

SEAXCHANGE: A BLOCKCHAIN DRIVEN APP FOR TUNA SUPPLY CHAIN MANAGEMENT

A Special Problem

Presented to

5 the Faculty of the Division of Physical Sciences and Mathematics

6 College of Arts and Sciences

7 University of the Philippines Visayas

Miag-ao, Iloilo

In Partial Fulfillment

¹⁰ of the Requirements for the Degree of

11 Bachelor of Science in Computer Science by

12 BAT-OG, Jeff Rouzel

13 CAHILIG, Maxinne Gwen

14 GANIT, Zyrex Djewel

15 Francis DIMZON, Ph.D.

16 Adviser

May 23, 2025

18

Approval Sheet

19

The Division of Physical Sciences and Mathematics, College of Arts and
Sciences, University of the Philippines Visayas

20

certifies that this is the approved version of the following special problem:

21

**SEAXCHANGE: A BLOCKCHAIN DRIVEN APP FOR
TUNA SUPPLY CHAIN MANAGEMENT**

22

Approved by:

23

Name	Signature	Date
------	-----------	------

Francis D. Dimzon, Ph.D. _____

(Adviser) _____

Christi Florence C. Cala-or _____

(Panel Member) _____

Kent Christian A. Castor _____

(Division Chair) _____

26 Division of Physical Sciences and Mathematics
27 College of Arts and Sciences
28 University of the Philippines Visayas

29 **Declaration**

30 We, Jeff Rouzel Bat-og, Maxinne Gwen Cahilig, and Zyrex Djewel Ganit,
31 hereby certify that this Special Problem has been written by us and is the record
32 of work carried out by us. Any significant borrowings have been properly acknowl-
33 edged and referred.

Name	Signature	Date
Jeff Rouzel Bat-og (Student)	_____	_____
Maxinne Gwen Cahilig (Student)	_____	_____
Zyrex Djewel Ganit (Student)	_____	_____

Dedication

36 This Special Problem, SeaXChange: A Blockchain-Driven App for Tuna Sup-
37 ply Chain Management, is dedicated to those who were instrumental in its creation
38 and purpose:

39 To our esteemed adviser and mentor, **Francis D. Dimzon, Ph.D.**, for his
40 invaluable guidance, expertise, and unwavering support throughout our research
41 journey.

42 To the members of the **Miagao community**, particularly the fishermen, sup-
43 pliers, and retailers from **Barangay Sapa**, and the consumers from **Barangay**
44 **Mat-y**, your participation and insights were crucial in assessing product trace-
45 ability for our project.

46 To **Engr. Noel Lucero**, whose valuable perspectives during interviews pro-
47 vided essential context for understanding current trends and issues in the tuna
48 supply chain and the fishing industry as a whole.

49 To our fellow Special Problem classmates and colleagues—for the collabora-
50 tion, shared challenges, and mutual support that made this academic endeavor
51 more enriching.

52 And most importantly, to our beloved families—for their constant love, encour-
53 agement, understanding, and sacrifices that enabled us to pursue and complete
54 this work.

55 This Special Problem paper stands as a testament to your support and as a
56 contribution toward addressing the challenges faced within the tuna supply chain.

57

Acknowledgment

58 Our deepest appreciation goes to our thesis adviser, **Francis D. Dimzon**,
59 **Ph.D.**, for his expertise, patient guidance, and unwavering support. We are also
60 grateful to **Christi Florence C. Cala-or**, panelist, for her insightful feedback
61 and suggestions that enriched this study.

62 We thank the **University of the Philippines Visayas**, particularly the **Di-**
63 **sion of Physical Sciences and Mathematics**, College of Arts and Sciences,
64 for providing the academic foundation and resources. We also acknowledge the
65 **Computer Science Faculty** for equipping us with essential skills in software
66 engineering.

67 Special thanks are extended to the **Institute of Marine Fisheries and**
68 **Oceanology**. We particularly recognize **Ricardo P. Babaran Ph.D.**, , for his
69 valuable insights into the state of the tuna supply chain, existing technologies,
70 and the importance of supporting marginalized groups within it. We also thank
71 **Carmelo Del Castillo, Ph.D.**, for sharing related insights from prior thesis
72 work on the tuna supply chain.

73 To our fellow Special Problem classmates and colleagues—thank you for the
74 shared learning experiences, collaboration, and mutual support that made this
75 academic journey more fulfilling.

76 The successful implementation of SeaXChange was made possible through the
77 use of several tools and platforms. We acknowledge **Google Cloud Platform**
78 for hosting our backend services and blockchain network, and the **Hyperledger**
79 **Fabric** open-source framework, along with its documentation and community, for

80 providing a robust foundation for our blockchain system.

81 We extend our gratitude to the community participants and organizations in
82 **Miagao** who generously shared their time and perspectives. Our sincere thanks
83 go to the respondents—fisherfolk, suppliers, retailers, and consumers—whose real-
84 world experiences were vital to this study. We especially thank **Jerome F. Ca-**
85 **batuan** and **Veronica Jeruta** for facilitating access and participation within
86 their respective communities. We also acknowledge **Jagnee Fishing Corpora-**
87 **tion** and **Engr. Noel Lucero** for sharing their knowledge of the fish industry,
88 fishing vessels, and the tuna supply chain.

89 Finally, and most importantly, we extend our heartfelt thanks to our families
90 for their unwavering love, encouragement, sacrifices, and patience throughout the
91 duration of this study.

92 This thesis is a culmination of the support and collaboration of all those men-
93 tioned. We hope that this work contributes meaningfully to improving trans-
94 parency and efficiency in the tuna supply chain.

95

Abstract

96 The tuna supply chain faces critical challenges regarding traceability, transparency
97 and sustainability, particularly due to certain issues such as illegal, unreported and
98 unregulated (IUU) fishing. Ensuring the traceability within the tuna supply chain
99 is a critical role in enhancing consumer confidence, transparency and promoting
100 adherence to environmental and legal standards. This research explores the appli-
101 cation of blockchain technology as a solution to these given issues. By combining
102 qualitative insights gathered from different key stakeholders across the supply
103 chain, the researchers evaluated the potential of blockchain to improve product
104 traceability, accountability, and trust. The findings suggests that blockchain offers
105 a secure and transparent method of recording the journey of tuna products from
106 catch to market, helping to combat IUU fishing and promote sustainable practices.
107 However, successful implementation requires overcoming barriers related to tech-
108 nological integration, cost, and stakeholder collaboration. This study provides
109 valuable insights into the feasibility and impact of blockchain adoption within
110 other fish supply chains, contributing to the development of more transparent,
111 responsible, and sustainable tuna industries.

112 Suggested keywords based on ACM Computing Classification system can be found
113 at https://dl.acm.org/ccs/ccs_flat.cfm

114 **Keywords:** Blockchain, Traceability, Smart Contract, Supply Chain

¹¹⁵ **Contents**

¹¹⁶ 1 Introduction	¹
¹¹⁷ 1.1 Overview	¹
¹¹⁸ 1.2 Problem Statement	²
¹¹⁹ 1.3 Research Objectives	³
¹²⁰ 1.3.1 General Objective	³
¹²¹ 1.3.2 Specific Objectives	³
¹²² 1.4 Scope and Limitations of the Research	⁴
¹²³ 1.5 Significance of the Research	⁴
¹²⁴ 2 Review of Related Literature	⁷
¹²⁵ 2.1 State of Tuna Industry in the Philippines	⁸
¹²⁶ 2.2 Fishing Regulations in the Philippines	⁹

127	2.3 Tuna and Fish Supply Chain	11
128	2.4 Tuna Supply Chain Stages and Roles	12
129	2.5 Factors Affecting the Tuna Supply Chain	14
130	2.6 Technology of Blockchain	16
131	2.7 Opportunities of Blockchain Technology for Supply Chain Management	17
133	2.8 Supply Chain Model with Blockchain Technology of Fishing Industry in Indonesia	20
134		
135	2.9 Existing Technology Intended for Traceability and Supply Chain .	21
136	2.10 Developing a Traceability System for Tuna Supply Chains	24
137	2.11 Chapter Summary	25
138	2.11.1 Comparison Table of Related Studies	26
139	2.11.2 Research Gaps and Problem	26
140	2.11.3 Summary	27
141	3 Research Methodology	29
142	3.1 Software Development	29
143	3.2 Research Activities	30
144	3.2.1 Feedback Collection Method	31

CONTENTS xi

145	3.2.2 Data Gathering	31
146	3.2.3 Designing and Developing the System	32
147	3.2.4 Implementing Algorithms and Services	33
148	3.2.5 Modeling the System Architecture	36
149	4 Results and Discussions	41
150	4.1 Overview	41
151	4.2 Smart Contract Deployment and Installation	42
152	4.2.1 Hyperledger Fabric Prerequisites	42
153	4.2.2 Invoking the Blockchain System	44
154	4.3 Backend Security Analysis (Hyperledger Fabric on GCP)	47
155	4.3.1 System Architecture and Deployment Overview	47
156	4.3.2 Blockchain Network Security	48
157	4.3.3 Smart Contract Automated Test Result	49
158	4.3.4 GCP Infrastructure Security	53
159	4.3.5 Threat Model and Mitigations	55
160	4.4 Mockups	55
161	4.5 Operational Flow of the Web Application	58

162	4.5.1 Landing Page	58
163	4.5.2 Sign Up Page	61
164	4.5.3 Login Page	61
165	4.5.4 Homepage	62
166	4.5.5 Profile	64
167	4.5.6 Logout	64
168	4.6 System Discussion	65
169	4.7 User Demonstration and Feedback Results	66
170	4.7.1 Demo Setup and Scenario	66
171	4.7.2 Summarized Feedback	69
172	4.7.3 Results and Analysis	73
173	5 Conclusion	75
174	5.1 Overview	75
175	5.2 Conclusion	75
176	5.3 Recommendations	76
177	5.4 Summary	77
178	6 References	79

CONTENTS

xiii

179	A Code Snippets	87
180	A.1 Hyperledger Fabric Deployment Instructions	99
181	A.1.1 Environment Setup	99
182	A.1.2 Network Startup and Smart Contract Deployment	100
183	A.1.3 Testing the Smart Contract	102
184	A.1.4 Important Notes	103
185	B Resource Persons	105
186	C Interview Request Letter	107
187	D App Demo Documentation	109

¹⁸⁸ **List of Figures**

¹⁸⁹	3.1 Blockchain Architecture of SeaXChange	37
¹⁹⁰	3.2 Overall System Architecture of SeaXChange	38
¹⁹¹	3.3 Use case diagram for SeaXChange.	40
¹⁹²	4.1 Query Smart Contract: Check assets	44
¹⁹³	4.2 Invoke Smart Contract: Create/Add new tuna asset	45
¹⁹⁴	4.3 Query Smart Contract: Check assets with new tuna asset	45
¹⁹⁵	4.4 Invoke Smart Contract: Transfer asset to a supplier	46
¹⁹⁶	4.5 Query Smart Contract: Check asset after transfer	46
¹⁹⁷	4.6 Invoke Smart Contract: Transfer asset to a retailer	46
¹⁹⁸	4.7 Query Smart Contract: Check asset after transfer	47
¹⁹⁹	4.8 Query Smart Contract: Check updated assets	47
²⁰⁰	4.9 Initialization Confirmation of the Ledger	50

201	4.10 Initialization Confirmation of the Ledger	51
202	4.11 SeaXChange Mockups showing the Authentication Page, Role-Based	
203	Homepage, Asset Transfer Interfaces for Fishermen, Suppliers, and	
204	Retailers, and Asset Viewing page for the Consumers	58
205	4.12 Landing Page	59
206	4.13 Mobile View: Landing Page	60
207	4.14 SignUp Page	61
208	4.15 LogIn Page	61
209	4.16 Fisher Homepage	62
210	4.17 Supplier Homepage	63
211	4.18 Retailer Homepage	63
212	4.19 Consumer Homepage	64
213	4.20 View Profile	64
214	4.21 Log Out	65
215	4.22 Add Catch (Asset)	67
216	4.23 Save Details	67
217	4.24 Send Asset	68
218	C.1 Scanned Interview Request Letter	108

LIST OF FIGURES

xvii

219	D.1 Respondents (Fishermen, supplier, and retailer) from Barangay	
220	Sapa, Miagao	109
221	D.2 Respondents (Retailer, and consumer) from Barangay Mat-Y, Miagao	110

List of Tables

223	2.1 Comparison of Studies on Technology for Traceability and Supply Chain Management	26
225	4.1 Potential Threat and Mitigation	55
226	4.2 Mean ratings and descriptions for functionality-related features per stakeholder group.	69
228	4.3 Mean ratings and descriptions for end-user needs-related features per stakeholder group.	70
230	4.4 Mean ratings and descriptions for performance-related features per stakeholder group.	70
232	4.5 Mean ratings and descriptions for usability-related features per stakeholder group.	71
234	4.6 Mean ratings and descriptions for ease of use-related features per stakeholder group.	72

236	4.7 Mean ratings and descriptions for feasibility-related features per	
237	stakeholder group.	72

²³⁸ Chapter 1

²³⁹ Introduction

²⁴⁰ 1.1 Overview

²⁴¹ The tuna supply chain faces critical issues that affect both the industry and its
²⁴² consumers. Illegal fishing, overfishing, and poor traceability threaten the sustain-
²⁴³ ability and ethical trade (Ambarsari et al., 2023; Nurhayati & Purnomo, 2021).
²⁴⁴ A lack of transparency compromises product quality and misleads consumers.
²⁴⁵ Blockchain technology can address these challenges by tracking the tuna's jour-
²⁴⁶ ney from ocean to plate through a secure, tamper-proof ledger (Zheng et al., 2017).
²⁴⁷ This system also holds stakeholders to comply to legal standards.

²⁴⁸ Although blockchain integration in the tuna sector is underway, gaps remain in its
²⁴⁹ implementation. This study helps to address the industry's need for transparent
²⁵⁰ and secure tracking of tuna products from ocean to consumer, while assessing the
²⁵¹ feasibility of implementing blockchain at scale in the seafood sector.

252 1.2 Problem Statement

253 Fish is one of the most consumed protein products in the Philippines. Before the
254 COVID-19 pandemic, fish was the most consumed animal protein in the Philip-
255 pines with annual per capita consumption of 40 kg (Agriculture and Agri-Food
256 Canada, 2022). Among the wide variety of fish, tuna stands out as a particularly
257 significant commodity due to its high demand and economic value (Kong et. al.,
258 2023). Although the nation is gifted with an abundance of aquatic resources, the
259 methods of dispersal of the product usually leads to inefficiency in terms of sales,
260 pricing, and overall product quality for the consumers. The complexity arises with
261 the introduction of a supply chain from several middlemen between the fish farmer
262 or producer and wholesaler in the coastal and aquatic regions towards the con-
263 sumers with less access to fresh fish. Consequently, both consumers and suppliers
264 struggle to ensure transparent and fair pricing, accurately track tuna products,
265 and maintain consistent quality (Adhikary et al., 2018; Stocco et al., 2022; Hoque
266 et al., 2021). These challenges stem from inefficient methods of product dispersal
267 and the presence of multiple intermediaries between fisherfolks in coastal munici-
268 palities and consumers. This fragmented supply chain introduces delays, obscures
269 product origin, and opens opportunities for mishandling or mislabeling. To ad-
270 dress this, the study focuses on the absence of a reliable, real-time traceability
271 system that allows stakeholders to verify the origin, novelty, handling processes,
272 and adherence to ethical sourcing practices of tuna at every stage. Without such
273 a system, the tuna supply chain and industry lack the transparency and account-
274 ability necessary to build consumer trust and improve supply chain efficiency. By
275 introducing a blockchain-driven solution, this study aims to streamline the trace-
276 ability process and reduce the negative impact of intermediary-heavy distribution.

²⁷⁷ **1.3 Research Objectives**

²⁷⁸ **1.3.1 General Objective**

²⁷⁹ The general objective of the study is to design and develop a blockchain-driven
²⁸⁰ application that improves the traceability of the tuna supply chain. Given the
²⁸¹ timely issues regarding illegal, unreported and unregulated (IUU) fishing and lack
²⁸² of product traceability and transparency, this study addresses these challenges
²⁸³ through an application that has the capability to provide an immutable ledger
²⁸⁴ and tamper-proof records. The result of this study serves as a framework for
²⁸⁵ integrating blockchain technology in the fish supply chain, specifically for tuna.
²⁸⁶ This also supports future researchers and developers facing similar challenges in
²⁸⁷ the industry.

²⁸⁸ **1.3.2 Specific Objectives**

²⁸⁹ To further specify the research objectives, the study focuses to:

- ²⁹⁰ 1. develop a smart contract framework using blockchain technology for data
²⁹¹ verification and transaction recording, ensuring secure and tamper-proof
²⁹² data for the stakeholders,
- ²⁹³ 2. design and develop a blockchain-driven application with a user-friendly in-
²⁹⁴ terface that allows stakeholders to access and input data while enhancing
²⁹⁵ traceability in the tuna supply chain through a tuna asset record for the
²⁹⁶ supply chain participants, and

297 3. deploy the application after completing all necessary preparations for soft-
298 ware development and to evaluate its overall results and effectiveness in en-
299 hancing the tuna supply chain, as discussed in the fourth and fifth chapters
300 of the study.

301 **1.4 Scope and Limitations of the Research**

302 The scope of this study focuses on how blockchain technology can be applied to
303 enhance traceability and transparency within the tuna supply chain. It involves
304 features such as smart contracts for recording the transactions and user interface
305 for stakeholders. The study also focuses exclusively on whole, small-sized caught
306 tuna products in the supply chain, excluding processed forms such as canned
307 or packaged tuna. The research examines the traceability of whole tuna from
308 capture to market sale, specifically centering on a supplier based in Miagao and
309 San Joaquin, Iloilo.

310 This study focuses solely on the supply chain within the specified municipalities
311 of Iloilo, so findings may not fully represent global practices. As it concentrates
312 on blockchain's role in traceability, other potential functions of the technology fall
313 outside of the scope of this research.

314 **1.5 Significance of the Research**

315 This study serves a significant purpose for several stakeholders in the tuna supply
316 chain. This study aims to solve the problems related to the management of tuna

³¹⁷ supply chain, particularly with regards to product traceability.

³¹⁸ • The Stakeholders

³¹⁹ – This study enhances transparency and accountability, allowing stakeholders such as fishers, suppliers, and retailers to access tamper-proof
³²⁰ and accurate information, thereby promoting a more ethical and authentic supply chain. By providing a digital record of the product's
³²¹ history, this study helps ensure compliance with environmental and
³²² legal standards. In cases of anomalies such as oil spills, red tide occurrences,
³²³ and illegal fishing activities, stakeholders can be involved in identifying and addressing these issues, fostering a collaborative approach to sustainability. Similarly, the record of a tuna asset can be
³²⁴ utilized for accountability purposes when problems such as damaged
³²⁵ products, mislabeling, or contamination arise, allowing stakeholders to
³²⁶ trace and resolve them efficiently.

³³¹ • The Consumers

³³² – This study helps consumers verify the history of the tuna product from its origin up until its journey to the consumers, therefore increasing
³³³ trust and transparency. By promoting traceability, it offers a more detailed and verifiable record of the supply chain, enabling consumers to
³³⁴ assess sustainability practices and identify stakeholders responsible for
³³⁵ any potential issues affecting the tuna product. This, in turn, encourages critical evaluation of the tuna supply chain, driving improvements
³³⁶ in accountability, resource management, and ethical sourcing.

340 • For Future Researchers

341 – As blockchain technology continues to grow, this study contributes to
342 the application of blockchain in the supply chain management and the
343 insights regarding its benefits and limitations. This research can be
344 helpful in the growing knowledge on digital solutions for traceability
345 and transparency for future research.

346 • The Policy Makers

347 – This study provides policy makers a reliable and data-driven founda-
348 tion in monitoring and regulating the tuna supply chain. By lever-
349 aging tamper-proof and transparent records, policymakers can more
350 effectively enforce compliance with fishing quotas and environmental
351 protections. This research also aids in lessening the illegal, unreported
352 and unregulated (IUU) fishing practices, contributing to the national
353 sustainability goals.

³⁵⁴ Chapter 2

³⁵⁵ Review of Related Literature

³⁵⁶ This chapter reviews existing literature related to the traceability of the tuna sup-
³⁵⁷ ply chain and the potential application of blockchain technology. In purchasing
³⁵⁸ goods, one important consideration is the product quality. An important factor
³⁵⁹ of determining the quality is to know the traceability of the supply chain. Trace-
³⁶⁰ ability refers to the ability of tracking the journey of the product from its source
³⁶¹ until its destination. The term “traceability” is now more utilized in both the
³⁶² food and production industry (Islam & Cullen, 2021). In the context of the tuna
³⁶³ supply chain, it can be used not only to promote transparency to consumers but to
³⁶⁴ also ensure compliance with environmental and legal standards. With blockchain
³⁶⁵ technology, the status of tuna at each stage could be recorded in the blockchain
³⁶⁶ which could be used for traceability. This paper aims to address the following
³⁶⁷ research question: How can blockchain technology improve the traceability of the
³⁶⁸ tuna supply chain management? To explore this, the chapter reviews literature
³⁶⁹ on the state of the tuna industry in the Philippines, fishing regulations, and the
³⁷⁰ structure of the tuna supply chain, including its stages and the roles of key actors.
³⁷¹ It also examines factors that affect the efficiency and transparency of the supply

372 chain. The discussion then turns to blockchain technology and its potential ap-
373 plications in supply chain management, with examples from the Philippines and
374 Indonesia. Finally, the chapter reviews existing traceability technologies and sys-
375 tems and concludes with key insights that inform the development of this study’s
376 proposed solution.

