INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

MESTRADO EM ENGENHARIA INFORMÁTICA

Bitcoin em Pagamentos Online: Expandindo a moeda digital em pagamentos digitais

RUTE COSTA SANTOS

junho de 2022

POLITÉCNICO
DO PORTO

Instituto Superior de Engenharia do Porto

Bitcoin in Online Payments

Moving towards digital currency for digital payments

Rute Costa Santos

**A dissertation submitted in partial fulfilment of
the requirements for the degree of Master of Informatics,
Specialisation Area of Software Engineering**

Supervisor: Isabel de Fátima Silva Azevedo

Porto, June 2022

To my mother who has always believed in me and supported my decisions.

Abstract

Technology has evolved exponentially, which has boosted the digitalization of payment systems. Thus, cryptocurrency emerged as a form of digital money and payment, whose adoption has been rising in recent years. On the other hand, the internet and technology also impacted e-commerce, which has potential for growth and development, consequently influencing online payment methods. While cryptocurrencies can attract and retain customers by offering an innovative, low-cost, and efficient payment alternative, there is still some resistance to market entry due to a lack of knowledge and trust.

Aiming toward expanding the knowledge of cryptocurrencies, and sharing how payments with them work, this project seeks to develop a prototype capable of processing online Bitcoin cryptocurrency transactions. This prototype performs the integration with cryptocurrency payment gateways, being easily adapted to be integrated with other applications and support more cryptocurrencies.

The evaluation conducted in this project is based on the Goal/Question/Metric approach, structuring the analysis concerning the quality attributes of Maintainability, Security and Reliability. The results are positive as the solution met all the expected values.

Keywords: Bitcoin, Cryptocurrency, E-commerce, Payment Gateway, Payment.

Resumo

A tecnologia tem evoluído exponencialmente o que potenciou a digitalização os sistemas de pagamento. Assim surgiu a criptomoeda como uma forma de dinheiro digital e pagamento, cuja adoção tem crescido nos últimos anos. Por outro lado, a internet e a tecnologia, têm também um grande impacto no comércio eletrónico que apresenta potencial de crescimento e desenvolvimento tendo assim influenciado os meios de pagamento online. Apesar de as criptomoedas puderem atrair e reter clientes ao oferecer uma alternativa de pagamento inovadora, de baixo custo e eficiente, ainda existe alguma resistência na entrada no mercado devido à falta de conhecimento e confiança.

Tendo como objetivo expandir o conhecimento das criptomoedas, e partilhar como funcionam os pagamentos com estas, este projeto pretende desenvolver um protótipo capaz de suportar a criptomoeda Bitcoin para processamento de transações online. Este protótipo realiza a integração com *gateways* de pagamento de criptomoedas, sendo facilmente adaptado para ser integrado com outras aplicações e suportar mais criptomoedas.

A avaliação realizada neste projeto é baseada na abordagem Goal/Question/Metric, estruturando a análise em relação aos atributos de qualidade de Manutenibilidade, Segurança e Confiabilidade. Os resultados finais são positivos visto que a solução cumpriu todos os valores expectáveis.

Palavras-chave: Bitcoin, Comércio Eletrónico, Criptomoeda, Gateway de Pagamento, Pagamento.

Acknowledgement

Firstly, I must thank with the deepest love my family and boyfriend for all the support, patience and conditions that provided me to achieve my goals.

I'd also like to thank my friends for the help and encouragement to overcome the difficulties faced throughout this stage of my life.

I have to express my sincere gratitude to my advisor Professor Isabel Azevedo, for the guidance and motivation that contributed to the quality of the present document.

Moreover, I want to acknowledge every teacher and colleague who worked with me during my academic journey at ISEP, providing me with the necessary skills to evolve academically and professionally.

On an ending note, I would like to forward my appreciation to all those who, in some way, contributed to making this project come to fruition.

