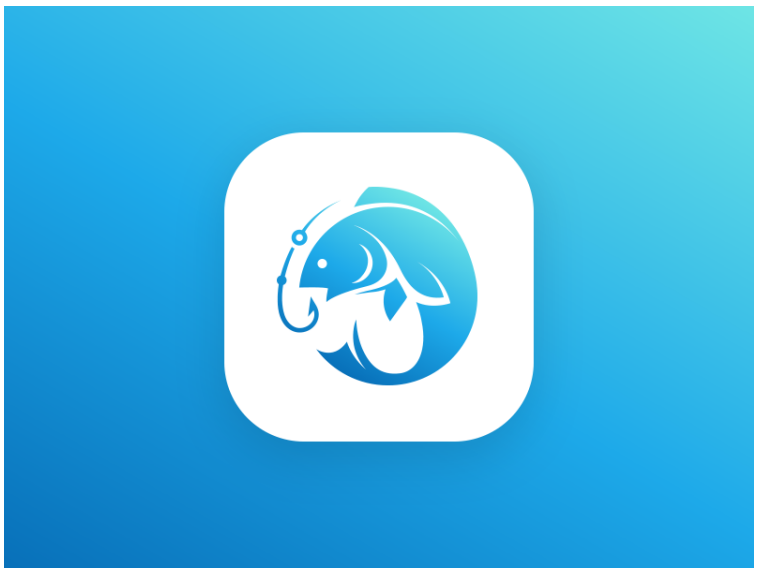
Department of Computer Science & Engineering
The University of Texas at Arlington

Architectural Design Specification
CSE 4316: Senior Design I
Fall 2022



Team Fish Quest
Fish Quest

Brandon Stibich
Kevin Phan
Mohammed Zakiuddin
Othman Kamel
William Sigala

Revision History

Revision	Date	Author(s)	Description
0.1	10.26.2022	BS	Document Creation
0.2	10.28.2022	BS, KP, MZ, OK, WS	Design ADS Diagram
0.3	10.29.2022	KP, MZ	Design Review
1.0	11.04.2022	KP	Section 1 & 2
1.0	11.04.2022	BS, KP, MZ, OK, WS	Section 3
1.0	11.04.2022	WS	Section 4
1.0	11.04.2022	KP, OK	Section 5
1.0	11.04.2022	MZ	Section 6
1.0	11.04.2022	BS	Section 7
1.1	11.05.2022	BS, KP, MZ, OK, WS	Complete Draft
1.2	12.13.2022	BS, KP, MZ, OK, WS	Updated Table Titles, Logo & Individual Subsection Diagrams

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

FishQuest is a mobile android application that revolves around fishing and is meant to improve the fishing experience for all anglers. FishQuest aims to make fishing more enjoyable by by making it a more social hobby and also through rewards to create a higher incentive to fish. The main features of FishQuest include a catch logger, a social feed page that displays recently logged catches, a map feature to display the locations of all logged catches, and fishing related missions that provide rewards when completed.

2 System Overview

The FishQuest application consists of four main layers. The layers are the Sensors, User Interface, Data Processing, and Database layers. The diagram below represents the layers and how they interact with one another.

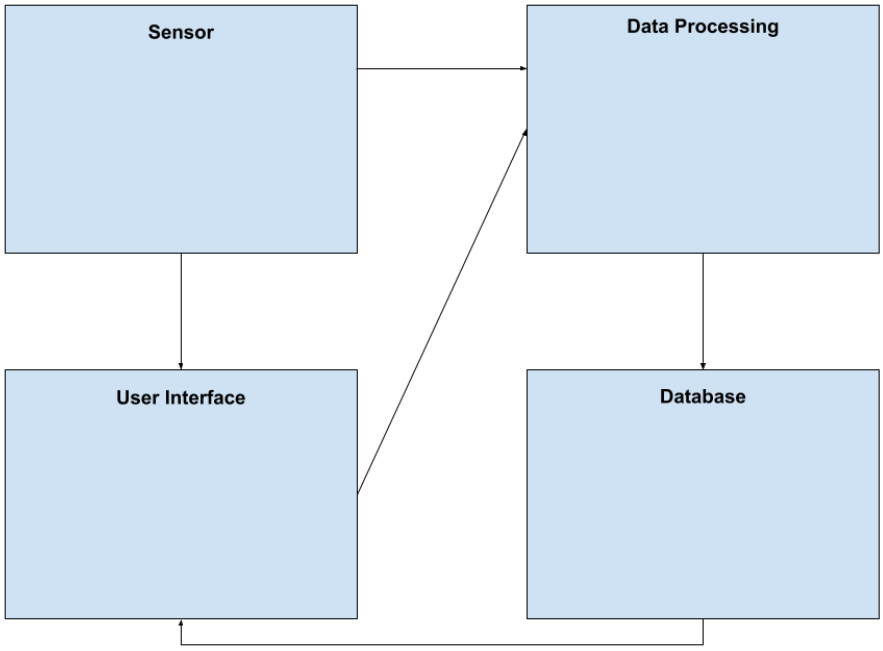


Figure 1: A simple architectural layer diagram

2.1 Sensor Layer

The main purpose of the sensor layer is to enable image input for the components in our application that are dependent on images. Taking pictures will be done through the camera subsystem. The main app components that will receive images from the sensor layer are the catch logger and fish identifying machine learning model.