377 In purchasing goods, one thing to consider is the quality of it. An important part
378 of determining the quality is to know the traceability of the supply chain. Trace-
379 ability refers to the ability of tracking the journey of the product from its source
380 until its destination. The term “traceability” is now more utilized in both the
381 food and production industry (Islam & Cullen, 2021). In the context of the tuna
382 supply chain, it can be used not only to promote transparency to consumers but to
383 also ensure compliance with environmental and legal standards. With blockchain
384 technology, the status of tuna at each stage could be recorded in the blockchain
385 which could be used for traceability. This paper aims to address the following
386 research question: *How can blockchain technology improve the traceability of the*
387 *tuna supply chain management?*

388 2.1 State of Tuna Industry in the Philippines

389 In 2014, the Philippines became the top global producer of tuna according to
390 Llanto, et. al. (2018). The tuna is caught in domestic and international fishing
391 grounds near the country through various fishing methods such as purse seines,
392 gill nets, handline (hook and line) and ring net. Among the tuna species, the
393 skipjack tuna accounted for the largest portion of the catch by 40%. The study of

394 PCMARD (1993 as cited in Nepomuceno et al., 2020) stated that skipjack tuna
395 are often caught out in open waters or in offshore areas. In addition, Nepomuceno
396 et al. (2020) mentioned in their study that the dominant production of skipjack
397 tuna, together with yellowfin tuna, was recorded in South Cotabato. The tuna
398 supply has declined since 2000 due to various factors including overfishing, climate
399 change, and the laws and regulations imposed by different governing bodies for
400 the tuna fishing ground such as the Regulation No. 56, released by the Indone-
401 sian Maritime Affairs and Fisheries Ministry in November 2014. The regulation
402 imposed a moratorium on issuance of fishing licenses from 3 November 2014 to 30
403 April 2015 to eliminate illegal, unreported, and unregulated fishing in Indonesian
404 waters near Mindanao where tuna are known to thrive (Llanto et al, 2018). The
405 regulation imposed for the protection of tuna fishing grounds in the western and
406 central pacific ocean also lead to the decline of local tuna production which re-
407 quires the fishing operators of the Philippines to invest in the manufacturing and
408 processing of fish particularly tuna in Indonesia which includes hiring Indonesian
409 crew to be deployed in the Philippine fishing vessels (Llanto et al, 2018).

410 2.2 Fishing Regulations in the Philippines

411 A study of Asche et al. (2018) divided the fishing management strategies that
412 include right-based fishery management like territorial use of rights, access rights
413 and harvest rights. It discussed that a rights-based system could support the sus-
414 tainability of global fisheries by taking in account the three pillars of sustainability
415 (economic development, social development, and environmental protection) rather
416 than focusing on their trade-offs. A restriction on the fisherman's behavior by har-

vest rights and catch shares could be a profit problem for them in the short-run but in the long-run, this could help both in the fish stock and the fishermen's profit. Lack of restriction could lead to overfishing. Access rights limit the entry to fishery through permits which can also reduce the effect of high harvest levels. A sustainable fishing management system in the Philippines is important in order to preserve marine resources. To preserve these resources and protect the livelihood of local communities, various fishing management strategies should be implemented. A collaboration between the fishermen, local government and other stakeholders often happens to manage marine resources (Pomeroy & Courtney, 2018). The study of Pomeroy and Courtney discussed that marine tenure refers to the rights and responsibilities in terms of who can access the marine and coastal resources. The 1998 Fisheries Code paved the way for local government units (LGUs) to be involved in the management of municipal waters. LGUs are given the responsibility to overlook and regulate fisheries and establish marine tenure rights for fishers within 15 km from shore and these rights are applicable for municipal fishers and their respective organizations that are listed in the registry (Pomeroy & Courtney, 2018). In this way, it resolved problems in terms of fishing rights between small-scale and commercial fishing.

According to the study conducted by Mullon et al. (2017), the five major species of tuna: yellowfin *Thunnus albacares*, bigeye *Thunnus obesus*, bluefin *Thunnus thynnus* or *Thunus orientalis*, albacore *Thunnus alalunga*, and skipjack *Katsuwonus pelamis* are harvested to meet the global supply chain demand which causes those group of tuna fishes to be heavily exploited and threatened. The study conducted by Paillin et al. (2022) states that there are multiple risk agents in the supply chain assessment of tuna, these include the lack of standard environmental man-

agement system, lack of maintenance management, and lack of quality control from suppliers. The usage of efficient boats and good quality catching technology can also lead to fisheries depletion which causes various agency such as BFAR (Bureau of Fisheries and Aquatic Resources), the local government units, and the Philippine Coast Guard to enable policies for upholding closed fishing season to restrict large scale fishing vessel to minimize the fishing activities in the identified areas (Macusi et al, 2023). The implementation of closed fishing season caused delay or lack of fish supply, which led to higher fish prices. The growing demands and depleting population of tuna fishes coupled with the rapid increase in fuel costs can have a negative impact on the future of the supply chain in tuna fisheries (Mullon et al., 2017). With factors concerning the slow decline of tuna catches in the Philippines and surrounding nations, the future of the global supply chain of tuna must be addressed.

2.3 Tuna and Fish Supply Chain

According to Macusi et al (2023), the implementation of traceability programs in the agricultural product commodities and value chain in the Philippines is slower than its competing nation for tuna production. The Philippines has been steadily responding to the market innovation and integration of cost-effective and smart technologies for the traceability of various commodities. Accurate catch data is crucial for determining the attributes of the fish health, size, volumes, and maturity (Grantham et al, 2022) which can be used as a basis for the transparency of the traceability of the fish product. Illegal, unreported, and unregulated (IUU) is another concern for the fish industry. In the 2000s, the persistent IUU became

465 a global crisis affecting the biological, ecological, and socio-economics factors re-
466 volving around marine livelihood in Southeast Asia (Malinee et al, 2020). IUU
467 fishing is known to cause short- and long-term problems in the socio-economic
468 opportunities which affects food security and results in the possible collapse of
469 the fish industry and stocks due to overfishing (Malinee et al, 2020).

470 The establishment of marine protected areas in the Davao Gulf (MPAs) affected
471 the management of small-scale fisheries due to the growing population and de-
472 mands for seafood products. The closure of a wide range of fishing areas hosting
473 diverse and marine organisms has affected the socio-economics and livelihood of
474 the local and small-scale fishermen (Macusi et al, 2023), this in turn resulted
475 in IUU fishing. To ensure that fish stocks in the gulf are sustainably managed,
476 the implementation of GPS for tracking the movement and activities of fishers
477 through logbook and habitat monitoring can provide data and insights for track-
478 ing, monitoring, and understanding the condition of the marine resources (Obura
479 et al, 2019; Macusi et al, 2023).

480 **2.4 Tuna Supply Chain Stages and Roles**

481 The study conducted by Delfino (2023) highlights the roles of different actors in-
482 volved in the supply, production, distribution, and marketing of skipjack tuna in
483 Lagonoy Gulf in the Philippines. The study showcased a total of eleven intercon-
484 nected value chains but are generalized into four major stages or roles - fishers,
485 wholesalers, retailers, and processors. The fishers are the initial players responsible
486 for catching fish using boats or fishing vessels equipped with purse seines, gillnets,

487 and handlines(hook and line). Wholesalers are the actors for selling freshly caught
488 fish locally and regionally, they receive the fish supply directly from the fishers.
489 The next stage after wholesalers are the retailers, these intermediaries sell the fish
490 product to local markets, house-to-house (*libod*" in Visayan languages), and other
491 local medium such as *talipapa* or fish stands. Another intermediary is the proces-
492 sors, they convert fresh skipjack tuna into products like smoked tuna. The given
493 stages also overlapped in some cases as there are fisher-wholesalers who catch and
494 sell the fishes directly to retailers and there are also retailer-processors that both
495 sell whole and processed products. Despite having a firm system to transport fish
496 from sea to table, all the actors face problems during seasonal challenges involving
497 the availability of the tuna product. The fishers also need to consider strict local
498 regulations such as RA 10654 and RA 8550. The strict implementation of RA
499 10654 and RA 8850 at the local level or the Fisheries Code of the Local Philip-
500 pines aims to curb the problem encountered during season of deficit tuna supply
501 by limiting fishing activities and implementation of 15-km boundary lines in the
502 municipal waters of each municipality (Delfino, 2023). The study suggests that
503 improving conditions for value chain actors, particularly through support services
504 and government involvement could lead to a stable and sustainable exchange of
505 skipjack tuna and other seafood products from sea to table.

506 A study of Digal et al. (2017) discussed one of the value chains which was the
507 purse seine or the skipjack tuna value chain in the Philippines. Purse seining
508 is the method of catching a school of fish wherein it uses a large net around it,
509 trapping them and pulling the bottom of the net like a purse-like (Digal et al.,
510 2017) . This type of catching often targets dense fishes like tuna. Skipjack fishes
511 that weigh 300 grams and above are often sold to canneries, while the smaller ones

512 are sold at local markets, often used for consumption by Filipinos. Purse seiners
513 are usually employees of a fishing company and they have a fixed salary. They
514 could reach international waters so they need to bring their passports with them.
515 Jamboleros, who act as distributors, often buy from different fishing companies
516 per *banyera* or tub. They will then pack the fish and sell it to traders/truckers
517 who go to General Santos fishport. These traders will deliver it to the retailers
518 across Kidapawan who contacted them. There is no formal contract between the
519 jambolero and traders/truckers. One of the issues of the retailer is for everyday
520 that a fish is not sold, they would have a ₱10.00 less per kilogram.

521 2.5 Factors Affecting the Tuna Supply Chain

522 The tuna supply chain faced several factors and challenges for the safety and
523 quality of the product (Mercogliano & Santonicola, 2019). Without the proper
524 handling of the tuna after catching it can lead to various food-borne diseases and
525 outbreaks. The most frequent and mitigated food-borne causing compound is
526 Histamine(HIS) which causes Scombrotoxin fish poisoning (SPF) outbreaks re-
527 lated to food allergies when consumed (EFSA, 2017; Peruzy et al., 2017). Tuna
528 species are known for having high presence of amino acid histidine concentrations
529 which are converted to HIS by bacterial enzyme histidine-decarboxylase or HDS
530 (Aponte et al., 2018; Verkhivker & Altman, 2018). To combat the risk of SPF
531 and other food-borne diseases caused by tuna consumption, several safety hazards
532 and protocols were imposed to the tuna supply chain management. The term cold
533 chain refers to the storing of fish in temperatures less than 5°C after it was caught
534 (Yang & Lin, 2017). According to the article published by Mercogliano and San-

tonicola (2019), implementing a cold chain from the time the fish is caught until it is consumed is crucial for mitigating the outbreak of HIS poisoning. Additionally, the article also states that using high-quality raw tuna, cold chain maintenance, pre-cooking, and cooking can also reduce HIS development.

According to the study conducted by Pacoma and Yap-Dejeto entitled “Health Risk Assessment: Total Mercury in Canned Tuna and in Yellowfin and Frigate Tuna Caught from Leyte Gulf and Philippine Sea”, examines mercury contamination in both canned and locally caught tuna in the Philippines. Methylmercury, a potent neurotoxin, presents risks especially to vulnerable groups like pregnant women and children. The study reveals that canned tuna generally has higher mercury levels (0.07 µg/g) than locally caught tuna (0.002–0.024 µg/g). Based on the Food and Agriculture Organization’s fish consumption data, the mercury intake from locally caught tuna is within the World Health Organization’s safe limits, whereas canned tuna may exceed these limits for some groups. This highlights the need for monitoring of the mercury levels in the Philippine tuna supply chain, as tuna is a dietary staple and economic asset in the region, to mitigate health risks from chronic exposure.

Risk management is important for tuna supply chains to analyze the root of the risk and to assess the probability of such cases through the information taken from the different locations or sorting states where the tuna product is handled before being purchased by the consumer or end-use state (Parenreng et al, 2016).

556 2.6 Technology of Blockchain

557 According to Zheng et al. (2017), the idea of blockchain was first introduced in
558 2008 and was first implemented in the Bitcoin business which deals with cryp-
559 tocurrencies. This kind of technology tracks transactions and stores it in a list of
560 blocks. According to Sarmah (2018) , it acts as a database of transactions which
561 is overseen and verified by distributed nodes. Blockchain works by linking blocks
562 (where data is stored). When a transaction is initiated, it is then broadcasted to
563 a network of computers that verifies the transaction and if the verification is suc-
564 cessful, it will then be grouped and linked with the previous transactions that will
565 be added to the blockchain (Zheng et al., 2017). It does not require a middleman
566 because it operates on a peer-to-peer transaction. This eliminates the traditional
567 way of a central authority like the bank (Sarmah, 2018).

568 Automated process of transactions is also one of the salient features of blockchain
569 which is executed automatically based on predefined rules involving no third par-
570 ties. These predefined rules are conditions that need to be met in order for the
571 transaction to proceed. Given this, blockchain is described to be a “trust-free
572 technology” since it reduces the number of trusted individuals instead they trust
573 the machine itself which is difficult to manipulate given its cryptographic security
574 (Ali et al., 2023). Blockchain also ensures immutability with its data. Once the
575 data has been added to the blockchain, it is difficult to change since each block has
576 a cryptographic link to other blocks, which guarantees that the data is tamper-
577 proof and permanent (Ali et al., 2023). This also brings back to the essence
578 of blockchain being trustworthy. Immutability and tamper-proof enhances data
579 management. It also helps prevent fraudulent activities especially with finances.

2.7. OPPORTUNITIES OF BLOCKCHAIN TECHNOLOGY FOR SUPPLY CHAIN MANAGEMENT

580 Transparency with transactions is one of the features of blockchain which makes
581 the chances of data being tampered less because it is accessible to anyone on the
582 network (Ali et al., 2023).

583 According to Nasurudeen Ahamed et al. (2020), Blockchain technologies are clas-
584 sified into 3 types: Public Blockchain, Private Blockchain,. Consortium Blockchain.
585 In Public Blockchain, all public peers can join together and have equal rights (for
586 example, read, write, and execute) on the public node. In Private Blockchain,
587 only Authorized Private Peers have access to the network. The access to the node
588 in the private peer is limited to the specific node. In Consortium Blockchain,
589 only the authorized team can access and join this blockchain, and all operations
590 in the node must adhere to the access. Their paper, Sea Food Supply Chain
591 Management Using Blockchain, gave the idea that with the purpose of creating a
592 blockchain-driven application, a public blockchain approach could be appropriate
593 for handling consumer-based information as users can verify non-sensitive data
594 like prices, freshness and availability. While handling sensitive information such
595 as internal works and logistics, a consortium blockchain where authorized users
596 such as fish owners, distributors, manufacturers, etc. can handle the core supply
597 chain operations, like tracking the movement of tuna from catch to market.

598 **2.7 Opportunities of Blockchain Technology for** 599 **Supply Chain Management**

600 Supply chain is the term used for understanding the business activities for design-
601 ing, developing, delivering, purchasing, and using a product or service (Hugos,

602 2024). Companies and various industries are heavily relying on supply chains to
603 achieve their business objectives. The purpose of supply chain began to be more
604 significant in the last century as firms discovered that supply chain can be used for
605 competitive advantage instead of just a cost driver as believed in the bygone days
606 (Snyder & Shen, 2019). Following the supply chain paradigm can demonstrate the
607 delivery of a product or service while strongly emphasizing the customer's speci-
608 fications. With the increasing studies conducted and published for supply chain,
609 many companies adopted this practice for the benefit of their longevity, as such
610 the term supply chain management has come into place. The Council of Supply
611 Management Professionals or CSCMP (2024) defines supply chain management
612 as the planning and management of all activities involved in sourcing and pro-
613 curement, conversion, and all logistics management activities; essentially, supply
614 chain management integrates supply and demand management within and across
615 the company. Supply chain management is also involved with the relationship
616 with collaborators and channel partners such as suppliers, intermediaries, third
617 party providers, and customers (CSCMP, 2024) .

618 In the article of Cordova et. al (Cordova et. al, 2021), the role of supply chain man-
619 agement and the growing opportunities for blockchain technology in supply chain
620 management was discussed. According to Cordova (2021), the recent innovation
621 and globalization has given rise to the idea of using a data innovation framework
622 for supply chain management. Technologies such as blockchains and enterprise
623 resource planning (ERP) are among the highly contested platforms for supply
624 chain management to operate in a seamless interaction and distribution with the
625 product while heavily relying on modern technology and innovations. The logistic
626 business of the supply chain market is wide and complex, the distribution and

2.7. OPPORTUNITIES OF BLOCKCHAIN TECHNOLOGY FOR SUPPLY CHAIN MANAGEMENT

flow of products is not a simple job, and it heavily relies on paperwork (Georgiou, 2019; Cordova et. al, 2021). The usage of paperwork for logistic business can be at higher risk for lack of transparency, complex or unreliable tracking, deficiency of information, and possible dispute due to the tendency of paper to disappear or tear down, this can in turn delay the process and delivery of the item/product. With the issues encountered in the supply chain market, businesses, people and enterprises are eyeing toward the application of blockchain technology on supply chain management (Cordova et. al, 2021).

Implementing blockchain innovation in ERP systems and companies that use digital platforms can provide opportunities and contribute greatly for business processes (EOS Costa Rica, 2019 as cited by Cordova et al, 2021). The ability of blockchain technology to append new transactions to an existing block containing data can be thought of as a decentralized ledger (Cole et al, 2019). The method of blockchain to behave like a decentralized ledger can serve as a single unified source of data which in turns create a clear and consistent audit trail involved in the manufacturing, assembly, supply, and maintenance processes. According to Cole et al (2019), blockchains provide data to the movement and relation of products from its origin, inventory, shipment, and purchase. One potential of blockchain for supply chain management (Hackius & Petersen, 2017) is the ease of paperwork processing, specifically in ocean freight. When IBM and Maersk settled for a permissioned blockchain solution, they were able to connect a global network of shippers, carriers, ports, and customs. Another potential of blockchain in SCM is to identify counterfeit products. In the pharmaceutical industry and healthcare setting, blockchain could improve patient safety and hazard through establishing supply chain transparency from manufacturers through wholesale and pharma-

652 cies to the individual patients (Hackius & Petersen, 2017). Using blockchain can
653 make it harder to tamper or alter the products chain of origin with illegal and
654 counterfeit products. Blockchain have the potential to facilitate origin tracking.
655 According to Cordova et al (2021) , blockchain allows organizations to input rel-
656 evant data inside a chain which would have constant updates and tracking, this
657 supports visibility and traceability of the origin of the product. Smart contracts,
658 an executable code and a feature of blockchain, serves as a computer protocol
659 made between participants to digitally facilitate, execute, verify, and enforce an
660 agreement or terms of contract which is then stored in the blockchain (Khan et
661 al, 2021).

662 **2.8 Supply Chain Model with Blockchain Tech- 663 nology of Fishing Industry in Indonesia**

664 Larissa and Parung (2021) who explored the application of blockchain and de-
665 signed a supply chain model based on it, specifically for the Indonesian fishing
666 industry, aimed to mitigate the challenges in the fishery industry such as product
667 quality (perishability), long shipping times ,and data manipulation. The model
668 they developed of using QR codes for each player in the supply chain then tracking
669 it by scanning the QR code, could inspire the approach of this study in building
670 a blockchain-driven application for the tuna supply chain in the Philippines.

671 **2.9 Existing Technology Intended for Traceabil-**

672 **ity and Supply Chain**

673 A study of Shamsuzzoha et al. (2023) discussed the feasibility of implementing
674 a blockchain driven application called ‘Tracey’ for monitoring the fish traceabil-
675 ity in supply chain management. The study utilized the theoretical framework
676 developed by Islam & Cullen (2021) for improving the understanding and effec-
677 tiveness of implementing a food traceability system. The framework consists of
678 four principles as a basis for the supply chain management: identification, data
679 recording, data integration, and accessibility (Islam & Cullen, 2021). The Tracey
680 application utilized a public-private hybrid blockchain-based conceptual frame-
681 work by Mantravadi and Srai (2023) to uphold the transparency, traceability, and
682 certification of the sea food produce, specifically shrimp. The prototype being
683 studied by Shamsuzzoha et al. (2023) called Tracey focuses on the mobile-based
684 solution approach, the study found that the most widely used smartphone type
685 in the Philippines is the android phone which is where the Tracey prototype is
686 intended to be used. The Tracey app allows fishermen to log their catch details
687 and buyers to verify and update transaction history (Shamsuzzoha et al., 2023).
688 The Tracey app uses a central database for storing fish trading data and a decen-
689 tralized ledger or blockchain for traceability purposes. The decentralized ledger
690 acts as a tamper-proof copy of the data recorded by fishermen and buyers. The
691 result of the study finds that fishermen are open to using digital methods for
692 payments and confidentiality which is required for exporting the fish product to
693 maintain high standards for traceability, catch certification, and product quality.
694 The usage of blockchain as exemplified by the Tracey project can be used for

695 upholding the restriction for IUU due to its ability to ensure transparent trade,
696 consistent records, and accessibility. The result and discussion of the study of
697 Shamsuzzoha et al. (2023) provides a solution for improving the sustainability of
698 tuna fishery and ensures that Filipino fishermen receive fair compensation. For
699 the study limitation of the Tracey project, although there is a high acceptabil-
700 ity of potentially using the app for fishermen, there are still constraints in terms
701 of proper incentives, connectivity issues, technology usability, and education for
702 using the app.

703 A study of Cocco and Mannaro (2021) proposed a blockchain-based technology
704 in the traceability of the supply chain management of a traditional Italian food
705 product, Carasau bread, which is made from durum wheat flour, salt, yeast and
706 water. Since the production of this product is traditional, consumers would de-
707 mand for transparency on the methods used in the Carasau bread production to
708 ensure authenticity. The proposed model involves the combination of the appli-
709 cation of Internet of Things (IoT), specifically the Radio Frequency Identification
710 (RFID) sensors and Interplanetary File System (IPFS) with Blockchain (Cocco
711 & Mannaro, 2021). RFID is a technology that uses radio frequencies to identify
712 and track a tagged object while IPFS allows files to be stored and tracked over a
713 decentralized and distributed file system. Cocco and Mannaro (2021) also stated
714 that using RFID tags that will be integrated with different sensors to monitor
715 food quality will be useful in their study. These sensors include freshness indica-
716 tors to monitor the food quality when packaged, biosensors to detect degradation
717 molecules, time temperature indicator to measure and record temperature and
718 humidity sensors to detect the amount of water vapor in the atmosphere. The
719 integration of IoT and blockchain technology can have a huge impact in increasing

720 traceability in agri-food supply chain. Moreover, this study proposed to have a
721 generic agri-food traceability system which will be based on Ethereum blockchain,
722 Radio-frequency identification (RFID), Near Field Communication (NFC), and
723 Interplanetary File System (IPFS) technology. Moreover, the model proposal also
724 includes sensor network devices, smart contacts, optical cameras and an external
725 database. Each IoT device will be connected to Raspberry Pi and interfaces with
726 blockchain implementing smart contracts and IPFS which authorities can inspect
727 every node and batch online through the uploaded files in IFPS; hashes of the
728 uploaded files on IFPS are also stored on blockchain (Cocco & Mannaro, 2021) .
729 In this way, users along the chain supply can view and trace each batch using the
730 NFC tags promoting transparency and traceability. Overall, the proposed model
731 is a combination of two subsystems. The first one is an on-chain system which is
732 the blockchain implementing smart contracts and will be developed using Solidity,
733 an object-oriented language. The second one is an off-chain system that will be
734 implemented in Javascript using Node.js (to interact with the smart contracts)
735 and Web3.js packages (to interact with blockchain) and these packages should be
736 installed on the Raspberry Pi. However, the purpose of this study is to exam-
737 ine the traceability systems of the agri-food industry and further provide possible
738 solutions.