Contents

1 Introduction 1

1.1 Context 1

1.2 Problem 2

1.3 Goal 3

1.4 Contributions 3

1.5 Research Methodology 3

1.6 Document Structure 4

2 Background 7

2.1 Distributed Ledger Technology 7

2.2 Blockchain 8

2.2.1 Types of Blockchain 9

2.2.2 Consensus Algorithms 10

2.3 Cryptocurrencies 11

3 Value Analysis 13

3.1 Value, Value Proposition and Perceived Value Concept Definition 13

3.2 Value Analysis Concept Definition 14

3.3 Problem and Opportunity Identification 14

3.3.1 Opportunity Identification 16

3.3.2 Opportunity Analysis 19

3.3.3 Idea Generation and Enrichment 22

3.3.4 Idea Selection 22

3.3.5 Concept Definition 28

3.4 Value Proposition 28

3.5 Problem Functional Analysis 30

3.6 Value Analysis Conclusion 32

4 State of The Art 33

4.1 E-commerce Platforms 33

4.2 Cryptocurrency Payment Gateways 34

4.3 Cryptocurrency Wallet 42

5 Solution Requirements and Analysis 45

5.1 Objectives 45

5.2 Requirements 45

5.3 Constraints and Concerns 48

5.4 Cryptocurrency Payment Gateways 48

5.5 Decision Criteria 49

5.6 Domain Model........ 50

6 Solution Design........ 53

6.1 Possible Approaches 53

6.1.1 Segregation of responsibilities 53

6.1.2 API Centralization........ 54

6.1.3 Design Solution Evaluation and Choice........ 55

6.2 Detailed Design........ 55

7 Solution Implementation........ 61

7.1 Authentication and Authorization 61

7.2 Exception Handling 63

7.3 Transaction Status 65

7.4 Functional Requirements 65

7.4.1 UC1: List Cryptocurrencies by Fiat currencies 66

7.4.2 UC2: Convert fiat currency to cryptocurrency 66

7.4.3 UC3: Create payment transaction 69

7.4.4 UC4: Process transaction........ 70

7.4.5 UC5: List transaction by identifier........ 73

8 Solution Assessment 75

8.1 Planning 75

8.2 Definition........ 75

8.3 Data Collection........ 76

8.4 Data Analysis and Interpretation........ 78

9 Conclusion 81

9.1 Achieved Objectives........ 81

9.2 Limitations and Future Work 82

9.3 Final Appreciation 83

List of Figures

Figure 1 - Centralized, Decentralized and Distributed Networks 8
Figure 2 - Flow diagram representing the three stages of the FEI process 15
Figure 3 - Representation of the New Concept Development "Wheel" Model........................ 15
Figure 4 - Percentage of market share of cryptocurrencies 17
Figure 5 - Total network volume of Bitcoin and other digital payment networks.................... 18
Figure 6 - Retail e-commerce sales worldwide from 2014 to 2024 in billion US Dollars 19
Figure 7 - Current Market of Cryptocurrency 20
Figure 8 - Percentage of responses saying they trust their local institutions organized by markets 21
Figure 9 - Reasons given for crypto-acceptance and corresponding percentage of responses 21
Figure 10 - Hierarchical Decision Tree describing the connections between each criterion and alternative 23
Figure 11 - Value Proposition Canvas integrated with Business Model Canvas 29
Figure 12 - Value Proposition Canvas of the product being developed 30
Figure 13 - FAST Diagram illustrating the How, Why and When questions.............................. 31
Figure 14 - Functional analysis represented in a FAST diagram... 32
Figure 15 - System functional requirements.. 46
Figure 16 - Decision regarding the cryptocurrencies payment gateway which will process the transaction 50
Figure 17 - Domain model diagram.. 51
Figure 18 - High-level design approach of the system with segregation of responsibilities, presented in a UML Component Diagram 54
Figure 19 - High-level design approach of the system with API centralization 55
Figure 20 - Logical View of component Cryptocurrency Payment API 56
Figure 21 - Process view of use cases UC1, UC2, UC3 and UC4 58
Figure 22 - UML Class Diagram for Transaction Aggregate.. 59
Figure 23 - Transaction states 65
Figure 24 - High-level flow of converting fiat currency to cryptocurrency 69
Figure 25 - Interface IException and its implementation classes), presented in a UML Class Diagram 109
Figure 26 - Flow of converting fiat currency to cryptocurrency (with Interaction Use), presented in a UML Sequence Diagram 111
Figure 27 - Flow of converting fiat currency to cryptocurrency (Interaction Use part), presented in a UML Sequence Diagram 112
Figure 28 - Services created to handle the conversion of fiat currency to cryptocurrency, presented in a UML Class Diagram 113
Figure 29 - Gantt Chart diagram providing an illustrative representation of the project planning phases 116
Figure 30 - Overview of SonarQube analysis results... 117

