ÇANKAYA UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

CENG-407

SOFTWARE REQUIREMENTS SPECIFICATION

Adopting Blockchain Technology in Scuba Diving

Deniz KAR 202011042

İclal Sezin GÜRSES 202111017

Arda Celal KAPLAN 202111013

Ahmet Berk EROĞLU 202011411

Mustafa Arda ERDİNÇ 202011073

1. INTRODUCTION

1.1. Purpose

The ScubaChain project is a blockchain-based platform for scuba diving certification verification, and this article outlines its needs. To provide a reliable platform to the diving community, the project aims to create a decentralized, transparent, and safe system for confirming diving licenses.

Even through a smartphone application, the system will let divers confirm their credentials. IPFS (InterPlanetary File System) and blockchain technology will be used to guarantee that data is reliable and immutable.

1.2. Scope of Project

In this project, we are creating ScubaChain, a revolutionary, blockchain-based platform to bring the scuba diving certification and dive log management ecosystem into the 21st century. Scuba Diving industry is still using the traditional systems based on the paper which are prone to loss, forgery, and limited accessibility. ScubaChain applies blockchain technology to offer a transparent, tamper-proof, and trustworthy digital solution to increase trust, data security, and user engagement among divers, dive masters, and dive centers.

The service will include two main parts: a web application and mobile application for dive centers, dive masters and divers. The web interface and the mobile app will let dive schools and dive masters issue and authenticate certificates, skills, and record dives on the blockchain. Meanwhile, they will allow divers to view and control their certifications and dive accounts, request validations, book dives and keep track of live updates such as dive maps and weather. They'll fix the current system and create a global, secure and standardized scuba diving ecosystem.

Certification management that is issuance and auditing of certifications in a permanently locked digital ledger and dive log management that helps divers keep their dive record digitally safe are its key features. Blockchain will be essential to ensure data authenticity and evasion of unauthorized modification or forgery. As it takes the place of traditional paper, certifications and dive records will be safer, more convenient and less prone to data loss with the new platform.

ScubaChain is tailored for 3 different user segments; divers, dive masters and dive centers. Diving customers will manage certifications, record dives, ask for verifications and join the community of diving users via social media and event participation on the mobile app. Certification verification, dive log checking, dive event management, and community events to be performed by dive masters. Dive centers will be administrative hubs that issue certifications

and keep a clout of diver knowledge and experience. Each user persona will see customized screens and components for ease of use and for their specific purposes. In addition to basic certification and log management, ScubaChain incorporates several advanced features to enrich the user experience.

The platform will provide geolocation-based dive site maps, real-time weather and water condition updates, and interactive gamification elements. Divers will have the opportunity to earn NFT-based rewards for achieving milestones or participating in events, adding a gamified dimension to the platform that encourages engagement and fosters a sense of accomplishment. Social features, such as the ability to share dive experiences and achievements on social media, will further strengthen community connections.

The objectives will be attained by investing in cutting edge technology, incorporating blockchain for safety and reliability of data, building a web platform for dive centers and dive masters, and creating mobile applications for Android and iOS smartphones. Additional cloud-based services will provide other features, including integration of up-to-date weather forecasts and geolocation mapping. This technology-driven approach will provide the platform the desired user experience which is secure and dependable.

The main outputs of the project include web based interactive platform for dive centers and dive masters, mobile application for divers, and secure certification and log management blockchain structure. Other functionalities such as, milestone-based NFT rewards, interactive maps for dive sites, and tools to engage with the community will further enrich the usefulness of the platform. The scope however, does not rule out the management and integration of logs and certification as the main scope because certifying agencies and registration services might in future broaden the view.

Some elements fall outside the scope of this project at this time. For example, the introduction of hardware devices like underwater tablets for dive logging is not part of the focus. Advanced level training simulations or e-learning modules for certification courses in scuba diving also fall out of the scope of this project although they may be proposed in later versions. Integration with systems unrelated to blockchain certification is as well eliminated at this point.

In summary, ScubaChain wants to make paper work in the scuba diving industry a thing of the past by making online processes secure, transparent and community centered. The framework emphasized will develop towards incorporating advanced features and standards to build and manage global certification and dive logs, and strengthen the global diving community system.