2.2 User Interface Layer

The user interface layer is the graphical side of the application that will allow users to navigate through the application. This layer is also responsible for handling user input from the different app components and then sending the data to the backend to be processed and stored. Aside from input, this layer also handles output by fetching data from the database and displaying it on the different pages in the app. The user interface layer will allow users to create a new account, log in to an existing account, log catches, interact with other users on the social feed page, view fishing missions, and access a map that displays all logged catches.

2.3 Data Processing Layer

The data processing layer receives submitted input from the different components in the user interface, processes the data, and then stores the data into the database. The majority of the data will be processed and stored to the database by invoking a backend API. The data processing layer also processes credential

data for unique users which will enable account registration and log in operations. Lastly, the data processing layer can also process fish images to identify the fish species through a machine learning model

2.4 Database Layer

The database layer is where all of FishQuest's data is stored. The database is divided into several partitions to group different types of data. The database contains partitions for users, catch posts, and logged catches.

3 Subsystem Definitions & Data Flow

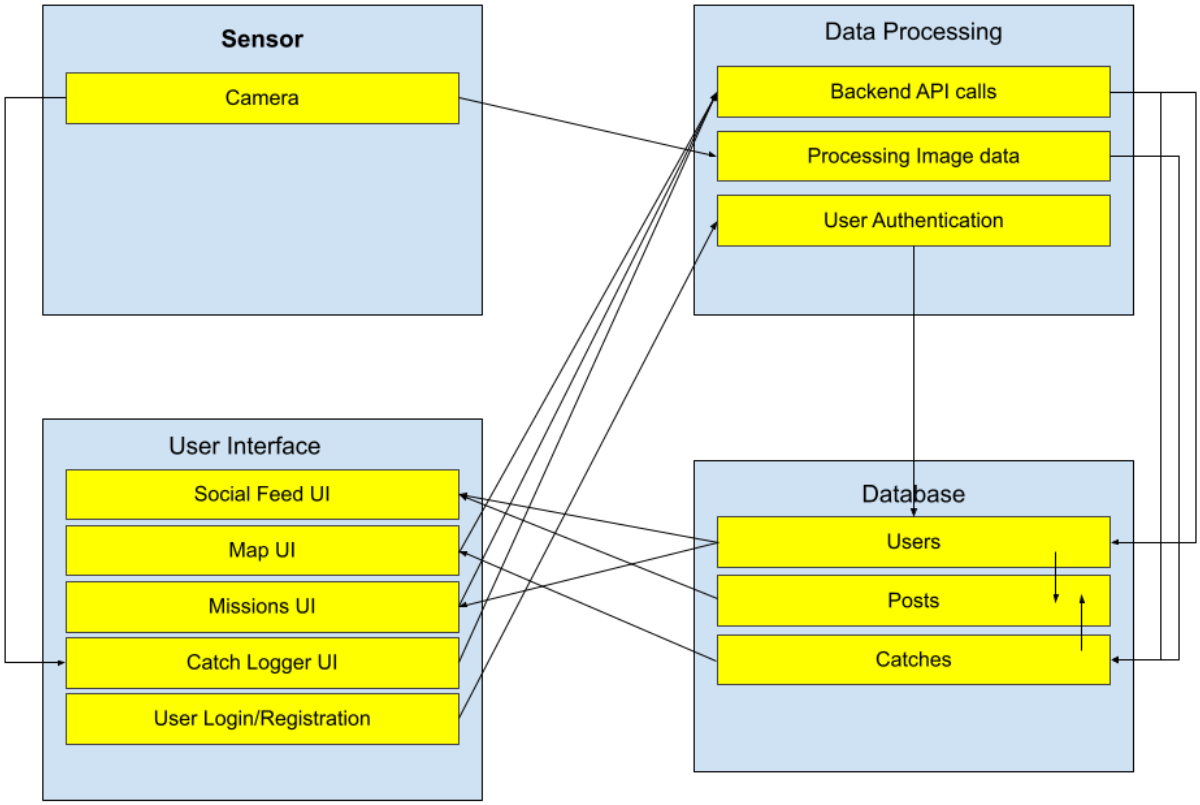


Figure 2: A simple data flow diagram

4 Sensor Layer Subsystems

The sensor layer is a core component in the functionality of the app. It provides the image data by capturing the input from the camera of the user's phone. Upon the user's submission of a catch, the image data from the camera is transferred over to the data processing layer.