Figure 31 - Summary of maintainability rating of SonarQube analysis results 118
Figure 32 - Summary of security rating of SonarQube analysis results 119
Figure 33 - Summary of reliability rating of SonarQube analysis results 120
Figure 34 - Overview of Snyk analysis results 121
Figure 35 - Acunetix report with vulnerabilities... 122
Figure 36 - Acunetix report with vulnerabilities specification 122
Figure 37 - Acunetix report without vulnerabilities 123

List of Tables

Table 1 - Overview of blockchain typologies 9
Table 2 - Saaty (Saaty, 1984) fundamental scale 24
Table 3 - Pairwise judgements for criterion 24
Table 4 - Saaty (Saaty, 1984) random judgements indices 25
Table 5 - Alternatives Pairwise Comparison Matrix in the context of Security criteria 27
Table 6 - Alternatives Pairwise Comparison Matrix in the context of Facility to support multiple currencies criteria 27
Table 7 - Alternatives Pairwise Comparison Matrix in the context of Implementation costs criteria 27
Table 8 - Comparison of different cryptocurrency payment gateways regarding some characteristics 40
Table 9 - Comparison of different cryptocurrency payment gateways (with free trial) regarding some characteristics 41
Table 10 - System functional requirements 46
Table 11 - Description of use cases 47
Table 12 - Description of solution’s quality attributes 47
Table 13 - Constraints of the system 48
Table 14 - Concerns 48
Table 15 - Goal/Question/Metric for each Quality Attribute 76
Table 16 - Vulnerabilities found on the Acunetix scan 77
Table 17 - Answers and conclusion of the assessment goals 79
Table 18 - Top websites that list the top cryptocurrency payment gateways by clients 106
Table 19 - Top cryptocurrency payment gateways by clients and their respective occurrence 107

List of Code Snippets

Code Snippet 1 - Function that handles the authentication/authorization of the merchant ... 62
Code Snippet 2 - Example of a function that reads an item from HTTPContext that was added during authentication/authorization of the merchant 63
Code Snippet 3 - Handler class responsible for catching and handling exceptions.................. 64
Code Snippet 4 - Function that gets the list of cryptocurrencies 66
Code Snippet 5 - Function responsible for providing the list of valid and accessible cryptocurrency payment gateways... 67
Code Snippet 6 - Function responsible for finding the first valid response from a list of cryptocurrency payment gateways... 68
Code Snippet 7 - Function responsible for validating the request, converting the currency, building the transaction, storing the transaction on the database, and building the response 68
Code Snippet 8 - Method that validates the request to create payment transaction use case 69
Code Snippet 9 - Function that gets the cryptocurrency payment gateway and calls it to create a transaction 70
Code Snippet 10 - Part of the function that calls COINQVEST to convert cryptocurrency 71
Code Snippet 11 - Part of the function responsible for calling BitPay to create a transaction . 71
Code Snippet 12 - Controller with the Webhooks for each cryptocurrency payment gateway 72
Code Snippet 13 - Function that gets the transaction from the database, validates it and creates the response 73
Code Snippet 14 - Function that validates the request to list a transaction by identifier......... 73

