1.INTRODUCTION

1.1 Purpose

The ScubaChain project seeks to revolutionize scuba diving certification verification by leveraging blockchain technology to create a reliable, decentralized, and transparent platform for managing diving credentials. The platform aims to eliminate issues associated with traditional paper-based systems, such as forgery and limited accessibility, by providing a secure and immutable digital ledger.

Through an integrated smartphone application, divers will have the ability to effortlessly confirm their certifications, ensuring ease of use and reliability. The system incorporates cutting-edge technologies such as the Interplanetary File System (IPFS) for decentralized storage and blockchain for data authenticity, enabling a seamless, tamper-proof solution tailored for the global diving community.

1.2 Scope

In this Software Design Document (SDD), we outline ScubaChain, a platform designed to revolutionize the scuba diving industry by transitioning from traditional paper-based systems to a secure, blockchain-based digital ecosystem. ScubaChain provides a transparent, tamper-proof solution for managing certifications and dive logs, ensuring data security, accessibility, and user engagement for divers, dive masters, and dive centers.

The platform consists of a web application for dive centers and dive masters and a mobile application for divers. These tools enable users to issue and verify certifications, log dives, and access real-time updates such as dive maps and weather conditions. Advanced features, including geolocation-based dive site maps, milestone based NFT rewards, and social sharing capabilities, enrich the user experience while fostering a global diving community.

By leveraging blockchain technology, ScubaChain ensures data authenticity and prevents unauthorized modifications or forgery. This approach not only enhances trust and reliability but also paves the way for a standardized and globally accessible scuba diving ecosystem. With future scalability in mind, ScubaChain is set to transform how the diving industry manages its critical processes, making them more secure, efficient, and community-driven.

1.3 Glossary

Term	Definition
Dive Center	A user entity responsible for verifying diver certifications, managing events, and updating dive locations.
Dive Master	A user entity with responsibilities such as managing dive events, verifying certifications, and assisting divers.
Diver	A primary user of the system, responsible for managing their diving certifications and participating in events.
License Hash	A unique identifier generated through blockchain technology to securely verify and store diving certifications.
Transaction_ID	An identifier for transactions logged in the blockchain to track operations like certification or NFT minting.
Smart Contract	Blockchain-based automated scripts that execute specific tasks like hashing licenses, NFTs, and dive logs.
System Information	The central module handling system-level operations such as user authorization, profile updates, and data requests.

1.4 Overview of Document

The Software Design Document (SDD) for the ScubaChain project outlines the architectural and design details essential for implementing a blockchain-based solution in the scuba diving industry. This document serves as a bridge between the requirements specified in the Software Requirement Specification (SRS) document and the actual development process by providing a detailed blueprint of the system's structure and components. The SDD for the

ScubaChain is structured to ensure a comprehensive understanding of the system's design, covering key aspects such as system architecture, data flow, and interaction patterns. This document is aimed at developers, project stakeholders, and anyone involved in the software development lifecycle, ensuring clarity and alignment across all phases of the project. The key objectives of this document include providing a clear understanding of the system's purpose and problem domain by highlighting the challenges in the scuba diving industry and how blockchain technology addresses them; defining the scope of the design by establishing the boundaries of the system and detailing the primary functionalities; describing the system architecture and its components by illustrating the high-level structure through diagrams and technical descriptions; specifying the technologies and methodologies used by identifying tools, frameworks, and best practices essential for the system’s implementation; and ensuring traceability between requirements and design by linking the design elements to the specific requirements outlined in the SRS document. By presenting a well-structured and detailed design framework, this document aims to facilitate the development of a robust, scalable, and secure blockchain system tailored to the needs of the scuba diving community. Each section of this document delves into specific aspects of the system, ensuring a holistic and methodical approach to design and implementation.

1.5 Motivation

This motivation section provides a comprehensive explanation of the origins of ScubaChain, its objectives, and its intended achievements. ScubaChain was developed with the aim of modernizing the scuba diving industry, which currently relies on fragile paper-based systems for managing certifications and dive logs. By leveraging blockchain technology, ScubaChain offers a secure, transparent, and tamper-resistant digital platform that enhances both integrity and accessibility. Additionally, ScubaChain seeks to support a sustainable diving ecosystem by integrating advanced features such as real-time dive area information and service location maps. Although ScubaChain was initially launched as a local application, our primary goal is to facilitate its adoption on a global scale.

2. SYSTEM DESIGN

The ScubaChain project leverages blockchain technology to provide a secure, transparent, and decentralized platform for scuba diving license verification. The system is designed to address existing inefficiencies and security risks in traditional license management processes by introducing a modern, user-centric digital solution. This document outlines the architectural design, functional components, explanation diagrams, and interaction flow that collectively define the ScubaChain system.

2.1 Architectural Design

This project includes totally six layer, consists of blockchain layer, smart contract layer, IPFS & database layer, backend layer, frontend layer and authentication & identity management layer. Blockchain serves as the backbone for secure issue and relates the objects with decentralized data storage (Which is IPFS component). This layer implements a public or permissioned blockchain (e.g., Ethereum) for managing license data records and verification transactions.

Smart contract layer automates license verification, renewal, and level-up processes based on predefined rules. In addition, it allows us to use functions (Minting, OpenSea publishing etc.) related to NFT items.

The IPFS & Database layer are divided into three separate parts. IPFS part handles the images about to converted into NFT and normal images. Blockchain sends to the PostgreSQL system and database handles the storage of dive logs and special hashes. General PostgreSQL Database stores the normal data types such as user credentials, location information, rating etc.

Backend layer manages the API & user requests; determines which user has access to which functions and processes API requests from other applications in the background. It determines at which layer the called functions will be processed and what outputs they will give to the user after processing.

The frontend & identity management layer indicates the application differently for each different user type, changes the appearance of the application and allowed functions. Also, this the identity management layer that it allows to sign up with Metamask account.

2.1.1 Problem Description

The ScubaChain project addresses critical issues within the scuba diving industry, which currently relies on outdated, paper-based systems for certification management and dive logging. These traditional methods are prone to loss, forgery, inefficiencies, and accessibility challenges, leading to a lack of trust and reliability. Divers face difficulties verifying their credentials, while dive centers struggle with secure data management and fraud prevention. The fragmented nature of the existing ecosystem further complicates collaboration among divers, dive masters, and organizations. ScubaChain aims to revolutionize this ecosystem by integrating blockchain technology to ensure transparency, security, and immutability. The platform will provide decentralized certification verification, enabling divers to manage their credentials seamlessly through mobile and web applications. Additionally, it facilitates digital dive logging, reducing dependency on physical records. By incorporating real-time data such as weather updates, geolocation-based dive site mapping, and gamified NFT rewards, the system enhances user engagement and fosters a connected diving community. Ultimately, ScubaChain seeks to establish a global, standardized, and secure digital framework, addressing current inefficiencies while paving the way for a more collaborative, trustworthy, and innovative scuba diving industry.

2.1.2 Technologies Used

The ScubaChain project incorporates a combination of advanced technologies to ensure security, scalability, and usability for now & future. Each technology is selected based on its ability to address the specific requirements of the system with most stable ones, including blockchain integration, decentralized identity management, and user-friendly interfaces. Below is a detailed breakdown of the technologies used:

1. **Blockchain Technology**

   Ethereum: Ethereum (public blockchain) is considered for their robust smart contract capabilities, security features, and widespread adoption. Ethereum enables public, decentralized verification, uses Proof of Stake (PoS) for consensus mechanism. [1]

   Smart Contracts: Written in Solidity (for Ethereum), smart contracts automate certification issuance, verification, and updates. Ensures immutable and tamper-proof record keeping. [2]

2. **Frontend Development**

   Web Application: Built using **React.js** for its component-based architecture, responsiveness, and scalability. Provides an intuitive interface for dive centers and certification organizations to validate licenses and manage certifications.

   Mobile Application: Developed using **Flutter** for cross-platform compatibility, ensuring a seamless experience on both Android and iOS devices. Enables divers to access their certifications, manage dive logs, and interact with the system easily. [3]

3. **Backend Development**

   Node.js: Used to build a robust, scalable backend server to handle API requests, manage business logic, and facilitate blockchain interactions. [4]

   Express.js: A lightweight web application framework for Node.js, streamlining the development of RESTful APIs.

