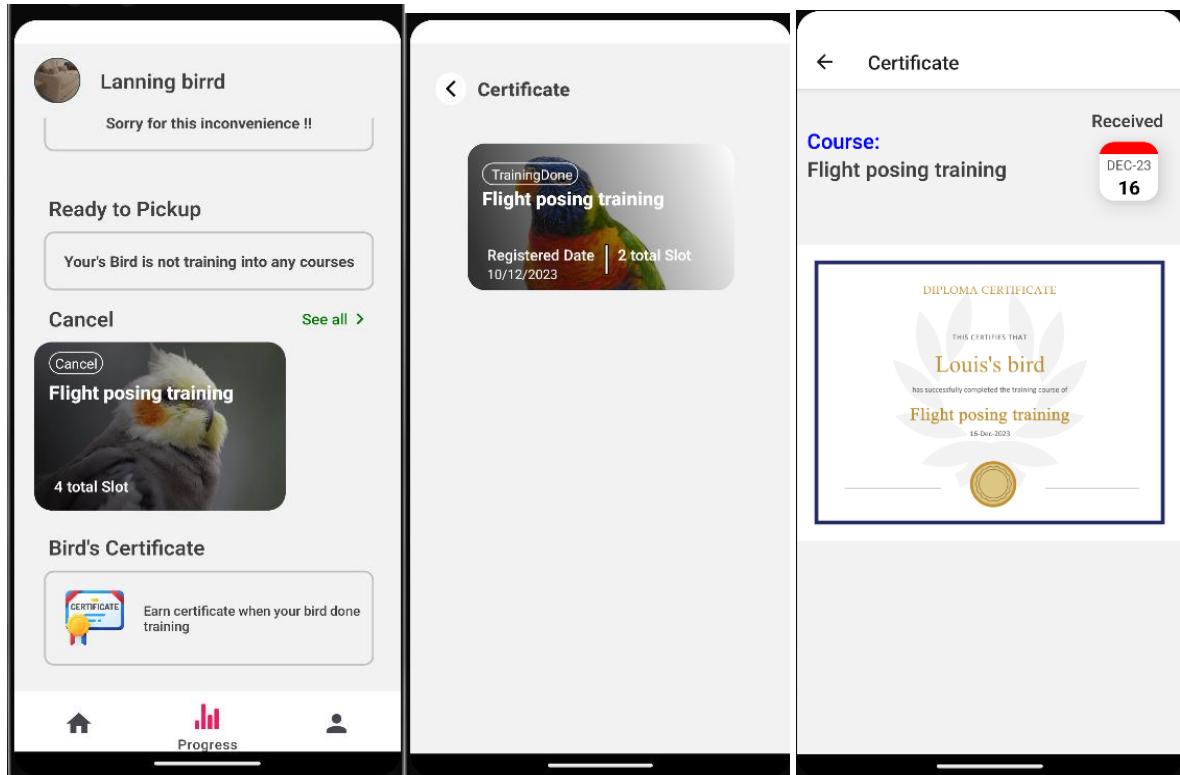
Flight training Trainer: Nicole Cowper Slot: 2