1.3. Glossary

Term	Definition
Blockchain	A decentralized and immutable database system. Used in ScubaChain for securely verifying and storing diving licenses.
Smart Contract	Automated code pieces running on the blockchain. Manages processes such as NFT minting and license verification.
IPFS (InterPlanetary File System)	A decentralized file storage system. Securely stores diving licenses and NFT photographs.
MetaMask	An Ethereum-based digital wallet used for logging into the system and signing blockchain transactions.
NFT (Non-Fungible Token)	Unique digital assets. Converts photos taken during dives into digital collectibles recorded on the blockchain.
Minting	The process of creating new NFTs by recording digital content (e.g., photos, certificates) onto the blockchain through smart contracts.
Diver	A primary system user who queries and verifies their own diving license.
Divemaster (Dive Center)	A user type responsible for verifying divers' licenses and providing dive center information on the map.
PADI/CMAS/SSI	International organizations providing diving licenses. The system uses their databases for license verification.
CID (Content Identifier)	A hash value that uniquely identifies files stored on IPFS.
Frontend	The visual interface that users interact with.
Backend	The server-side system managing databases and API operations.

API (Application Programming Interface)	An interface facilitating data exchange between applications, such as querying diving licenses from the PADI/CMAS database.
GPS (Global Positioning System)	A system used for identifying the locations of dive centers on the world map.
Pinning	A mechanism used to ensure permanent storage of files on IPFS.
Public Key	Represents a user's identity on the blockchain. Used for license verification and login operations.
Hash	An algorithmic output that encrypts data uniquely on the blockchain, ensuring the accuracy of license information.
JSON (JavaScript Object Notation)	A lightweight and human-readable format used for data exchange in API calls and metadata files.
WalletConnect	A protocol enabling integration of digital wallets, like MetaMask, with mobile devices.
Proof of Stake (PoS)	A transaction validation mechanism for blockchain networks such as Ethereum, operating with low energy consumption.
Authentication	The process of securely logging users into the system, typically done via MetaMask wallet.
Decentralization	Refers to the operation of a system without reliance on a central authority, a key feature of blockchain and IPFS.

1.4. Overview of the Document

This document covers the ScubaChain project's technical and functional aspects, including how users will interact with the system and how it will operate. It's divided into two primary parts of the document, each aimed at a distinct user:

General Overview and Product Viewpoint: This part provides an overview of the main characteristics of the product, user profiles, and the technologies that "will be used" in its development.

Requirement Specification: Technical information on hardware and software interfaces, system performance standards, security standards, and interfaces with other systems is provided in this part.

2. OVERALL DESCRIPTION

2.1. Product Perspective

ScubaChain is a digital platform that aims to develop the diving industry. Diving certification, skill verification and diving records are transferred to the digital environment with Blockchain technology in a secure, transparent and unalterable way with advanced technology. With this life-making method, it streamlines all processes by strengthening the connections between individual divers, dive masters and dive centers. ScubaChain allows diving centers and dive leaders to verify certificates and record dives on the blockchain, allowing divers to easily check their personal records using the mobile application and instantly track their dive transactions and certificates. At the same time, whether divers are suitable for diving or not can be viewed through this application. ScubaChain aims to develop the diving industry by digitizing the registration processes and making diving easier and safer.

2.1.1. Development Methodology

ScubaChain's development principle is based on agile, a software development philosophy that can respond quickly and easily to changing project requirements. This methodology specifically emphasizes collaboration, includes User-Centered Development processes, and receives continuous feedback, uses this feedback to improve the application, and enables rapid adjustments to the project.

The initial phase of the development process begins with determining the needs of divers, dive leaders and dive centers, and continues with a detailed analysis of what will be required by determining what the platform will be like, what its features will be, what work the platform will be used for. This analysis will form a basis to maximize the user experience and satisfaction of the platform. Afterwards, the development team focuses on how the design of the platform will be and the functionality of the platform.

Agile methodology facilitates seamless communication and collaboration between divers, dive leaders and dive centers who will use the application. Thus, it enables the development team to detect potential problems in advance and intervene easily. At the same time, agile

methodology ensures that the most up-to-date and best technologies are used in the project. This approach ensures that the application keeps up with future technologies and prevents any problems when updating it, and contributes to ScubaChain being a platform that complies with industry standards. During the development period, with the contribution of the iterative structure of the agile methodology, each component is quickly prototyped and then tested. Since this event will be held at regular intervals, it receives feedback from all kinds of users, ensuring that the product improves at every stage and creates a positive impact for users. This process aims to create a reliable, user-friendly and scalable platform while also making people love diving with ScubaChain.