4.1 Camera

The camera from the user's device will capture image data when the camera view is open. The captured image data is then displayed to the user.

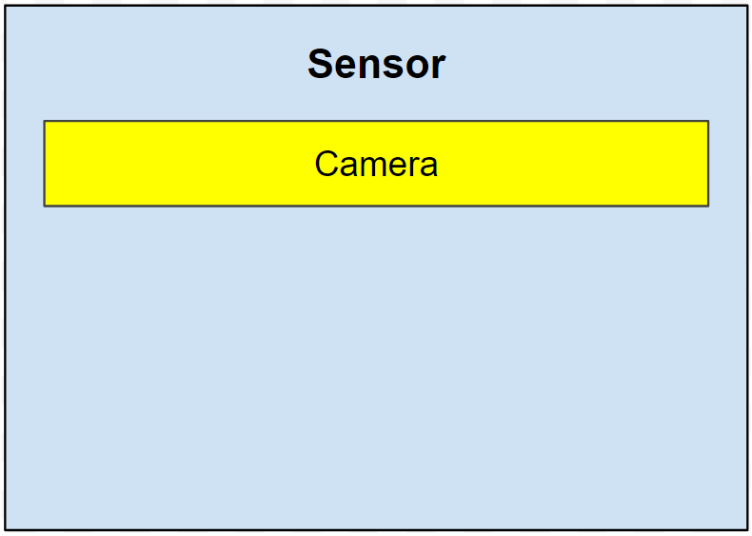


Figure 3: Camera Description Diagram

4.1.1 Assumptions

The supported devices will have cameras that are compatible and accessible to the app. The design of the app is not exclusive to one particular type of camera.

4.1.2 Responsibilities

The camera is responsible for capturing the frames of the scene from the device and delivering the image data to the app. It is not the responsibility of the camera to ensure that the scene contains a fish.

4.1.3 Camera Interface

The input from the camera sensor is obtained from the captured scene. The output is the image data which is to be used at the application layer.

Table 2: Camera interfaces

ID	Description	Inputs	Outputs
#1	Camera	Scenes from the camera's lens	Image Data

5 User Interface Layer Subsystems

The user interface layer deals with the the graphical portion of the app that allows users to interact with the app. The subsystems in this layer consist of user interfaces for the social feed, map, missions menu, catch logger, logging in, and creating an account.

5.1 Social Feed User Interface

The social feed user interface allows users to interact with the social features of the application. While logging a catch, users can provide an image, a caption, a location, and other details about the catch which which will be synthesized into a post format and displayed on the feed page for the public to see. Other users can like, dislike, and comment on posts.

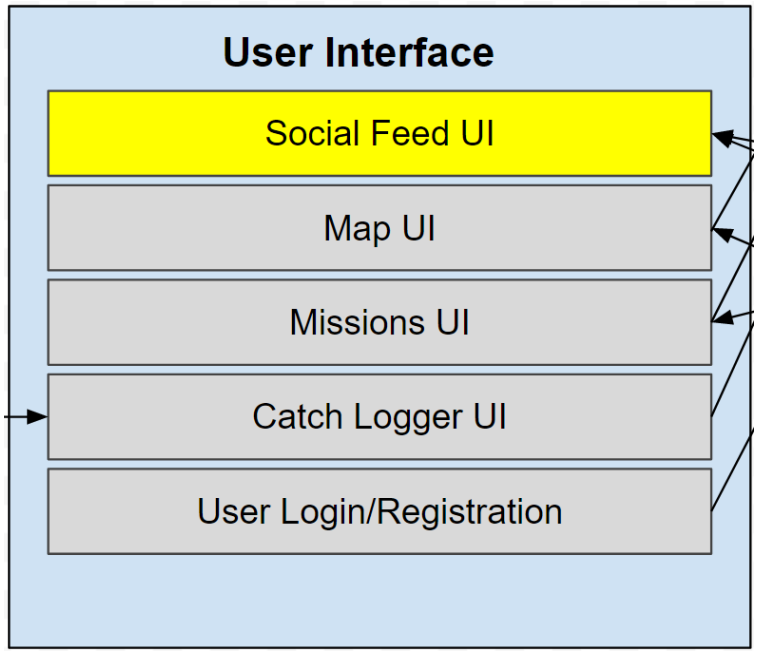


Figure 4: Social Feed UI Description Diagram

5.1.1 Assumptions

Users must have an existing account and be logged in before they can interact with the social features.