Pose training Trainer: Bao Nguyen Slot: 2

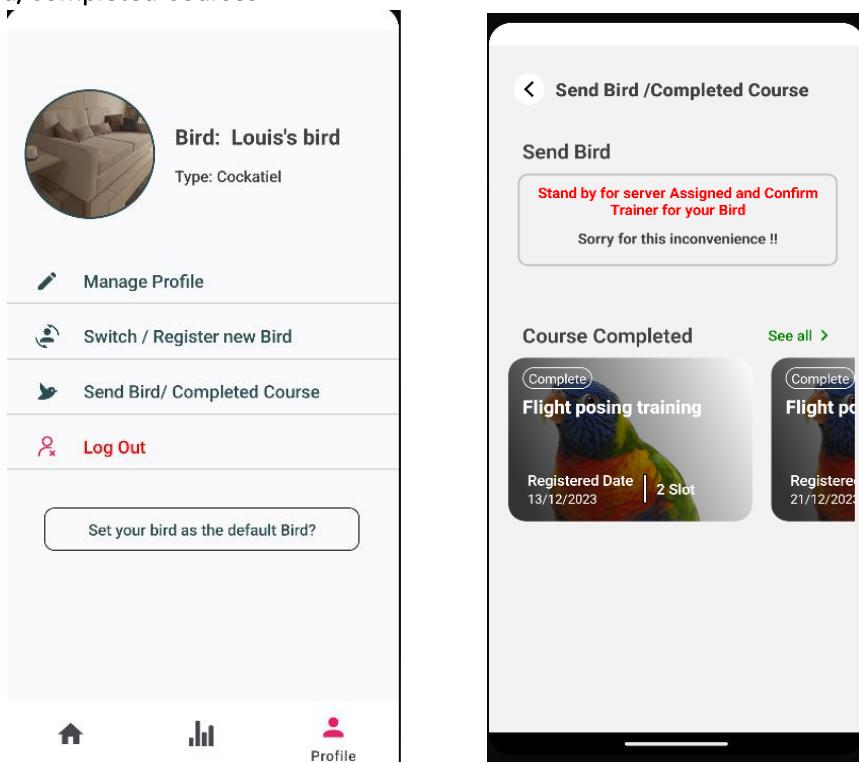
Progress: 0%

Your's Bird is still training this skills No feedback yet

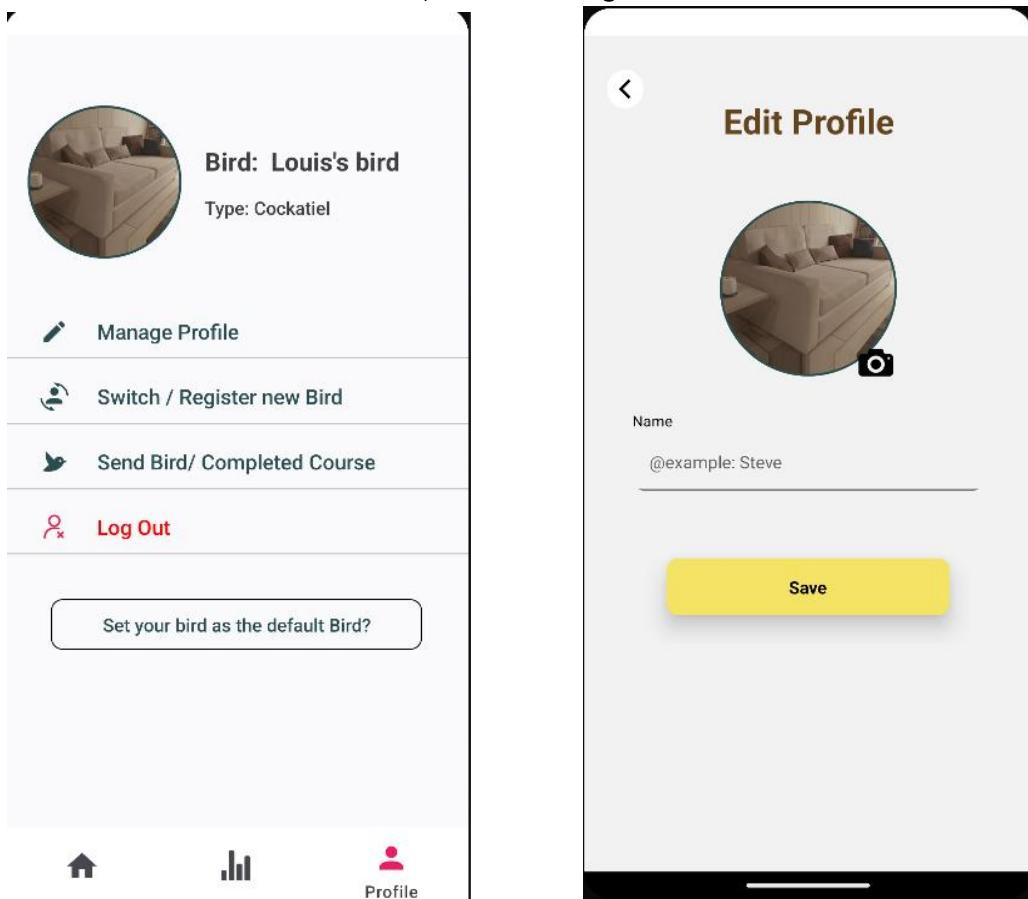
3.3.5.9 <Customer> after there bird complete training can view certificate in **Bird's Certificate**



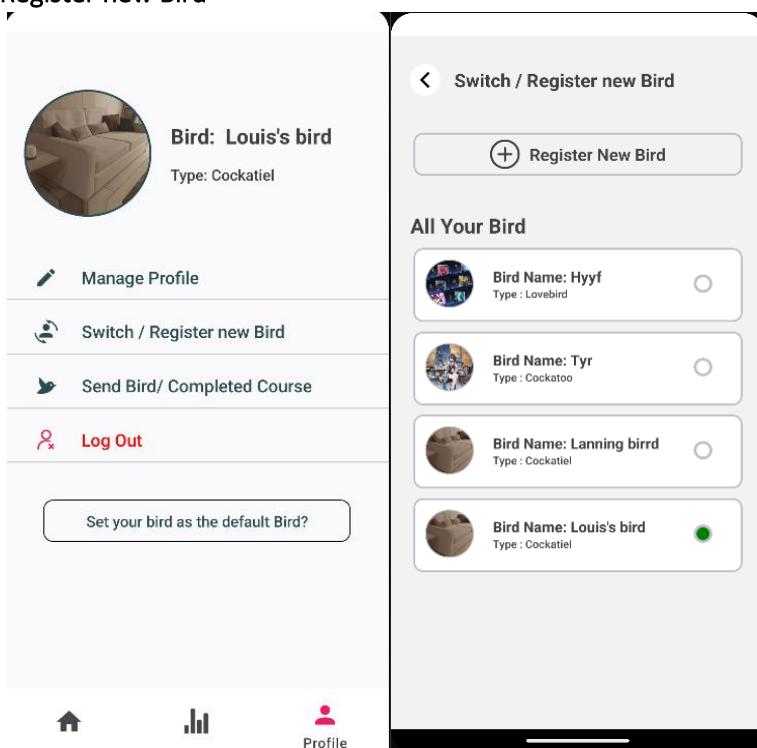
3.3.5.10. <Customer> in profile can view list send bird or completed Course in section **SendBird/Completed Courses**



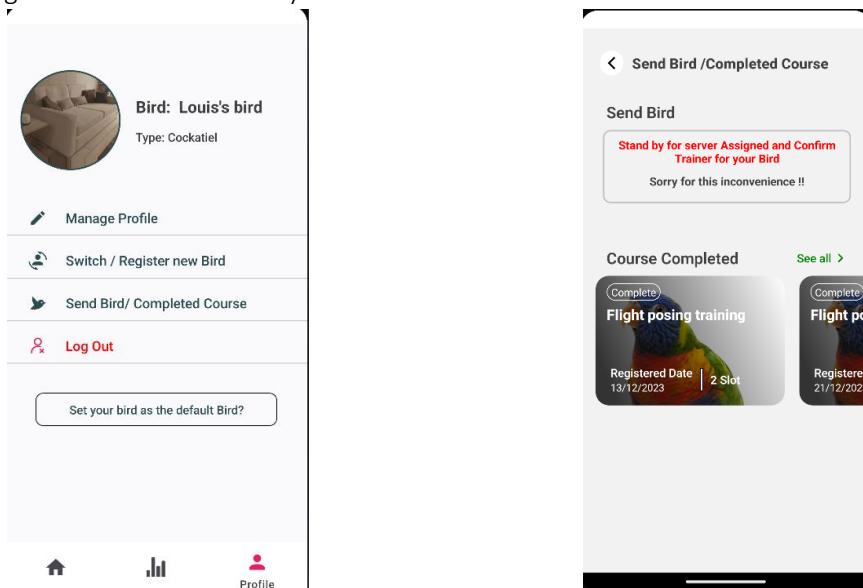
3.3.5.11. <Customer> can edit Bird profile in **Manage Profile**