2.2. User Characteristic

2.2.1. Divers

Divers form the primary user group of the platform. This group includes both beginner divers and experienced professionals. Their main needs are to manage diving certifications securely, log dives, and engage with the diving community. Additionally, community organizers and individuals planning diving events are included in this category. These users require:

- **Certification Management**: They need tools to verify existing diving certifications, upload new ones, and securely store them on the blockchain.
- **Dive Logging and Monitoring**: Divers want to record their past dives, manage details, and track equipment information using user-friendly tools.
- **Community Engagement**: Features like social media integration, NFT minting, and participation in competitions enhance their interaction with the community.
- **Mobile Application Usage**: A seamless and accessible mobile app experience for quick actions, sharing dive logs, and receiving reminders is essential.
- **Multilingual Support**: Supporting multiple languages is critical for the platform's global user base.

2.2.2 Divemaster

Dive masters represent the platform's second major user group. Their primary needs revolve around verifying divers’ certifications, organizing dives, and simplifying their operational workflows.

- **Certification Verification**: They require a quick and reliable way to authenticate divers' certifications during training or events.
- **Dive Organization**: They need tools to manage divers' records, assign divemasters, and oversee event details efficiently.
- **Map and Location Management**: Dive masters need map-based tools to add new dive sites, update existing locations, and share these details with divers.

- **Blockchain Security**: Leveraging blockchain technology for reliable storage and verification of certifications is a key requirement.
- **Brand Profile Management**: Customizable profile pages to promote their services and showcase their offerings are highly valued.

3. REQUIREMENTS SPECIFICATION

3.1. External Interface Requirements

3.1.1. User Interfaces

1. Diver Interface

The interface for divers is designed to provide a seamless and engaging user experience while addressing their specific needs, as outlined in the user characteristics.

- **Mobile Application Dashboard**: The home screen of the mobile application offers a personalized dashboard. Divers can view their certification status, recent dive logs, notifications for upcoming certifications, and suggestions for dive sites based on their preferences.
- **Certification Management**: A dedicated section allows divers to upload and verify certifications using blockchain technology. The interface guides users step-by-step through the verification process and provides a clear status update on the certification's validity.
- **Dive Logbook**: An intuitive digital logbook enables divers to record dives, including details like location, depth, dive duration, and equipment used. The interface includes a timeline view to help users track their diving history visually.
- **NFT Minting for Dive Memories**: Divers can upload images or videos from their dives and mint them as NFTs directly from the app. This feature integrates seamlessly into the dashboard, ensuring an easy and quick minting process.
- **Social Integration**: The interface supports social sharing, allowing divers to post dive logs or achievements directly to platforms like Instagram or Twitter with one click.
- **Multilingual Support**: The user interface includes a language selection feature to accommodate a global user base, ensuring accessibility for non-English-speaking divers.

2. Dive Center and Divemaster

This interface caters to the operational needs of dive centers and divemasters, ensuring efficiency and accuracy.

- **Certification Verification Panel**: The interface provides a streamlined tool for verifying divers' certifications by scanning QR codes or entering blockchain-based certification IDs.
- **Event and Dive Management**: A calendar-based interface allows dive centers to schedule dives, assign divemasters, and manage participant details efficiently.
- **Location and Map Tools**: Dive centers can add or update dive site information through an integrated map tool. Users can pin dive locations, add descriptions, and upload images to create a comprehensive dive map.
- **Profile Management**: Dive centers can customize their profiles by adding branding elements, services offered, and promotional content. This profile is visible to divers when they search for dive centers.
- **Blockchain Interaction**: Tools for securely uploading and managing certification data using blockchain technology are integrated into the interface. The system ensures that all interactions are intuitive and do not require advanced technical knowledge.
- **Analytics Dashboard**: Dive centers can view analytics related to their operations, such as the number of certifications verified, participants per event, and feedback from divers.

3. Common Features

Both divers and divemasters share access to certain common functionalities designed to enhance user experience.

- **Login and Authentication**: Secure login via MetaMask or WalletConnect ensures data privacy. The authentication process is simple and user-friendly, allowing quick access for all users.
- **Notifications and Reminders**: The notification center alerts users about expiring certifications, upcoming dives, or new features added to the platform.
- **Help and Support**: An integrated support system provides FAQ sections to assist users with technical or operational issues.
- **Mobile-First Design**: Both interfaces are optimized for mobile devices, ensuring responsive and visually appealing layouts for smartphones and tablets.