4. **Database Management**

   PostgreSQL: A relational database system used for storing off-chain data, such as user profiles and certification metadata. Offers strong consistency, scalability, and integration capabilities with blockchain-based applications. [5]

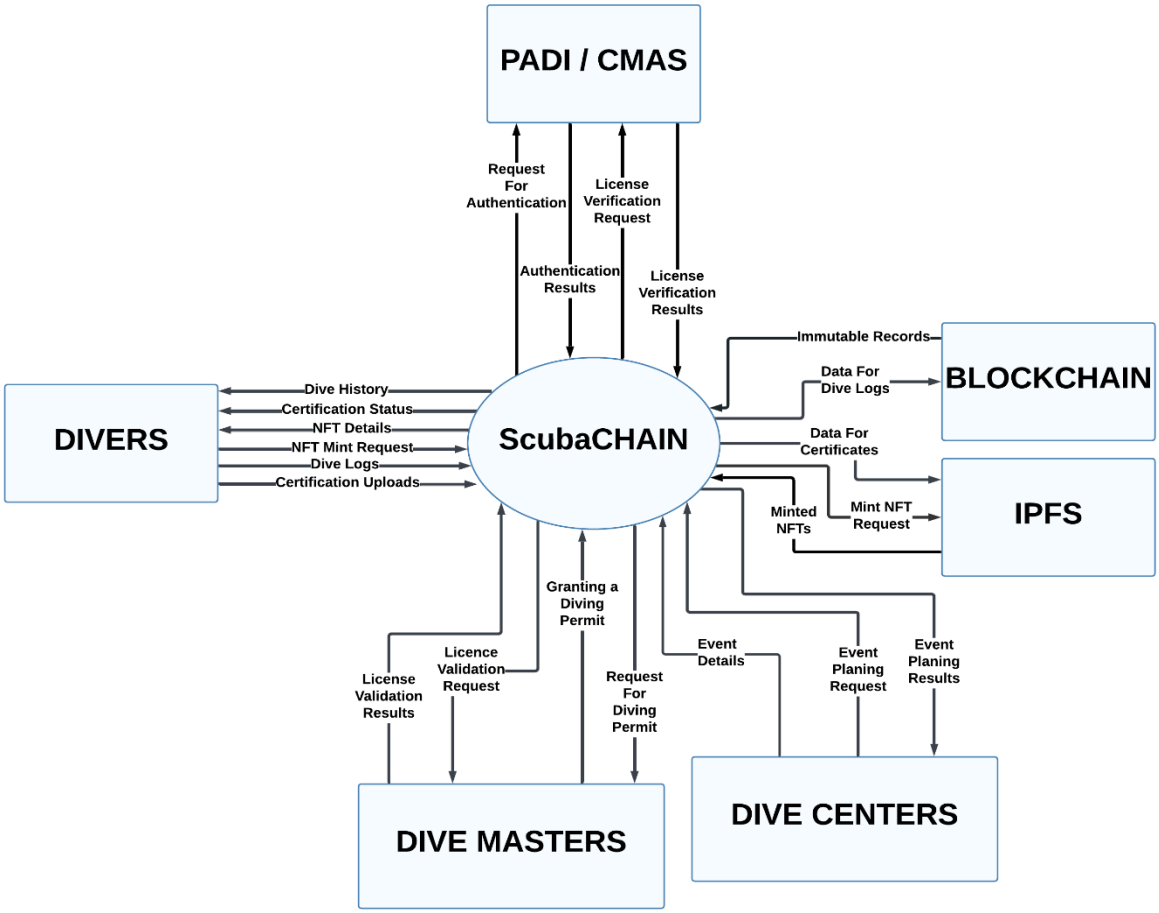
   IPFS (InterPlanetary File System): Used for decentralized storage of certification-related documents, ensuring data integrity and accessibility. [6]

5. **Authentication and Identity Management**

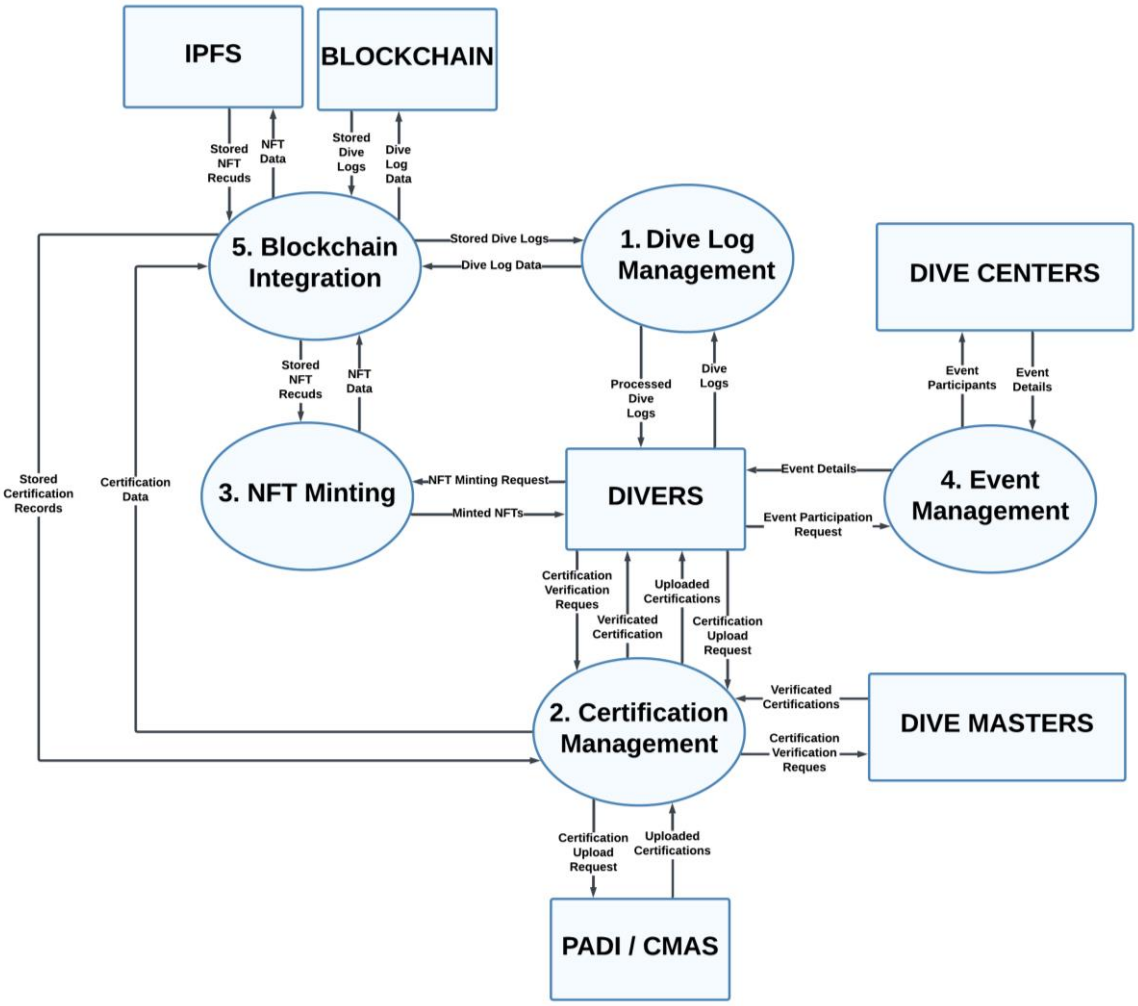
   Digital Wallet Integration: Supports blockchain wallets like **MetaMask** for secure user authentication and interaction with the blockchain. [7]

2.1.3 Data Flow Diagram (DFD)

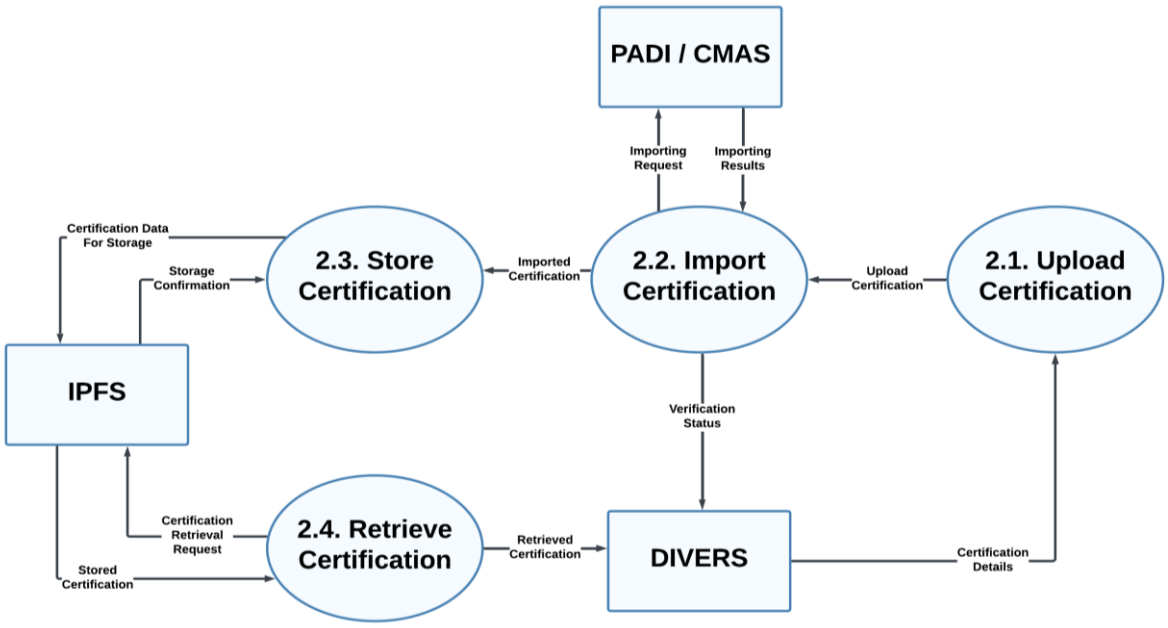
Level 0 (Context Diagram)