5.1.2 Responsibilities

The responsibility of the social features is to provide users with a platform where they can virtually interact with other members of the fishing community and remain interconnected. The social feed must also fetch and display data that has been submitted by the catch logger.

5.1.3 Subsystem Interfaces

The inputs from this social feed subsystem are acquired from the catch logger whenever the user submits data about the catch. The social feed will receive inputs such as the fish's physical traits, an image, the fish's species, location, and a caption. The output will display this data in a post format. Inputs for the social feed subsystem will also come from user interactions such as commenting, liking, and disliking. The output will display these inputs on the post.

Table 3: Social Feed User Interface

ID	Description	Inputs	Outputs
#1	Social Feed Post	Fish Physical Traits Image Species Location Caption	This data will be displayed in a post format on the feed page
#2	Commenting	Comments	Comments will be displayed under the corresponding post
#3	Liking/Disliking	Like Dislike	The like and dislike counters will update under the corresponding post

5.2 Catch Logger User Interface

The catch logger allows users to

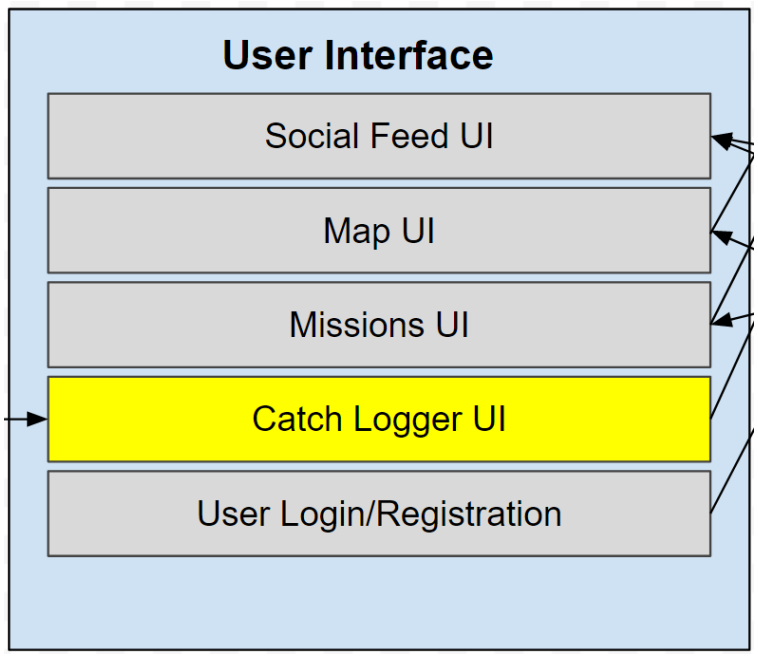


Figure 5: Catch Logger UI Description Diagram

5.2.1 Assumptions

Users must have an existing account and be logged in before they can log catches. Users must also have their location services enabled so that the catch logger can save the location of the catch. The catch logging feature will also function under the assumption that users are honest and submit legitimate catch data.

5.2.2 Responsibilities

The responsibility of the catch logger is to enable users to upload their catches to the platform. The backend API must process the submitted catch data and properly store it into the database so that other application components that are dependent on the catch data can utilize it.

5.2.3 Subsystem Interfaces

The catch logger will allow users to enter data about their catch such as the fish's physical traits, an image, the fish's species, location, and a caption. This input data will be saved to the database and be displayed in various formats throughout the other subsystems that rely on the catch data.

Table 4: Catch Logger User Interface

ID	Description	Inputs	Outputs
#1	Logging Catch	Fish Physical Traits Image Species Location Caption	Catch data will be displayed on the feed page after user submits catch log

5.3 Map

The map will allow users to view all the catches by other users on a map.