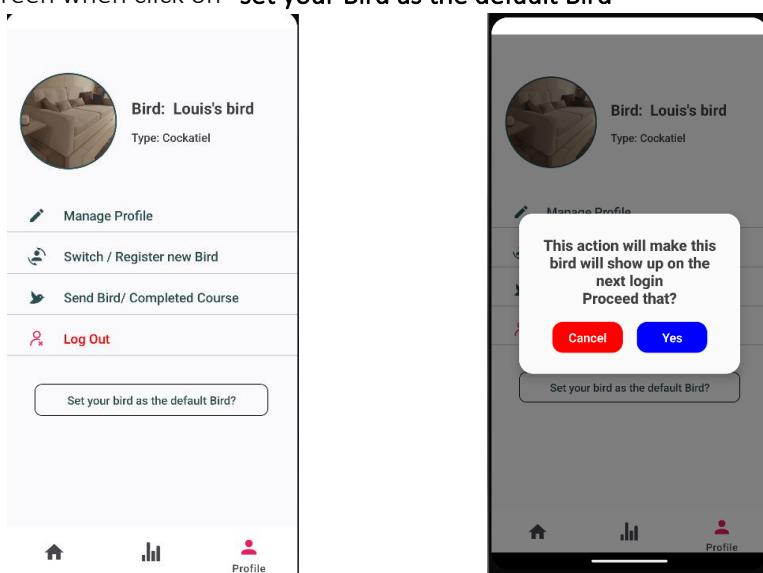
3.3.5.12. <Customer> can switch bird they already create before or Register New one in **Switch/Register new Bird**



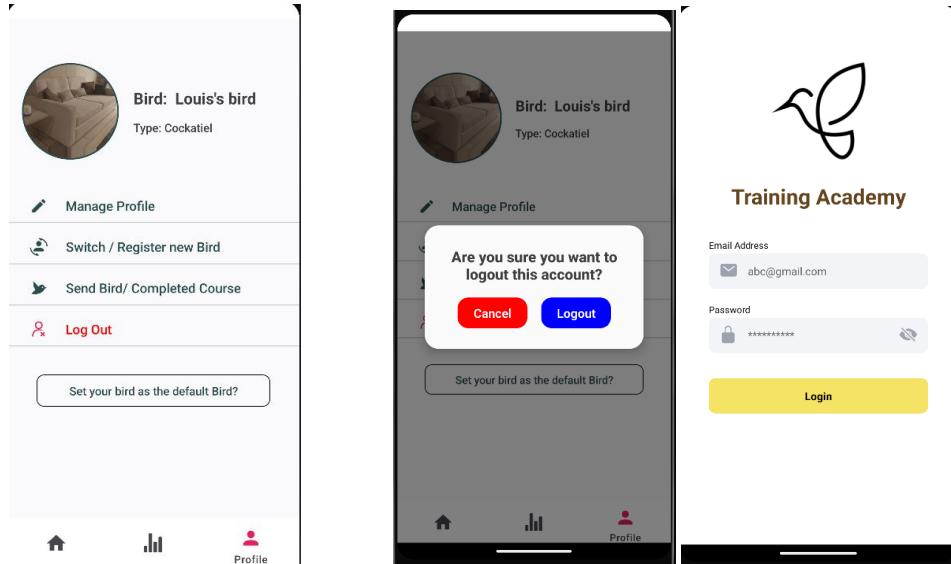
3.3.5.13. <Customer> can check if they can ready to bring the bird to center and all Complete Course including return Bird Successfully



3.3.5.14. <Customer> can set bird as default so as the next login into account the bird will exist as main screen when click on “set your Bird as the default Bird”



3.3.5.15. <Customer> can **Log out** account and login into different one



## 3.4 <Trainer> Trainer timetable

### 3.4.1 <Trainer> Timetable Overview

TRAINER SCHEDULE						
	14 Sun	15 Mon	16 Tue	17 Wed	18 Thu	19 Fri
8:00 AM	Training bird at the center.	Training bird at the center.	Host workshop class slot	Training bird at the center.	Host workshop class slot	Training bird at the center.
9:00 AM			Host workshop class slot		Training bird at the center.	
10:00 AM		Training bird at the center.	Training bird at the center.		Training bird at the center.	
11:00 AM	Training bird at the center.		Training bird at the center.		Training bird at the center.	
12:00 PM						
1:00 PM		Consulting Customer		Training bird at the center.		
2:00 PM						
3:00 PM						
4:00 PM						
5:00 PM						

1. Click on "Timetable" tab in the left navigation pane.

### 3.4.2 <Trainer> Consultation in Timetable

**Bird Training Center Management**

**Trainer** Consulting Trainer

- Dashboard
- Consultant
- Timetable
- Academy

**TRAINER SCHEDULE**

Today	Back	Next	December 24 – 30					Agenda	Day	Week	Month
			24 Sun	25 Mon	26 Tue	27 Wed	28 Thu	29 Fri	30 Sat		
8:00 AM											
9:00 AM											
10:00 AM					Consulting Customer	Training bird at the center			Consulting Customer		
11:00 AM											
12:00 PM											
1:00 PM					Consulting Customer				Consulting Customer		
2:00 PM											
3:00 PM											
4:00 PM											
5:00 PM											

- Click on “Timetable” tab in the left navigation pane.

**Bird Training Center Management**

**Trainer** Consulting Trainer

- Dashboard
- Consultant
- Timetable
- Academy

**Ticket Detail**

**CUSTOMER INFORMATION**

Name:	Pham Nhat An
Email:	customer1@gmail.com
Phone Number:	(+84) 908941767



**TICKET INFORMATION**

Trainer:	Consulting Trainer	Consulting Type:	Training bird at home
Google Meet Link	https://meet.google.com/roa-pnxu-ipd	Address:	tết thgg
Appointment Date:	04/01/2024	Distance:	1Km
More Detail	thong dang tết	Price:	55000VND
		Status:	Finished

**Evidence**

- Click on the “Consulting Customer” Tab in the Schedule to view detail.