739 **2.10 Developing a Traceability System for Tuna 740 Supply Chains**

741 The study of Kresna et al. (2017), proposed an IT-based traceability system for
742 tuna supply chain as opposed to the traditional paper based traceability system
743 which has several limitations such as the potency to be manipulated, error by
744 the human, language barrier, and physical damage. The architecture comprises
745 several layers: infrastructure, data, application, communication, and user lay-
746 ers. The infrastructure layer includes computer hardware, network infrastructure,
747 and sensing devices like CCTV, GPS, and RFID for data acquisition. The data
748 layer serves as the system's database, featuring both a main system database and
749 an emergency database for critical situations. The application layer consists of
750 various modules—admin, tracing, transporter, supplier, and government—that
751 registered actors can access through different interfaces. Finally, the user layer
752 consists of the registered actors who utilize the system.

753 The journal article of Tiwari (2020) called Application of Blockchain in Agri-
754 Food Supply Chain conducted two case studies for a blockchain driven app built
755 for supply chain related to food, fishing, and agriculture. The first case-study is
756 the usage and effectiveness of the Provenance system for tuna tracking certifica-
757 tion. The objective of the Provenance system is to enhance transparency in the
758 tuna supply chain by ensuring certification and standard compliance across all
759 roles(e.g. supplier, retailer) in the chain. The system is built using six modular
760 programs: registering, standards, production, manufacturing, tagging, and user-
761 interface. The usage of blockchain in the Provenance system allows transactions to
762 be recorded to allow shared ledger for transparency and smart contracts for secure

763 exchanges of money or information. The usage of the Provenance system is to
764 solve the issues encountered in the tuna fishing industry affected by various factors
765 such as illegal, unregulated, unauthorized (IUU) fishing, fraud, and human rights
766 abuses. The solution of the Provenance system is to allow tracking, tracing, and
767 certification of tuna using blockchain. The Provenance system has a smart tag-
768 ging feature that allows fishermen to use SMS for digital assets on the blockchain
769 to track where the fish, in return, all supply chain stakeholders can access the
770 data that was sourced from the SMS. The second case-study is the usage of the
771 IBM Food Trust for transparency in the food supply chain. The IBM Food Trust
772 aims to solve the problems in the food supply chain, specifically in product safety.
773 Locating supply chain items in real-time using identifiers like GTIN or UPC is
774 the primary feature of the IBM Food Trust. The app also provides end-to-end
775 product provenance, real-time location and status, and facilitates rapid product
776 recalls. The IBM Food Trust also provides insights and visibility for the freshness
777 of the product to reduce losses and spoilage. Lastly, the IBM Food Trust provides
778 certifications from the information taken when handling and managing the prod-
779 ucts in the supply chain. The case studies conducted by Tiwari (2020) illustrates
780 the potential of blockchain technology in improving transparency, efficiency, and
781 ethical practices within supply chains.

782 2.11 Chapter Summary

2.11.1 Comparison Table of Related Studies

Study	Technology Used	Focus Area	Application	Key Findings	Limitations
Shamsuzzoha et al. (2023)	Blockchain (hybrid model)	Fish traceability	Mobile app (Tracey) for fishermen and buyers	Demonstrated feasibility for traceability, certification, transparency; high user acceptability among fishermen	Connectivity issues, technology usability, need for proper incentives and education
Cocco & Mannaro (2021)	Blockchain, IoT (RFID, IPFS), NFC, Ethereum	Italian traditional food (Carasau bread)	Agri-food supply chain traceability system	Proposed a model combining IoT and blockchain to enhance traceability and transparency	High complexity; requires integration of multiple technologies (RFID, IPFS, NFC)
Kresna et al. (2017)	IT-based system, RFID, CCTV, GPS	Tuna supply chain	Digital traceability system (IT-based)	Identified weaknesses in traditional paper-based systems; proposed IT layers for comprehensive tracking	Infrastructure requirements (CCTV, GPS, RFID); limited detail on blockchain
Tiwari (2020) - Provenance	Blockchain, smart contracts, SMS	Tuna tracking certification	Provenance system for tuna supply chain	Enhanced transparency, certification compliance; addressed IUU fishing, fraud, human rights abuses	Limited to specific regions and roles within supply chain
Tiwari (2020) - IBM Food Trust	Blockchain, GTIN/UPC identifiers	Food supply chain	Real-time tracking, product recalls	Improved transparency, efficiency; enabled real-time status, freshness insights	Focused on general food products; high dependency on IBM ecosystem

Table 2.1: Comparison of Studies on Technology for Traceability and Supply Chain Management

783 2.11.2 Research Gaps and Problem

784 Given the advanced existing technologies in blockchain-based traceability systems
 785 for agri-food supply chains, significant gaps remain in understanding the user ex-
 786 perience and integration challenges faced by the fishermen. While the study by
 787 Shamsuzzoha et al. (2023) highlighted the feasibility and benefits of the mobile
 788 application, Tracey, they overlooked possible issues related to technology adop-

⁷⁸⁹ tion, digital literacy, and connectivity.

⁷⁹⁰ Furthermore, existing technologies have primarily focused on large-scale imple-
⁷⁹¹ mentations and theoretical frameworks without adequately considering the prac-
⁷⁹² tical implications and user experience needed for effective system integration. This
⁷⁹³ study aims to address these gaps by exploring real-world challenges faced by users,
⁷⁹⁴ especially fishermen, in adopting blockchain technology for traceability.

⁷⁹⁵ 2.11.3 Summary

⁷⁹⁶ The literature reviewed highlighted the critical challenges and opportunities re-
⁷⁹⁷ garding the tuna supply chain, particularly in the areas of traceability and sustain-
⁷⁹⁸ ability. Existing supply chain technologies, especially those utilizing blockchain,
⁷⁹⁹ present solutions but also come with limitations in terms of blockchain adoption.
⁸⁰⁰ The application of blockchain technology in the tuna supply chain has shown po-
⁸⁰¹ tential for enhancing traceability from ocean to consumer.

⁸⁰² Through this paper, a blockchain-driven solution could contribute to providing a
⁸⁰³ more efficient and transparent supply chain. However, further studies are neces-
⁸⁰⁴ sary to assess the long-term sustainability of blockchain in such systems.

⁸⁰⁵ **Chapter 3**

⁸⁰⁶ **Research Methodology**

⁸⁰⁷ This chapter outlines a clear and detailed description of the research methods and
⁸⁰⁸ processes used in the development and evaluation of SeaXChange: A Blockchain
⁸⁰⁹ Driven App for Tuna Supply Chain Management. The algorithms, systems, theo-
⁸¹⁰ ries, framework and models are described in detail in which this chapter establishes
⁸¹¹ the foundation of this study .This chapter also explains the data collection method
⁸¹² used ensuring the validity and reliability of the results.In addition, the chapter
⁸¹³ discusses the considerations and potential limitations of this study. Overall, this
⁸¹⁴ will serve as a guide for the readers in understanding the structured process of
⁸¹⁵ developing the SeaXChange.

⁸¹⁶ **3.1 Software Development**

⁸¹⁷ Scrum is a framework within the Agile development that prioritizes flexibility.
⁸¹⁸ It is an iterative software development approach that lets a project be broken
⁸¹⁹ down into phases and emphasizes continuous improvements. For this study, the

820 researchers opted to use Scrum because it involved many stakeholders and it
821 operated in a ever-changing environment. Scrum allowed the team to adapt to
822 new requirements through structured sprint planning, weekly reports, and sprint
823 reviews, ensuring continuous alignment with project goals.

824 **3.2 Research Activities**

825 For this study, the researchers opted for interviews because it enabled in-depth
826 exploration of stakeholder perspectives and experiences. The identified fisher and
827 supplier client interface was tested within the perimeters of Barangay Sapa, Mi-
828 agao, Iloilo, Philippines. The identified retailer testers were the vendors who
829 reside in Barangay Mat-y and Barangay Sapa in Miagao. The identified con-
830 sumer testers were situated in Miagao. The specific choice of these stakeholders
831 and locations within Miagao was made to align with the study's defined scope,
832 which concentrates on applying blockchain technology to enhance traceability and
833 transparency within the small-scale tuna supply chain and localized market in the
834 specified barangay of Miagao, Iloilo. This focus includes examining the trace-
835 ability of whole, small-sized caught tuna products from capture through the local
836 supply chain to consumers within this area. This localized approach facilitated a
837 practical assessment of the SeaXChange application's potential effectiveness and
838 user experience within a specific operational context.

839 3.2.1 Feedback Collection Method

840 The feedback was collected through a combination of structured interviews and
841 assessment forms. Participants were asked to complete an assessment rubric evalu-
842 ating the SeaXChange app across key criteria such as functionality, end-user needs,
843 performance, usability, ease of use and feasibility. Moreover, follow-up interviews
844 were conducted to gather deeper qualitative insights and obtain suggestions for
845 system improvement.

846 The feedback gathered from fishermen, suppliers and retailers, and consumers
847 was analyzed based on the SeaXChange assessment rubric, which evaluated six
848 major categories: Functionality, End-user Needs, Performance, Usability, Ease of
849 Use and Feasibility. The collected data were analyzed using descriptive statistics,
850 through the computation of mean scores for each assessment criterion. These
851 mean values were used to summarize stakeholder perceptions of the system. Mean
852 ratings were calculated based on the 1-5 Likert Scale where 1 = Poor and 5 =
853 Very Good.

854 3.2.2 Data Gathering**855 • Primary Data:**

- 856 – Stakeholder(Fishermen, Supplier, Retailers, and Consumers) interviews
857 were conducted to identify the use-case and user requirements, interface
858 usability, and adoption challenges.
- 859 – Observations were made of existing tuna supply chain processes in local
860 settings.

861 • **Secondary Data:**

- 862 – Literature review was conducted on blockchain applications in supply
863 chain management and product traceability.
- 864 – Industry reports and regulatory documents related to tuna fishing and
865 supply chain operations.

866 **3.2.3 Designing and Developing the System**

867 1. **Software Development Methodology:** The project followed a Scrum
868 framework to ensure continuous iteration, stakeholder involvement, and flex-
869 ibility in adapting to feedback.

870 2. **Technology Stack:**

871 • **Front-end Development:** Used React for creating a secure and user-
872 friendly interface for stakeholders, prioritizing simple and responsive
873 user-interface.

874 • **Back-end Development:** Used Node.js along with Express for managing
875 back-end processes and API integration. Express is a flexible web applica-
876 tion framework for Node.js used to build APIs for web applications.
877 Docker for containerization of the project and Windows Subsystem for
878 Linux (Ubuntu as the Linux distribution) for setting up the network.

879 • **Cloud Infrastructure:** Used Google Cloud to host backend services and
880 manage the databases, where the app could be accessed globally. It
881 also ensured the app could scale smoothly as more data and users were
882 added.

- 883 • Blockchain Framework: Used Go language for developing smart con-
884 tracts and providing an immutable ledger for transaction data.
- 885 • Database for Accounts: Used Firebase managing user accounts and
886 authentication.

887 **3. Blockchain Development Platform:**

- 888 • Used Hyperledger Fabric for its permissioned nature and scalable ar-
889 chitecture.
- 890 • The open-sourced resources and timely updates of Hyperledger Fabric
891 components is ideal for creating a distributed ledger for tuna supply
892 chain.

893 **3.2.4 Implementing Algorithms and Services**

894 The system for this study is built on top of a Hyperledger Fabric project, it also
895 utilized combinations of algorithms to facilitate the work flow of data or asset
896 as well as ensuring high security with encryption and decryption configuration
897 techniques.

898 **1. Consensus Algorithm**

899 The project followed Raft(Leader-based consensus) for handling organiza-
900 tions or nodes. Raft is intended for managing a replicated log in a blockchain
901 network. Raft is a Crash Fault Tolerant (CFT) protocol, is designed to han-
902 dle non-malicious node failures (e.g., hardware crashes, network issues) In
903 Raft, one node is elected as the leader, and it coordinates the ordering of

904 transactions (Xu et al, 2022) . The leader replicates log entries (transactions)
905 to follower nodes, ensuring consistency across the network.

906 **2. Cryptographic Algorithm**

907 The project employed several cryptographic algorithms to ensure security
908 and privacy. These cryptographic data served as digital signatures and iden-
909 tity verification for the project. ECDSA (Elliptic Curve Digital Signature
910 Algorithm) was used for generating digital signatures while X.509 certifi-
911 cates are intended for identity management and authentication of partic-
912 ipants (Anitha & Sankarasubramanian, n.d.) . For the encryption, AES
913 (Advanced Encryption Standard) was used for encrypting data at rest and
914 in transit. TLS (Transport Layer Security) secured communication between
915 network nodes. SHA-256 (Secure Hash Algorithm-256) ensured data in-
916 tegrity by generating unique hashes for blocks and transactions.

917 **3. Membership Service**

918 The implementation of the Membership Service Provider (MSP) requirement
919 involved a set of folders added to the network configuration. These folders
920 defined an organization both internally, by specifying its administrators,
921 and externally, by enabling other organizations to verify the authority of
922 entities attempting specific actions. While Certificate Authorities (CAs)
923 are responsible for generating the certificates that represent identities, the
924 MSP included a list of permitted identities. The MSP specified which Root
925 CAs and Intermediate CAs are authorized to define members of a trust
926 domain. This was achieved by either listing the identities of their members
927 or identifying the CAs allowed to issue valid identities for those members.

928 **4. Ordering Service**

929 The ordering service in this study played a crucial role in maintaining the
930 integrity and functionality of the blockchain network. Its primary responsibilities
931 included ensuring that transactions are processed in the correct
932 sequence (transaction ordering), grouping transactions into blocks based on
933 configurable parameters like size or timeout (block creation), and distributing
934 these ordered blocks to peers for validation and commitment (block
935 distribution) (Nassar et al, 2024). Additionally, the ordering service provided
936 fault tolerance to ensure the network remains operational even in the
937 presence of node failures through Raft.

938 **5. Endorsement Policy**

939 Fabric employs endorsement policies to specify which peers must validate a
940 transaction before it's committed. The algorithm involved multi-signature
941 schemes where a transaction is valid if it receives endorsements from the
942 required peers as per the policy.

943 **6. Chaincode (Smart Contract)**

944 The handling and flow of business logic agreed to by members of the tuna
945 supply chain in the blockchain network is executed by a chaincode or smart
946 contract. The chaincode of the app was written in Go language. Docker
947 container was used for enabling the chaincode to securely run along with the
948 overall hyperledger fabric configurations. Chaincode initializes and manages
949 ledger state through transactions submitted by applications (Hyperledger
950 Fabric Documentation, 2024) . The chaincode followed the object-oriented
951 paradigm for creating classes and objects necessary for the tuna supply
952 chain.

953 3.2.5 Modeling the System Architecture

954 The system architecture of the project were consisted of many nodes that com-
955 municated with each other. The chaincode enabled the system to run algorithms,
956 particularly, holding state and ledger data, and executes transactions such as asset
957 transfer in the tuna supply chain.

958 • **Blockchain Architecture**

959 The project involved peer, ordering services, ledger, and client application
960 to perform various transaction such as tracing the origin and the stop points
961 of a tuna asset. Peers are nodes in the blockchain network that maintained
962 a copy of the distributed ledger and execute chaincode (smart contracts).
963 The ordering service is the central component of the blockchain for ordering
964 transactions and creating blocks to distribute to peers through consensus
965 mechanism. The ledger is the immutable record of all transaction in the tuna
966 supply chain network, stored across all peers. The client application is the
967 interface through which users or tuna supply chain participants interact with
968 the blockchain network. Figure 3.1 below shows the blockchain architecture
969 of SeaXChange.

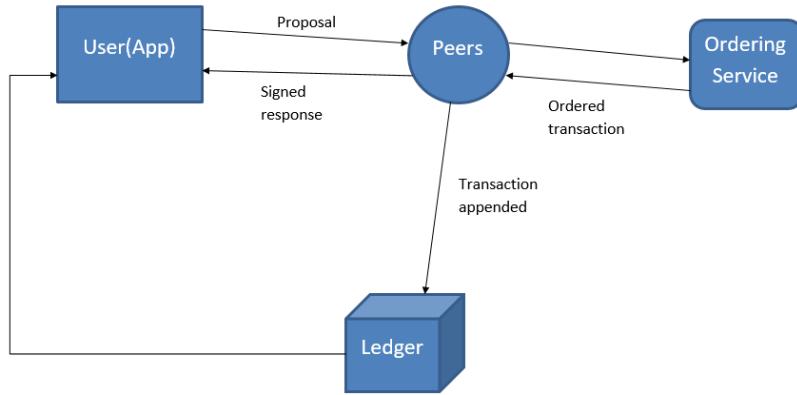


Figure 3.1: Blockchain Architecture of SeaXChange

• Overall System Architecture

The overall system architecture contains a web application built with Next.js for the frontend, utilizing Firebase for user authentication and account management. As shown in Figure 3.2, the application follows a role-based access model (Fisher, Supplier, Retailer, Consumer) where each role has specific permissions for interacting with tuna assets in the supply chain. The backend runs on Google Cloud Platform, consisting of an Express.js API that interfaces with a Hyperledger Fabric blockchain network (containerized in Docker) which stores and manages the immutable record of tuna assets and their transfers between supply chain participants. This architecture enables secure tracking of tuna from creation by fishers through the supply chain to consumers, with appropriate viewing and transfer capabilities assigned to each role in the process.

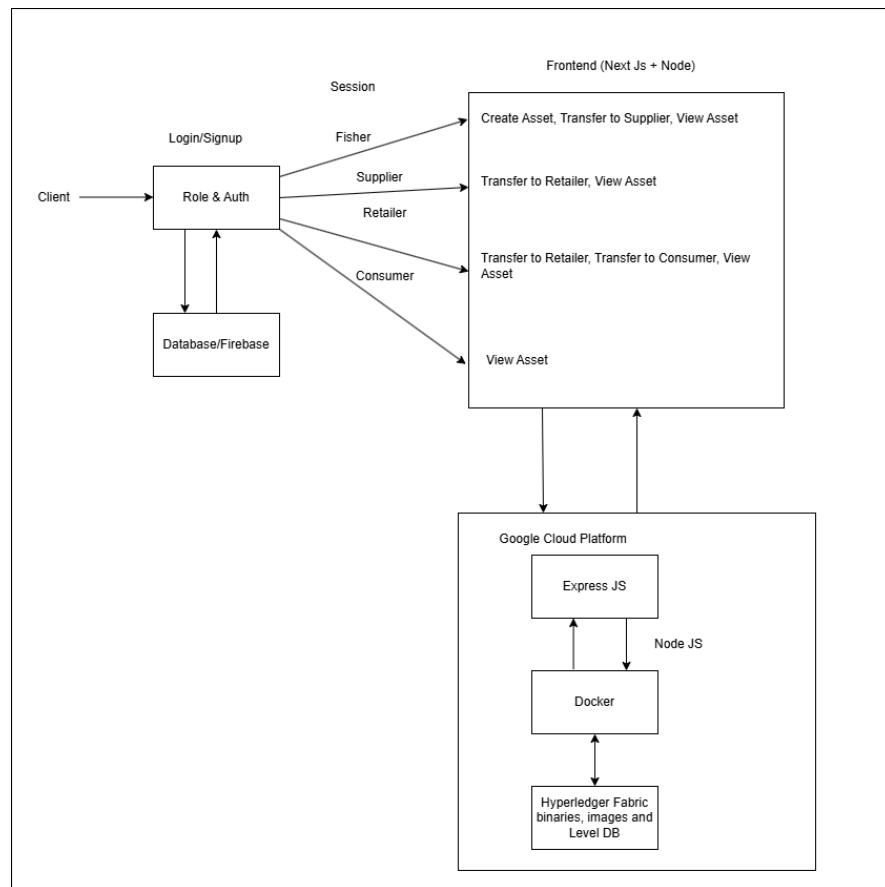


Figure 3.2: Overall System Architecture of SeaXChange

983 • **Use Case**

984 The use case shows the outline on how the user will interact with the SeaX-
 985 Change App. It followed the major stages or participants in the tuna supply
 986 chain. Figure 3.3 shows the use case diagram for SeaXChange.

987 1. **Fisher**

- 988 - Encodes tuna I.D. of fish.
- 989 - Encodes the date when the fish was captured.
- 990 - Encodes the location where the fish was captured.
- 991 - Encodes the fishing method used.

992 - Query the origin and exchange of the tuna asset.

993 **2. Supplier**

994 - Encodes when the product was transferred from fisher to supplier.

995 - Query the origin and exchange of the tuna asset.

996 - Generate supplier's location during retrieval of tuna asset.

997 **3. Retailer**

998 - Encodes when the product was retrieved from the supplier or another
999 retailer.

1000 - Query the origin and exchange of the tuna asset.

1001 - Generate retailer's location during retrieval of tuna asset.

1002 **4. Consumer**

1003 - Retrieve data from retailer.

1004 - Query the origin and exchange of the tuna asset.