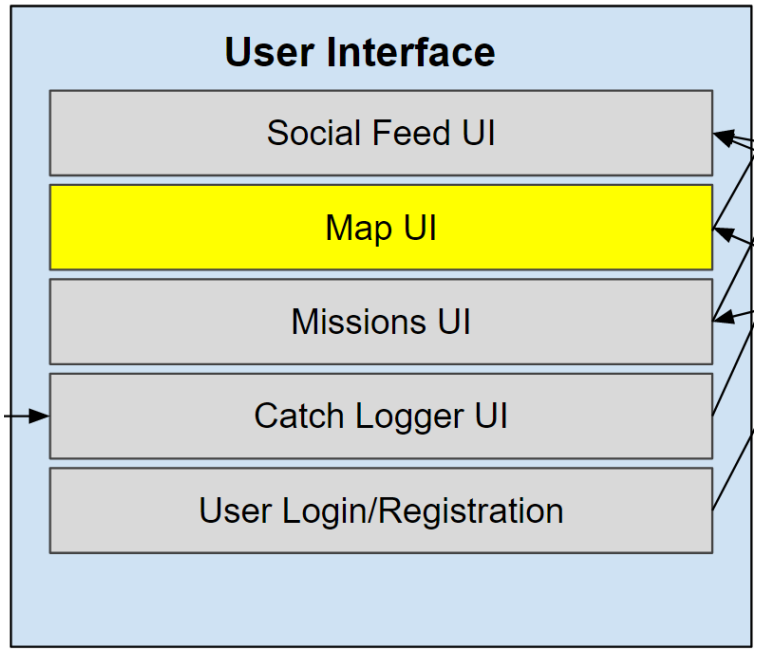


Figure 6: Map UI Description Diagram

5.3.1 Assumptions

The users enable location permission on their devices.

5.3.2 Responsibilities

The map page allows users to view a map of the United States with markers on each location where a fish was caught by a user. When a user clicks on a specific marker, it will allow them to view detailed information about the fish caught.

5.3.3 Subsystem Interfaces

Table 5: Map User Interface

ID	Description	Inputs	Outputs
#1	Map Markers	Click on marker	Metadata about the fish

5.4 Missions

The missions will allow users to earn tokens and points by completing certain tasks.

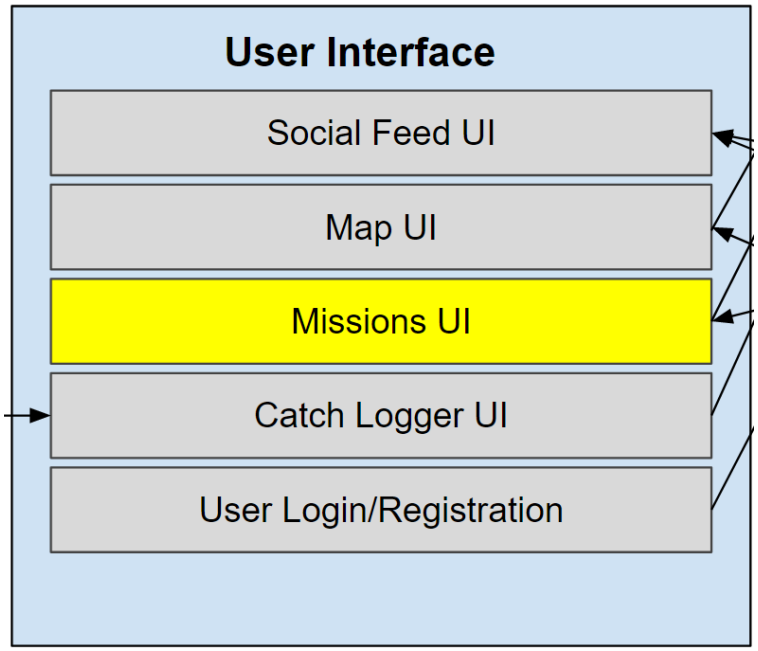


Figure 7: Missions UI Description Diagram

5.4.1 Assumptions

The users enable camera permission on their devices.

5.4.2 Responsibilities

The missions page will allow users to complete challenges and medals and XP points that will increase the user level after a certain amount, and there will also be a timed mission for all users to make fishing a competitive experience.

5.4.3 Subsystem Interfaces

Table 6: Missions User Interface

ID	Description	Inputs	Outputs
#1	Weekly Mission	Picture of the fish	- medal - XP increase