Acronyms

List of Acronyms

AHP	Analytic Hierarchy Process
API	Application Programming Interface
B2B	Business to Business
B2C	Business to Consumer
BMC	Business Model Canvas
C2B	Consumer to Business
C2C	Consumer to Consumer
CSP	Content Security Policy
DB	Database
DeFi	Decentralized Finance
DLT	Distributed Ledger Technology
DSR	Design Science Research
EMV	Europay, Mastercard and Visa
FAST	Function Analysis System Technique
FEI	Front End of Innovation
GQM	Goal/Question/Metric
HSTS	HTTP Strict Transport Security
IaaS	Infrastructure as a Service
ISEP	*Instituto Superior de Engenharia do Porto*
IT	Information Technology
MEI	*Mestrado em Engenharia Informática*
NCD	New Concept Development
NFT	Non-Fungible Token

PaaS	Platform as a Service
PoS	Proof of Stake
PoW	Proof of Work
REST	Representational State Transfer
SaaS	Software as a Service
SEPA	Single Euro Payments Area
SSL	Secure Sockets Layer
TLS	Transport Layer Security
TMDEI	*Tese/Dissertação/Estágio*
UI	User Interface
UML	Unified Modeling Language
URL	Uniform Resource Locator
VA	Value Analysis
XSS	Cross-site scripting

Symbols List

$\overline{X}$	Median

1 Introduction

The present document analyses and exposes approaches for processing payments with cryptocurrencies, namely Bitcoin.

This chapter aims to give an overview of the project, so it begins by presenting the inherent context of the thesis, the problem addressed and the research methodology for carrying out the project. Next, the goal and contributions are described. Finally, the report's structure is defined, where the remaining chapters are succinctly presented.

1.1 Context

The fast globalization and evolution of technology have stimulated remarkable attention for the area of information technology (IT) by today's society and market. The mainstream use of the internet has driven the speedy growth of digitalization and consequently contributed to its adoption in the financial industry (Christine, 2019; Inzirillo and Mat, 2021).

The digitalization of payment systems is accelerating and unstoppable, changing the way services, goods, capital, and assets are handled and exchanged. Thus, and to respond to the financial system’s weakness after the 2008 financial crisis, the need arose to create a form of digital money and financial payment - cryptocurrency.

New forms of digital money are the latest innovation in an evolving landscape for the way payments are made in the economy. They can contribute to faster, cheaper and more efficient payments (Bank of England, 2021).

Since 2008, when the first cryptocurrency was devised, and the concept of blockchain was introduced, the industry and adoption of both cryptocurrencies and blockchain have evolved at high speed in recent years. There are already over 17.000 cryptocurrencies (CoinMarketCap, 2021) and several use case projects of the application of blockchain, for example, secure sharing of medical data, voting mechanisms and supply chain and logistics monitoring (Daley, 2021).

The internet and technology also had and have a great impact on commerce and how it is carried out. Commerce, which means the buying and selling of products or services, became possible through the internet, thus emerging the concept of electronic commerce (e-commerce). E-commerce is carried out worldwide, and new technologies make transactions easier, safer and more sophisticated (Adewole, Saxena and Bhadauria, 2020).

Over the past few decades, e-commerce has expanded globally and continues to show great potential for growth and development. This massive growth has driven various online payment methods, for example, credit/debit card payment, direct debit payment, electronic funds transfer, electronic wallet payment, and smart cards (Adewole, Saxena and Bhadauria, 2020), with recent research indicating that blockchain, and consequently cryptocurrencies, will have a major impact on this industry (Treiblmaier and Sillaber, 2021).