### 3.4.3 <Trainer> Workshop Class hosted in Timetable

### 3.4.4 <Trainer> Academy Training Slot in Timetable

## 3.5 <Manager><Staff> Service Management

### 3.5.1 <Manager> Workshop Management

#### 3.5.1.1 <Manager> <Staff> View workshops

The screenshot shows a list of workshops under the 'Workshop' tab. There are two tabs at the top: 'ACTIVE' (selected) and 'INACTIVE'. The table has columns: Image, Title, Register period, Total Slot, Price (VND), and Status (with a checked checkbox). The data includes:

Image	Title	Register period	Total Slot	Price (VND)	Status
	Bird Training Academy 301	7 day(s)	2	100,000	<input checked="" type="checkbox"/>
	tét report 5	1 day(s)	1	200,000	<input checked="" type="checkbox"/>
	ádasdsa	1 day(s)	2	123,123,123,123	<input checked="" type="checkbox"/>
	For the Love of Bluebirds	7 day(s)	2	200,000	<input checked="" type="checkbox"/>
	Bird Training Academy 101	7 day(s)	2	100,000	<input checked="" type="checkbox"/>
	Bird Training Academy 201	7 day(s)	2	100,000	<input checked="" type="checkbox"/>

1. Click "Workshop" in left navigation pane
2. Click "ACTIVE" or "INACTIVE" to filter the workshops based on activation status

#### 3.5.1.2 <Manager> Create new workshop

The screenshot shows the 'Create new workshop' form. It has sections for General information, Registration information, Description (with a rich text editor), Pictures, and a 'CREATE WORKSHOP' button.

**General information**  
Title \*    Total Slot \*    Price (VND) \*

**Registration information**  
Register Period \*    Minimum Registered \*    Maximum Registered \*

**Description**  
[Rich Text Editor]

**Pictures**  
UPLOAD IMAGE(S)  
No file is provided!

**CREATE WORKSHOP**

1. From step 3.5.1.1, click on "CREATE WORKSHOP" button
2. Fill in information
3. Click button "CREATE WORKSHOP"

### 3.5.1.3 <Manager> Refund Policy Management

Bird Training Center Management

Manager Manager

- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

X Refund Policies Management

ADD

No	Duration before start	Refund rate	
1	13	0	UPDATE
2	29	0.5	UPDATE
3	-1	1	UPDATE
4	0	0	UPDATE
5	30	0.75	UPDATE

Add new refund policy

Duration before start

0

Refund rate

0

ADD CANCEL

## Modify refund policy:

Duration before start

13

Refund rate

0

RESET

CONFIRM

CANCEL

1. From step 3.5.1.1, click on “PRICE POLICIES MANAGEMENT” button
2. Click “ADD” button to show add modal]
3. Click “UPDATE” button to show update modal

### 3.5.1.4 <Manager> View workshop detail

#### Workshop: For the Love of Bluebirds

- (i) Workshop detail
- (i) Classes
- (+) New class

Bird Training Center Management

Manager Manager

Dashboard Consultant Online Course Timetable Workshop Academy

Workshop: For the Love of Bluebirds - Detail

Workshop Title: For the Love of Bluebirds Slot(s): 2  
Register Period: Day(s): 7 Total Slot: VND 200000  
Price:

Registered Amount:  
Minimum: 6 Maximum: 11

Description: What will you learn?  

- Learn about the plants, shrubs, and trees that attract bluebirds
- Learn exciting facts about the bluebird's history and how it was nearly pushed to extinction
- Learn the benefits of encouraging bluebirds to nest on your property

Status:  Active EDIT WORKSHOP INFORMATION

Slot 1 No detail selected! Slot 2

UPLOAD NEW IMAGE(S)



Upload new image(s) for workshop: For the Love of Bluebirds



Uploaded image(s) will permanently replace previous image(s)

UPLOAD IMAGE(S)



No file is provided!

CONFIRM

CANCEL

## Modify workshop: For the Love of Bluebirds

## General information

- Title \*

## For the Love of Bluebirds

- Price (VND) \*

200000

## Registration information

- Register Period \*

7

Minimum Registered \*

6

- Maximum Registered \* -

11

### Description



What will you learn?

- Learn about the plants, shrubs, and trees that attract bluebirds
  - Learn exciting facts about the bluebird's history and how it was nearly pushed to extinction
  - Learn the benefits of encouraging bluebirds to nest on your property

CONFIRM

CANCELI

Slot 1

Slot 2

Learning about the Eastern Bluebird and how to attract them to your property.

SAVE CHANGES

- From step 3.5.1.1, right click on a workshop row, then select “Workshop Detail”.
- In “Workshop Detail” view, click “UPLOAD NEW IMAGE(S)” to open upload image modal.
- In “Workshop Detail” view, click “EDIT WORKSHOP INFORMATION” to open edit workshop modal.
- In “Workshop Detail” view, select a slot in the bottom left to view slot detail and modify it.
- In “Workshop Detail” view, click in status checkbox in workshop information pane to alternate status between “ACTIVE” and “INACTIVE”

### 3.5.1.5 <Manager> Workshop Class Management

Workshop: For the Love of Bluebirds

(i) Workshop detail

(i) Classes

[+] New class

Bird Training Center Management

Manager Manager

Dashboard Consultant Online Course Timetable Workshop Academy

Workshop: For the Love of Bluebirds - Classes

UPLOAD NEW IMAGE(S)

Title: For the Love of Bluebirds Slot(s): 2

Register Period: Day(s): 7 Price: VND 200000

Registered Amount:

Description: What will you learn?

- Learn about the plants, shrubs, and trees that attract bluebirds
- Learn exciting facts about the bluebird's history and how it was nearly pushed to extinction
- Learn the benefits of encouraging bluebirds to nest on your property

Status:  Active EDIT WORKSHOP INFORMATION

PENDING	OPEN REGISTRATION	CLOSED REGISTRATION	ON GOING	COMPLETED	CANCELLED
Open Registration	Closed Registration	Registration	Location	Status	
No record!					