3.1.2. Hardware Interfaces

To access this platform, the necessary hardware features are up-to-date, tablets, laptops and desktop computers that can access any website, as well as smart devices such as any phone or tablet that can download current applications from its own store application.

3.1.3. Software Interfaces

The software interfaces for the ScubaChain project will be as follows:

• Mobile Application (Front-end):

1. It will be developed using Flutter.
2. It will provide separate interfaces for divers and dive centers.
3. License validation, viewing blockchain hashes and IPFS connections features will be added.

• Server (Back-end):

1. It will be developed using Node.js (Express.js).
2. PostgreSQL will be used to manage user and license data.
3. RESTful APIs will handle communication between the front-end and back-end and will perform validation with PADI/CMAS databases.

• Blockchain and IPFS Integration:

1. The Ethereum blockchain will be used to validate license data through smart contracts written in Solidity.
2. License documents will be stored on IPFS, which ensures decentralized storage by providing access via Content Identifier (CID).

3.1.4 Communication Interfaces

Even though this application works on two different device / platform types, its general functional structure is the same. However, examining these platforms under two separate headings will highlight which tools are required for each device type. To deal with blockchain integration, all systems will be developed in Solidity, which is a programming language used for the Ethereum blockchain platform.

3.1.4.a Web-Based Platform

General user types in this Project are divers and dive masters. They would use these functionalities that:

1. Verification of certifications and recording dive logs

2. Issuing new certifications digitally and storing objects
3. User management and access control.

As it is mentioned in literature review, classical Web 2.0 tools (HTML, CSS) and Web 3.0 technologies (React.js, Node.js, Express.js) are enough for developing web platform of this project. Access control issue is solved via HTTPS, user authentication protocols (OAuth or digital signatures) for now.

3.1.4.b Mobile Platform

With the same user profiles mentioned in 3.1.4.a Web-based Platform, it's just enough for determining which tools will be used. Flutter for cross-platform support, NFC and QR-Code technology for basic verifications. Then, the mobile app connects to the blockchain database via APIs to fetch and send real-time data.

3.1.4.c API Integrations & Security Layer

Third parties, such that government databases, weather services, and social media platforms, will be used for API integrations. The usage purposes are verifying/requesting certifications through government or official databases and integrating real-time weather and geolocation data. Data transfer formats are in JSON/XML formats.

For security and encryption layer, this application's system have to protecting user data and digital identities through encryption (SSL/TLS) and ensuring data integrity with digital signatures. It can be provided with using OpenZeppelin, it mentioned in detail in literatüre review.