1005 There are four (4) types of users that will use the app. The first user
1006 type is the Fisher, which will be the starting point of the blockchain.
1007 It will encode the catch details of a tuna product such as the date of
1008 capture, location, and fishing method. The second user type is the
1009 Supplier, which will encode when the product was transferred from
1010 the fisher to the supplier, as well as generate their location during
1011 the retrieval of the tuna asset. The third type is the Retailer, which
1012 will encode when the product was transferred from the supplier to the
1013 retailer or in the case of multiple retailers, from the previous retailer to
1014 the current retailer, their location is also generated during the retrieval
1015 of the tuna asset. Lastly, the Consumers, which can only query the
1016 origin and exchange of tuna assets.



Figure 3.3: Use case diagram for SeaXChange.

₁₀₁₇ **Chapter 4**

₁₀₁₈ **Results and Discussions**

₁₀₁₉ **4.1 Overview**

₁₀₂₀ This chapter presents the results of the system. Included in this chapter are
₁₀₂₁ screenshots and the discussion of results. The tuna supply chain management
₁₀₂₂ smart contract on Hyperledger Fabric has been initiated and tested within a con-
₁₀₂₃ trolled blockchain environment. Results indicated that the system was function-
₁₀₂₄ ally robust and reliable, having managed assets, transaction integrity, and the
₁₀₂₅ ability to query and update the ledger in the blockchain. This chapter presents
₁₀₂₆ the details of the major steps executed during the process, results for those steps,
₁₀₂₇ and the current status of the system's operations.

1028 **4.2 Smart Contract Deployment and Installa-**

1029 **tion**

1030 **4.2.1 Hyperledger Fabric Prerequisites**

1031 Before executing a smart contract framework and blockchain system, it is cru-
1032 cial to first install and set up the necessary tools and technologies. This includes
1033 setting up Hyperledger Fabric, which involves installing the Fabric binaries, con-
1034 figuring the network, and ensuring all necessary dependencies like Docker, Docker
1035 Compose, and Node.js are installed and properly configured. Additionally, setting
1036 up the required certificates, defining the channel configurations, and ensuring that
1037 peer nodes and orderers are correctly connected and synchronized are all essential
1038 steps in preparing the environment for blockchain and smart contract operations.

1039 • **Software Requirements:**

- 1040 – **Docker and Docker Compose:** Hyperledger Fabric needs to have
1041 Docker installed and running on the system. Docker is required to run
1042 the peer and ordering services of the blockchain system.
- 1043 – **Node.js:** Required for the Fabric SDK for client application integra-
1044 tion with JavaScript libraries such as react.
- 1045 – **Go:** Ensure Go is installed, and the GOPATH environment variable
1046 is set up. This is essential for building and running chaincode(smart
1047 contract) written in Go.
- 1048 – **Fabric Samples:** Clone the official Hyperledger Fabric's fabric-samples
1049 repository from GitHub:

```
1050      git clone -b release-2.4 --single-branch  
1051      https://github.com/hyperledger/fabric-samples  
1052      cd fabric-samples/test-network  
1053
```

1054 – **Binaries and Docker Images:**

```
1055 curl -sSL https://bit.ly/2ysbOFE | bash -s
```

```
1056
```

1057 • **Network Setup:**

- ```
1058 – Run the test-network script to start the Hyperledger Fabric test net-
1059 work:
```

```
1060 ./network.sh up
```

```
1061
```

```
1062 This script starts a peer, an ordering service, and a CA (Certificate
1063 Authority) on the local machine.
```

- ```
1064 – After starting the network to docker in the same directory (test-network),  
1065 a channel must be created:
```

```
1066 ./network.sh createChannel
```

```
1067
```

1068 • **Deploying Chaincode (Smart Contract):**

- ```
1069 – Step 1:
```

```
1070 export PATH=${PWD}/../bin:$PATH
```

```
1071
```

1072 – Step 2:

```
1073 export FABRIC_CFG_PATH=$PWD/../../config/
```

1074

1075 — Step 3:

```
1076 export CORE_PEER_TLS_ENABLED=true
```

```
1077 export CORE_PEER_LOCALMSPID="Org1MSP"
```

```
1078 export CORE_PEER_TLS_ROOTCERT_FILE=${PWD}/organizations
```

1079 /peerOrganizations/org1.example.com/peers/peer0.org1.example.com

1080 /tls/ca.crt

```
1081 export CORE_PEER_MSPCONFIGPATH=${PWD}/organizations
```

1082 /peerOrganizations/org1.example.com/users

1083 /Admin@org1.example.com/msp

```
1084 export CORE_PEER_ADDRESS=localhost:7051
```

1085

## 4.2.2 Invoking the Blockchain System

After setting up the prerequisites, including Docker containers, the test network, and chaincode, the tuna supply chain system can now be invoked for creating, transferring, and querying tuna assets. The figures provided below demonstrate the processes involved in invoking the blockchain system.

```
zyx@zxy002#APTOR-093000# /mnt/c/SpecialProblem/hyperledger-fabric/#fabric-samples/test-network$ peer chaincode query -C mychannel -n fabric-test -v 1.0 -c '{"function": "GetAllAssets"}'
[{"id": "tuna", "Species": "Yellowfin", "Weight": 8.5, "CatchLocation": "Palawan", "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "tuna", "Species": "Bull", "Weight": 5.5, "CatchLocation": "Antique", "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "tuna", "Species": "Skipjack", "Weight": 7.5, "CatchLocation": "Antique", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "jagne", "Species": "Jagne", "Weight": 1.5, "CatchLocation": "Davao", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "tuna", "Species": "Albacore", "Weight": 6, "CatchLocation": "Davao", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "tuna", "Species": "Skipjack", "Weight": 4, "CatchLocation": "Davao", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}, {"id": "tuna", "Species": "Jagne", "Weight": 2, "CatchLocation": "Davao", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Supplier", "Retailer": "Santos"}]
```

Figure 4.1: Query Smart Contract: Check assets

1091     ● **Adding new tuna assets:**

1092     Here, a new tuna asset is created and registered on the blockchain. This in-  
 1093     volves invoking the smart contract to add a new entry, which includes details  
 1094     such as the type of tuna, quantity, and any other relevant information. This  
 1095     step ensures that newly caught or acquired tuna can be tracked throughout  
 1096     the supply chain.

```
ryxx2082@LAPTOP-Q93UQUB8:/mnt/c/SpecialProblems/hyperledger-fabric-samples/test-network$ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tlsRootCertFiles $PWD/organizations/ordererOrganizations/example.com/tlsca/tlsca.example.com-cert.pem -C mychannel -n mychannel -c '{"function": "CreateAsset", "args": [{"ID": "tuna4", "Species": "Skipjack", "Weight": 8.5, "CatchDate": "2024-12-01", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierA", "RetailerY": "RetailerZ"}, {"ID": "tuna5", "Species": "Tuna", "Weight": 8.4, "CatchDate": "2024-12-02", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierB", "RetailerY": "RetailerZ"}, {"ID": "tuna6", "Species": "Bluefin", "Weight": 8.5, "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierC", "RetailerY": "RetailerZ"}]}'
ryxx2082@LAPTOP-Q93UQUB8:/mnt/c/SpecialProblems/hyperledger-fabric-samples/test-network$
```

Figure 4.2: Invoke Smart Contract: Create/Add new tuna asset

1097     ● **Check all assets after adding a new tuna asset:**

1098     After adding a new tuna asset, the smart contract is queried again to verify  
 1099     that the asset has been successfully added. This step confirms that the new  
 1100     asset is part of the current inventory and that no discrepancies exist in the  
 1101     recorded data.

```
ryxx2082@LAPTOP-Q93UQUB8:/mnt/c/SpecialProblems/hyperledger-fabric-samples/test-network$ peer chaincode query -C mychannel -n basic -c '{"Args": ["GetAllAssets"]}'
[{"ID": "tuna1", "Species": "Skipjack", "Weight": 8.5, "CatchDate": "2024-12-01", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierA", "RetailerY": "RetailerZ"}, {"ID": "tuna2", "Species": "Tuna", "Weight": 8.4, "CatchDate": "2024-12-02", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierB", "RetailerY": "RetailerZ"}, {"ID": "tuna3", "Species": "Bluefin", "Weight": 8.5, "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierC", "RetailerY": "RetailerZ"}, {"ID": "tuna4", "Species": "Skipjack", "Weight": 8.5, "CatchDate": "2024-12-01", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierA", "RetailerY": "RetailerZ"}, {"ID": "tuna5", "Species": "Tuna", "Weight": 8.4, "CatchDate": "2024-12-02", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierB", "RetailerY": "RetailerZ"}, {"ID": "tuna6", "Species": "Bluefin", "Weight": 8.5, "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Jaguar", "Supplier": "SupplierX", "Retailer": "RetailerY", "SupplierX": "SupplierC", "RetailerY": "RetailerZ"}]
```

Figure 4.3: Query Smart Contract: Check assets with new tuna asset

1102     ● **Transfer tuna asset to Supplier:**

1103     This step involves transferring ownership of a tuna asset from the current  
 1104     holder (e.g., a fisherman or a trader) to a supplier. The smart contract is  
 1105     invoked to facilitate the transfer, ensuring that the transaction is securely  
 1106     recorded on the blockchain and that the asset's new owner is updated ac-  
 1107     cordingly.

```
1108 Error: error getting broadcast client: orderer client failed to connect to orderer.example.com:7050: failed to create new connection: context deadline exceeded
1109 崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tlsCaFile ./crypto-config/ordererOrganizations/example.com/orderers/orderer.example.com/tlsca.example.com-cert.pem" --channelName mychannel --peerAddresses localhost:9051 --tlsRootCertFiles "./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt" -c '{"function":"transferAsset","args":["tuna6","Supplier","SupplierA"]}'
1110 崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tlsCaFile ./crypto-config/ordererOrganizations/example.com/orderers/orderer.example.com/tlsca.example.com-cert.pem" --channelName mychannel --peerAddresses localhost:9051 --tlsRootCertFiles "./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt" -c '{"function":"transferAsset","args":["tuna6","Supplier","SupplierA"]}'"
1111 崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode query -c '{"Args": ["ReadAsset","tuna6"]}'
1112 崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$
```

Figure 4.4: Invoke Smart Contract: Transfer asset to a supplier

1108     • Check the updated tuna asset:

1109       After the transfer, the smart contract is queried once more to check if the  
1110       asset details have been updated correctly. This step verifies that the asset's  
1111       new owner is now the supplier and that all relevant information is correctly  
1112       updated on the blockchain.

```
1113崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -C mychannel -n basic -c '{"Args":["TransferAsset","tuna6","Supplier","SupplierA"]}' orderer endpoint: orderer.example.com:7050
1114崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ Error: error getting broadcast client: orderer client failed to connect to orderer.example.com:7050: failed to create new connection: context deadline exceeded
1115崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tlsCaFile ./crypto-config/ordererOrganizations/example.com/orderers/orderer.example.com/tlsca.example.com-cert.pem" --channelName mychannel --peerAddresses localhost:9051 --tlsRootCertFiles "./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt" -c '{"function":"TransferAsset","args":["tuna6","Supplier","SupplierA"]}'"
1116崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode query -C mychannel -n basic -c '{"Args":["ReadAsset","tuna6"]}'"
1117崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$
```

Figure 4.5: Query Smart Contract: Check asset after transfer

1113     • Transfer tuna asset to Retailer:

1114       Similar to the supplier transfer, this step involves transferring the tuna asset  
1115       from the supplier to a retailer. The smart contract facilitates this transfer,  
1116       ensuring that ownership is correctly updated and that the retailer has control  
1117       over the tuna asset. This step is crucial for the supply chain as it moves the  
1118       tuna from bulk supply to retail.

```
1119崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -C mychannel -n basic -c '{"Args":["TransferAsset","tuna6","Supplier","SupplierA"]}' orderer endpoint: orderer.example.com:7050
1120崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ Error: error getting broadcast client: orderer client failed to connect to orderer.example.com:7050: failed to create new connection: context deadline exceeded
1121崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tlsCaFile ./crypto-config/ordererOrganizations/example.com/orderers/orderer.example.com/tlsca.example.com-cert.pem" --channelName mychannel --peerAddresses localhost:9051 --tlsRootCertFiles "./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt" -c '{"function":"TransferAsset","args":["tuna6","Supplier","SupplierA"]}'"
1122崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode query -C mychannel -n basic -c '{"Args":["ReadAsset","tuna6"]}'"
1123崖xx2002LAPTOP-Q93UQH6:/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$
```

Figure 4.6: Invoke Smart Contract: Transfer asset to a retailer

1119     • Check the updated tuna asset:

1120       After the transfer to the retailer, another query is made to verify the updated  
1121       asset details. This step ensures that the transaction was successful and that

### 4.3. BACKEND SECURITY ANALYSIS (HYPERLEDGER FABRIC ON GCP)47

1122 the retailer now has ownership of the tuna asset. It confirms that the asset  
1123 has moved through the supply chain correctly.

```
root@DESKTOP-Q93UQUB:~/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode query -C mychannel -n basic -c '{"Args": ["ReadAsset", "tuna0"]}'
{"ID": "tuna0", "Species": "Skipjack", "Weight": 6, "CatchLocation": "2024-12-05", "CatchDate": "Antique", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "SupplierA", "Retailer": "RetailerA"}
root@DESKTOP-Q93UQUB:~/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$
```

Figure 4.7: Query Smart Contract: Check asset after transfer

#### 1124 • **Query Smart Contract and check updated assets:**

1125 The final step involves querying the smart contract to get a complete overview  
1126 of all the assets in the supply chain. This includes all tuna assets from fishing  
1127 to retail, allowing stakeholders to monitor and manage inventory effectively.  
1128 It provides traceability in the supply chain, helping to maintain freshness  
1129 and authenticity of the tuna as it moves through the market.

```
root@DESKTOP-Q93UQUB:~/mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network$ peer chaincode query -C mychannel -n basic -c '{"Args": ["DetailAssets"]}'
[{"ID": "tuna0", "Species": "Skipjack", "Weight": 6.5, "CatchLocation": "Antique", "CatchDate": "2024-12-01", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "SupplierX", "Retailer": "RetailerX"}, {"ID": "tuna1", "Species": "Yellowfin", "Weight": 8.5, "CatchLocation": "Palawan", "CatchDate": "2024-12-02", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "SupplierY", "Retailer": "RetailerY"}, {"ID": "tuna2", "Species": "Bluefin", "Weight": 8.5, "CatchLocation": "Philippines", "CatchDate": "2024-12-03", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "SupplierZ", "Retailer": "RetailerZ"}, {"ID": "tuna3", "Species": "Albacore", "Weight": 5, "CatchLocation": "Antique", "CatchDate": "2024-12-04", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "MAA", "Retailer": "RetailerA"}, {"ID": "tuna4", "Species": "Skipjack", "Weight": 5, "CatchLocation": "Antique", "CatchDate": "2024-12-05", "FishingMethod": "Longline", "Vessel": "Davao", "Supplier": "SupplierA", "Retailer": "RetailerA"}, {"ID": "tuna5", "Species": "Skipjack", "Weight": 6, "CatchLocation": "2024-12-05", "CatchDate": "Antique", "FishingMethod": "Longline", "Vessel": "Jagnee", "Supplier": "SupplierX", "Retailer": "RetailerX"}]
```

Figure 4.8: Query Smart Contract: Check updated assets

## 1130 4.3 Backend Security Analysis (Hyperledger Fab- 1131 ric on GCP)

### 1132 4.3.1 System Architecture and Deployment Overview

1133 The backend of the system's tuna assets was developed using a containerized  
1134 Hyperledger Fabric deployed on Google Cloud Platform (GCP). The network of

<sub>1135</sub> Hyperledger Fabric consists of a peer node, an ordering node, and Certificate  
<sub>1136</sub> Authorities (CAs).

<sub>1137</sub> **4.3.2 Blockchain Network Security**

<sub>1138</sub> The blockchain network leverages Hyperledger Fabric's security model to ensure  
<sub>1139</sub> authenticated transactions and controlled access. A Membership Service Provider  
<sub>1140</sub> (MSP) manages identities and issues certificates based on a Public Key Infrastruc-  
<sub>1141</sub> ture (PKI) model, ensuring that only verified participants can interact with the  
<sub>1142</sub> network.

<sub>1143</sub> Key security features include:

<sub>1144</sub> **Channel Privacy**

<sub>1145</sub> Channels act as private communication subnets, isolating transaction data so that  
<sub>1146</sub> only authorized organizations can access and process specific transactions.

<sub>1147</sub> **Policies and Access Control**

<sub>1148</sub> Policies, including endorsement policies and access control lists (ACLs), govern  
<sub>1149</sub> how transactions are validated, how channel resources are accessed, and how  
<sub>1150</sub> changes to the network are approved. Endorsement policies specifically define  
<sub>1151</sub> which peer nodes must approve a transaction before it is committed to the ledger.

**1152 Secure Communication**

**1153** Transport Layer Security (TLS) is enforced across node communications to protect  
**1154** data in transit. Mutual TLS is used for operational endpoints like monitoring  
**1155** services.

**1156 Identity and Role Management**

**1157** Every network participant—peer nodes, orderer nodes, client applications (SeaX-  
**1158** Change Web Application)—has a cryptographically verifiable identity, with roles  
**1159** defined within the framework to control access and permissions within channels.

**1160 Hardware Security Modules (HSMs)**

**1161** Critical cryptographic operations, such as signing transactions under the blockchain  
**1162** assets invocation, can optionally be handled by HSMs to secure private keys out-  
**1163** side of the software environment.

**1164** These layered mechanisms collectively ensure the confidentiality, integrity, and  
**1165** authenticity of transactions in the blockchain network.

**1166 4.3.3 Smart Contract Automated Test Result**

**1167** To validate the security and functionality of the deployed smart contracts on the  
**1168** Hyperledger Fabric network, an automated testing script (app.js) under asset-  
**1169** transfer-basic directory was executed. The script interacted with the blockchain

1170 network through the gateway application, utilizing the defined channel (mychannel)  
 1171 and chaincode (basic). The automated tests performed the following opera-  
 1172 tions:

1173 **InitLedger Transaction**

1174 The ledger was initialized by creating a predefined set of tuna asset entries. The  
 1175 transaction was successfully committed, confirming the proper initialization of  
 1176 asset data. (See Figure 4.9 for initialization confirmation.)

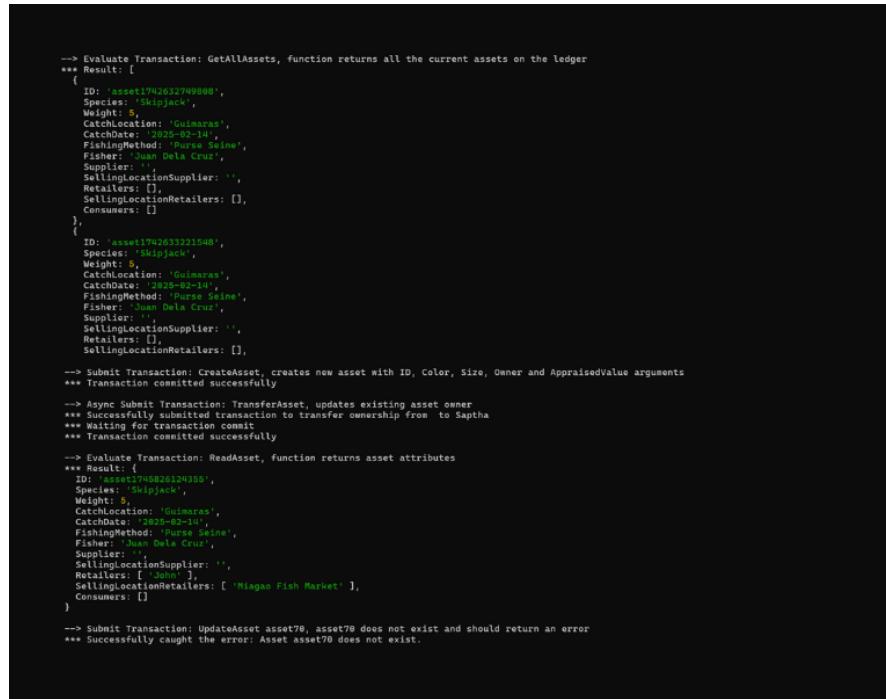
```
x:\www\Hyperledger\APITest\Q3\Q3Q3\bin>cd ..\..\fabric-samples\asset-transfer-basic\application-gateway-javascript\src$ node app.js
channelName: mychannel
chaincodeName: basic
rootCertPath: ./certs
cryptoPath: ./crypto
keyDirectoryPath: ./mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network/organizations/peerOrganizations/org1.example.com/users/User1@org1.example.com/msp
keyCertPath: ./mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network/organizations/peerOrganizations/org1.example.com/users/User1@org1.example.com/msp
certDirectoryPath: ./mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network/organizations/peerOrganizations/org1.example.com/users/User1@org1.example.com/msp
signcerts
signcertPath: ./mnt/c/SpecialProblem/hyperledger-fabric3/fabric-samples/test-network/organizations/peerOrganizations/org1.example.com/users/User1@org1.example.com/msp
/tls
/cacerts
peerEndpoint: localhost:7981
peerHostAlias: peer0.org1.example.com
--> Submit Transaction InitLedger, function creates the initial set of assets on the ledger
*** Transaction committed successfully
```

Figure 4.9: Initialization Confirmation of the Ledger

1177 **GetAllAssets Query**

1178 A query operation retrieved all existing assets recorded on the ledger. The results  
 1179 displayed multiple tuna asset entries with details such as species, weight, catch  
 1180 location, catch date, fishing method, fisher, supplier, retailers, selling locations,  
 1181 and consumers. (See Figure 4.10 for the asset retrieval output.)