- From step 3.5.1.1, right click on a workshop row, then select “Classes”.
- In classes view, <Manager> can upload new image(s) and edit workshop information as step 3.5.1.4.
- <Manager> can manage classes as feature inherited from <Staff>, detailed below in 3.5.2

### 3.5.2 <Staff> Workshop Class Management

#### 3.5.2.1 <Staff> Workshop Class Management

The screenshot shows a web-based application for managing workshops. On the left is a sidebar with the title 'Bird Training Center Management' and a navigation menu including 'Staff', 'Dashboard', 'Consultant', 'Online Course', 'Timetable', 'Workshop' (which is selected), and 'Academy'. The main content area has a yellow header bar with the text '< Workshop: For the Love of Bluebirds - Classes'. Below this is a large image of a bluebird. To the right of the image are several input fields: 'Workshop Title' (For the Love of Bluebirds), 'Slot(s)' (2), 'Total Slot' (2), 'Register Period' (7), 'Price' (VND 200000), 'Minimum Registered' (6), 'Maximum Registered' (11), and a 'Description' section with bullet points about bluebirds. At the bottom, there's a status field with a checked 'Active' checkbox. Below this is a filter bar with tabs: PENDING, OPEN REGISTRATION, CLOSED REGISTRATION, ON GOING, COMPLETED, and CANCELLED. A table below the filter shows columns for Open Registration, Closed Registration, Registration, Location, and Status. A note says 'No record!' and there is a plus sign icon.

- From step 3.5.1.1, right click on a workshop row, then select “Classes”.
- Click between class statuses pane such as “PENDING”, “OPEN REGISTRATION”, ... to filter classes with status.

#### 3.5.2.2 <Staff> New Class Dialog

The dialog box has a title 'Add new class for workshop: For the Love of Bluebirds'. It contains a message 'Enter open registration date for class.' and a warning message '⚠ All slots must be fulfilled to public the class!'. There are two input fields: 'Location \*' and 'Date' (set to 01/15/2024). At the bottom are 'CREATE' and 'CANCEL' buttons.

- From step 3.5.2.1, click on icon in filtered classes pane to open new class dialog
- Fill in information
- Click “CREATE” or “CANCEL” button

### 3.5.2.3 <Staff> Class Detail

The screenshot shows the 'Workshop: For the Love of Bluebirds - Classes' page. On the left is a sidebar with 'Bird Training Center Management' and links for Staff, Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy. The main area has sections for 'Open Registration' (01/15/2024), 'Closed Registration' (01/22/2024), 'Hosted Location' (Center hall, FPTU), 'Registered Limitation' (Minimum: 6, Maximum: 11), 'Registered Amount' (0/11), and 'Status' (OpenRegistration). Buttons include 'MODIFY', 'CLOSE REGISTRATION', 'SET ONGOING', 'SET COMPLETE', and 'CANCEL'. Below these are fields for 'Slot Date' (Not picked), 'Slot Time' (Not picked), and 'Trainer' (Not picked). A note says 'Update For The Slot' and 'No slot selected!'. There are also two buttons for 'Slot 1' and 'Slot 2'.

- From step 3.5.2.1, click on a class in filtered classes pane to view selected class detail
- Click "MODIFY" button to open edit class information modal for editing, declared below in step 3.5.2.4
- Click "CLOSE REGISTRATION" button to close the class registration.
- Click "SET ONGOING" button to manually set class to status on-going.
- Click "SET COMPLETE" button to manually set class to status completed.
- Click "CANCEL" button to manually cancel the class.

### 3.5.2.4 <Staff> Class Information Management

The modal dialog is titled 'Modify class hosted information'. It contains three input fields: 'Location' (Center hall, FPTU), 'Expected Open' (01/15/2024), and 'Expected Closed' (01/20/2024). At the bottom are three buttons: 'RESET' (orange), 'CANCEL' (red), and 'CONFIRM' (yellow).

- From step 3.5.2.3, click "MODIFY" button to open class information management modal.

### 3.5.2.5 <Staff> Class Slots Management

Slot Date: 01/23/2024

Slot Time: 08:00

Trainer: trainer2@gmail.com

Slot 1

Slot 2

Update For The Slot

Date: 01/23/2024

Choose Slot

Choose Trainer

Description: Learning about the Eastern Bluebird and how to attract them to your property.

**SAVE**

- From step 3.5.2.3, class slots management pane is within at the bottom of the class detail screen.
- Click on any slot in the left pane to show the slot information as above.
- Modify information in the right pane and click “SAVE” button.

### 3.5.3 <Staff> Consultation Booking Management

#### 3.5.3.1 <Staff> Approve Offline Consulting Ticket

Bird Training Center Management

- Staff
- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

**Ticket Detail**

**Basic Information**

Ticket ID:	102	Service:	Offline
Date:	31/01/2024	Time:	09:00-10:00

**Detail Information**

Customer:	Pham Nhat An	Phone:	906941767
Email:	customer1@gmail.com		
Address:	Demo	Distance:	21 Km

**More Detail**

Trainer:	Nicole Cowper	Type:	About the Center
Price:	75000VND	Status:	WaitingForApprove

**Buttons:** ASSIGN, CANCEL

### 3.5.3.2<Staff> Approve Online Consulting Ticket

The screenshot shows the 'Ticket Detail' page of the Bird Training Center Management system. The left sidebar includes links for Staff, Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy. The main content area has tabs for LIST NOT ASSIGNED, LIST ASSIGNED, LIST HANDLED, PRICE POLICY, and CONSULTANT TYPE. The 'LIST ASSIGNED' tab is selected. The 'Ticket Detail' section contains fields for Basic Information (Ticket ID: 103, Date: 27/01/2024, Service: Online), Detail Information (Customer: Pham Nhat An, Phone: 908941767, Email: customer1@gmail.com), and More Detail (Trainer: Consulting Trainer, Type: Other, Price: 55000VND, Status: WaitingForApprove). Buttons for CONFIRM and CANCEL are at the bottom.

### 3.5.4 <Manager> Online Course Management

#### 3.5.4.1 <Manager> Add new Courses

The screenshot shows the 'Create New Online Course' form. The left sidebar includes links for Manager, Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy. The main content area has a 'Create New Online Course' header and sections for General information (Title \* and Price \*), Short Description (with a rich text editor), Pictures (with an UPLOAD IMAGE button and a message indicating no file is provided), and a CREATE ONLINE COURSE button.

### 3.5.4.2 <Manager> Add new Section for specific Courses

The screenshot shows a web-based management system for bird training. On the left sidebar, under the 'Management' section, there is a 'Manager' role icon. The main content area displays a list of 'Training Steps' for a course titled 'testing part 2'. A modal window titled 'Add More Section:' is open, prompting the user to enter a 'Section Title \*'. The 'Description' field contains a rich text editor with a toolbar. Below the editor is a button labeled 'UPLOAD RESOURCE FILE' and a note stating 'No file is provided!'. At the bottom of the modal are 'CANCEL' and 'CREATE NEW SECTION' buttons. To the right of the modal, the course details are visible, including the URL <https://storage.googleapis.com/birdtrainingsystem.appspot.com/online-course-resources/04be4484-19fb-4009-89b3-6abcc3ae9c79.pdf>.