Cryptocurrencies can be a major differentiating factor from competitors, attracting and retaining customers by offering an innovative, low-cost, and efficient payment alternative (Christine, 2019; Jonker, 2019). On the other hand, cryptocurrencies open doors to new online business opportunities in developing countries. This is justified by the fact that access to financial institutions or services in these countries is not easy or too risky, and cryptocurrencies, being a peer-to-peer service, are a safer and viable alternative (Mohamed, Almasri and Lasheen, 2018).

1.2 Problem

Money can be defined as a medium of exchange, further serving as a unit of account and store of value (University of Minnesota, 2016). Cryptocurrencies already adopt the characteristics of money to some extent (Eikmanns and Sandner, 2015):

(1) they are considered a medium of exchange since there are already merchants who accept cryptocurrencies in exchange for their products or services;
(2) cryptocurrencies have to be commonly accepted as a medium of exchange to be used to set prices or make mathematical calculations and consequently be considered a unit of account;
(3) a store of value means that an item holds value over time, this will only be possible through trust in the effectiveness as a medium of exchange and as a store of value.

With this, it is clear that cryptocurrencies have the potential to obey the characteristics of money, and this only depends on their recurrent use and thus be applied in the context for which they were conceived, for payments and not only investment or trading.

As already mentioned, the current usage also comes from the population's trust in this payment method. Trust is achieved through knowledge and understanding of the technology that adheres to cryptocurrencies. Individuals who do not understand the cryptocurrency payment mechanism and its technology are resistant to entering this market due to a lack of trust in the system (Jaag and Bach, 2015).

Additionally, the volatility of cryptocurrencies is also considered a major obstacle by those who do not yet use cryptocurrencies (Neureuter, 2021). This volatility can be influenced by the fact (a) that it is still an emerging market, (b) that it does not depend on third parties meaning that its price is determined by supply and demand, (c) that the technologies are still being developed and improved, (d) that there are speculative bets that increase market uncertainty, (e) that the press affects investments and negotiations, and finally, (f) that it is possible for anyone to be an investor, even those with little experience and knowledge (Khan and Hakami, 2021).

As one might expect, traders may not be willing to take price fluctuation risks. Nevertheless, high volatility may only be a temporary property for cryptocurrencies. The stability of cryptocurrencies is likely to be achieved not only as more companies and customers use them but also as regulatory guidelines become clearer (Jaag and Bach, 2015).

To conclude, the value and stability of cryptocurrencies depend on the trust shown by users and, consequently, on their effective use as a medium of exchange. The problem of trust will not be directly addressed in this thesis, since it is a broad issue. However, trust can be achieved by secure payments and the knowledge acquired on how cryptocurrency payments work.

1.3 Goal

The context of this thesis lies in the exploration activity of an emerging technological area to help merchants integrate Bitcoin cryptocurrency payment into their e-commerce websites.

The main goal is the development of a prototype capable of supporting the cryptocurrency Bitcoin. This prototype must be easily adaptable to be integrated with other applications and support more cryptocurrencies.

1.4 Contributions

This thesis proposes to expand the knowledge of cryptocurrencies and their adherent technology, as well as possible ways to process payments with them, thus contributing to increasing state of the art. On the other hand, this project intends to boost confidence in cryptocurrencies in payments by sharing knowledge and demonstrating how this type of transaction works, eventually contributing to more merchants' adoption of this payment method.

1.5 Research Methodology

This thesis aims to solve a problem by developing a solution. To this end, the Design Science Research (DSR) method was used to guide the process of rigorous research as well as to help in the recognition and legitimization of such research and respect for its objectives, processes and outputs (Peffers *et al.*, 2006). This methodology is divided into six activities, which are then applied to the present research context (Peffers *et al.*, 2007).

Activity 1: Problem identification and motivation, consists of identifying and defining the problem to be solved, justifying the value that the solution will bring, and the reasons associated with solving the problem. In this document, the problem is described in Section 1.2, and the value of the solution is analysed in Chapter 3.