### 4.3. BACKEND SECURITY ANALYSIS (HYPERLEDGER FABRIC ON GCP)51



```
--> Evaluate Transaction: GetAllAssets, function returns all the current assets on the ledger
*** Result: [
 {
 ID: 'asset1742632749908',
 Species: 'Skipjack',
 Weight: 5,
 CatchLocation: 'Guimaras',
 CatchDate: '2025-02-14',
 FishingMethod: 'Purse Seine',
 Fisher: 'Juan Dela Cruz',
 Supplier: '',
 SellinglocationSupplier: '',
 Retailers: [],
 SellinglocationRetailers: [],
 Consumers: []
 },
 {
 ID: 'asset1742631221548',
 Species: 'Skipjack',
 Weight: 5,
 CatchLocation: 'Guimaras',
 CatchDate: '2025-02-14',
 FishingMethod: 'Purse Seine',
 Fisher: 'Juan Dela Cruz',
 Supplier: '',
 SellinglocationSupplier: '',
 Retailers: [],
 SellinglocationRetailers: []
 }
]
--> Submit Transaction: CreateAsset, creates new asset with ID, Color, Size, Owner and AppraisedValue arguments
*** Transaction committed successfully
--> Async Submit Transaction: TransferAsset, updates existing asset owner
*** Successfully submitted transaction to transfer ownership from to Captha
*** Waiting for transaction commit
*** Transaction committed successfully
--> Evaluate Transaction: ReadAsset, function returns asset attributes
*** Result: {
 ID: 'asset174263124388',
 Species: 'Skipjack',
 Weight: 5,
 CatchLocation: 'Guimaras',
 CatchDate: '2025-02-14',
 FishingMethod: 'Purse Seine',
 Fisher: 'Juan Dela Cruz',
 Supplier: '',
 SellinglocationSupplier: '',
 Retailers: ['John'],
 SellinglocationRetailers: ['MiaGao Fish Market'],
 Consumers: []
}

--> Submit Transaction: UpdateAsset asset70, asset70 does not exist and should return an error
*** Successfully caught the error: Asset asset70 does not exist.
```

Figure 4.10: Initialization Confirmation of the Ledger

#### 1182 CreateAsset Transaction

1183 A new asset was successfully created and appended to the blockchain. The trans-  
1184 action was committed without errors, validating the chaincode's ability to handle  
1185 new data insertion.

#### 1186 TransferAsset Transaction

1187 Ownership transfer of an existing asset was simulated. The transaction was suc-  
1188 cessfully submitted and committed, demonstrating the correct application of asset  
1189 updates in the blockchain ledger.

**1190 ReadAsset Query**

1191 The updated asset was retrieved and verified to ensure the correctness of the  
1192 transfer. The retrieved asset data reflected the changes made through the previous  
1193 transaction, confirming data consistency.

**1194 UpdateAsset Error Handling**

1195 An attempt to update a non-existent asset (asset70) was performed to test the  
1196 smart contract's error-handling mechanism. The application correctly caught and  
1197 reported the error, verifying that improper transactions are adequately handled  
1198 and rejected.

**1199 Summary of Results**

1200 All positive transactions (initialization, creation, transfer, and reading) were suc-  
1201 cessfully executed and committed. The smart contract exhibited robust error  
1202 handling during invalid operations. Endorsement policies and Membership Ser-  
1203 vice Provider (MSP) enforcement ensured transaction authenticity and integrity  
1204 during the process. These tests confirm the functional reliability and transac-  
1205 tion security of the smart contracts used within the tuna supply chain blockchain  
1206 network.

1207 **4.3.4 GCP Infrastructure Security**

1208 The Hyperledger Fabric network deployment on Google Cloud Platform (GCP)  
1209 was secured by leveraging multiple layers of Google's infrastructure security model,  
1210 following best practices in network, identity, and data protection.

1211 **Firewall Rules and Network Control**

1212 Only essential ports (e.g., 7051 for peer communication and 7050 for the ordering  
1213 service) were opened to minimize network exposure. GCP's VPC firewall rules,  
1214 ingress and egress controls, and options like VPC Service Controls help further  
1215 isolate services and prevent unauthorized access. Traffic between virtual machines  
1216 and Google APIs is securely routed through Google's internal network infrastruc-  
1217 ture using Private Google Access when available.

1218 **IAM Roles and Access Management**

1219 The principle of least privilege was enforced by assigning minimal permissions to  
1220 users and services through GCP Identity and Access Management (IAM). Access  
1221 decisions involve authentication, authorization checks, and enforcement of policies  
1222 through centralized services, helping reduce the risk of unauthorized actions or  
1223 privilege escalation.

**1224 Encryption**

1225 GCP ensures that data is encrypted both at rest and in transit by default. Storage  
1226 systems use multiple layers of encryption, with cryptographic keys managed by  
1227 Google. For additional control, Cloud Key Management Service (KMS) enables  
1228 customers to manage their own encryption keys. Data in transit between services  
1229 is protected using Application Layer Transport Security (ALTS), and all external  
1230 communication with Google services is encrypted via TLS.

**1231 Access Control**

1232 Access to virtual machines and services was restricted using secure access methods.  
1233 Identity-Aware Proxy (IAP) or VPN connections were employed to safeguard SSH  
1234 and administrative access. GCP's zero-trust model emphasizes verifying identity  
1235 and device security rather than relying solely on network location, aligning with  
1236 best practices for modern infrastructure protection.

**1237 Infrastructure and Operational Security**

1238 GCP's underlying infrastructure benefits from Google's proprietary hardware de-  
1239 signs, including the Titan security chip, secure boot mechanisms, and service iden-  
1240 tity enforcement. Google's physical data centers use multi-layered defenses such  
1241 as biometrics and intrusion detection systems. Operational security practices, in-  
1242 cluding binary authorization and extensive monitoring, reduce insider risks and  
1243 enforce software integrity throughout the lifecycle.

<sub>1244</sub> By deploying the blockchain network on GCP, the project leveraged a highly se-  
<sub>1245</sub> cure environment, benefiting from Google's layered security architecture across  
<sub>1246</sub> networking, identity management, encryption, access control, and operational  
<sub>1247</sub> practices.

### <sub>1248</sub> 4.3.5 Threat Model and Mitigations

<sub>1249</sub> Potential threats to the system were identified and mitigation strategies were applied, as summarized in Table 4.1.

| Threat                          | Mitigation                                    |
|---------------------------------|-----------------------------------------------|
| Unauthorized access to network  | Use of MSP and Certificate Authorities        |
| Tampering with transactions     | Endorsement policies and consensus mechanisms |
| Denial of Service (DoS) attacks | GCP Firewall and rate limiting rules          |
| Data leakage                    | Private channels and access controls          |

Table 4.1: Potential Threat and Mitigation

<sub>1250</sub>

## <sub>1251</sub> 4.4 Mockups

<sub>1252</sub> The mockups represent the preliminary design for the SeaXChange web appli-  
<sub>1253</sub> cation, created using Figma to facilitate collaboration and incorporate feedback  
<sub>1254</sub> efficiently (see Figures showing initial mockup designs). The visual design features  
<sub>1255</sub> a teal-based color scheme to evoke an oceanic theme, aligning with the app's focus  
<sub>1256</sub> on tuna products supply chain. This aesthetic choice reinforces the app's identity  
<sub>1257</sub> and enhances user engagement.

<sub>1258</sub> Upon launching the app, users are first directed to the Login or Sign-Up page,

1259 where authentication is required to access any data. This ensures security and  
1260 role-specific access within the blockchain system.

1261 Once authenticated, users are redirected to their personalized homepages, which  
1262 include their profile information and a dashboard showing relevant tuna product  
1263 assets. The user experience is role-dependent:

- 1264 • All users can view available assets on the blockchain.
- 1265 • Fishermen are the only users who can create new assets, representing newly  
1266 caught tuna.
- 1267 • Suppliers and Retailers have the ability to pass on assets down the supply  
1268 chain, updating the product's status, location, or freshness.

1269 This role-based structure ensures traceability, accountability, and clarity across  
1270 the supply chain, while maintaining a clean and intuitive interface tailored to  
1271 each user type.

#### 4.4. MOCKUPS

57

**Login Page**

SeaXChange

Email: \_\_\_\_\_  
Password: \_\_\_\_\_  
**Log In**

Don't have an account? Sign up

**Fisher Homepage**

SeaXChange

Logout Tuna ID Profile Sign Out

+ ADD CATCH

Picture

TUNA1 Dec 1, 2024 Available

Picture

TUNA1 Dec 1, 2024 Available

Picture

TUNA1 Dec 1, 2024 Sold

Picture

TUNA1 Dec 1, 2024 Available

Picture

TUNA1 Dec 1, 2024 Available

Picture

TUNA1 Dec 1, 2024 Available

**Fisher Add Catch Page**

SeaXChange

TUNA1

Species: Skipjack ✓  
Weight (kg): 5.5 ✓  
Catch Location: Africa ✓  
Catch Date: 2024-12-01 ✓  
Fishing Method: Longline ✓  
Vessel: Japene ✓  
Supplier: NA  
Retailer: NA  
Consumer: NA

**Fisher Add Catch Page 2**

SeaXChange

TUNA1

Species: Skipjack ✓  
Weight (kg): 5.5 ✓  
Catch Location: Africa ✓  
Catch Date: 2024-12-01 ✓  
Fishing Method: Longline ✓  
Vessel: Japene ✓  
Supplier: NA  
Retailer: NA  
Consumer: NA

You won't be able to sell the tuna selected for it later.

**Supplier Homepage**

SeaXChange

Logout Tuna ID Profile Sign Out

Enter Tuna ID:

Picture

TUNA1 Dec 1, 2024 Available

**Supplier Search/Click Result Page**

SeaXChange

TUNA1

Species: Skipjack  
Weight (kg): 5.5  
Catch Location: Africa  
Catch Date: 2024-12-01  
Fishing Method: Longline  
Vessel: Japene  
Supplier: John Doe ✓  
Retailer: NA  
Consumer: NA

**Retailer Search/Sell Page**

SeaXChange

TUNA1

Species: Skipjack  
Weight (kg): 5.5  
Catch Location: Africa  
Catch Date: 2024-12-01  
Fishing Method: Longline  
Vessel: Japene  
Supplier: Uncle Bob  
Retailer: Once Upon A Time  
Consumer: NA

SEND TUNA MARK AS SOLD

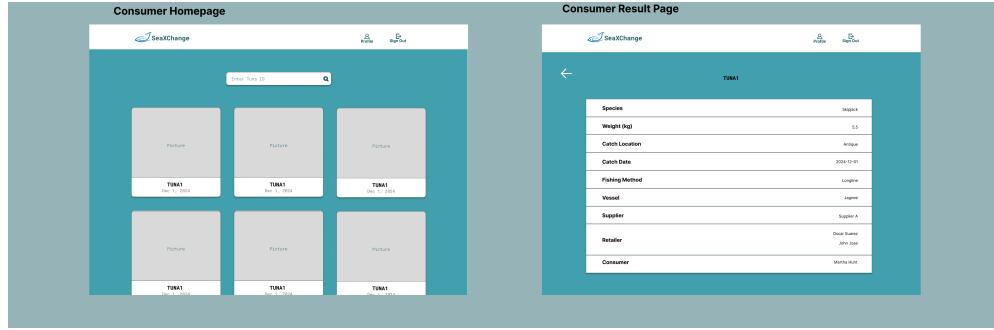


Figure 4.11: SeaXChange Mockups showing the Authentication Page, Role-Based Homepage, Asset Transfer Interfaces for Fishermen, Suppliers, and Retailers, and Asset Viewing page for the Consumers

## <sup>1272</sup> 4.5 Operational Flow of the Web Application

<sup>1273</sup> This section will discuss the flow in using the SeaXChange Web Application. It  
<sup>1274</sup> will show the respective interface for every page and how the users can interact  
<sup>1275</sup> with it.

### <sup>1276</sup> 4.5.1 Landing Page

<sup>1277</sup> Users are be greeted with the landing page (Figure 4.12), where it shows a ocean  
<sup>1278</sup> visuals and a tagline “Discover the Journey your tuna made from the ocean to  
<sup>1279</sup> your dinner plate”. Users are given the option to Login, where they are redirected  
<sup>1280</sup> to the login page or Get Started, where they are redirected to the sign up page.



Figure 4.12: Landing Page

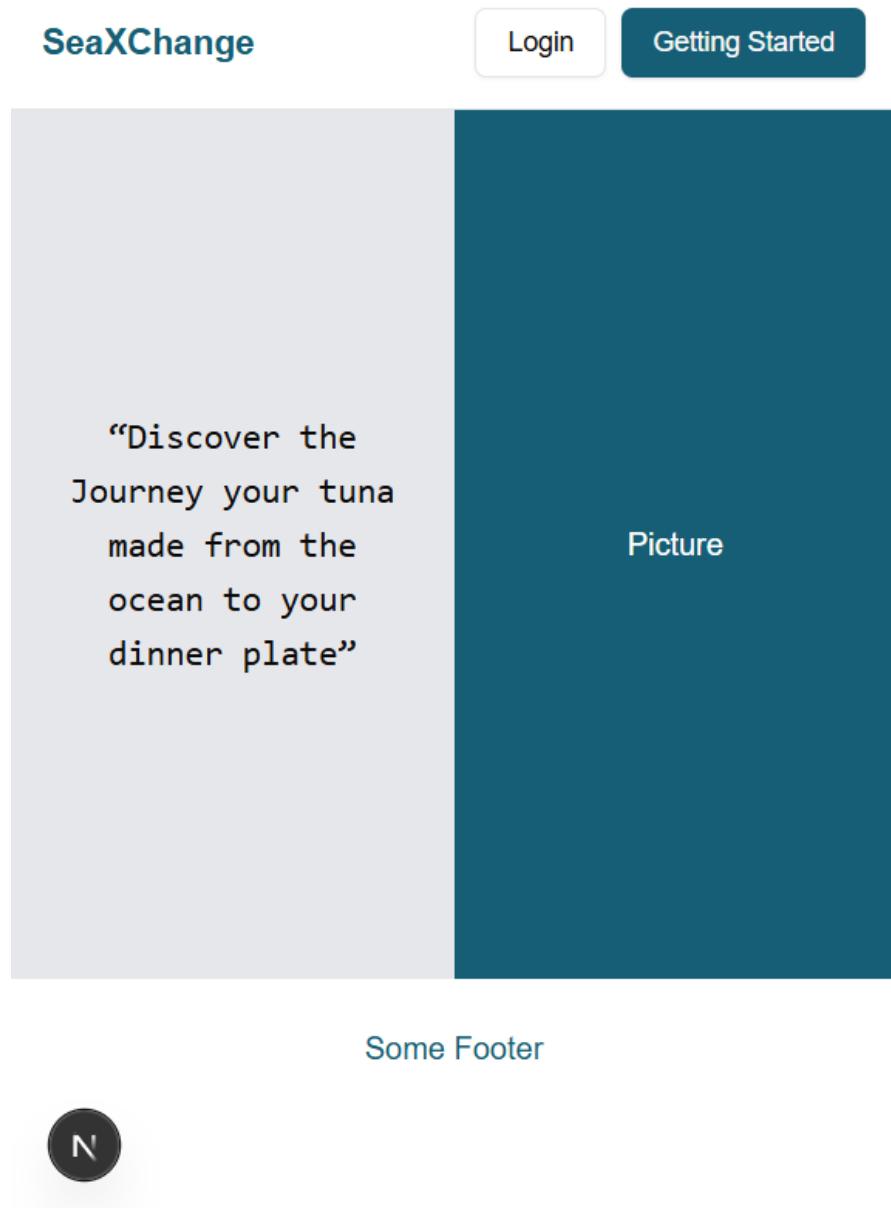


Figure 4.13: Mobile View: Landing Page

1281 **4.5.2 Sign Up Page**

1282 First time users will be required to create an account (Figure 4.14). They are to  
1283 provide their name, email and password. For their user type, there are four roles  
1284 to choose from: Fisher, Supplier, Retailer and Consumer.

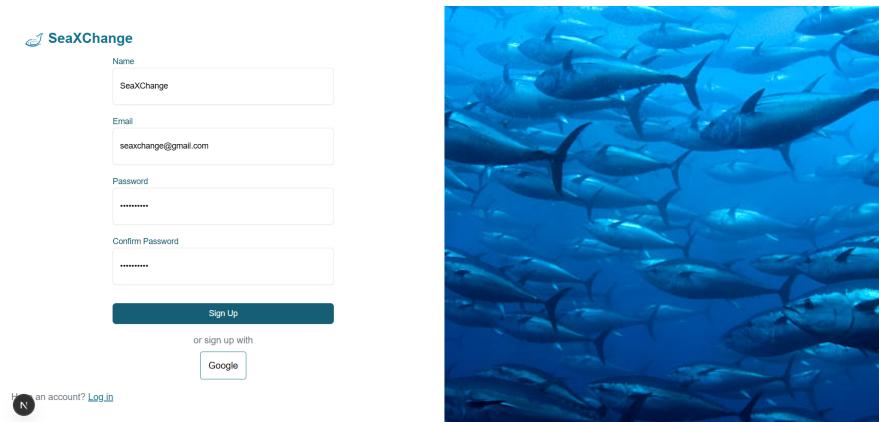


Figure 4.14: SignUp Page

1285 **4.5.3 Login Page**

1286 Returning users are required to login with their email and password (Figure 4.15).

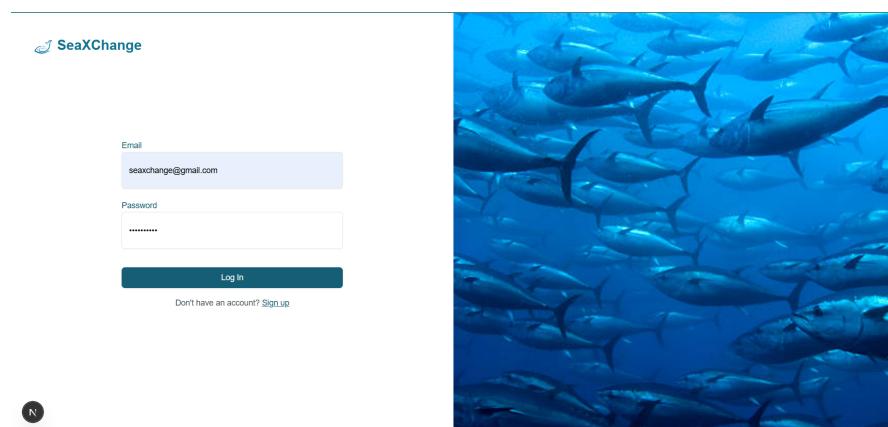


Figure 4.15: LogIn Page

#### <sup>1287</sup> 4.5.4 Homepage

<sup>1288</sup> Each user type has their own respective homepages and features, as shown in the  
<sup>1289</sup> following interface designs.

- <sup>1290</sup> • **Fisher** Fishers can add a fish catch using the "Add catch" button (Figure  
<sup>1291</sup> 4.16), where they are to input the species of the fish, weight in kg, catch  
<sup>1292</sup> location, catch date, fishing method used and vessel name. The remaining  
<sup>1293</sup> text fields like the Supplier name, Retailer name and Consumer name are  
<sup>1294</sup> left null and cannot be edited because they will be filled out by the other  
<sup>1295</sup> users receiving the tuna asset. Users can send a tuna asset to the Supplier.  
<sup>1296</sup> Users can also browse existing tuna assets.

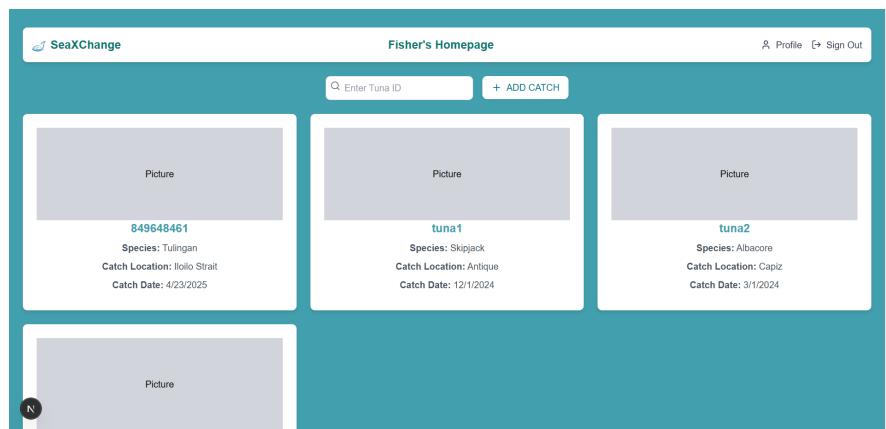


Figure 4.16: Fisher Homepage

- <sup>1297</sup> • **Supplier** Suppliers can browse existing tuna assets (Figure 4.17). Upon  
<sup>1298</sup> clicking a tuna asset, the user can only edit the Supplier text field. They  
<sup>1299</sup> can send the tuna asset to the Retailer.

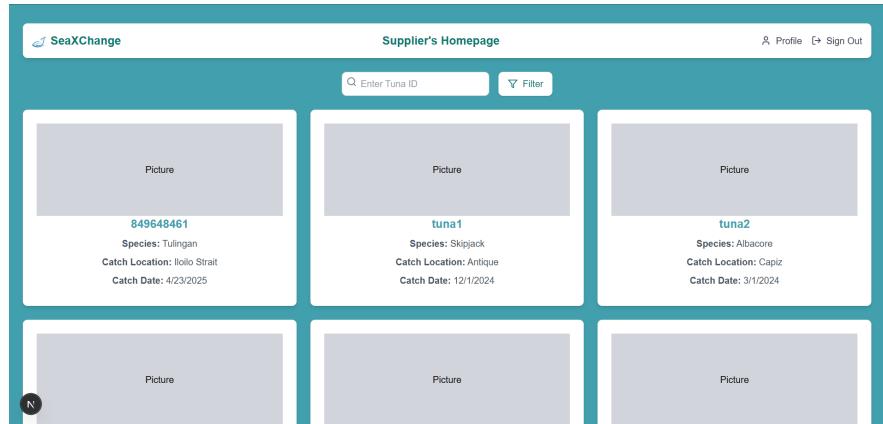


Figure 4.17: Supplier Homepage

- 1300 • **Retailer** Retailers can browse existing tuna assets and can send it to the  
1301 Consumer (Figure 4.18).