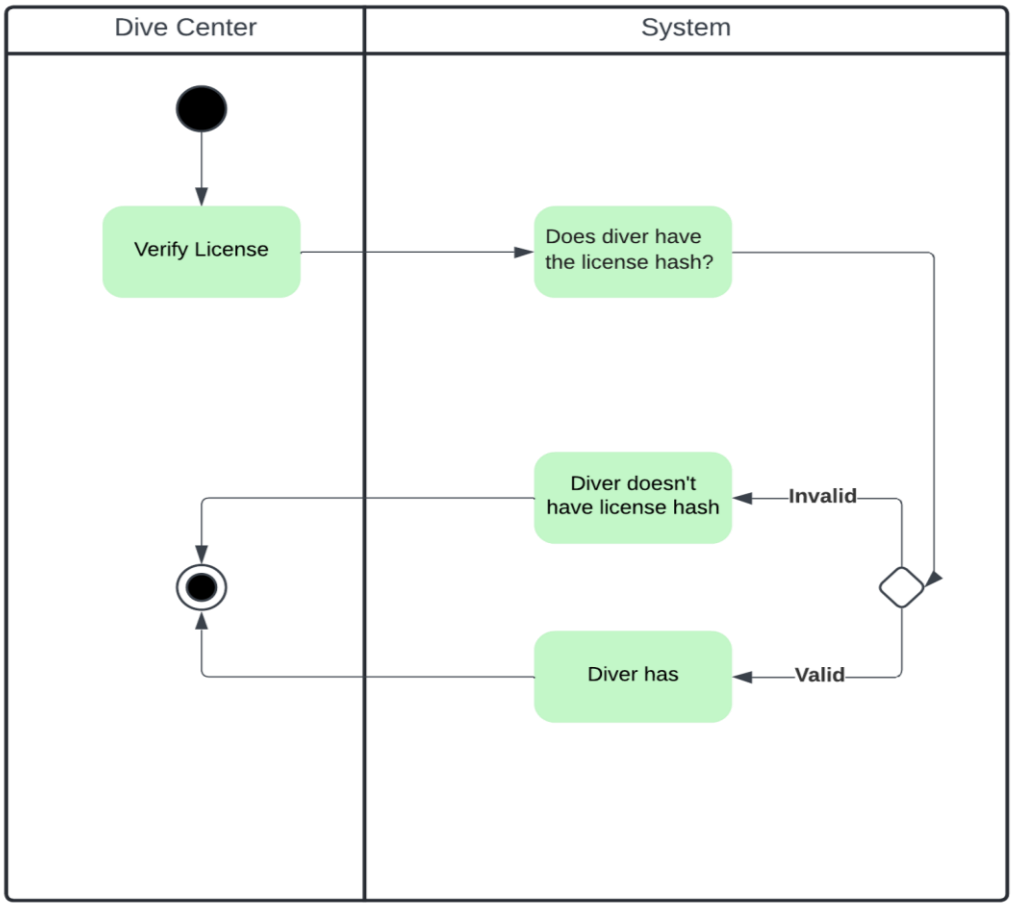
Level 1

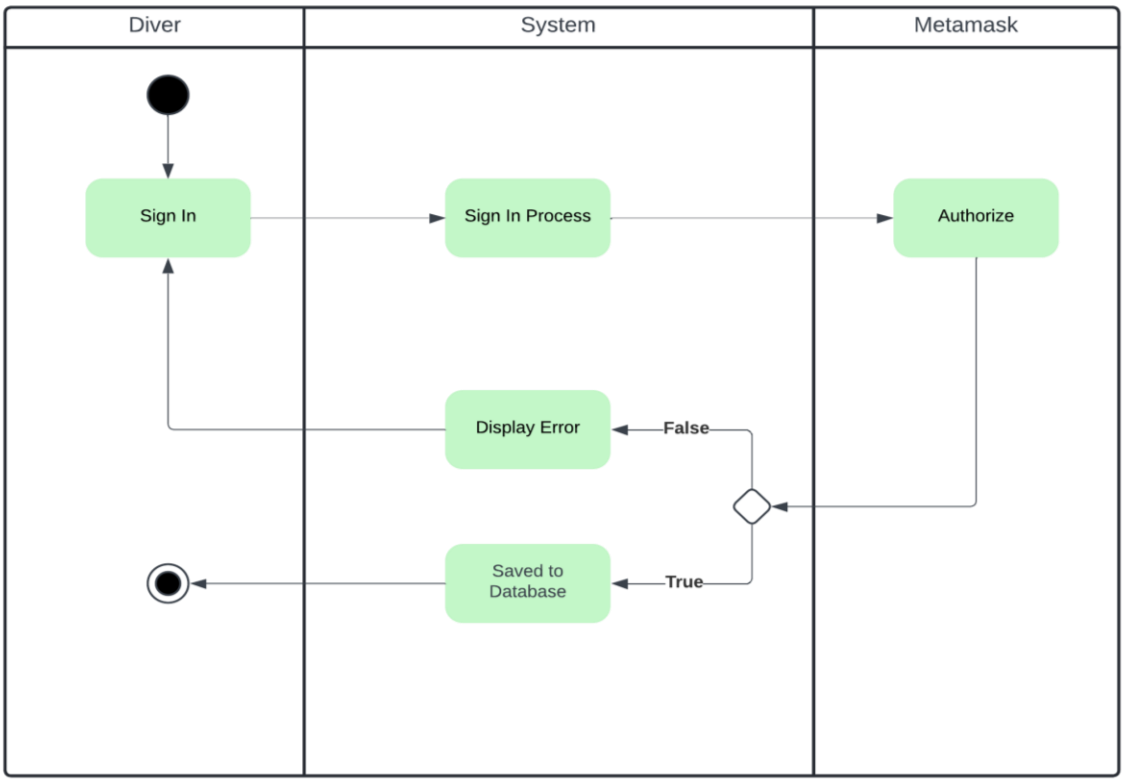

IPFS
BLOCKCHAIN
Stored NFT Recuds
NFT Data
Stored Dive Logs
Dive Log Data
5. Blockchain Integration
Stored Dive Logs
Dive Log Data
1. Dive Log Management
DIVE CENTERS
Stored NFT Recuds
NFT Data
Processed Dive Logs
Dive Logs
Event Participants
Event Details
Stored Certification Records
Certification Data
3. NFT Minting
NFT Minting Request
Minted NFTs
DIVERS
Event Details
Event Participation Request
4. Event Management
Certification Verification Reques
Verificated Certification
Uploaded Certifications
Certification Upload Request
Verificated Certifications
2. Certification Management
Certification Verification Reques
DIVE MASTERS
Certification Upload Request
Uploaded Certifications
PADI / CMAS