Activity 2: Define the objectives for a solution, as the name indicates, is the phase where the purposes are inferred based on the problem previously identified. The state of the problems and the current solutions are also expected to know herein. The objectives are detailed in Section 5.1 and in Chapter 4 the state of the art is reviewed.

Activity 3: Design and development, this activity includes solution design and implementation. Chapters 6 and 7 cover this activity.

Activity 4: Demonstration, constitutes a set of activities (e.g., experimentation, simulation, case study, etc.) that prove how effective the solution solves the problem. It is available in Chapter 8.

Activity 5: Evaluation, aims to measure the implementation to validate how well it solves the problem by comparing the objectives with the results of the previous activity. Chapter 8 also addresses this activity.

Activity 6: Communication is the last activity and aims to convey the entire research process, from the problem and its relevance to the solution and how it effectively combats the problem and meets the expected value. This activity is detailed in Chapter 9 and the final presentation to the jury and present audience.

1.6 Document Structure

This document is divided into ten chapters, starting with the Introduction, the current chapter, which intends to give an overview of the project in context.

In the second chapter, Background, the key concepts that this thesis addresses are described, explaining the Blockchain and Cryptocurrency technology, as well as its advantages and disadvantages.

Then in chapter three, Value Analysis, the business and research value of this project is described, identifying, and analysing the opportunity, idea generation and selection, and concept definition using the New Concept Development model integrated into the Front End of Innovation process.

The fourth chapter, State of The Art, reviews current technologies for processing cryptocurrency transactions, as well as e-commerce applications that already provide this payment method.

Solution Requirements and Analysis, the fifth chapter, specifies the established requirements, constraints, and concerns of the project. Moreover, the analysis of the realized solution is described, explaining the selection of the cryptocurrency payment gateways to be integrated with the solution and the decision criterion that should be considered. Lastly, it presents the business concepts.

Chapter six, Solution Design, explains the approaches studied to develop the underlying system. It focuses on its architecture and design choices and identifies the key components for the final prototype.

Following seven, Solution Implementation describes the implementation of what was detailed in the application’s requirements, alongside snippets of code to reinforce some features that were implemented.

For Solution Assessment, chapter eight details the validation of the outcome solution according to the established requirements and metrics.

The last chapter, Conclusion, provides an overview of what was discussed in the document, reviewing the achieved solution with the proposed objectives and advising for future works, such as improvements.

2 Background

This chapter entails the introduction of some important concepts, giving a technological overview and establishing an understanding of cryptocurrencies. It defines distributed ledger and blockchain technology, typifies blockchain networks and describes some consensus algorithms used in the blockchain network. Moreover, it is described cryptocurrencies and some related concepts.

2.1 Distributed Ledger Technology

A ledger is a database that records transactions in chronological order. Figure 1 illustrates the three ledger network models (Krause, Natarajan and Gradstein, 2017; Lenz, 2019):

(1) Centralized, where the authority to change the ledger is exclusive to one participant (also referred to as nodes) in the network;
(2) Decentralized, where the authority to change the ledger is limited to several participants in the network; and
(3) Distributed, where the authority to change the ledger is with all participants in the network.