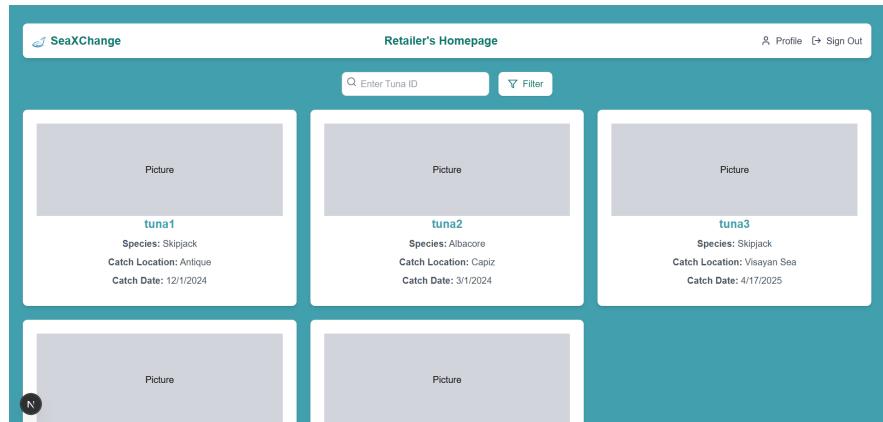


Figure 4.18: Retailer Homepage

- 1302 • **Consumer** Consumers can only view the tuna asset and cannot edit anything  
1303 else (Figure 4.19).

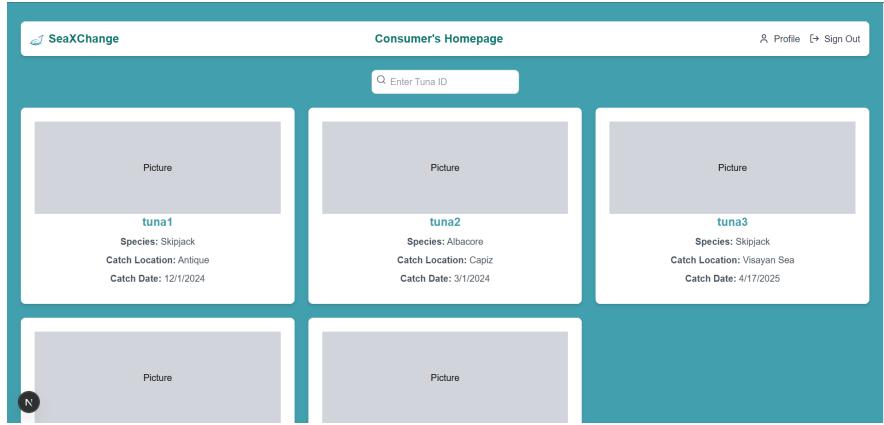


Figure 4.19: Consumer Homepage

### <sup>1304</sup> 4.5.5 Profile

<sup>1305</sup> The user's profile information is shown on the homepage through a pop-up (Figure  
<sup>1306</sup> 4.20). It shows the user's name and role.

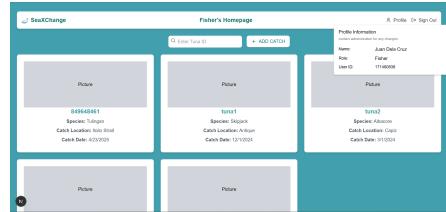


Figure 4.20: View Profile

### <sup>1307</sup> 4.5.6 Logout

<sup>1308</sup> Users can logout of their accounts and is redirected to the Signup Page (Figure  
<sup>1309</sup> 4.21).

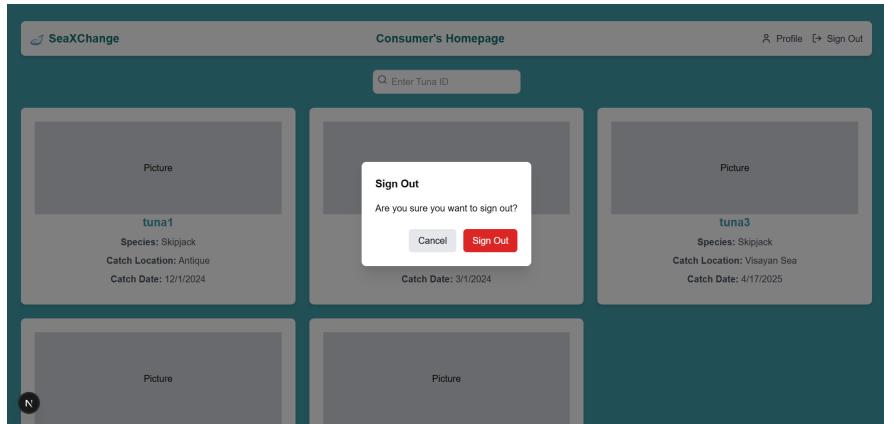


Figure 4.21: Log Out

## 1310 4.6 System Discussion

1311 After modifying the Hyperledger Fabric smart contract to assess necessary pro-  
1312 cesses involved in the tuna supply chain, the blockchain is ready to be invoked  
1313 wherein the smart contract can be activated. To start, a new tuna asset is added  
1314 and registered to the blockchain. Each tuna asset has its attributes or details.  
1315 Before proceeding to the transfer of tuna asset, the smart contract is queried to  
1316 verify if the creation of the asset is successful and if it is part of the current in-  
1317 ventory. After that, the tuna asset can be transferred from fisher to supplier and  
1318 the asset's owner is updated. The smart contract is queried again to verify if the  
1319 asset details have been updated successfully. With the same process, the tuna as-  
1320 set is transferred from supplier to retailer using the smart contract and the owner  
1321 is updated again. To ensure that the asset details are successfully updated, the  
1322 smart contract is queried again. The final step is to query the smart contract to  
1323 show the overview of all the assets in the supply chain. With this, it can be seen  
1324 all the tuna assets from fishing to retail. Overall, the steps and process provides

<sub>1325</sub> transparency and traceability in the tuna supply chain.

## <sub>1326</sub> 4.7 User Demonstration and Feedback Results

### <sub>1327</sub> 4.7.1 Demo Setup and Scenario

<sub>1328</sub> During the demonstration of the system, the participants had a brief introduction  
<sub>1329</sub> of the key functionalities of the SeaXChange app. They were shown how to  
<sub>1330</sub> create an account, input and send tuna assets from one stakeholder to another.

<sub>1331</sub> The demonstration included the asset creation process (Figure 4.22), saving asset  
<sub>1332</sub> details (Figure 4.23), and transferring assets between stakeholders (Figure 4.24).

<sub>1333</sub> Participants were also shown how real-time updates were reflected on the app.

<sub>1334</sub> Finally, they were introduced on how to view transaction histories and traceability  
<sub>1335</sub> information on each tuna asset. Throughout the demonstration, participants

<sub>1336</sub> were encouraged to ask questions and provide feedback on the usability and functionality  
<sub>1337</sub> of the system. After the demonstration, they were given feedback forms

<sub>1338</sub> in order to assess the SeaXChange app.

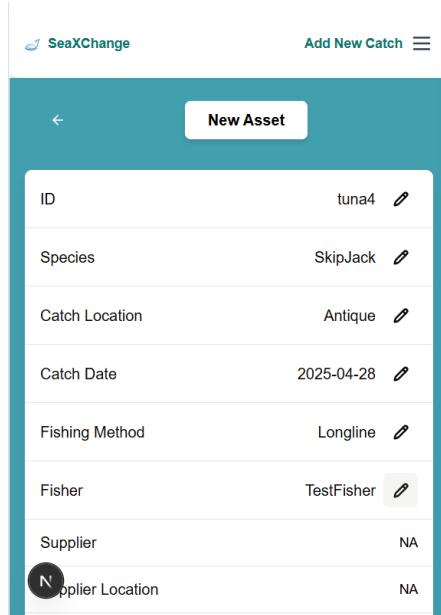


Figure 4.22: Add Catch (Asset)

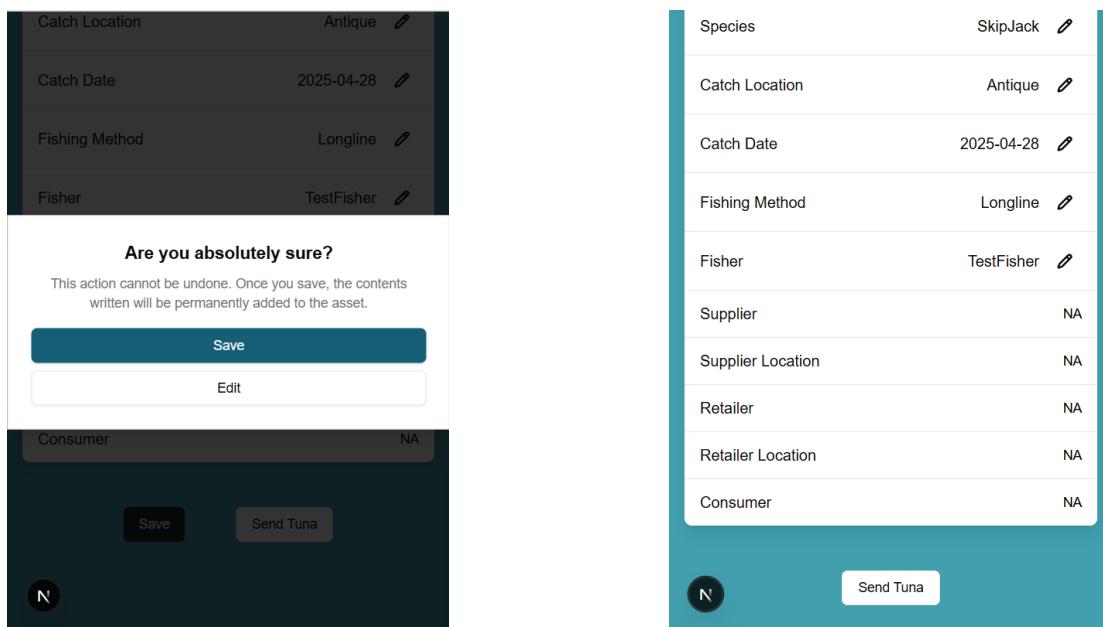


Figure 4.23: Save Details

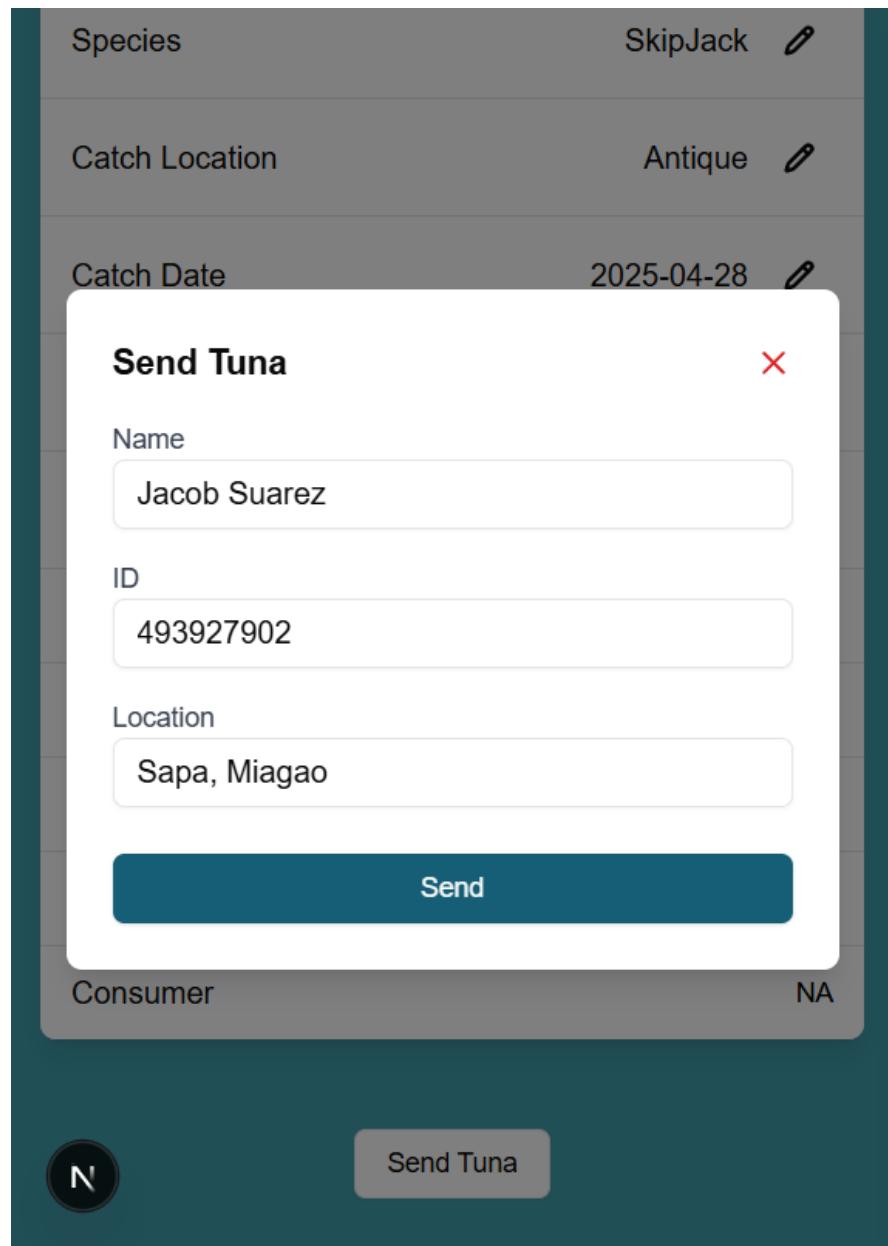


Figure 4.24: Send Asset

<sup>1339</sup> **4.7.2 Summarized Feedback**

| Functionality                   | Stakeholder            | Mean | Description |
|---------------------------------|------------------------|------|-------------|
| Track assets                    | Entire Group           | 3.67 | Average     |
|                                 | Fishermen              | 4.0  | Good        |
|                                 | Supplier and Retailers | 3.0  | Average     |
|                                 | Consumers              | 4.0  | Good        |
| Verify tuna assets              | Entire Group           | 3.67 | Average     |
|                                 | Fishermen              | 3.33 | Average     |
|                                 | Supplier and Retailers | 4.0  | Good        |
|                                 | Consumers              | 3.67 | Average     |
| Support real-time updates       | Entire Group           | 3.56 | Average     |
|                                 | Fishermen              | 3.78 | Average     |
|                                 | Supplier and Retailers | 4.0  | Good        |
|                                 | Consumers              | 4.0  | Good        |
| Enable smart contract execution | Entire Group           | 3.42 | Average     |
|                                 | Fishermen              | 2.33 | Fair        |
|                                 | Supplier and Retailers | 3.25 | Average     |
|                                 | Consumers              | 4.67 | Good        |

Table 4.2: Mean ratings and descriptions for functionality-related features per stakeholder group.

<sup>1340</sup> When taken as a whole, the respondents have average feedback in asset tracking  
<sup>1341</sup> but when classified by stakeholder, the fishermen ( $M = 4.0$ ) and consumers ( $M$   
<sup>1342</sup>  $= 4.0$ ) had good feedback in tracking , while the supplier and retailers have an  
<sup>1343</sup> average rating ( $M = 3.0$ ). For verifying tuna assets, the entire group has an average  
<sup>1344</sup> feedback. When classified by stakeholder, the fishermen ( $M = 3.33$ ) and consumers  
<sup>1345</sup> ( $M = 3.67$ ) have average ratings. For real-time updates, the respondents, when  
<sup>1346</sup> taken as a whole, have an average feedback. When classified by stakeholder, the  
<sup>1347</sup> fishermen ( $M = 3.78$ ) have an average rating, while both supplier and retailers ( $M$   
<sup>1348</sup>  $= 4.0$ ) and consumers ( $M = 4.0$ ) have good ratings. For smart contract execution,  
<sup>1349</sup> the respondents, when taken as a whole, also have an average feedback. When  
<sup>1350</sup> classified according to stakeholder, the fishermen have a fair rating ( $M = 2.33$ ),

- <sup>1351</sup> the supplier and retailers have average ratings ( $M = 3.25$ ) and the consumers have  
<sup>1352</sup> good ratings ( $M = 4.67$ ).

| End-user Needs                   | Stakeholder            | Mean | Description |
|----------------------------------|------------------------|------|-------------|
| Provide transparency in tracking | Entire Group           | 3.56 | Average     |
|                                  | Fishermen              | 2.67 | Fair        |
|                                  | Supplier and Retailers | 4.0  | Good        |
|                                  | Consumers              | 4.0  | Good        |
| Provide seamless interaction     | Entire Group           | 3.77 | Average     |
|                                  | Fishermen              | 1.33 | Poor        |
|                                  | Supplier and Retailers | 3.0  | Average     |
|                                  | Consumers              | 4.0  | Good        |

Table 4.3: Mean ratings and descriptions for end-user needs-related features per stakeholder group.

- <sup>1353</sup> The respondents, when taken as a whole, had an average feedback in transparency.  
<sup>1354</sup> When classified by stakeholder, The fishermen have fair ratings ( $M = 2.67$ ), while  
<sup>1355</sup> both supplier and retailers ( $M = 4.0$ ) and consumers ( $M = 4.0$ ) have good ratings.  
<sup>1356</sup> In evaluating the seamless interaction of the app, the entire group has an average  
<sup>1357</sup> feedback ( $M = 3.77$ ). When classified by stakeholder, the fishermen ( $M = 1.33$ )  
<sup>1358</sup> have poor feedback, the supplier and retailers have average feedback ( $M = 3.0$ )  
<sup>1359</sup> and the consumers have good feedback ( $M = 4.0$ ) in seamless interaction.

| Performance                         | Stakeholder            | Mean | Description |
|-------------------------------------|------------------------|------|-------------|
| Processes transactions efficiently  | Entire Group           | 3.81 | Average     |
|                                     | Fishermen              | 3.67 | Average     |
|                                     | Supplier and Retailers | 3.75 | Average     |
|                                     | Consumers              | 4.0  | Good        |
| Ensures data integrity and security | Entire Group           | 3.31 | Average     |
|                                     | Fishermen              | 2.67 | Fair        |
|                                     | Supplier and Retailers | 3.25 | Average     |
|                                     | Consumers              | 4.0  | Good        |

Table 4.4: Mean ratings and descriptions for performance-related features per stakeholder group.

1360 As a whole, the respondents have an average feedback on efficient transactions ( $M$   
 1361 = 3.81). If evaluated per stakeholder, both fishermen ( $M = 3.67$ ), supplier and  
 1362 retailers ( $M = 3.75$ ) evaluated average while consumers had good feedback ( $M =$   
 1363 4.0). For data security, the entire group has an average feedback ( $M = 3.31$ ). The  
 1364 fishermen have fair evaluation ( $M = 2.67$ ), supplier and retailers ( $M = 3.25$ ) have  
 1365 an average and consumers have solid scores ( $M = 4.0$ ).

| Usability                                      | Stakeholder            | Mean | Description |
|------------------------------------------------|------------------------|------|-------------|
| Provides intuitive interface                   | Entire Group           | 3.83 | Average     |
|                                                | Fishermen              | 4.0  | Good        |
|                                                | Supplier and Retailers | 3.5  | Average     |
|                                                | Consumers              | 4.0  | Good        |
| Allows cross-platform accessibility            | Entire Group           | 4.14 | Good        |
|                                                | Fishermen              | 4.0  | Good        |
|                                                | Supplier and Retailers | 3.75 | Average     |
|                                                | Consumers              | 4.67 | Good        |
| Clear, structured, and visually appealing info | Entire Group           | 3.80 | Average     |
|                                                | Fishermen              | 3.33 | Average     |
|                                                | Supplier and Retailers | 3.75 | Average     |
|                                                | Consumers              | 4.33 | Good        |

Table 4.5: Mean ratings and descriptions for usability-related features per stakeholder group.

1366 It shows the frequency of intuitive interface among the respondents when taken  
 1367 as a whole is average ( $M = 3.83$ ). When classified according to stakeholder,  
 1368 both fishermen ( $M = 4.0$ ) and consumers ( $M = 4.0$ ) have good ratings, while the  
 1369 supplier and retailers ( $M = 3.5$ ) have average ratings. For cross-platform usage,  
 1370 the entire group rated good ( $M = 4.14$ ). When classified according to stakeholder,  
 1371 both fishermen ( $M = 4.0$ ) and consumers ( $M = 4.1$ ) also have good ratings, while  
 1372 supplier and retailers ( $M = 3.75$ ) have average. For visual clarity, the entire group  
 1373 rated average ( $M = 3.80$ ). When classified by each stakeholder, both fishermen  
 1374 ( $M = 3.33$ ) and supplier and retailers ( $M = 3.75$ ) have average ratings, while

<sup>1375</sup> consumers ( $M = 4.33$ ) have good ratings.

| Ease of Use                      | Stakeholder            | Mean | Description |
|----------------------------------|------------------------|------|-------------|
| Clear instructions for new users | Entire Group           | 3.89 | Average     |
|                                  | Fishermen              | 4.0  | Good        |
|                                  | Supplier and Retailers | 4.0  | Good        |
|                                  | Consumers              | 3.67 | Average     |
| Uses clear and simple language   | Entire Group           | 3.31 | Average     |
|                                  | Fishermen              | 4.03 | Good        |
|                                  | Supplier and Retailers | 3.75 | Average     |
|                                  | Consumers              | 4.33 | Good        |

Table 4.6: Mean ratings and descriptions for ease of use-related features per stakeholder group.

<sup>1376</sup> When taken as a whole, the respondents ( $M = 3.89$ ) rated instruction clarity as  
<sup>1377</sup> average. When classified by stakeholder, both fishermen ( $M = 4.0$ ) and supplier  
<sup>1378</sup> and retailers ( $M = 4.0$ ) have good feedback regarding instruction clarity, while the  
<sup>1379</sup> consumers ( $M = 3.67$ ) have average feedback. The entire group rated language  
<sup>1380</sup> clarity as average ( $M = 3.31$ ). When evaluated by each stakeholder, both fisher-  
<sup>1381</sup> men ( $M = 4.03$ ) and consumers ( $M = 4.33$ ) have good feedback, while supplier  
<sup>1382</sup> and retailers ( $M = 3.75$ ) have average feedback.

| Feasibility                         | Stakeholder            | Mean | Description |
|-------------------------------------|------------------------|------|-------------|
| Integration with tuna industry      | Entire Group           | 4.06 | Good        |
|                                     | Fishermen              | 4.0  | Good        |
|                                     | Supplier and Retailers | 4.5  | Good        |
|                                     | Consumers              | 3.67 | Average     |
| Consumer use to track tuna products | Entire Group           | 4.03 | Good        |
|                                     | Fishermen              | 4.0  | Good        |
|                                     | Supplier and Retailers | 3.75 | Average     |
|                                     | Consumers              | 4.33 | Good        |

Table 4.7: Mean ratings and descriptions for feasibility-related features per stakeholder group.