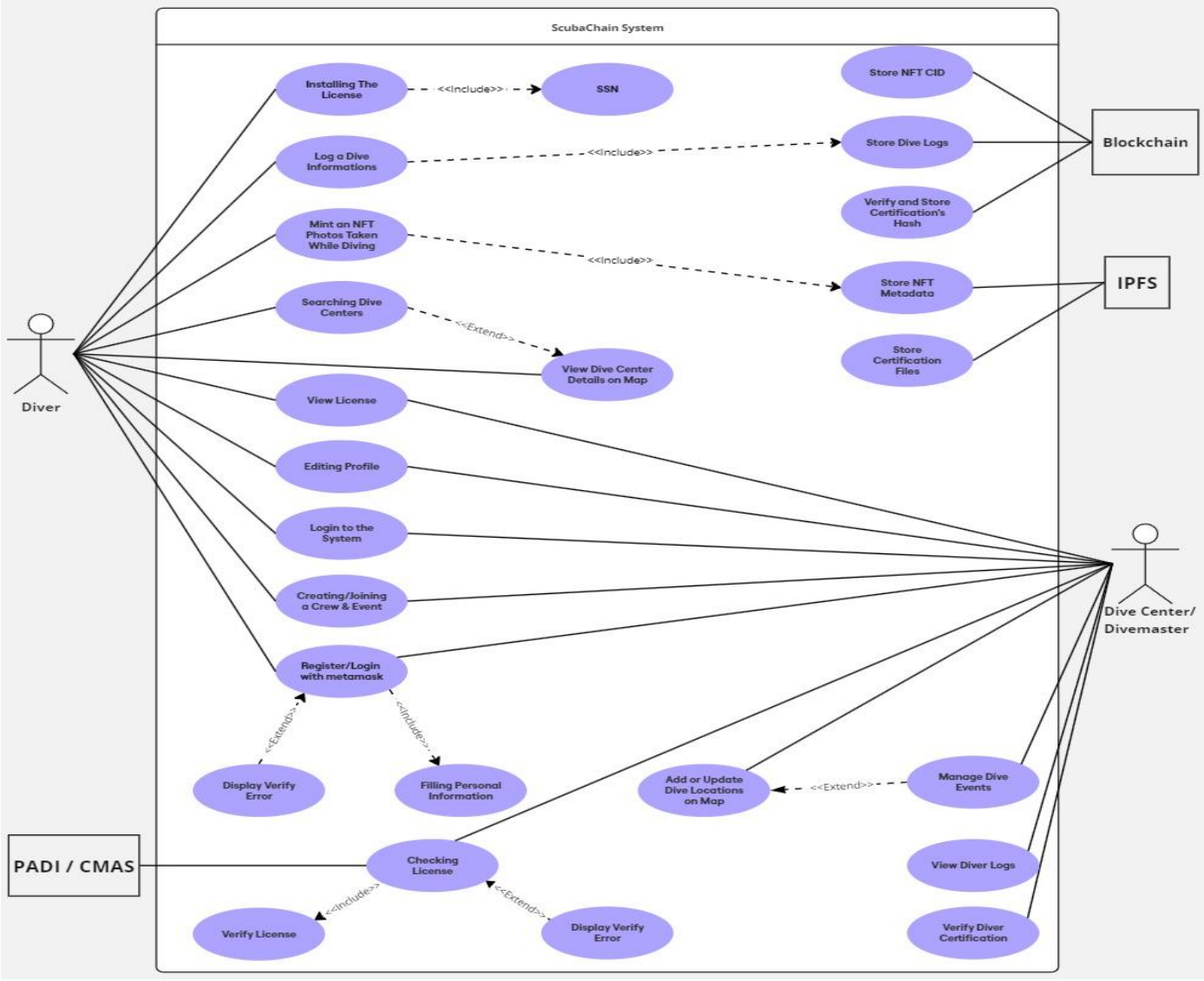
3.2. Functional Requirements

Use Cases:

- Register/Login with Metamask
- Installing the License
- Log a Dive Information Logout
- Mint on NFT Photos Token While Diving
- Searching Dive Centers
- View License
- Editing Profile

- Login to the System
- Creating/Joining a Crew & Event
- Verify Diver Certification
- View Diver Logs
- Manage Dive Events
- Add or Update Dive Locations on Map

Diagram:



1.Register/Login with Metamask

Use Case Number	Use Case 1
Use Case Name	Register/Login with Metamask
Actor	Diver
Description	The diver registers or logs into the system using their Metamask wallet.
Precondition	Diver has a Metamask wallet installed.
Scenario	1. Diver selects Metamask login. 2. Diver connects wallet and verifies identity. 3. System grants access.
Postcondition	Diver successfully registers or logs in via Metamask.
Exceptions	Wallet connection failure, authentication error.

2. Installing the License

Use Case Number	Use Case 2
Use Case Name	Installing the License
Actor	Diver
Description	The diver enters their SSN (Social Security Number) to upload the license to the system. The system verifies the license with PADI or CMAS. If successful, the license is confirmed, and its hash is stored on the Blockchain.
Precondition	Diver has a valid license.
Scenario	1. Diver enters SSN and license details. 2. The system verifies the license with PADI/CMAS. 3. License is validated and stored on Blockchain.
Postcondition	Diver's license is verified and added to the system.
Exceptions	Invalid SSN or license details, system verification failure.

3. Log a Dive Information

Use Case Number	Use Case 3
Use Case Name	Log a Dive Information
Actor	Diver
Description	The diver logs dive details (e.g., location, duration, depth) into the system. The data is stored securely on Blockchain.
Precondition	Diver is logged into the system.

Scenario	1. Diver enters dive details. 2. System saves the data to Blockchain.
Postcondition	Dive information is securely stored in the system.
Exceptions	Invalid data input, Blockchain connectivity issues.

4. Mint on NFT Photos Token While Diving

Use Case Number	Use Case 4
Use Case Name	Mint on NFT Photos Token While Diving
Actor	Diver
Description	The diver uploads photos taken during the dive. The system converts them to NFTs, stores their CID on Blockchain, and the image on IPFS.
Precondition	Diver is logged into the system with access to IPFS.
Scenario	1. Diver uploads photos. 2. System converts photos to NFT. 3. CID is stored on Blockchain and file on IPFS.
Postcondition	Photos are successfully minted as NFTs.
Exceptions	Photo upload failure, Blockchain or IPFS connectivity issues.