5.5 Login/Registration

The login/registration will allow users to create and manage their accounts

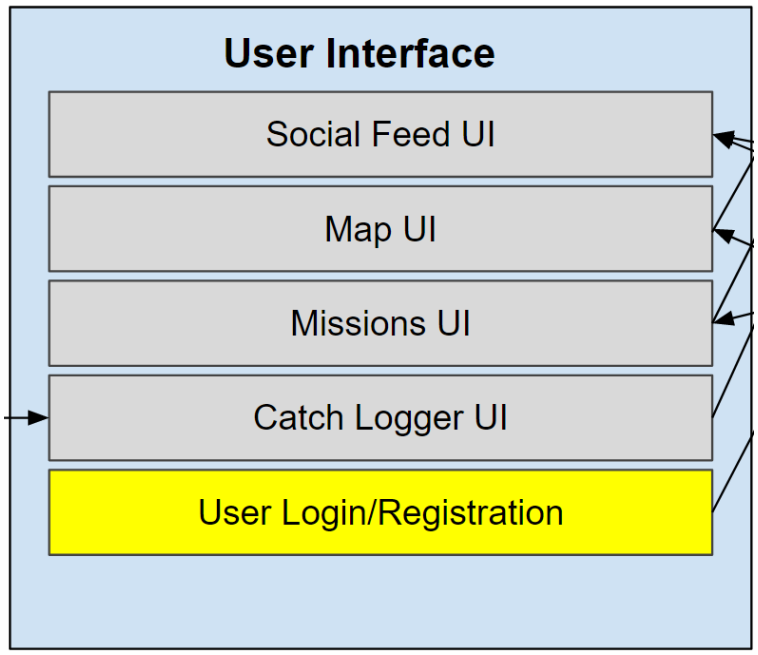


Figure 8: Login/Registration UI Description Diagram

5.5.1 Assumptions

For registration, the user must have internet connection. For login, the user must have an account registered.

5.5.2 Responsibilities

Will allow the user to access all parts of the application and manage their profile

5.5.3 SUBSYSTEM INTERFACES

Table 7: Login/Registration User Interface

ID	Description	Inputs	Outputs
#1	Registration	Username Email Password	Homepage of the application (Social Feed)
#2	Login	Username Password	Homepage of the application (Social Feed)

6 Data Processing Layer Subsystems

The Data Processing System Layer is the layer of the application that is responsible for managing the Back-End API calls, User Authentication & Processing Image Data. The Back-End API calls will respond to the user's request in accordance with whether it involves logging in for user authentication or registering as a new user; in these cases, the data is either transferred to the database or is fetched from the API calls and will assist the user in the authentication process. The same is true for processing image data, whether the user is attempting to retrieve image details or wants to store the information in the database.

6.1 Back-End API Calls

Back-End API calls layer is responsible for executing all of the user's requests.

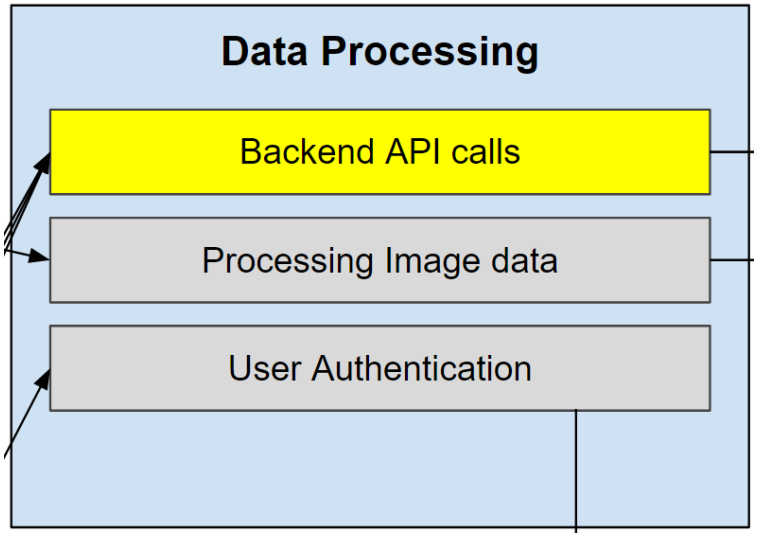


Figure 9: Backend API Calls Description Diagram

6.1.1 Assumptions

- Users who request data through the Back-End API calls must be registered as Fish Quest users.

6.1.2 Responsibilities

- The application should allow users to register.
- Once registered, users should be able to request data from the Back-End APIs.

6.1.3 Back-End API Calls Interfaces

Table 8: Back-End API Calls Interfaces

ID	Description	Inputs	Outputs
#1	User Requests	Request ID	Fetches Data
#2	Fetch Data	Request ID	Outputs Data

6.2 User Authentication

User authentication is a method of granting registered users access to the Fish Quest Application by verifying the user's identity and distinguishing the user's permissions. This method also protects against unauthorized access.

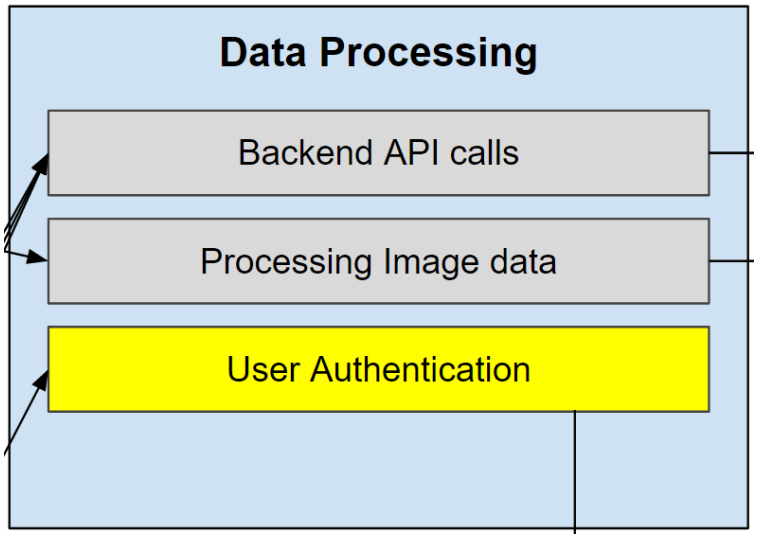


Figure 10: Backend User Authentication Description Diagram

6.2.1 Assumptions

- Users can request for Authentication to log in to the application.