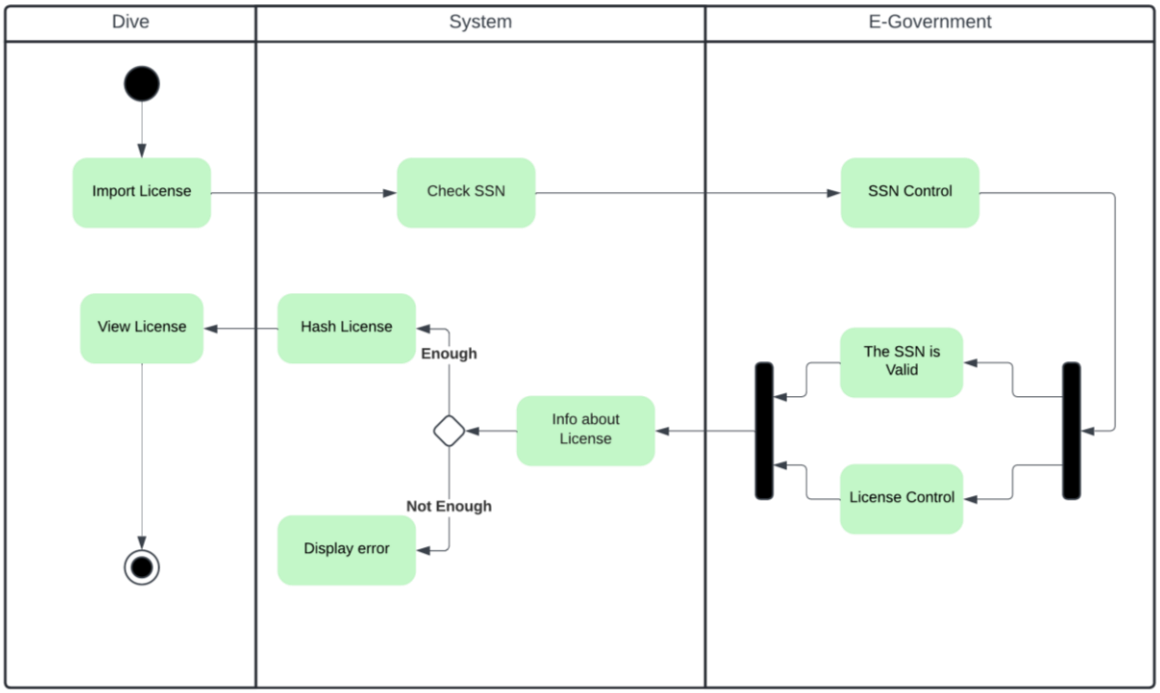
Level 2

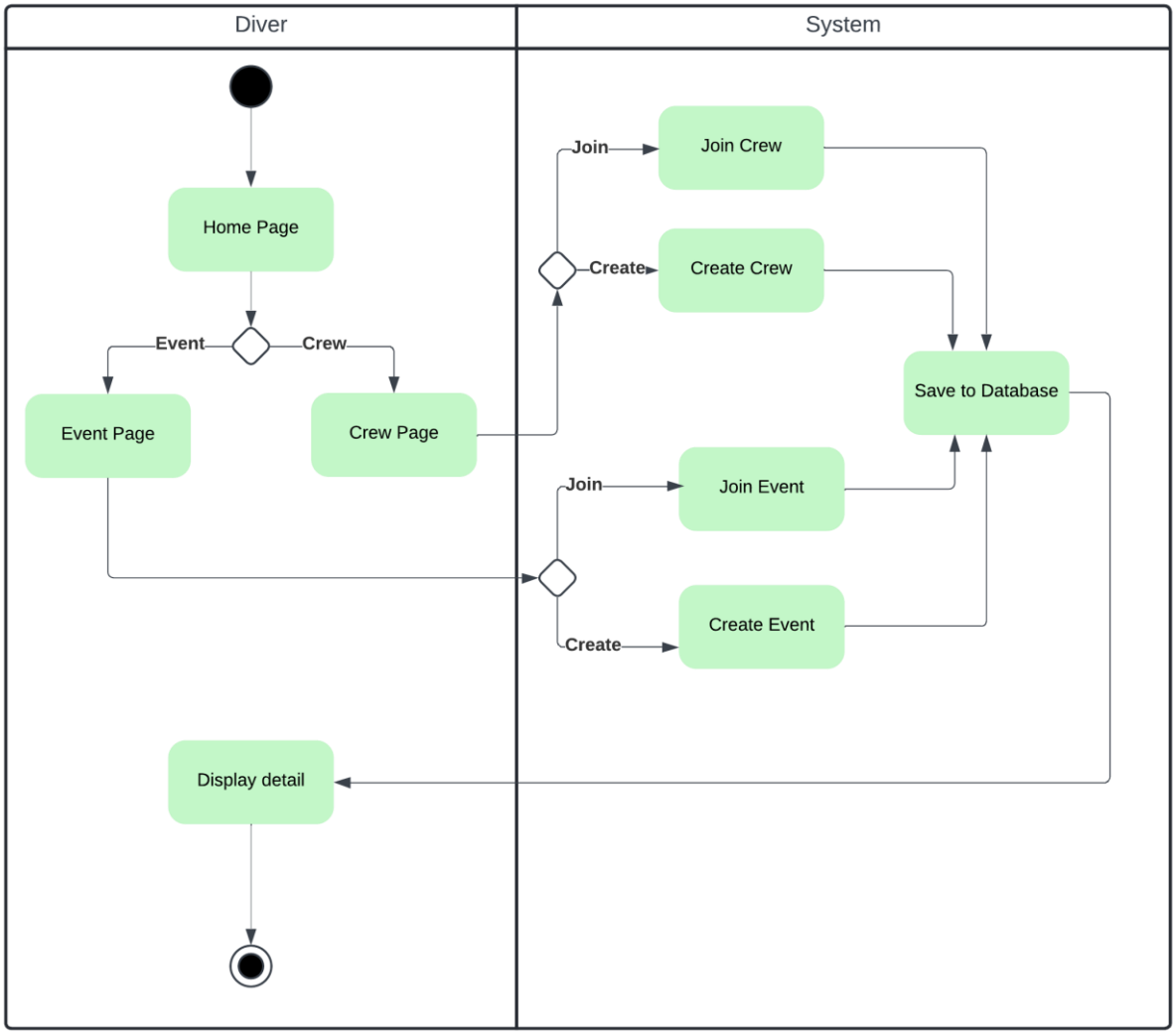


2.1.4 Activity Diagram

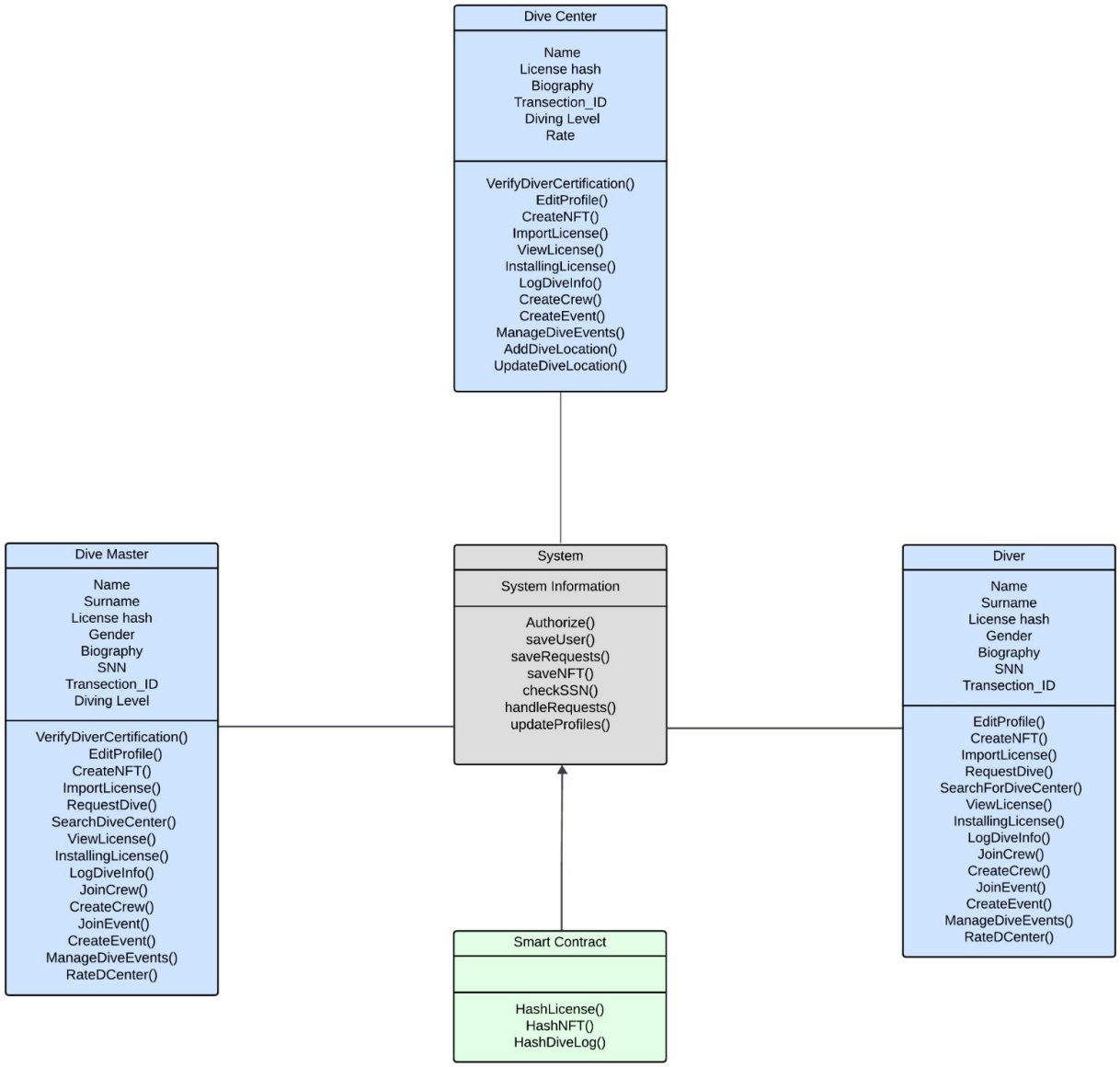



Diver
System
Metamask
Sign In
Sign In Process
Authorize
Display Error
False
Saved to Database
True