5.Searching Dive Centers

Use Case Number	Use Case 5
Use Case Name	Searching Dive Centers
Actor	Diver
Description	The diver searches for nearby or specific region dive centers on a map and views their details.
Precondition	System access with map integration.
Scenario	1. Diver searches for dive centers. 2. System displays relevant centers and their details.
Postcondition	Diver can access information about desired dive centers.
Exceptions	No results found, system or map integration issues.

6. View License

Use Case Number	Use Case 6
Use Case Name	View License
Actor	Diver

Description	The diver can view their uploaded license through a visual interface.
Precondition	License information is already uploaded and stored on Blockchain.
Scenario	1. Diver selects the option to view their license. 2. System retrieves and displays license details.
Postcondition	Diver successfully views their license details.
Exceptions	License not found, Blockchain retrieval failure.

7.Editing Profile

Use Case Number	Use Case 7
Use Case Name	Editing Profile
Actor	Diver
Description	The diver updates personal profile information such as name, surname, and contact details.
Precondition	Diver is logged into the system.
Scenario	1. Diver selects the option to edit their profile. 2. Diver updates profile details. 3. System saves the changes.
Postcondition	Profile information is successfully updated.
Exceptions	Invalid input, system update failure.

8.Login to the System

Use Case Number	Use Case 8
Use Case Name	Login to the System
Actor	Diver
Description	The diver logs into the system to access their personal account.
Precondition	Diver has valid login credentials.
Scenario	1. Diver enters username and password. 2. System verifies credentials. 3. Diver accesses their account.
Postcondition	Diver successfully logs into the system.
Exceptions	Incorrect credentials, system login failure.

9.Creating/Joining a Crew & Event

Use Case Number	Use Case 9
Use Case Name	Creating/Joining a Crew & Event

Actor	Diver
Description	The diver creates or joins a crew or a diving event through the system.
Precondition	Diver is logged into the system..
Scenario	1. Diver creates or searches for an event. 2. System displays event details. 3. Diver joins or creates an event.
Postcondition	Diver successfully joins or creates an event.
Exceptions	Event creation failure, joining restrictions.

10. Verify Diver Certification

Use Case Number	Use Case 10
Use Case Name	Verify Diver Certification
Actor	Dive Center/DiveMaster
Description	The dive center or instructor verifies the diver's certification through organizations like PADI or CMAS.
Precondition	Diver certification details are available.
Scenario	1. Dive center initiates certification verification. 2. System contacts PADI/CMAS for verification. 3. Certification is validated.
Postcondition	Diver's certification is successfully verified.
Exceptions	Verification failure, invalid certification.

11.View Diver Logs

Use Case Number	Use Case 11
Use Case Name	View Diver Logs
Actor	Dive Center/DiveMaster
Description	The dive center or instructor views the diver's logged dive records.
Precondition	Dive records are stored on Blockchain.
Scenario	1. Dive center requests diver logs. 2. System retrieves logs from Blockchain. 3. Logs are displayed.
Postcondition	Dive center successfully views the diver's logs.
Exceptions	Log retrieval failure, Blockchain connectivity issues.

12.Manage Dive Events

Use Case Number	Use Case 12

Use Case Name	Manage Dive Events
Actor	Dive Center/DiveMaster
Description	The dive center or instructor manages diving events, including creating, editing, or canceling events.
Precondition	Access to event management system.
Scenario	1. Dive center creates or edits event details. 2. System saves or updates event information. 3. Event is published or updated.
Postcondition	Diving events are successfully managed.
Exceptions	Event creation or update failure, system access issues.

13.Add or Update Dive Locations on Map

Use Case Number	Use Case 13
Use Case Name	Add or Update Dive Locations on Map
Actor	Dive Center/DiveMaster
Description	The dive center or instructor adds or updates dive locations on a map.
Precondition	Access to map and location management system.
Scenario	1. Dive center adds or updates a location. 2. System saves location details. 3. Map is updated with new location.
Postcondition	Dive locations are successfully updated on the map.
Exceptions	Location addition failure, map integration issues.

3.3 Performance Requirements

The performance requirements for ScubaChain focus on ensuring the system operates efficiently and reliably. The platform should provide low-latency responses for basic user functionalities but it would have one hour tolerance for certification verifications and dive log updates. Generally, it should handle increasing user numbers, limited requests' size per user and transaction volumes without degradation in overall performance. Also, with minimum corruption rate, the system should storage and save the all data with zero incidents of any unauthorized modification. The system should offer a smooth and fast user interface for both web and mobile applications.