6.2.2 Responsibilities

- Once registered, users should be able to log in.
- Unauthorized access is avoided by using user authentication.

6.2.3 User Authentication Interfaces

Table 9: User Authentication Interfaces

ID	Description	Inputs	Outputs
#1	User Registration	User ID	New User
#2	User Authentication	User ID	Logs In

6.3 Processing Image Data

The image data provided from the camera will be processed to ensure that the image submissions are appropriate and valid. Information about the image data will be produced such as the type of fish and other features that can describe the image.

6.3.1 Assumptions

Some of the processing will be done on the user's device. The user's device is capable of processing the image data in a reasonable amount of time.

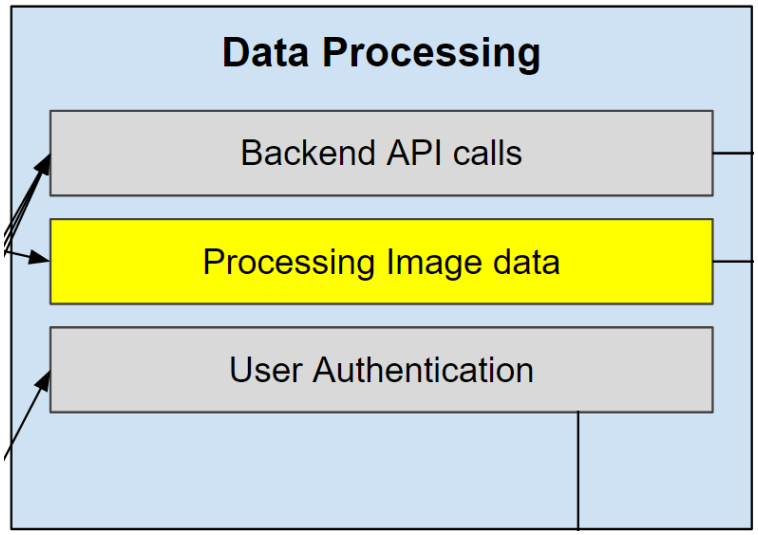


Figure 11: Backend Image Processing Description Diagram

6.3.2 Responsibilities

This subsystem is responsible for classifying the fish in the image and identifying features. If there is no fish in the image, then the submission is rejected. If it is not rejected, then identify features to be used in determining the authenticity of the image. Information gathered from processing the image will be passed over to the other layers.

6.3.3 Processing Image Data Interfaces

Table 10: Processing Image Data Interfaces

ID	Description	Inputs	Outputs
#1	Fish Classification	Image data	Fish Class
#2	Features Identification	Image data	Image Features

7 Database Layer Subsystems

The database layer contains 3 core layers which each represent a section of the database with a few sub-tables in each that will not be described in detail here. The 3 sections of tables within the database will pertain to Users, Posts, and Catches. Users containing user information and account information, Posts regarding information about the social media aspect of the app, and teh catches portion will be in regards to the catch logger and catch map.

7.1 Users

The Users layer is the layer of the database that will contain all information about the users and their accounts. Sub layers of this layer will include the tables: User, Achievements, AchievementsNames

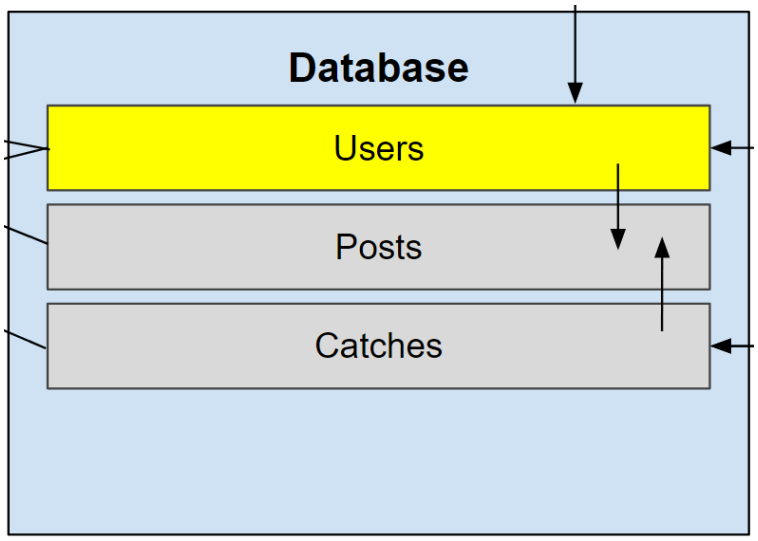


Figure 12: Database Users Description Diagram

7.1.1 Responsibilities

The responsibilities of this layer is store any information related to the user and their account. This also means log in information to be pulled when the user wants to log in to their account.