Dive
System
E-Government
Import License
Check SSN
SSN Control
View License
Hash License
Enough
The SSN is Valid
Info about License
Not Enough
Display error
License Control

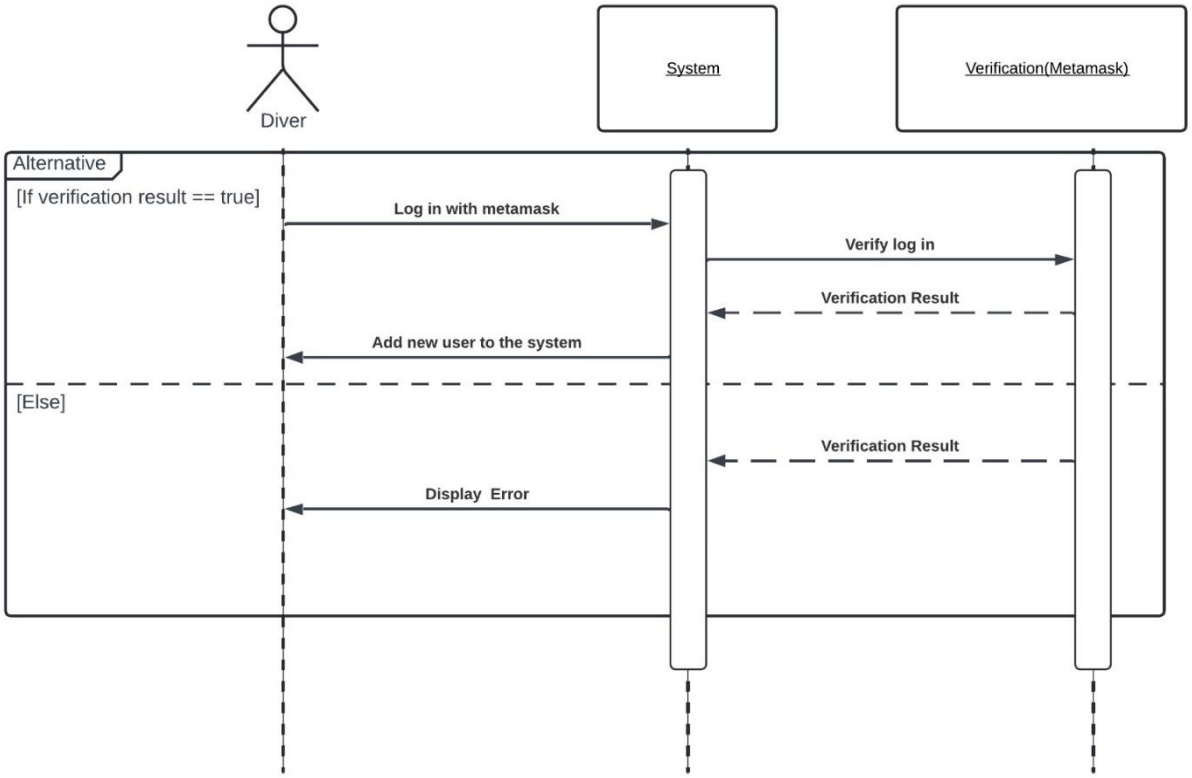

Diver
System
Home Page
Event
Crew
Event Page
Crew Page
Join
Join Crew
Create
Create Crew
Save to Database
Join
Join Event
Create
Create Event
Display detail

2.1.5 Class Diagram

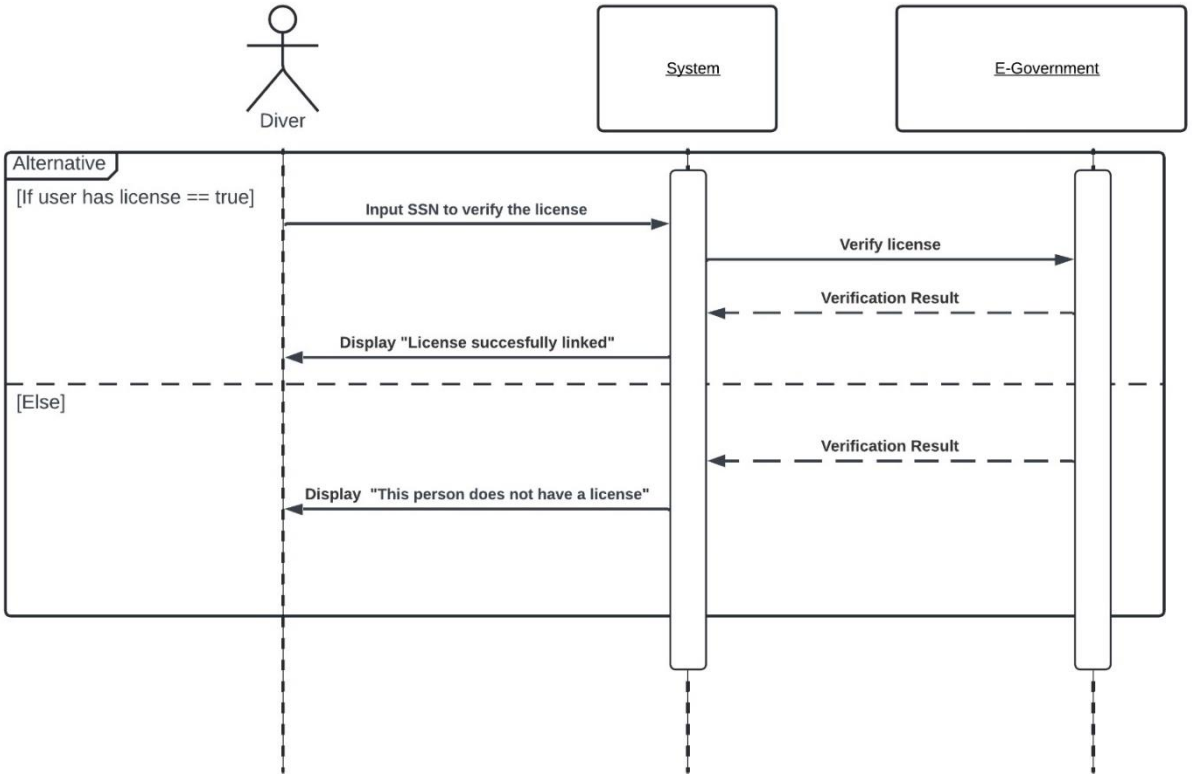


2.1.6 Sequence Diagram

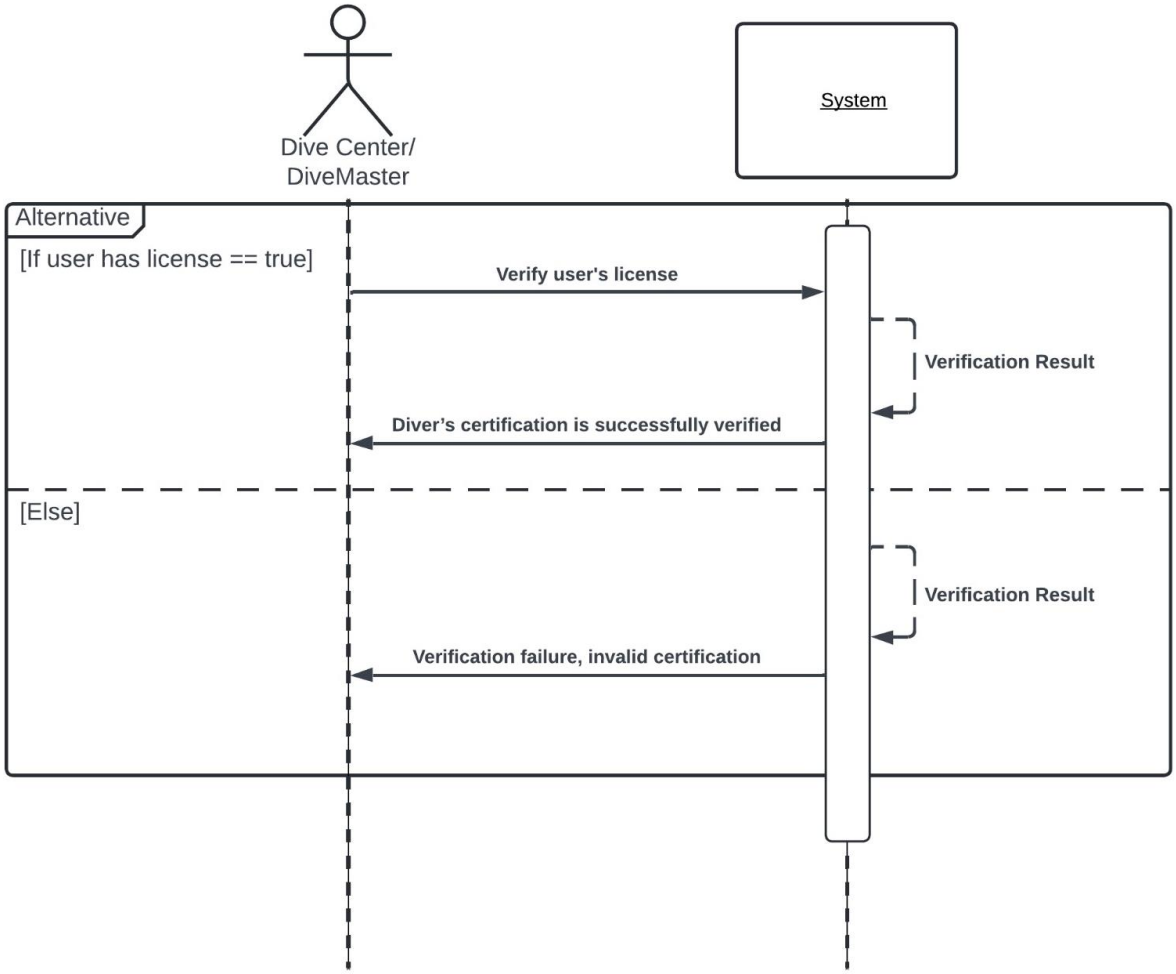
Sequence Diagram of Use Case 1 - Register/Login with Metamask