About the issue, efficiency & total cost, the system should provide to optimize energy consumption and total fee, especially for operations involving blockchain consensus mechanisms and

blockchain transactions/storage operations. There is no expected desirement level for transaction throughput, latency, availability and fault tolerance for now, due to uncertainity of development stage.

3.4 Software System Attributes

3.4.1 Portability

The system’s mobile application should support both IOS and Android platforms and updated versions using a single codebase (like Flutter) and web platform should be compatible with major browsers (Chrome, Firefox, Safari, Edge etc.). Also, the blockchain integration must support different blockchain platforms (Ethereum) to adapt to user preferences or future technological shifts.

3.4.2 Scalability

For scalability of this project, off-chain storage solutions like IPFS for large data files (e.g., dive logs, certificates) to reduce on-chain congestion. Elastic cloud infrastructure (like Amazon Cloud Services) also would be used for the web platform to manage high user loads and user requests.

3.4.3 Adaptability

Possible requirements are constructing modular architecture using APIs and microservices to easily incorporate new features (e.g., additional certification bodies, blockchain elements like NFTs)and support for smart contract upgrades to address evolving certification standards or regulations.

3.4.4 Usability

The main issue in this case, determining which users can interact with the system. Intuitive user interfaces for both web and mobile platforms, designed with divers, dive centers, and instructors in mind. In future, also multilingual support can be added for the international diving community.

3.4.5 Performance

Although there is no expectation about performance desirement criteria, the general mission is receiving low-latency blockchain transactions, high throughput for certification and dive log updates and minimal downtime with a target of high and stable availability.

3.5. Safety Requirement

1. Blockchain Ensures Immutable Data Storage

Name	**Blockchain Ensures Immutable Data Storage**
Purpose/Description	Enhance security by ensuring data is stored immutably on the blockchain.
Inputs	User-provided data (e.g., license information, NFT data).
Process	Data is recorded on the blockchain and becomes immutable. All transactions are verified with digital signatures.
Output	Data is securely stored on the blockchain and can be tracked with public logs.

2. Secure Transactions with Encryption and Digital Signatures

Name	Secure Transactions with Encryption and Digital Signatures
Purpose/Description	Ensure secure data transmission and transaction verification using encryption and digital signatures.
Inputs	User's digital signature, transaction data.
Process	All data transfers are encrypted and digital signatures verified. Unauthorized access is prevented.
Output	Transactions are securely executed, and user data is protected.

3. Access Control Mechanisms and Unauthorized Access Prevention

Name	**Access Control Mechanisms and Unauthorized Access Prevention**
Purpose/Description	Provide access control mechanisms to protect user data and prevent unauthorized access.
Inputs	User session information (e.g., authentication tokens, user roles).

Process	Authorize logged-in users. Unauthorized access attempts are blocked, and incidents are logged.
Output	Only authorized users access data, and the system is protected from unauthorized entries.

4. Public Logs for All Blockchain Transactions

Name	**Public Logs for All Blockchain Transactions**
Purpose/Description	Ensure transparency by storing all transactions in publicly accessible logs.
Inputs	Blockchain transaction data (e.g., transaction ID, date, user address, transaction details).
Process	Each transaction is recorded as a log and made publicly accessible. These logs are stored on the blockchain.
Output	Transactions are transparently tracked and auditable.

Reliability

1. Data Accessibility Guaranteed via IPFS Pinning Services

Name	**Data Accessibility Guaranteed via IPFS Pinning Services**
Purpose/Description	Use pinning services to ensure continuous accessibility of data on IPFS.
Inputs	User-uploaded data on IPFS (e.g., files, hash values).
Process	Data remains continuously accessible on IPFS, ensuring uninterrupted user experience.
Output	Data is always accessible on IPFS, providing seamless user experience.

2. Monthly Backup and Recovery Mechanisms for System Failures

Name	**Monthly Backup and Recovery Mechanisms for System Failures**

Purpose/Description	Provide regular backup and recovery mechanisms to address system failures or data loss.
Inputs	Stored system data (e.g., user information, transaction records, blockchain, and IPFS data).
Process	Data is securely backed up monthly. In case of a system failure, the latest backup is used for data recovery.
Output	Risk of data loss is minimized, and the system is quickly operational post-failure.