7.1.2 Subsystem Interfaces

.

Table 11: Subsystem interfaces

ID	Description	Inputs	Outputs
#1	User authentication	Login attempt	Correct or Incorrect
#2	Social Feed UI	Request for user info	Username and Picture
#3	Missions UI	Request for user info	Achievements Progress

7.2 Posts

The Posts subsection will be responsible for all the table that regard the social media posts. This includes the tables: Post, Reaction, and Comment

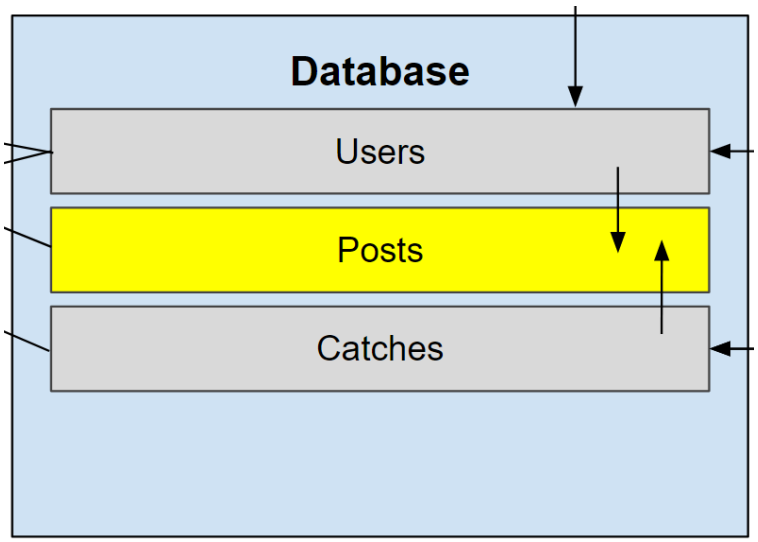


Figure 13: Database Posts Description Diagram

7.2.1 Responsibilities

This layer will be responsible for keeping track of all social media posts and their relevant information.

7.2.2 Subsystem Interfaces

Table 12: Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Social Feed UI	Request for post data	Post Data
#2	Social Feed UI	Request to post	Confirmation of post

7.3 Catches

This layer is responsible for all data pertaining to catches. This primarily means the table Catch.

7.3.1 Responsibilities

This table is responsible for keeping track of any and all data related to a user's particular catch including who caught, where they caught it, when they caught it.

7.3.2 Subsystem Interfaces

Table 13: Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Processing Image data	Image and species	New table entry
#2	Map UI	Request for catches	All catches requested

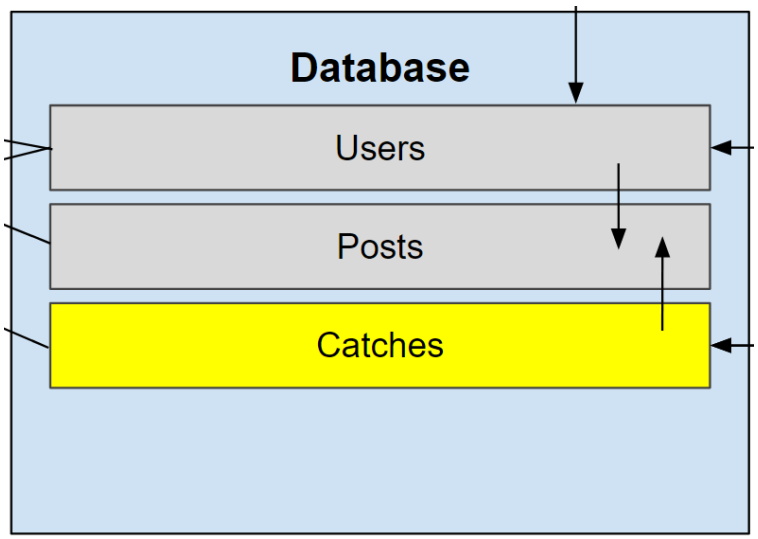


Figure 14: Database Catches Description Diagram

REFERENCES