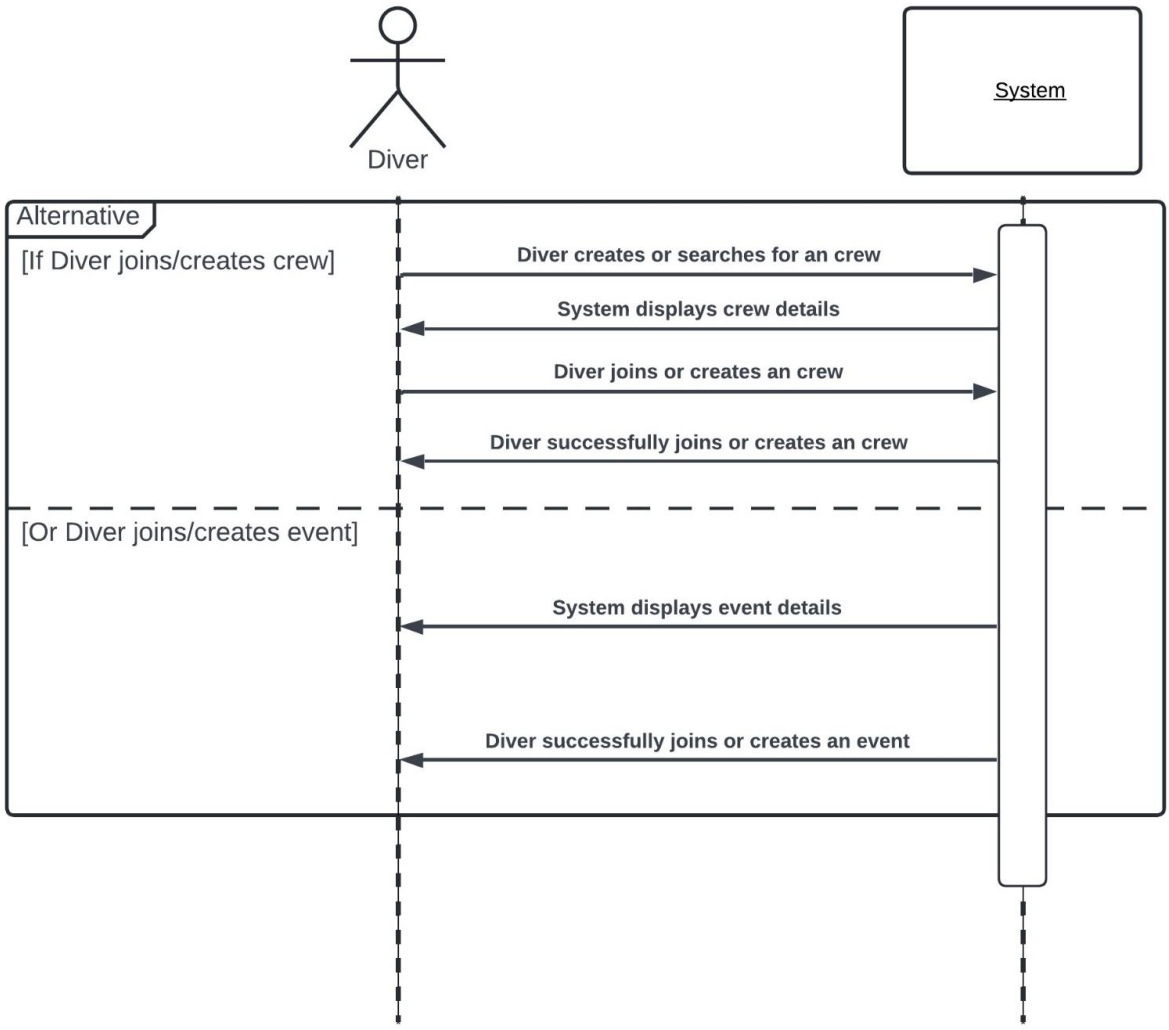
Sequence Diagram of Use Case 2 - Installing the License



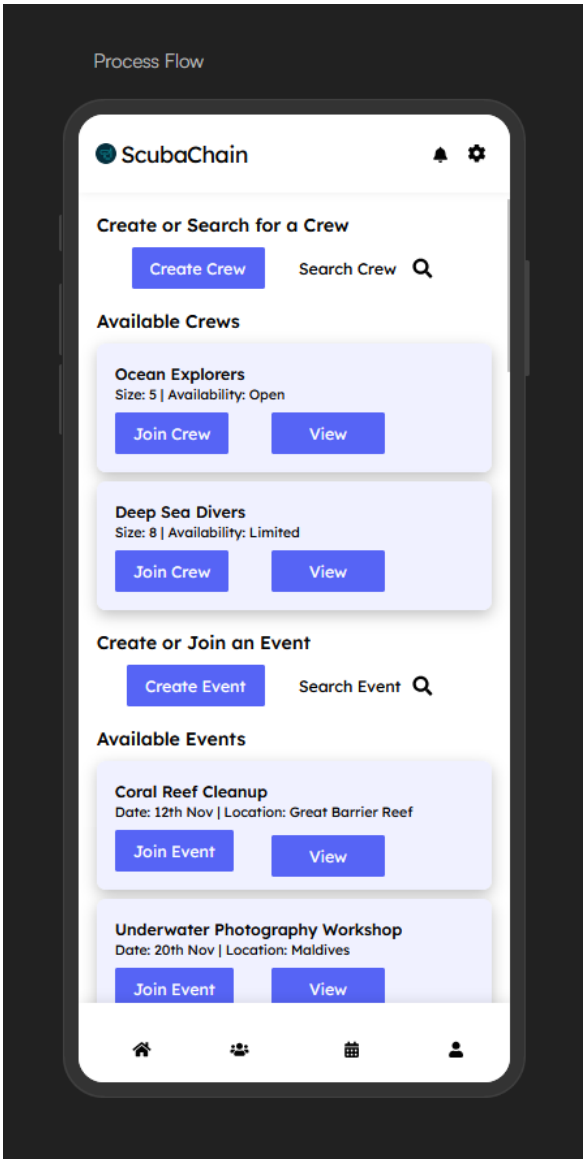
Sequence Diagram of Use Case 10 -Verify Diver Certification