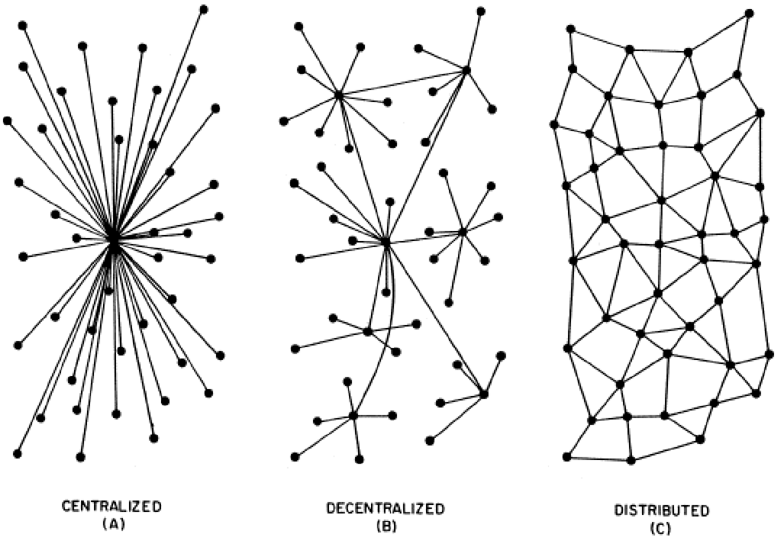


Figure 1 - Centralized, Decentralized and Distributed Networks
(Source: (Baran, 1964))

In a distributed ledger technology (DLT) based infrastructure, the control over the ledge lies with several network nodes, consequently which allow users to interact without necessarily trusting each other or without the need for a trusted third party (Hancock and Vaizey, 2016). DLT generally rely on other technologies to add transparency, traceability, and security to the network. These technologies are (El Ioini and Pahl, 2018; Lenz, 2019):

(1) public-key cryptography, where each node holds a pair of keys (one public, one private), which enforces the ownership's control over the data managed by the ledger;
(2) distributed peer-to-peer networks, where nodes are linked together with equal permissions for processing transactions, avoiding a single point of failure, and preventing a small group take charge of the network;
(3) consensus algorithms permit all nodes to agree on a single version of the truth.

2.2 Blockchain

Blockchain is a peer-to-peer, decentralized and distributed network first introduced in 2008 by Satoshi Nakamoto as part of a proposal for Bitcoin (Nakamoto, 2008; Iansiti and Lakhani, 2017). Bitcoin is the first use case for blockchain, nevertheless, its usage goes beyond the financial application, for instance, decentralized finance (DeFi) applications, non-fungible tokens (NFTs), and smart contracts.

Blockchain is essentially a digital ledger, one type of DTL, that facilitates the process of recording transactions that are grouped into blocks and distributed across the entire network of computer systems. Each block is linked via cryptography to the previous one after being verified by the consensus of most of the participants in the network (Crosby, 2016; Yaga *et al.*, 2018; Gupta, 2020).

Since blockchain is decentralized, in other words, does not need a trusted third party, the necessary trust is ensured through the four characteristics of blockchain technology (Yaga *et al.*, 2018): (1) Ledger, blockchain technology uses an append-only ledger to provide full transactional history, where data can never be changed or erased; (2) Secure, blockchains are cryptographically secure, ensuring that the information contained within the ledger has not been tampered with and is attestable; (3) Shared, the ledger is shared amongst multiple participants across the blockchain network, providing, therefore, transparency within; (4) Distributed, the blockchain can be distributed (where there are several nodes connected without a central node of control (Sheth and Dattani, 2019)), allowing for scaling the number of nodes of a blockchain network to make it more resilient to attacks.

2.2.1 Types of Blockchain

Different types of blockchain can be distinguished to better satisfy the requirements within a project built on the foundation of blockchain. These typologies determine what parts or actions of the blockchain are restricted to its users (Meijer, 2017).

One can characterize these typologies accordingly to two dimensions (Gerth and Heim, 2020): (1) Access, which corresponds to the user rights regarding reading and writing rights and execution of transactions (public vs. private); (2) Validation, which determines the users that have rights to participate in the consensus mechanism (permissionless vs. permissioned). By combining these two dimensions, four blockchain types are recognized and defined (Meijer, 2017; GSMA, 2018; Lenz, 2019; Brown, 2020; Wegrzyn and Wang, 2021), as represented in Table 1 and following described.