### 3.5.4.3 <Manager> can update, delete sections (can only delete when no lessons inside)

This screenshot shows the 'COURSE DETAIL: testing' page. The sidebar includes a 'Manager' role icon. The main area shows a table of 'Sections' with columns for 'Title' and 'Description'. One section is highlighted with the title 'update lai' and description 'aaa'. A modal window titled 'Update Section:' is open over this row. It contains fields for 'Section Title \*' and 'Description', both with rich text editors. Below these are 'UPLOAD RESOURCE FILE' and 'No file is provided!' buttons. At the bottom of the modal are 'CANCEL' and 'UPDATE' buttons. The background table shows a single row with a price of '200.000(vnd)' and a checked 'Status' checkbox. The URL <https://storage.googleapis.com/birdtrainingsystem.appspot.com/online-course-resources/5d2bb8a7-a02f-4f7c-bf57-3134535dd5b.jpg> is also visible.

**COURSE DETAIL: testing tao moi tiep**

Image	Title	Short Description	Price (vnd)	Status
	testing tao moi tiep	thontoinhthi	200.000(vnd)	<input checked="" type="checkbox"/>

**Sections**

Title	Description	Resource File
update lai	aaa	<a href="https://storage.googleapis.com/bird-training-course-resources/93bf5382-5f46-4577-a214-8c53f329cf9b.jpg">https://storage.googleapis.com/bird-training-course-resources/93bf5382-5f46-4577-a214-8c53f329cf9b.jpg</a>
<b>test delete section</b>	a	

**Do you want to delete this section:**

**cancel** **DELETE**

**UPDATE** **DELETE**

**+ ADD NEW SECTION**

#### 3.5.4.4 <Manager> can add new Lesson inside Sections

**Add More Lesson:**

**Lesson Title \***

**Description**

**B I U S E E E E E Normal**  
Normal

**UPLOAD VIDEO**

**No file is provided!**

**CANCEL** **CREATE NEW LESSON**

### 3.5.4.5 <Manager> can update to modify data inside lesson

The screenshot shows a modal dialog titled "Update Lesson:" over a list of lessons. The dialog has fields for "Lesson Title" (containing "Lesson Title \*") and "Description" (with a rich text editor). An "UPLOAD VIDEO" button is present. A message "No file is provided!" is displayed below the input fields. At the bottom are "CANCEL" and "UPDATE" buttons. The background lists several lessons, including "testing report 5" which is currently selected.

### 3.5.4.6 <Manager> can delete lesson

The screenshot shows a confirmation dialog box asking "Do you want to delete this lesson?". It includes "CANCEL" and "DELETE" buttons. The background shows the same list of lessons, with "testing report 5" still selected. The URL for the video is visible as https://storage.googleapis.com/3293-4835-b98c-bcc10c06384.mp4.

### 3.5.4.7 <Manager> can change status to active courses or deactivate courses

**COURSE DETAIL: testing tao moi tiep**

Image	Title	Short Description	Price (vnd)	Status
	testing tao moi tiep	thohothothoth	200,000(vnd)	<input type="checkbox"/>

**Sections**

Title	Description	Resource File
update lai	aaa	<a href="https://storage.googleapis.com/birdtrainingsystem.appspot.com/online-course-resources/936f5382-5f46-4577-a214-8c53f529cf9b.jpg">https://storage.googleapis.com/birdtrainingsystem.appspot.com/online-course-resources/936f5382-5f46-4577-a214-8c53f529cf9b.jpg</a>

**Actions:** UPDATE, DELETE

**test delete section** a

**Actions:** UPDATE, DELETE

**Buttons:** + ADD NEW SECTION

### 3.5.5 <Manager> Training Course Management

#### 3.5.5.1 <Manager> can create new training courses for bird to training

**CREATE NEW TRAINING COURSE**

Image	Title	Bird Species	Total Slot	Price (VND)	Status	Action
	eetestset12	Cockatoo	0	123,123,123	<input type="checkbox"/>	ACTIVE
	teting	Finch	0	122,000	<input type="checkbox"/>	ACTIVE
	Flight posing training	Cockatiel	4	100,000	<input type="checkbox"/>	ACTIVE
	Flight posing training	Cockatiel	2	100,000	<input checked="" type="checkbox"/>	DISABLE
	First training for new bird	Cockatiel	10	100,000	<input checked="" type="checkbox"/>	DISABLE
	Monk bird	Quaker Parrot (Monk Parakeet)	0	50,000	<input type="checkbox"/>	ACTIVE
	Lovebird flight training	Lovebird	5	100,000	<input checked="" type="checkbox"/>	DISABLE

Bird Training Center Management

Manager

- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

**Create Training Course**

General information

Pictures

Choose Species

UPLOAD IMAGE(S)

No file is provided!

Title \*

Price \*

Description

CANCEL CONFIRM CREATE COURSE

### 3.5.5.2 <Manager> can Active/Disable course

Bird Training Center Management

Manager

- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

**CREATE NEW TRAINING COURSE**

Image	Title	Bird Species	Total Slot	Price (VND)	Status
	eetestse12	Cockatoo	0	123,123,123	<input type="checkbox"/> ACTIVE <button>VIEW DETAIL</button>
	teting	Finch	0	122,000	<input type="checkbox"/> ACTIVE <button>VIEW DETAIL</button>
	Flight posing training	Cockatiel	4	100,000	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>
	Flight posing training	Cockatiel	2	100,000	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>
	First training for new bird	Cockatiel	10	100,000	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>
	Monk bird	Quaker Parrot (Monk Parakeet)	0	50,000	<input type="checkbox"/> ACTIVE <button>VIEW DETAIL</button>
	Lovebird flight training	Lovebird	5	100,000	<input checked="" type="checkbox"/> DISABLE <button>VIEW DETAIL</button>

Disable successfully! ×

Active successfully! ×

3.5.5.3. <Manager> can update Details information, add and remove bird skill of a specific course

**Bird Training Center Management**

**Training Course Detail**

**General information**

Bird Species Name: Cockatiel

Title: Flight posing training

Price: 100000

Slot: 4

**Pictures**

No file is provided.