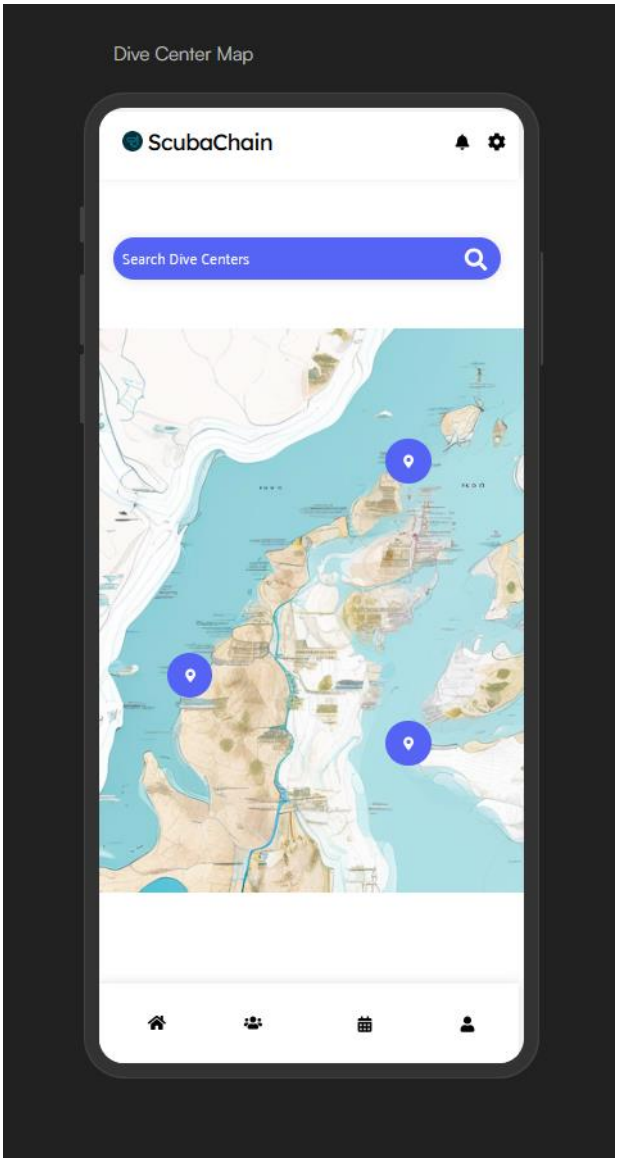


Sequence Diagram of Use Case 9 -Creating/Joining a Crew & Event

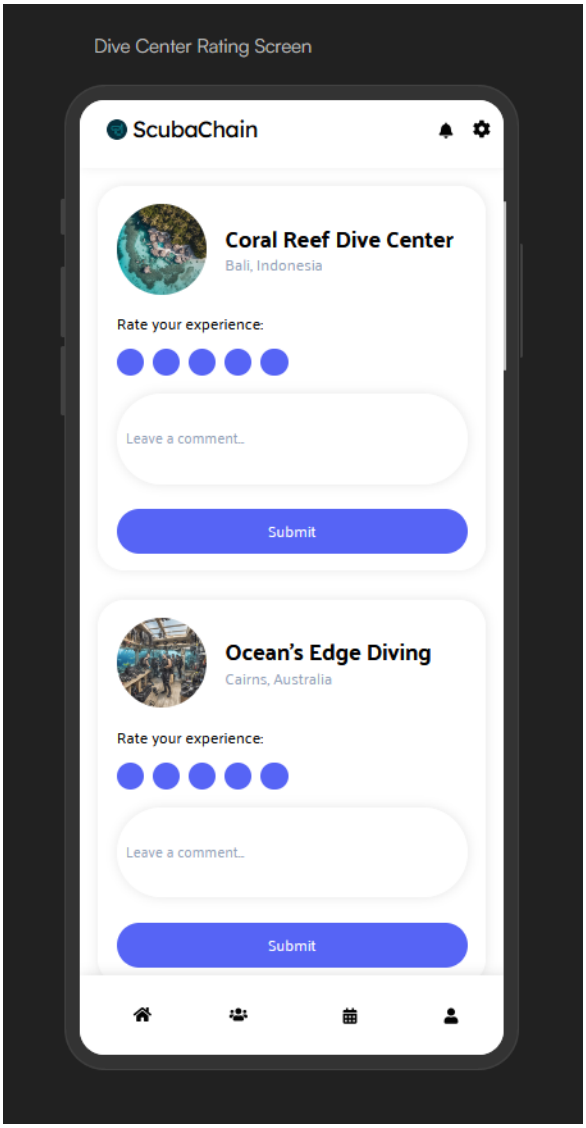


3. USER INTERFACE (UI) DESIGN

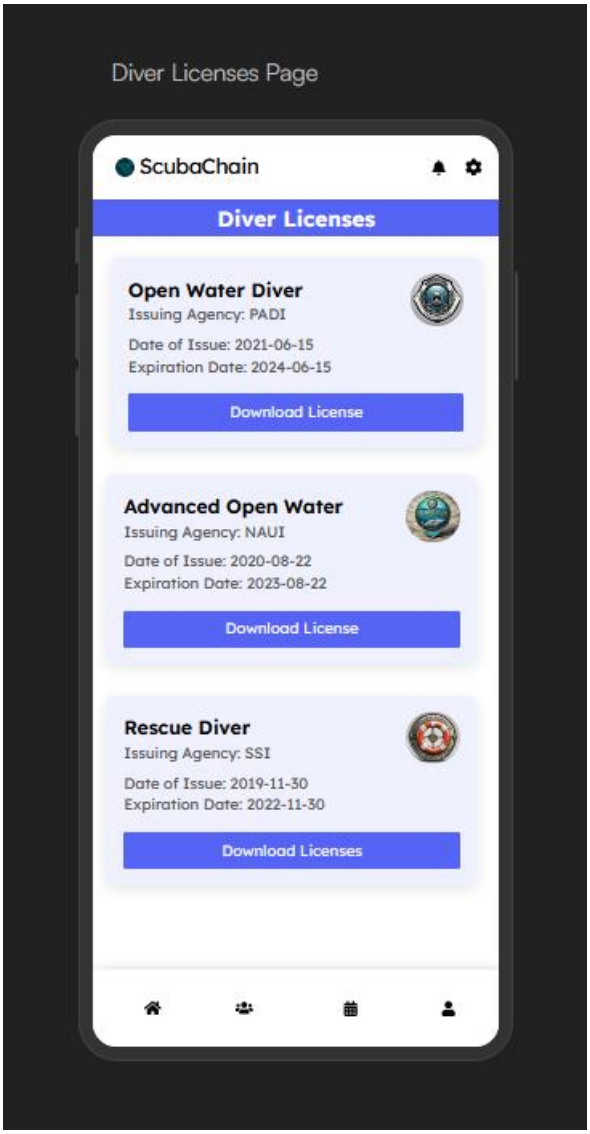




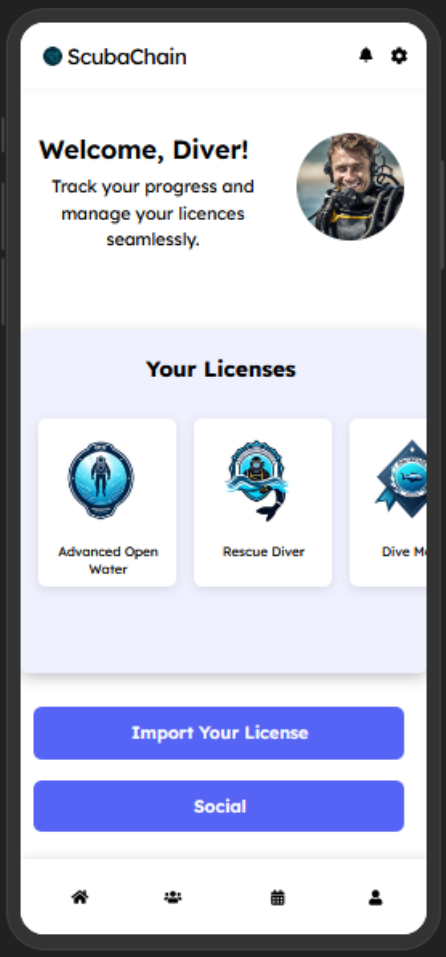
Dive Center Rating Screen



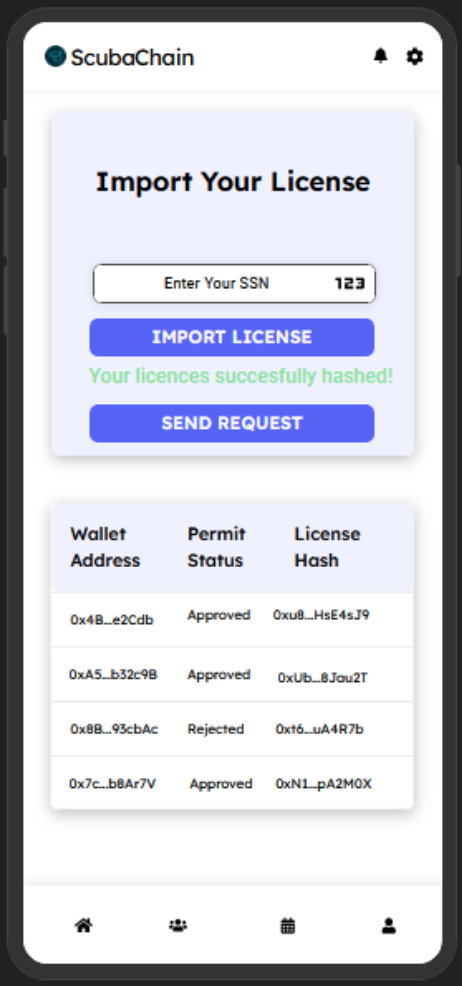
Diver Licenses Page

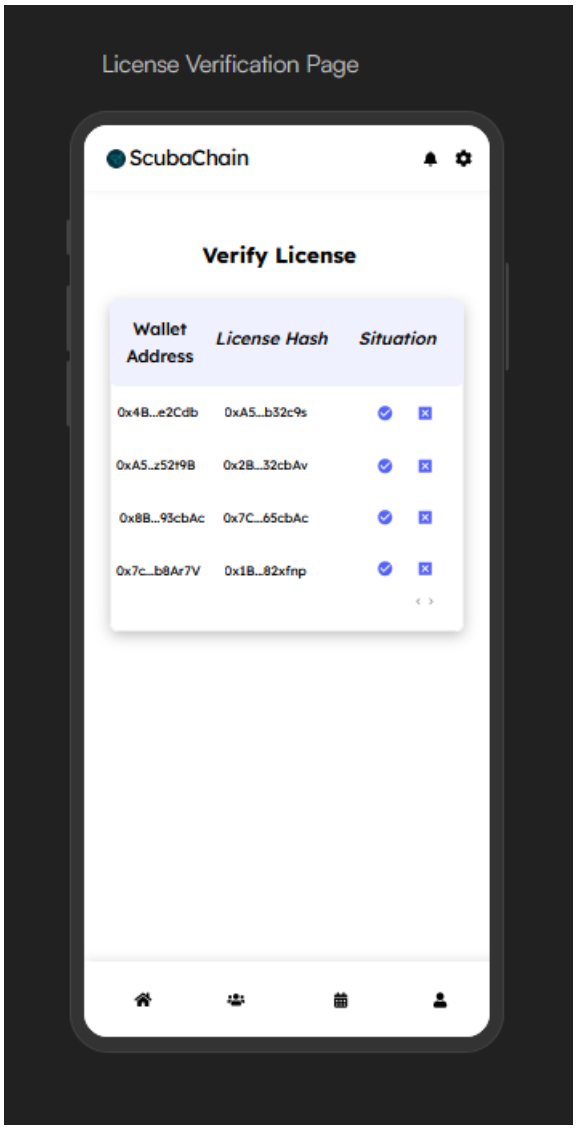


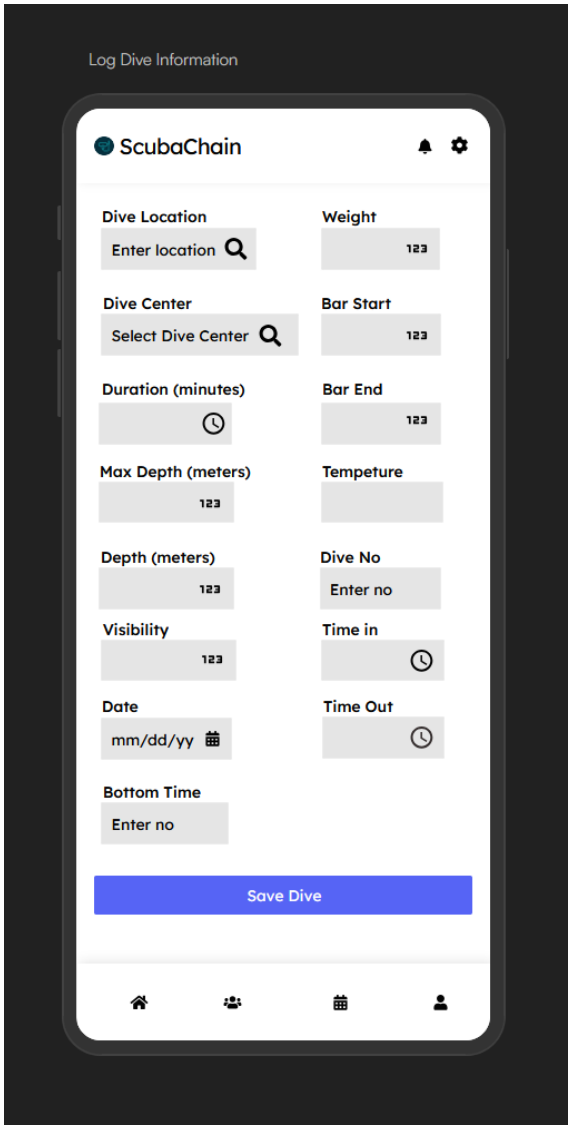
Home Page

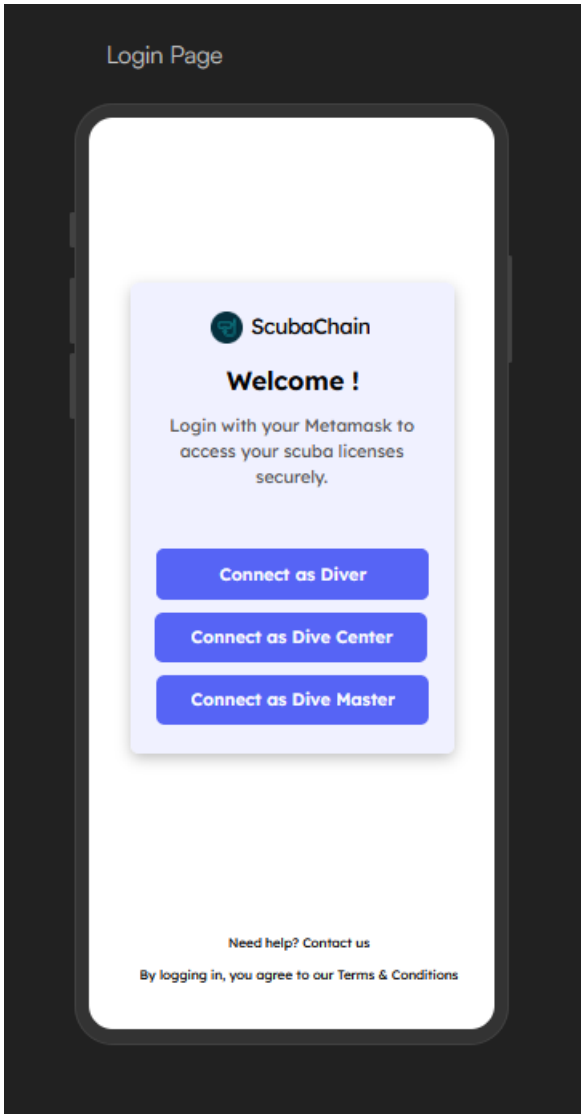


Import License Page




License Verification Page
ScubaChain
Verify License
Wallet Address
License Hash
Situation
0x4B...e2Cdb
0xA5...b32c9s
0xA5..z52t9B
0x2B...32cbAv
0x8B...93cbAc
0x7C...65cbAc
0x7c...b8Ar7V
0x1B...82xfnp


Log Dive Information
ScubaChain
Dive Location
Enter location
Weight
123
Dive Center
Select Dive Center
Bar Start
123
Duration (minutes)
Bar End
123
Max Depth (meters)
123
Tempeture
Depth (meters)
123
Dive No
Enter no
Visibility
123
Time in
Date
mm/dd/yy
Time Out
Bottom Time
Enter no
Save Dive



4. REFERENCES

[1] Ethereum.org, “Networks,” ethereum.org, https://ethereum.org/en/developers/docs/networks/ (accessed Dec. 24, 2024).

[2] “Solidity” Solidity, https://docs.soliditylang.org/ (accessed Dec. 24, 2024).

[3] “Flutter documentation” Docs, https://docs.flutter.dev/ (accessed Dec. 24, 2024).

[4] “Node.js V23.5.0 documentation,” Index | Node.js v23.5.0 Documentation, https://nodejs.org/docs/latest/api/ (accessed Dec. 24, 2024).

[5] “Main page,” PostgreSQL wiki, https://wiki.postgresql.org/wiki/Main_Page (accessed Dec. 25, 2024).

[6] A. Shikalgar, (PDF) a review on “ipfs based decentralized social media platform,” https://www.researchgate.net/publication/371158121_A_Review_on_IPFS_Based_Decentralized_Social_Media_Platform (accessed Dec. 25, 2024).

[7] “Integrate your dapp with the metamask wallet,” MetaMask developer documentation, https://docs.metamask.io/wallet/ (accessed Dec. 25, 2024).

Official results for JS Web Frameworks Benchmark, https://krausest.github.io/js-framework-benchmark/index.html (accessed Dec. 25, 2024).