<sup>1383</sup> When taken as a whole, it shows that the respondents have good feedback in the

1384 system integration. When classified by each stakeholder, both the fishermen ( $M =$   
1385 4.0) and supplier and retailers ( $M = 4.5$ ) have good feedback in system integration,  
1386 while the consumers( $M = 3.67$ ) have an average rating. The frequency of consumer  
1387 use among stakeholders, when taken as a whole, have good feedback ( $M = 4.03$ ).  
1388 When analyzed individually, both the fishermen ( $M = 4.0$ ) and consumers ( $M =$   
1389 4.33) have a good rating, while an average rating for the supplier and retailers ( $M$   
1390 = 3.75).

### 1391 4.7.3 Results and Analysis

1392 For functionality, fishermen rated asset tracking positively but identified chal-  
1393 lenges in real-time updates and smart contract execution. Suppliers and retail-  
1394 ers similarly found strengths in verification and real-time updates, although as-  
1395 set tracking averaged slightly lower. Consumers consistently rated functionality  
1396 highly, particularly appreciating the system's ability to support smart contracts.

1397 Regarding end-user needs, fishermen provided the lowest ratings, highlighting  
1398 difficulties with seamless interaction, considering the digital barrier that they ex-  
1399 perience while interacting with the app. Suppliers, retailers, and consumers gave  
1400 higher scores, suggesting that while transparency was generally achieved, the sys-  
1401 tem's ease of interaction required further improvement for all users.

1402 In terms of performance, fishermen and suppliers reported average transaction  
1403 efficiency but noted concerns regarding data integrity and security. Consumers,  
1404 on the other hand, expressed confidence in both areas.

1405 For usability, consumers provided the highest ratings, emphasizing the intuitive

<sup>1406</sup> design and accessibility of the interface. Fishermen and suppliers also rated us-  
<sup>1407</sup> ability positively but suggested improvements, such as better visual appeal and  
<sup>1408</sup> language localization. Feedbacks suggested incorporating the Karay-a language  
<sup>1409</sup> since most of the potential users uses that language in their everyday lives. An-  
<sup>1410</sup> other suggestion was to use capitalization (UI/UX) for the name of the tuna to  
<sup>1411</sup> give more emphasis to it.

<sup>1412</sup> Ease of use was generally rated positively across all groups. However, feedback  
<sup>1413</sup> highlighted the need for additional user guidance features, including clearer button  
<sup>1414</sup> descriptions and localized instructions for broader accessibility.

<sup>1415</sup> Finally, feasibility was strongly affirmed by all groups. Stakeholders believed the  
<sup>1416</sup> system could be effectively integrated into the existing tuna supply chain and  
<sup>1417</sup> accessed by consumers as a means to monitor tuna products.

# <sup>1418</sup> Chapter 5

## <sup>1419</sup> Conclusion

### <sup>1420</sup> 5.1 Overview

<sup>1421</sup> This chapter presents the key findings and conclusion of the study. It also discusses  
<sup>1422</sup> how the system addresses the challenges of traceability within the tuna supply  
<sup>1423</sup> chain. This chapter also provides recommendations in enhancing the system's  
<sup>1424</sup> functionality and usability, ensuring that the SeaXChange app continues to meet  
<sup>1425</sup> evolving needs of its users.

### <sup>1426</sup> 5.2 Conclusion

<sup>1427</sup> This study aimed to develop and evaluate SeaXChange, which is a blockchain-  
<sup>1428</sup> driven app designed to enhance transparency, traceability and accountability  
<sup>1429</sup> within the tuna supply chain. Through the adaption of Scrum, the team was  
<sup>1430</sup> able to develop a functional prototype that was based on iterative development

<sup>1431</sup> to achieve goals.

<sup>1432</sup> The results from the gathered data suggests that the app has effectively addressed  
<sup>1433</sup> key challenges in traceability and accountability, especially among suppliers and  
<sup>1434</sup> consumers. The stakeholders consistently rated the system as good. However,  
<sup>1435</sup> some areas need improvement, especially in ensuring seamless interaction and  
<sup>1436</sup> data security for fishermen.

<sup>1437</sup> Overall, SeaXChange demonstrates strong potential as a technological solution for  
<sup>1438</sup> promoting transparency, combating illegal, unreported, and unregulated (IUU)  
<sup>1439</sup> fishing, and empowering stakeholders across the tuna supply chain.

### <sup>1440</sup> 5.3 Recommendations

<sup>1441</sup> After analyzing and interpreting the gathered data, the researchers had identi-  
<sup>1442</sup> fied some improvements and recommendations for the further development and  
<sup>1443</sup> implementation of the SeaXChange app.

#### <sup>1444</sup> 1. Incorporation of Local Language

<sup>1445</sup> Since most of the target users are within Miagao, the system could provide  
<sup>1446</sup> multi-language support, including the Karay-a language. This will improve  
<sup>1447</sup> guidance through clearer button descriptions.

#### <sup>1448</sup> 2. Utilization of IoT

<sup>1449</sup> The system could use Internet of Things (IoT) in verifying the fisherman's lo-  
<sup>1450</sup> cation. This will add more accountability in tracing the fisherman's current  
<sup>1451</sup> location. Suitable IoT devices include temperature sensors (like DS18B20,

1452 DS1922L) to monitor the fish's temperature during transportation and stor-  
1453 age and GPS chips to track the location of fishing boats and transport  
1454 vehicles.

1455 **3. Inclusion of User Manual**

1456 To further enhance the experience of its users, the researcher could provide  
1457 printed or digital copies on the system's functionalities. This will help users  
1458 navigate through the system without being lost.

1459 **5.4 Summary**

1460 In conclusion, the development of the SeaXChange app highlights the critical role  
1461 of emerging technologies in providing solution regarding the traceability, trans-  
1462 parency and accountability within the tuna supply chain. While the system has  
1463 demonstrated strong potential, continuous improvements are still needed to en-  
1464 sure its effectiveness. Moreover, further development and usability enhancements  
1465 will be essential in attaining SeaXChange's goal of creating a more traceable,  
1466 transparent and accountable tuna industry.



# <sup>1467</sup> Chapter 6

## <sup>1468</sup> References

- <sup>1469</sup> Adhikary, R., Kar, S., Faruk, A., Hossain, A., Bhuiyan, M., & Abdulla-Al-Asif. (2018). Contribution of aquaculture on livelihood development of fish farmer at noakhali, bangladesh. *Asian-Australasian Journal of Bioscience and Biotechnology*. Retrieved from [https://www.researchgate.net/publication/327578781\\_Contribution\\_of\\_aquaculture\\_on\\_livelihood\\_development\\_of\\_fish\\_farmer\\_at\\_Noakhali\\_Bangladesh](https://www.researchgate.net/publication/327578781_Contribution_of_aquaculture_on_livelihood_development_of_fish_farmer_at_Noakhali_Bangladesh)
- <sup>1470</sup> <sup>1471</sup> Ahamed, N. N., Karthikeyan, P., Anandaraj, S., & Vignesh, R. (2020). *Sea Food Supply Chain Management Using Blockchain*. <https://ieeexplore.ieee.org/abstract/document/9074473>.
- <sup>1472</sup> <sup>1473</sup> Ali, V., Norman, A. A., & Azzuhri, S. R. B. (2023). *Characteristics of Blockchain and Its Relationship With Trust*. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10041154>.
- <sup>1474</sup> <sup>1481</sup> Ambarsari, K. T., Rahmansyah, I., Abidin, A. M. F., & Putri, A. D. A. (2023). Concept of illegal fishing for indonesian regulations and unclos. *Yuridika*. Retrieved from <https://e-journal.unair.ac.id/YDK/>

- 1484                   article/view/38045
- 1485     Anitha, R., & Sankarasubramanian, R. (2006). Verifiable Encryption of Digital  
 1486                   Signatures Using Elliptic Curve Digital Signature Algorithm and its Imple-  
 1487                   mentation Issues. In *Innovations in information technology: Research and*  
 1488                   *application*. doi: 10.4018/978-1-59904-168-1.ch010
- 1489     Aponte, M., Anastasio, A., Marrone, R., Mercogliano, R., Peruzy, M. F., & Murru,  
 1490                   N. (n.d.). *Impact of Gaseous Ozone Coupled to passive refrigeration system*.
- 1491     Asche, F., Garlock, T. M., Anderson, J. L., Bush, S. R., Smith, M. D., Anderson,  
 1492                   C. M., ... Vannuccini, S. (2018). *Three Pillars of Sustainability in Fisheries*.  
 1493                   <https://www.pnas.org/doi/abs/10.1073/pnas.1807677115>.
- 1494     Cocco, L., & Mannaro, K. (2021). *Blockchain in Agri-Food Traceability Systems: A Model Proposal for a Typical Italian Food Product*. <https://ieeexplore.ieee.org/abstract/document/9425984>.
- 1495     Cole, R., Stevenson, M., & Aitken, J. (2019). *Blockchain Technology: Implications for Operations and Supply Chain Management*. <https://www.emerald.com/insight/content/doi/10.1108/SCM-09-2018-0309/full/html#sec006>.
- 1501     Cordova, R. S., Maata, R. L. R., Epoc, F. J., & Alshar'e, M. (2021). *Chal-  
 1502                   lenges and Opportunities of Using Blockchain in Supply Chain Management*.  
 1503                   <http://www.gbmrrjournal.com/pdf/v13n3/V13N3-18.pdf>.
- 1504     CSCMP. (2024). *CSCMP Supply Chain Management Definitions and Glossary*. [https://cscmp.org/CSCMP/CSCMP/Educate/SCM\\_Definitions\\_and\\_Glossary\\_of\\_Terms.aspx](https://cscmp.org/CSCMP/CSCMP/Educate/SCM_Definitions_and_Glossary_of_Terms.aspx).
- 1507     Delfino, A. N. (2023). *Analysing the Value Chain of Skipjack Tuna (*Katsuwonus pelamis*) in Partido District, Camarines Sur, Philippines*. <https://www.inderscienceonline.com/doi/epdf/10.1504/IJVCM.2023.129271>.

- 1510 Digal, L. N., Placencia, S. G. P., & Balgos, C. Q. (2017). *Market Assessment*  
1511 *on the Incentives and Disincentives for the Adoption of Sustainable Prac-*  
1512 *tices Along the Tuna Value Chain in Region 12, Philippines.* [https://www.science](https://www.sciencedirect.com/science/article/abs/pii/S0308597X17301197)  
1513 [.direct.com/science/article/abs/pii/S0308597X17301197](https://www.sciencedirect.com/science/article/abs/pii/S0308597X17301197).
- 1514 Grantham, A., Pandan, M. R., Roxas, S., & Hitchcock, B. (2022). *Overcoming*  
1515 *Catch Data Collection Challenges and Traceability Implementation Barriers*  
1516 *in a Sustainable, Small-Scale Fishery.* [https://www.mdpi.com/2071-1050/](https://www.mdpi.com/2071-1050/14/3/1179)  
1517 [14/3/1179](https://www.mdpi.com/2071-1050/14/3/1179).
- 1518 Hackius, N., & Petersen, M. (2017). *Blockchain in Logistics and Supply Chain:*  
1519 *Trick or Treat?* [https://www.research](https://www.researchgate.net/publication/318724655_Blockchain_in_Logistics_and_Supply_Chain_Trick_or_Treat)  
1520 [.gate.net/publication/318724655\\_Blockchain\\_in\\_Logistics\\_and\\_Supply\\_Chain\\_Trick\\_or\\_Treat](https://www.researchgate.net/publication/318724655_Blockchain_in_Logistics_and_Supply_Chain_Trick_or_Treat)
- 1521
- 1522 Hoque, M. S., Roy, S., Mukit, S. S., Rahman, M. B., & Akter, S. (2021). Effects of  
1523 pangasius (pangasius hypophthalmus) and skipjack tuna (sarda orientalis)  
1524 mince blend on the quality of fish products: Ways to utilize resources and  
1525 nutrition in bangladesh. *Food Science & Nutrition*. Retrieved from <https://pmc.ncbi.nlm.nih.gov/articles/PMC8645732/>
- 1526
- 1527 Hugos, M. H. (2024). *Essentials of Supply Chain Management* (Edi-  
1528 tion Number (5) ed.). Hoboken, New Jersey: Wiley. Re-  
1529 trieved from [https://books.google.com.ph/books?hl=en&lr=&id=zpz0EAAAQBAJ&oi=fnd&pg=PP7&dq=supply+chain+&ots=jAuHDxF99j&sig=z10Tue18LKt13pIQWcr2uZT4pRw&redir\\_esc=y#v=onepage&q=supply%20chain&f=false](https://books.google.com.ph/books?hl=en&lr=&id=zpz0EAAAQBAJ&oi=fnd&pg=PP7&dq=supply+chain+&ots=jAuHDxF99j&sig=z10Tue18LKt13pIQWcr2uZT4pRw&redir_esc=y#v=onepage&q=supply%20chain&f=false)
- 1530
- 1531
- 1532
- 1533 Hyperledger Foundation. (2024). Hyperledger Fabric Documentation  
1534 [Computer software manual]. Retrieved from <https://hyperledger-fabric.readthedocs.io/> (Available at <https://hyperledger-fabric.readthedocs.io/>)

- 1536 .readthedocs.io/)
- 1537 Islam, S., & Cullen, J. M. (2021). *Food Traceability: A Generic Theoretical Framework*. <https://doi.org/10.1016/j.foodcont.2020.107848>.
- 1538
- 1539 Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., & Bani-Hani, A.
- 1540 (2021). *Blockchain Smart Contracts: Applications, Challenges, and Future Trends*. <https://doi.org/10.1007/s12083-021-01127-0>.
- 1541
- 1542 Kong, J., Hu, X.-M., Cai, W.-W., Wang, Y.-M., Chi, C.-F., & Wang, B. (2023).
- 1543 Bioactive peptides from skipjack tuna cardiac arterial bulbs (ii): Protective function on uvb-irradiated hacat cells through antioxidant and anti-apoptotic mechanisms. *Marine Drugs*, 21(2). Retrieved from <https://www.mdpi.com/1660-3397/21/2/105>
- 1544
- 1545
- 1546
- 1547 Kresna, B. A., Seminar, K. B., & Marimin, M. (2017). *Developing a Traceability System for Tuna Supply Chains*. <http://ijis-scm.bsne.ch/ojs.excelingtech.co.uk/index.php/IJSCM/article/view/1584/0.html>.
- 1548
- 1549
- 1550 Larissa, S., & Parung, J. (2021). *Designing Supply Chain Models with Blockchain Technology in the Fishing Industry in Indonesia*. <https://iopscience.iop.org/article/10.1088/1757-899X/1072/1/012020>.
- 1551
- 1552
- 1553 Llanto, G. M., Ortiz, M. K. P., & Madriaga, C. A. D. (2018). *The Philippines' Tuna Industry*. [https://www.eria.org/uploads/media/RURB\\_2018\\_FullReport.pdf#page=221](https://www.eria.org/uploads/media/RURB_2018_FullReport.pdf#page=221).
- 1554
- 1555
- 1556 Macusi, E. D., Castro, M. M. C., Nallos, I. M., & Perales, C. P. (2023). *Fishers' Communication as a Critical Factor for Tuna Catches and Potential Benefits of Traceability Draws Small-Scale Fishers to Program*. <https://www.sciencedirect.com/science/article/pii/S0964569123003873>.
- 1557
- 1558
- 1559
- 1560 Macusi, E. D., da Costa-Neves, A. C., Tipudan, C. D., & Babaran, R. P. (2023). *Closed Season and the Distribution of Small-Scale Fisheries Fishing Effort*
- 1561

- 1562           in Davao Gulf, Philippines. <https://www.mdpi.com/2673-4060/4/1/4>.
- 1563   Mercogliano, R., & Santonicola, S. (2019). *Scombroid Fish poisoning: Factors Influencing the Production of Histamine in Tuna Supply Chain. A Review.* <https://www.sciencedirect.com/science/article/pii/S0023643819307169>.
- 1564
- 1565
- 1566
- 1567   Mullon, C., Guillotreau, P., Galbraith, E. D., Fortilus, J., Chaboud, C., Bopp, L., ... Kaplan, D. M. (2017). *Exploring Future Scenarios for the Global Supply Chain of Tuna.* <https://doi.org/10.1016/j.dsr2.2016.08.004>.
- 1568
- 1569
- 1570   Nassar, M., Rottenstreich, O., & Orda, A. (2024, February). Cfto: Communication-Aware Fairness in Blockchain Transaction Ordering. *IEEE Transactions on Network and Service Management*, 21(1), 490–506. doi: 10.1109/TNSM.2023.3298201
- 1571
- 1572
- 1573
- 1574   Nepomuceno, L. T., Bacordo, R. S., Camu, D. G. Y., & Ramiscal, R. V. (2020). *Abundance, Distribution, and Diversity of Tuna Larvae (family Scombridae) in the Philippine Waters.* [https://www.nfrdi.da.gov.ph/tpjf/vol27\\_2/Abundance,%20Distribution,%20and%20Diversity%20of%20Tuna%20Larvae%20\(Family%20Scombridae\)%20in%20the%20Philippine%20waters.pdf](https://www.nfrdi.da.gov.ph/tpjf/vol27_2/Abundance,%20Distribution,%20and%20Diversity%20of%20Tuna%20Larvae%20(Family%20Scombridae)%20in%20the%20Philippine%20waters.pdf).
- 1575
- 1576
- 1577
- 1578
- 1579
- 1580   Nurhayati, A., & Purnomo, A. (2021). Adaptation of west java tuna supply chain in the covid-19 pandemic era. *Economic and Social of Fisheries and Marine Journal*. Retrieved from <https://ecsofim.ub.ac.id/index.php/ecsofim/article/view/339>
- 1581
- 1582
- 1583
- 1584   Obura, D. O., Aeby, G., Amornthammarong, N., Appeltans, W., Bax, N., Bishop, J., ... Wongbusarakum, S. (2019). *Coral Reef Monitoring, Reef Assessment Technologies, and Ecosystem-Based Management.* <https://www.frontiersin.org/journals/marine-science/>
- 1585
- 1586
- 1587

- 1588       articles/10.3389/fmars.2019.00580/full.
- 1589   Pacoma, A. U., & Yap-Dejeto, L. G. (2019). *Health Risk Assessment: Total  
1590       Mercury in Canned Tuna and in Yellowfin and Frigate Tuna Caught  
1591       from Leyte Gulf and Philippine Sea.* [https://www.researchgate.net/publication/340827869\\_Health\\_Risk\\_Assessment\\_Total\\_Mercury\\_in\\_Canned\\_Tuna\\_and\\_in\\_Yellowfin\\_and\\_Frigate\\_Tuna\\_Caught\\_from\\_Leyte\\_Gulf\\_and\\_Philippine\\_Sea](https://www.researchgate.net/publication/340827869_Health_Risk_Assessment_Total_Mercury_in_Canned_Tuna_and_in_Yellowfin_and_Frigate_Tuna_Caught_from_Leyte_Gulf_and_Philippine_Sea).
- 1592  
1593  
1594
- 1595   Paillin, D., Tupan, J., Paillin, J., Latuny, W., & Lawalata, V. (2022).  
1596       *Risk Assessment and Risk Mitigation in a Sustainable Tuna Supply  
1597       Chain.* [https://www.actalogistica.eu/issues/2022/I\\_2022\\_06\\_Paillin\\_Tupan\\_Paillin\\_Latuny\\_Lawalata.pdf](https://www.actalogistica.eu/issues/2022/I_2022_06_Paillin_Tupan_Paillin_Latuny_Lawalata.pdf).
- 1598
- 1599   Parenrenge, S. M., Pujawan, N., Karningsih, P. D., & Engelseth, P. (2016).  
1600       *Mitigating Risk in the Tuna Supply Through Traceability System Develop-  
1601       ment.* [https://himolde.brage.unit.no/himolde-xmlui/bitstream/handle/11250/3052893/Cristin-id\\_1318891\\_engelseth.pdf?sequence=1&isAllowed=y](https://himolde.brage.unit.no/himolde-xmlui/bitstream/handle/11250/3052893/Cristin-id_1318891_engelseth.pdf?sequence=1&isAllowed=y).
- 1602  
1603
- 1604   Peruzy, M. F., Murru, N., Perugini, A. G., Capuano, F., Delibato, E., Mercogliano,  
1605       R., ... Proroga, Y. T. R. (2017). *Evaluation of Virulence Genes in Yersinia  
1606       Enterocolitica Strains Using SYBR Green Real-Time PCR.* <https://www.sciencedirect.com/science/article/pii/S0740002016304555>.
- 1607
- 1608   Pomeroy, R., & Courtney, C. A. (2018). *The Philippines Context for Ma-  
1609       rine Tenure and Small-Scale Fisheries.* <https://sci-hub.st/https://doi.org/10.1016/j.marpol.2018.05.030>.
- 1610
- 1611   Sarmah, S. S. (2018). *Understanding Blockchain Technology.* <http://article.sapub.org/10.5923.j.computer.20180802.02.html>.
- 1612
- 1613   Shamsuzzoha, A., Marttila, J., & Helo, P. (2023). *Blockchain-enabled Traceability*

- 1614        *System for the Sustainable Seafood Industry.* <https://www.tandfonline.com/doi/full/10.1080/09537325.2023.2233632#d1e340>.
- 1615
- 1616        Smithrithee, M., Sato, A., Wanchana, W., Tongdee, N., Sulit, V. T., & Saraphaivanich, K. (2020). *Pushing for the Elimination of IUU Fishing in the Southeast Asian Region.* <http://repository.seafdec.org/handle/20.500.12066/6610>.
- 1617
- 1618
- 1619
- 1620        Snyder, L. V., & Shen, Z.-J. M. (2019). *Fundamentals of Supply Chain Theory* (Edition Number (2) ed.). New York: Wiley. Retrieved from [https://books.google.com.ph/books?hl=en&lr=&id=sJSaDwAAQBAJ&oi=fnd&pg=PR21&dq=supply+chain+&ots=IDNDcy0t37&sig=ssnh-6IDRAi1JzBRpohxT-hiwTE&redir\\_esc=y#v=onepage&q=supply](https://books.google.com.ph/books?hl=en&lr=&id=sJSaDwAAQBAJ&oi=fnd&pg=PR21&dq=supply+chain+&ots=IDNDcy0t37&sig=ssnh-6IDRAi1JzBRpohxT-hiwTE&redir_esc=y#v=onepage&q=supply)
- 1621
- 1622
- 1623
- 1624
- 1625        Stocco, L., Cezarino, L., Liboni, L., & Venkatesh, V. (2022). Wholesaler echelon and industry 4.0 in circular supply chains – a systematic review. *Modern Supply Chain Research and Applications*, 4(2). Retrieved from <https://www.emerald.com/insight/content/doi/10.1108/mscra-10-2021-0019/full/html>
- 1626
- 1627
- 1628
- 1629
- 1630        Tiwari, U. (2020). *Application of Blockchain in Agri-Food Supply Chain.* <https://doi.org/10.33258/bioex.v2i2.233>.
- 1631
- 1632        Verkhivker, Y., & Altmann, E. (2018). *Influence Parameters of Storage on Process of Formation the Histamine in Fish and Fish Products.* <https://www.sciencepublishinggroup.com/article/10.11648/j.wros.20180701.12>.
- 1633
- 1634
- 1635
- 1636        Xu, J., Wang, W., Zeng, Y., Yan, Z., & Li, H. (2022). Raft-plus: Improving Raft by Multi-Policy Based Leader Election with Unprejudiced Sorting. *Symmetry*, 14(6), 1122. Retrieved from <https://doi.org/10.3390/sym14061122>
- 1637
- 1638
- 1639

- 1640 Yang, Y.-C., & Lin, H.-Y. (2017). *Cold Supply Chain of Longline Tuna and*  
1641 *Transport Choice.* <https://www.emerald.com/insight/content/doi/10.1108/mabr-11-2017-0027/full/html>.
- 1642
- 1643 Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). *An Overview*  
1644 *of Blockchain Technology: Architecture, Consensus, and Future Trends.*  
1645 <https://ieeexplore.ieee.org/abstract/document/8029379>.