**Description**

Flying and posing training for bird

**Bird Skills in Course**

Picture	Name	Description	Train Slot	Action
	Flight training	Flight training for birds is an instinctive and vital process in which young avians learn to master the art of flying	2	<b>REMOVE</b>
	Pose training	Pose training for birds involves teaching them to assume specific positions on command	2	<b>REMOVE</b>

**Bird Skills out Course**

Picture	Name	Description	Action
	Speech training	Develops vocabulary and mimicking	<b>ADD</b>
	Behavior training	Modifies behaviors through positive reinforcement	<b>ADD</b>
	Agility training	Improves coordination through obstacle courses	<b>ADD</b>

### 3.5.5.4 <Manager> Create new Training Policy in training course

The screenshot shows a web-based application interface for 'Bird Training Center Management'. The left sidebar has a 'Manager' role selected. The main navigation tabs are 'ACADEMY MANAGEMENT', 'TRAINING COURSE MANAGEMENT', and 'PRICE POLICY MANAGEMENT', with the latter being active. A modal window titled 'Create Training Price Policy' is open, containing fields for 'Name' (with a required asterisk) and 'Charge rate' (set to 0). Below the form are 'CANCEL' and 'CONFIRM CREATE POLICY' buttons.

### 3.5.5.5 <Manager> can Active/Disable Price Policy

The screenshot shows the 'PRICE POLICY MANAGEMENT' table. It lists five policies with columns for Name, Charge rate, and Status. The first policy, 'Bird's health not good', has a charge rate of 0.25 and is marked as 'ACTIVE'. The other four policies have a charge rate of 1 and are marked as 'DISABLE'. Each row includes a 'VIEW DETAIL' button. To the right of the table, two success messages are displayed: 'Disable successfully!' and 'Active successfully!'. The left sidebar shows the 'Manager' role is selected.

Name	Charge rate	Status	
Bird's health not good	0.25	<input type="checkbox"/> ACTIVE	<button>VIEW DETAIL</button>
Success Requested	1	<input checked="" type="checkbox"/> DISABLE	<button>VIEW DETAIL</button>
Center Requested	0	<input checked="" type="checkbox"/> DISABLE	<button>VIEW DETAIL</button>
Customer Requested	1	<input checked="" type="checkbox"/> DISABLE	<button>VIEW DETAIL</button>

### 3.5.5.6 <Manager> can Update Price Policy

The screenshot shows the 'Training Price Policy Detail' page. The left sidebar is titled 'Bird Training Center Management' and includes a 'Manager' section with a user icon and the role 'Manager'. Other menu items include Dashboard, Consultant, Online Course, Timetable, Workshop, and Academy. The main content area has tabs for ACADEMY MANAGEMENT, TRAINING COURSE MANAGEMENT, and PRICE POLICY MANAGEMENT. The current tab is PRICE POLICY MANAGEMENT. The sub-page title is 'Training Price Policy Detail'. It contains a 'General information' section with a 'Name\*' field containing 'Bird's health not good' and a 'Charge rate\*' field containing '0.25'. At the bottom are 'CANCEL' and 'CONFIRM UPDATE POLICY' buttons.

### 3.5.5.7 <Manager> can show Bird List that customer register there's bird to our system

The screenshot shows the 'Customers' page. The left sidebar is identical to the previous one. The main content area has tabs for ACADEMY MANAGEMENT, TRAINING COURSE MANAGEMENT, and PRICE POLICY MANAGEMENT. The current tab is TRAINING COURSE MANAGEMENT. The sub-page title is 'Customers'. It lists six customers with their names, emails, phone numbers, and avatars. Each row has 'VIEW REQUEST' and 'SHOW BIRD LIST' buttons. Below this, the 'Customer Bird' page is shown. Its sidebar and tabs are the same. The sub-page title is 'Customer Bird'. It lists four birds registered by customer 'thg55': 'tao de test' (Cockatoo), 'Añ' (Quaker Parrot (Monk Parakeet)), and 'the hell' (Cockatiel). Each entry includes a bird picture and a 'VIEW RECEIVED BIRD SKILLS' button.

### 3.5.5.8 <Manager> can check bird's skill of each bird after it's completed Training

#### Bird Skill Received For Bird 2

Bird Name	Bird Skill Name	Bird Skill Description	Received Date	Bird Skill Picture
dol iamge nek	Flight training	Flight training for birds is an instinctive and vital process in which young avians learn to master the art of flying	11/12/2023	
dol iamge nek	Speech training	Develops vocabulary and mimicking	11/12/2023	

BACK

### 3.5.5.9 <Manager> can change status of each bird registered course

#### Registered TrainingCourse

Bird Name	Owner Name	Course Title	Registered	Training start date	Training done date	Status	
Lanning birrd	Louis Lane	Flight posing training	21-12-2023	18-12-2023	19-12-2023	Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Imitating human speech	19-12-2023	12-01-2024		Checkin	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Greeting people	19-12-2023	27-12-2023		Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Cockatiel entertainment performance	19-12-2023	19-12-2023		Complete	<a href="#">VIEW DETAILS</a>
Lanning birrd	Louis Lane	Flight posing training	13-12-2023	11-01-2024		Cancel	<a href="#">CONFIRM</a>
Lanning birrd	Louis Lane	Flight posing training	13-12-2023	20-12-2023	19-12-2023	Complete	<a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	Flight posing training	10-12-2023	19-12-2023	19-12-2023	TrainingDone	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	Flight posing training	10-12-2023	20-12-2023		Checkin	<a href="#">CHECK OUT</a> <a href="#">VIEW DETAILS</a>
Louis's bird	Louis Lane	First training for new bird	09-01-2024			Registered	<a href="#">CONFIRM CANCEL</a>
Lanning birrd	Louis Lane	First training for new bird	09-01-2024			Registered	<a href="#">CONFIRM CANCEL</a>

### 3.5.5.10 <Manager> can re-assign trainer, modify specific trainer slot and Date in View Training Details

**Bird Training Center Management**

- Manager Manager
- Dashboard
- Consultant
- Online Course
- Timetable
- Workshop
- Academy