Scalability

1. Scalable Infrastructure for Growing Users and Data

Name	**Scalable Infrastructure for Growing Users and Data**
Purpose/Description	Provide scalable infrastructure to accommodate increasing user and data demands.
Inputs	User sessions, transaction requests, database queries.
Process	The system dynamically allocates resources to respond to increased demand. Load balancing and high availability mechanisms are implemented.
Output	Performance is maintained, and uninterrupted service is provided during growth.

2. High-Volume Transactions Managed Without Performance Degradation

Name	**High-Volume Transactions Managed Without Performance Degradation**
Purpose/Description	Support infrastructure capable of handling high transaction volumes without performance loss.
Inputs	User transactions (API calls, database operations, blockchain requests).
Process	High-volume transactions are handled via parallel processing, optimization, and efficient resource management. Load balancing

	and queuing mechanisms are employed as needed.
Output	Transactions are processed smoothly, even under high loadconditions.conditions.

Performance

1. License Verification and IPFS Data Access Latency Below %d Seconds

Name	**License Verification and IPFS Data Access Latency Below %d Seconds**
Purpose/Description	Ensure quick completion of license verification and data access requests on IPFS.
Inputs	License verification requests, IPFS data access requests.
Process	The system uses high-speed connections and optimization techniques to maintain latency under %d seconds.
Output	Requests are processed with minimal delay, ensuring fast response times.

2. Interaction Response Times Under %d Seconds

Name	**Interaction Response Times Under %d Seconds**
Purpose/Description	Improve user experience by responding to interactions within %d seconds.
Inputs	User inputs (e.g., form submissions, button clicks, queries).
Process	The system optimizes for fast response times, utilizing high-speed processing infrastructure as needed.
Output	Users experience seamless interaction with prompt response times.

Usability

1. %d-Minute Learning Curve for Application Understanding

Name	**%d-Minute Learning Curve for Application Understanding**
Purpose/Description	Provide design and guides to help users quickly understand the application's functions and usage.
Inputs	User feedback, guide materials (e.g., videos, documentation, help texts).
Process	Intuitive user interface design is supported by educational materials and guides. Functions are simplified for user-focused experience.
Output	Users can learn the application’s core functions within %d minutes and use it effectively.

2. Accessibility Features for Users with Disabilities

Name	**Accessibility Features for Users with Disabilities**
Purpose/Description	Ensure accessibility for users with disabilities through tailored features.
Inputs	User interactions and accessibility needs (e.g., screen reader support, keyboard navigation).
Process	The application is designed in compliance with WCAG standards. Features include screen reader compatibility, high contrast mode, and alternative text support. Full compatibility with keyboards and assistive devices is ensured.
Output	Users with disabilities can use the application without difficulty, accessing all features.

Interoperability

1. Support for Data Exchange Protocols with Third-Party Applications and Devices

Name	**Support for Data Exchange Protocols with Third-Party Applications and Devices**
Purpose/Description	Ensure seamless data exchange between the application and third-party applications and devices by supporting standard protocols.
Inputs	Data exchange requests (e.g., RESTful API, JSON, XML).
Process	The system communicates with third-party systems using standard data exchange protocols. API integrations and protocol support are provided.
Output	Data is shared and received in a compatible manner with third-party applications and devices.

Maintainability

1. Easier Maintenance and Updates with Modular Architecture

Name	**Easier Maintenance and Updates with Modular Architecture**
Purpose/Description	Provide a modular architecture to simplify maintenance and updates of the application.
Inputs	Software updates, bug fixes, new feature requests.
Process	The system is developed with a modular structure, so changes made to a specific module do not affect the entire system. Updates can be applied independently.
Output	Updates are implemented quickly and effectively, minimizing system downtime during maintenance.

2. Well-Documented Codebase and Version Control

Name	**Well-Documented Codebase and Version Control**
Purpose/Description	Facilitate software development and maintenance processes by having a well-documented codebase and effective version control system.

Inputs	Code comments, technical documentation, version history.
Process	Code is written with explanatory comments, and comprehensive technical documentation is prepared. Version control systems (e.g., Git) are used to track code changes, and previous versions can be reverted if necessary.
Output	Developers and maintenance teams can easily understand the code and respond quickly. The software development process becomes more organized and efficient.

4. References

[1] https://ieeexplore.ieee.org/document/278253

[2] https://blockchain.ieee.org/standards

[3] https://miro.com