1646 **Appendix A**

1647 **Code Snippets**

```
1648
1649 const checkAssetAccess = (
1650 asset, userIdentifier, role
1651) => {
1652
1653 switch (role.toLowerCase()) {
1654 case 'fisher':
1655 if (asset.Fisher === userIdentifier)
1656 return { hasAccess: true, accessType
1657 : 'full' };
1658
1659 case 'supplier':
1660 if (asset.Supplier ===
1661 userIdentifier)
1662 return { hasAccess: true, accessType
1663 : 'full' };
1664
1665 case 'retailer':
```

```
1665 if (asset.Retailers?.includes(
1666 userIdentifier))
1667 return { hasAccess: true, accessType:
1668 : 'full' };
1669 break;
1670 case 'consumer':
1671 if (asset.Consumers?.includes(
1672 userIdentifier))
1673 return { hasAccess: true, accessType:
1674 : 'full' };
1675 break;
1676 }
1677
1678 if (role.toLowerCase() === 'consumer') {
1679 return { hasAccess: true, accessType:
1680 : 'readonly' };
1681 }
1682
1683 return { hasAccess: false, accessType: 'none'
1684 };
1685 };
1686 }
```

Listing A.1: Role-based access logic for tuna asset visibility

```
1687
1688 const grpc = require('@grpc/grpc-js');
1689 const { connect, hash, signers } = require(''
1690 @hyperledger/fabric-gateway'');
1691 const crypto = require('node:crypto');
```

```
1692: const fs = require('node:fs/promises');
1693: const path = require('node:path');
1694: const { TextDecoder } = require('node:util');
1695:
1696: const channelName = envOrDefault('CHANNEL_NAME', 'mychannel');
1697:
1698: const chaincodeName = envOrDefault('CHAINCODE_NAME', 'basic');
1699:
1700: const mspId = envOrDefault('MSP_ID', 'Org1MSP');
1701:
1702: // Path to crypto materials.
1703: const cryptoPath = envOrDefault(
1704: 'CRYPTO_PATH',
1705: path.resolve(
1706: __dirname,
1707: '..',
1708: '..',
1709: '..',
1710: 'test-network',
1711: 'organizations',
1712: 'peerOrganizations',
1713: 'org1.example.com'
1714:)
1715:);
1716:
1717: const keyDirectoryPath = envOrDefault(
1718: 'KEY_DIRECTORY_PATH',
```

```
1719) path.resolve(
1720) cryptoPath,
1721) 'users',
1722) 'User1@org1.example.com',
1723) 'msp',
1724) 'keystore'
1725))
1726) ;
1727)
1728) const certDirectoryPath = envOrDefault(
1729) 'CERT_DIRECTORY_PATH',
1730) path.resolve(
1731) cryptoPath,
1732) 'users',
1733) 'User1@org1.example.com',
1734) 'msp',
1735) 'signcerts'
1736))
1737) ;
1738)
1739) const tlsCertPath = envOrDefault(
1740) 'TLS_CERT_PATH',
1741) path.resolve(cryptoPath, 'peers', 'peer0.org1.
1742) example.com', 'tls', 'ca.crt')
1743))
1744)
1745) const peerEndpoint = envOrDefault('PEER_ENDPOINT', '
```

```

1746 localhost:7051');
1747
1748 const peerHostAlias = envOrDefault('PEER_HOST_ALIAS',
1749 'peer0.org1.example.com');
1750
1751 const utf8Decoder = new TextDecoder();
1752
1753 const assetId = asset${String(Date.now())};

```

Listing A.2: Node.js configuration for Hyperledger Fabric Gateway

```

1753
1754 type Asset struct {
1755
1756 ID
1757
1758 Species
1759
1760 Weight
1761
1762 CatchLocation
1763
1764 CatchDate
1765
1766 FishingMethod
1767
1768 Fisher
1769
1770 Supplier
1771
1772 SellingLocationSupplier

```

```

1773: Retailers
1774: [] string `json:"Retailers"
1775: "
1776: SellingLocationRetailers []
1777: string `json:"SellingLocationRetailers"
1778: Consumers
1779: [] string `json:"Consumers"
1780: "
1781: }
1782

```

Listing A.3: Asset Data Structure

```

1783
1784: func (s *SmartContract) CreateAsset(ctx contractapi.
1785: TransactionContextInterface, id string, species
1786: string, weight float64, catchLocation string,
1787: catchDate string, fishingMethod string, fisher
1788: string) error {
1789:
1790: exists, err := s.AssetExists(ctx, id)
1791: if err != nil {
1792:
1793: return err
1794: }
1795:
1796: if exists {
1797:
1798: return fmt.Errorf("the asset %s
1799: already exists", id)
1800:
1801: }
1802:
1803: asset := Asset{
1804:
1805: ID:
1806: id,
1807: Species:
1808: species,
1809: Weight:
1810: weight,
1811: CatchLocation:
1812: catchLocation,
1813: CatchDate:
1814: catchDate,
1815: FishingMethod:
1816: fishingMethod,
1817: Fisher:
1818: fisher,
1819: Status:
1820: "Available",
1821: LastUpdated:
1822: time.Now(),
1823: }
1824:
1825: err = s.insertAsset(ctx, asset)
1826: if err != nil {
1827:
1828: return err
1829: }
1830:
1831: return nil
1832:
1833: }

```

```

1800: Species: species,
1801: Weight: weight,
1802: CatchLocation: catchLocation,
1803: CatchDate: catchDate,
1804: FishingMethod: fishingMethod,
1805: Fisher: fisher,
1806: Retailers: [] string{},
1807: SellingLocationRetailers: []
1808: string{},
1809: Consumers: [] string{},
1810: }
1811: assetJSON, err := json.Marshal(asset)
1812: if err != nil {
1813: return err
1814: }
1815:
1816: return ctx.GetStub().PutState(id, assetJSON)
1817:
1818}

```

Listing A.4: CreateAsset Function

```

1819
1820: func (s *SmartContract) TransferAsset(ctx
1821: contractapi.TransactionContextInterface, id
1822: string, role string, newParticipant string,
1823: newLocation string) (string, error) {
1824: asset, err := s.ReadAsset(ctx, id)
1825: if err != nil {
1826: return "", fmt.Errorf("failed to

```

```
1827 fetch asset: %v", err)
1828 }
1829
1830 switch role{
1831 case "Supplier":
1832 oldSupplier := asset.Supplier
1833 asset.Supplier = newParticipant
1834 asset.SellingLocationSupplier =
1835 newLocation
1836 return oldSupplier, s.SaveAsset(ctx,
1837 id, asset)
1838 case "Retailer":
1839 if !contains(asset.Retailers,
1840 newParticipant){
1841 asset.Retailers = append(
1842 asset.Retailers,
1843 newParticipant)
1844 asset.
1845 SellingLocationRetailers
1846 = append(asset.
1847 SellingLocationRetailers,
1848 newLocation)
1849 }
1850 return "", s.SaveAsset(ctx, id,
1851 asset)
1852 case "Consumer":
1853 if !contains(asset.Consumers,
```

```

1854 newParticipant) {
1855 asset.Consumers = append(
1856 asset.Consumers,
1857 newParticipant)
1858 }
1859 return "", s.SaveAsset(ctx, id,
1860 asset)
1861 default:
1862 return "", fmt.Errorf("invalid role
1863 specified: %s", role)
1864 }
1865 }
1866 }
```

Listing A.5: TransferAsset Function

```

1867
1868 func (s *SmartContract) ReadAsset(ctx contractapi.
1869 TransactionContextInterface, id string) (*Asset,
1870 error) {
1871
1872 assetJSON, err := ctx.GetStub().GetState(id)
1873
1874 if err != nil {
1875
1876 return nil, fmt.Errorf("failed to
1877 read from world state: %v", err)
1878
1879 }
1880
1881 if assetJSON == nil {
1882
1883 return nil, fmt.Errorf("the asset %s
1884 does not exist", id)
1885
1886 }
1887
1888 var asset Asset
```

```

1881) err = json.Unmarshal(assetJSON, &asset)
1882) if err != nil {
1883) return nil, err
1884) }
1885) if asset.Consumers == nil {
1886) asset.Consumers = []string{}
1887) }
1888)
1889) return &asset, nil
1890)
1891}

```

Listing A.6: ReadAsset Function

```

1892)
1893) func (s *SmartContract) GetAllAssets(ctx contractapi.
1894) .TransactionContextInterface) ([]*Asset, error) {
1895) resultsIterator, err := ctx.GetStub().
1896) GetStateByRange("", "")
1897) if err != nil {
1898) return nil, err
1899) }
1900) defer resultsIterator.Close()
1901)
1902) var assets []*Asset
1903) for resultsIterator.HasNext() {
1904) queryResponse, err :=
1905) resultsIterator.Next()
1906) if err != nil {
1907) return nil, err

```

```

1908: }
1909:
1910: var asset Asset
1911: err = json.Unmarshal(queryResponse.
1912: Value, &asset)
1913: if err != nil {
1914: return nil, err
1915: }
1916: if asset.Consumers == nil {
1917: asset.Consumers = []string{}
1918: }
1919: assets = append(assets, &asset)
1920: }
1921:
1922: return assets, nil
1923: }
1924

```

Listing A.7: GetAllAssets Function

```

1925
1926: async function main() {
1927: displayInputParameters();
1928:
1929: const client = await newGrpcConnection();
1930:
1931: const gateway = connect({
1932: client,
1933: identity: await newIdentity(),
1934: signer: await newSigner(),

```

```
1935) hash: hash.sha256 ,
1936) evaluateOptions: () => ({ deadline:
1937) Date.now() + 5000 }),
1938) endorseOptions: () => ({ deadline:
1939) Date.now() + 15000 }),
1940) submitOptions: () => ({ deadline:
1941) Date.now() + 5000 }),
1942) commitStatusOptions: () => ({
1943) deadline: Date.now() + 60000 }),
1944) });
1945)
1946) try {
1947) const network = gateway.getNetwork(
1948) channelId);
1949) const contract = network.getContract
1950) (chaincodeName);
1951)
1952) await initLedger(contract);
1953) await getAllAssets(contract);
1954) await createAsset(contract);
1955) await transferAssetAsync(contract);
1956) await readAssetByID(contract);
1957) await updateNonExistentAsset(
1958) contract);
1959) } finally {
1960) gateway.close();
1961) client.close();
```

```

1962) }
1963) }
1964)
1965) main().catch((error) => {
1966) console.error('** FAILED to run the
1967) application:', error);
1968) process.exitCode = 1;
1969) });
1970)

```

Listing A.8: `main()` Function and Gateway Logic

## 1971 A.1 Hyperledger Fabric Deployment Instructions

1972 The following steps outline the process for deploying and testing the fish supply  
 1973 chain smart contract on Hyperledger Fabric using Google Cloud Platform.

### 1974 A.1.1 Environment Setup

#### 1975 1. Open GCP and access the VM instance:

- 1976     • Navigate to Console → Compute Engine → VM instances → start →  
       1977           click SSH
- 1978     • Alternatively: Virtual Machine → start → instance → SSH

#### 1979 2. Connect to the instance:

Listing A.9: Connect to GCP VM via `gcloud`

```

1981 gcloud compute ssh instance
1982 -20250322-102900 --zone=us-central1-c
1983

```

1984     3. Navigate to the test network directory:

1985           Listing A.10: Navigate to Compose Directory

```

1986 cd ~/fabric-samples/test-network/
1987
1988 compose

```

1989     A.1.2 Network Startup and Smart Contract Deployment

1990     1. Start the Hyperledger Fabric network:

1991           Listing A.11: Start Fabric Network

```

1992 sudo docker-compose -f
1993 compose-test-net.yaml
1994
1995 start

```

1996     • Deploy the chaincode:

1997           Listing A.12: Deploy Chaincode

```

1998 cd ../
1999
2000 ./network.sh deployCC -ccn
2001 basic -ccp ../asset-
2002 transfer-basic/chaincode-
2003 go -ccl go

```

2004     • Set environment path variables:

Listing A.13: Path Environment Variables

```

2005
2006 export PATH=${PWD}/../bin:
2007 $PATH
2008
2009 export FABRIC_CFG_PATH=$PWD
2010 /../config/

```

- **Configure organization environment variables:**

Listing A.14: Org1 Environment Configuration

```

2012
2013 # Environment variables for
2014 Org1
2015 export CORE_PEER_TLS_ENABLED
2016 =true
2017 export CORE_PEER_LOCALMSPID=
2018 "Org1MSP"
2019 export
2020 CORE_PEER_TLS_ROOTCERT_FILE
2021 =${PWD}/organizations/
2022 peerOrganizations/org1.
2023 example.com/peers/peer0.
2024 org1.example.com/tls/ca.
2025 crt
2026 export
2027 CORE_PEER_MSPCONFIGPATH=$
2028 ${PWD}/organizations/
2029 peerOrganizations/org1.
2030 example.com/users/
2031 Admin@org1.example.com/

```

```

2032 msp
2033 export CORE_PEER_ADDRESS=
2034 localhost:7051
2035

```

### 2036 A.1.3 Testing the Smart Contract

#### 2037 1. Initialize the ledger:

Listing A.15: Invoke InitLedger

```

2038
2039 peer chaincode invoke -o localhost:7050
2040 --ordererTLSHostnameOverride orderer
2041 .example.com --tls --cafile "${PWD}/
2042 organizations/ordererOrganizations/
2043 example.com/orderers/orderer.example.
2044 com/msp/tlscacerts/tlsca.example.com-
2045 cert.pem" -C mychannel -n basic --
2046 peerAddresses localhost:7051 --
2047 tlsRootCertFiles "${PWD}/
2048 organizations/peerOrganizations/org1.
2049 example.com/peers/peer0.org1.example.
2050 com/tls/ca.crt" --peerAddresses
2051 localhost:9051 --tlsRootCertFiles "${
2052 PWD}/organizations/peerOrganizations/
2053 org2.example.com/peers/peer0.org2.
2054 example.com/tls/ca.crt" -c '{"
2055 function":"InitLedger","Args":[]}'
2056

```

2057        2. Query assets:

2058                  Listing A.16: Query Fish Asset

```
2059 # Query a specific fish asset
2060 peer chaincode query -C mychannel -n
2061 basic -c '{"Args": ["ReadAsset", "tuna1
2062 "]}]',
2063
2064 # Query all assets in the ledger
2065 peer chaincode query -C mychannel -n
2066 basic -c '{"Args": ["GetAllAssets"]}'
2067
```

2068        3. Shut down the network:

2069                  Listing A.17: Stop Fabric Network

```
2070 sudo docker-compose -f compose-test-net
2071 .yaml stop
2072
```

2073        **A.1.4 Important Notes**

- 2074        • Ensure proper network connectivity when working with Google Cloud Plat-  
2075              form.
- 2076        • The environment variables must be set correctly for the organization context.
- 2077        • TLS certificates are required for secure communication between nodes.
- 2078        • The commands assume the blockchain network is already provisioned on  
2079              GCP.



<sup>2080</sup> **Appendix B**

<sup>2081</sup> **Resource Persons**

<sup>2082</sup> **Dr. Ricardo P. Babaran**

<sup>2083</sup> Professor of Fisheries

<sup>2084</sup> Institute of Marine Fisheries and Oceanology

<sup>2085</sup> University of the Philippines Visayas

<sup>2086</sup> rpbabaran@upv.edu.ph

<sup>2087</sup> **Engr. Noel Lucero**

<sup>2088</sup> Engineer

<sup>2089</sup> Jagnee Fishing Corp.

<sup>2090</sup> JagneeFishingCorp@outlook.com

<sup>2091</sup> **Ms. Veronica Jeruta**

<sup>2092</sup> Barangay Kagawad

<sup>2093</sup> Sapa Barangay Hall

<sup>2094</sup> veronicanave9@gmail.com



<sup>2096</sup> **Appendix C**

<sup>2097</sup> **Interview Request Letter**

<sup>2098</sup> Here is the scanned copy of the letter sent to the interviewees (Figure C.1).

Division of Physical Science and Mathematics  
College of Arts and Sciences  
University of the Philippines Visayas  
Miag-ao, Iloilo

Dear Ma'am/Sir,

Maayong adlaw! We are a group of Computer Science students from the University of the Philippines currently working on our project entitled "SeaXChange: A Blockchain-Driven App for Tuna Supply Chain Management". As part of the development and testing phase, we would like to interview selected fishermen, suppliers, retailers and consumers here in Barangay Mat-Y to gather feedback and evaluate the usability and practicality of our application. The interview will be conducted from April 21-25, 2025.

The goal of our system is to help track tuna products from the ocean to the consumers and to improve the traceability of the tuna supply chain. We believe that the insights from the fishermen, suppliers, retailers and consumers will be extremely valuable in making it more effective and user-friendly.

We respectfully request your permission to lend us your precious time to participate in our short interviews or discussions. Rest assured that all information collected will remain confidential and will be used for academic and research purposes only.

We hope for your favorable response. Thank you for your time and support!

Sincerely,  
The student researchers

  
Jeff Rouzel Bat-og  
Student Researcher

  
Maxinne Gwen Cahilig  
Student Researcher

  
Zyrex Djewel Ganit  
Student Researcher

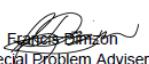
  
Francis Dimzon  
Special Problem Adviser

Figure C.1: Scanned Interview Request Letter

2099 **Appendix D**

2100 **App Demo Documentation**

2101 As shown in Figure D.1, the respondents from Barangay Sapa, Miagao actively en-  
2102 gaged in the app demonstration. Similarly, Figure D.2 illustrates the involvement  
2103 of respondents from Barangay Mat-Y, Miagao during the same activity.

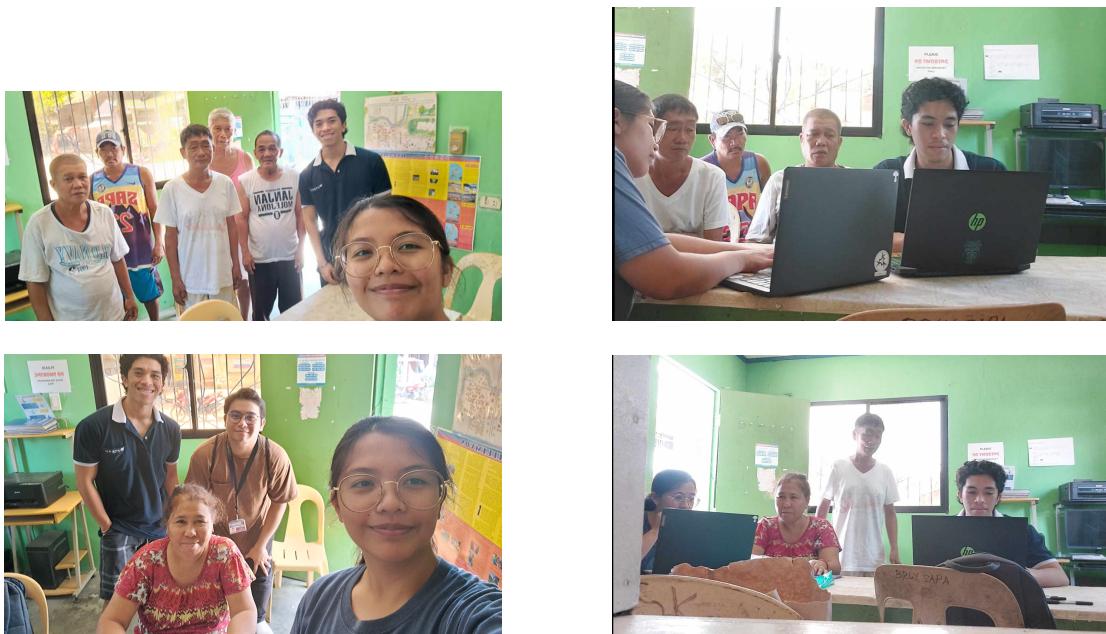


Figure D.1: Respondents (Fishermen, supplier, and retailer) from Barangay Sapa, Miagao



Figure D.2: Respondents (Retailer, and consumer) from Barangay Mat-Y, Miagao