**ACADEMY MANAGEMENT TRAINING COURSE MANAGEMENT PRICE POLICY MANAGEMENT**

**X Training Details By BirdSkill**

No	Bird Skill Name	Trainer Name	Trainer Email	Training Progression	Total Training Slot	Status	
0	Flight training	Nicole Cowper	trainer2@gmail.com	<div style="width: 0%;">0% Complete</div>	4	Assigned	<a href="#">RE-ASSIGN TRAINER</a> <a href="#">VIEW TRAINING DETAILS</a>
1	Target Training	Rachel Riley	trainer3@gmail.com	<div style="width: 0%;">0% Complete</div>	3	Assigned	<a href="#">RE-ASSIGN TRAINER</a> <a href="#">VIEW TRAINING DETAILS</a>

**Trainer**

Name	Email	
Rachel Riley	trainer3@gmail.com	<a href="#">ASSIGN</a>
Nicole Cowper	trainer2@gmail.com	<a href="#">ASSIGN</a>
Bảo Nguyễn	consultingtrainer1@gmail.com	<a href="#">ASSIGN</a>

The screenshot shows a web-based application titled "Bird Training Center Management". The left sidebar has a "Manager" role selected. The main content area is titled "Training Details By BirdSkill". It displays a table with columns: Slot, Date, Trainer Name, Trainer Email, and Status. Each row has a "MODIFY" button. The data in the table is as follows:

Slot	Date	Trainer Name	Trainer Email	Status
4	12/01/2024	Nicole Cooper	trainer2@gmail.com	Not Yet
3	13/01/2024	Nicole Cooper	trainer2@gmail.com	Not Yet
4	14/01/2024	Nicole Cooper	trainer2@gmail.com	Not Yet
3	15/01/2024	Nicole Cooper	trainer2@gmail.com	Not Yet

### 3.5.5.11 <Trainer> can upload evidences of assigned bird skill if any bird completed training

The screenshot shows the same application interface. A modal window is open over the list of training skills. The modal title is "Upload training done evidences video". It contains a "Videos" section with an "UPLOAD VIDEO(S)" button and a message "No file is provided!". Below it is a "Choose Status \*" dropdown set to "Pass" with a "CONFIRM" button. The background table lists training skills with their status and "VIEW TRAINING DETAILS" links. The "Speech training" row has "UPLOAD EVIDENCES" and "VIEW TRAINING DETAILS" buttons.

ASSIGNED TRAINING SKILL				
Bird Skill Name	Trainer Name	Training Progression	Total Training Slot	Status
Flight training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Pose training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Target Training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Target Training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Flight training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Pose training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	1	Pass
Speech training	Rachel Riley	<div style="width: 100%;">100% Complete</div>	3	Pass

## 3.6 <Administrator> Administration Management

### 3.6 <Administrator>.1 User Management

**Bird Training Center Management**

**Administrator** Administrator

**Admin**

**CUSTOMER TRAINER STAFF MANAGER ADMINISTRATOR FUNCTION**

**ADD NEW USER**

**Users**

Search for user (Email)

ID	Name	Email	Phone Number	Password	Role	Membership	Status
1	Trung	williamthanhtrunq2@gmail.com	908941761	123	Customer	Gold	Active
22	John Doe	john.doe@email.com	123456789	123	Customer	Standard	Active
23	Jane Smith	jane.smith@email.com	987654321	123	Customer	Standard	Active
24	Alice Johnson	alice.johnson@email.com	11233445	123	Customer	Standard	Active
4	Ace	golddace@gmail.com	908000000	123	Customer	Standard	Active
8	Pham Nhat An	customer1@gmail.com	908941767	123	Customer	Standard	Active
9	Bruce Wayne	customer2@gmail.com	908941768	123	Customer	Standard	Active
10	Clark Kent	customer3@gmail.com	908941769	123	Customer	Standard	Active
11	Louis Lane	customer4@gmail.com	908941770	123	Customer	Standard	Active
12	test1	test1@gmail.com	904560264	123	Customer	Standard	Active
15	Nhat An	test12@gmail.com	904560269	123	Customer	Standard	Active

### 3.6 <Administrator>.2 Function Management

**Bird Training Center Management**

**Administrator** Administrator

**Admin**

**CUSTOMER TRAINER STAFF MANAGER ADMINISTRATOR FUNCTION**

**Function**

**Slot**

Slot	Start Time	End Time
1	08:00:00	09:00:00
2	09:00:00	10:00:00
3	10:00:00	11:00:00
4	11:00:00	12:00:00
5	13:00:00	14:00:00
6	14:00:00	15:00:00
7	15:00:00	16:00:00
8	16:00:00	17:00:00

**Membership Rank**

ID	Name	Discount	Requirement
1	Standard	0	0
2	Gold	0.1	500000
3	Platinum	0.3	1000000

Duration Minute **UPDATE**

3.6 <Administrator>.2.1 Update Slot

**Slot**

Slot	Start Time	End Time
1	08:00:00	09:00:00
2	09:00:00	10:00:00
3	10:00:00	11:00:00
4	11:00:00	12:00:00
5	13:00:00	14:00:00
6	14:00:00	15:00:00
7	15:00:00	16:00:00
8	16:00:00	17:00:00

Duration:

Minute  
45

**UPDATE**

3.7 <Administrator>.2.2 Update Membership Rank

1. Select membership rank that you want to update.

**Membership Rank**

ID	Name	Discount	Requirement
1	Standard	0	0
2	Gold	0.1	5000000
3	Platinum	0.3	10000000

2. Update the filed you want to update then click UPDATE

**Bird Training Center Management**

CUSTOMER TRAINER STAFF MANAGER ADMINISTRATOR **FUNCTION**

### Function

Slot	Start Time	End Time	Membership Rank
1	08:00:00		
2	09:00:00		
3	10:00:00		
4	11:00:00		
5	13:00:00		
6	14:00:00		
7	15:00:00		
8	16:00:00	17:00:00	

Duration: Minute

**Platinum**

Name	Discount	Requirement
Platinum	0,3	10000000
Gold	0,1	5000000
Standard	0	0

**Membership Rank**

Name	Discount	Requirement
Standard	0	0
Platinum	0,3	10000000
Gold	0,1	5000000