Table 1 - Overview of blockchain typologies

		Validation	
		Permissionless	Permissioned
Access	Public	Public permissionless blockchain	Public permissioned blockchain
	Private	Private permissionless blockchain	Private permissioned blockchain

Public permissionless blockchains, usually known as simply public blockchains, allow anyone to join and contribute to the network and there are no restrictions on who can read data. These are designed to be fully decentralized and are primarily used for exchanging and mining cryptocurrency. Bitcoin, Ethereum, and Litecoin are examples of these blockchains.

Private permissioned blockchains, commonly known as private blockchains, are more centralized than public blockchains since it restricts the access for trusted members of the respective consortium and the transactions are only visible to them. These are valuable for internal enterprise solutions within industries that do not want their sensitive business data visible on a public blockchain, for instance, Hyperledger Fabric, and the business-to-business virtual currency exchange network like Bytecoin.

Public permissioned blockchains are sometimes entitled to hybrid blockchains, combining the privacy benefits of a permissioned and private blockchain with the security and transparency benefits of a public blockchain. It means that only authorised entities can join the network, but it provides the flexibility to choose what data to share and be made public, while the remaining is kept private. Dragonchain is a case of this blockchain.

Private permissionless blockchains are currently not used in practice as it provides a combination of properties that do not complement each other. These do not limit who can join the network and therefore participate in the consensus mechanism. However, the transactions are not visible to everyone.

2.2.2 Consensus Algorithms

As mentioned in sections 2.1 and 2.2, a block needs to be verified to be able to be added to the chain. In a more detailed description, any participant within a blockchain network can propose adding more information to the blockchain. However, this must be validated to verify its legitimacy (Houben and Snyers, 2018). In this way, the participants must agree on the validity of new transactions according to a set of rules, being achievable through a consensus mechanism. A consensus mechanism is a predefined specific cryptographic validation method that guarantees the correct sequence and authenticity of transactions on the blockchain, preventing the control of the blockchain by evildoers (especially in cases of permissionless blockchain) (Krause, Natarajan and Gradstein, 2017).

The consensus method is usually specified in the blockchain algorithmic design and can vary depending on the blockchain type, purpose, and asset, each having its advantages and disadvantages.

Consensus methods can be further classified according to the participant reward mechanism (Ferdous *et al.*, 2020): (a) Incentivized Consensus, which are consensus algorithms that reward nodes for creating and adding a block to the blockchain. These mechanisms are commonly used in public permissionless blockchains, thus encouraging the participation of nodes in the transaction validation; (b) Non-incentivised Consensus, which are consensus algorithms that do not reward nodes. They are used in private permissioned blockchains as nodes are considered trusted since only authorized nodes can participate in the consensus method process.

Numerous consensus mechanisms are currently being published, being Proof of Work and Proof of Stake the ones vastly used nowadays.

In the Proof of Work (PoW) mechanism, a node needs to provide a ‘proof’ that it performed some ‘work’. In other words, a node must solve a computationally intensive puzzle to be able to add the next block to the blockchain (Yaga *et al.*, 2018). These puzzles are designed to be hard to solve (in terms of computing power and processing time) but easy to verify (Krause, Natarajan and Gradstein, 2017). So, this makes it difficult to change and add blocks but easily detect any system abuse, preventing malicious users from manipulating the chain. All nodes can try to solve the puzzle, but only the first one to successfully complete it is rewarded with a digital form of value. To increase the chance of creating a new block, many participants aggregate their resources, and the reward is then divided among them. This leads to the centralization of the blockchain, going against the principles of blockchain technology (Ferdous *et al.*, 2020). This mechanism is not recommended for huge and growing networks which have the requirement of a massive number of transactions every second because of the time-consuming involved (Verma, Jain and Doriya, 2021). The cryptocurrencies Bitcoin, Litecoin and Bitcoin Cash are based on a PoW consensus mechanism (Houben and Snyers, 2018).

The Proof of Stake (PoS) mechanism was proposed to counteract the limitations of any PoW algorithm (Ferdous *et al.*, 2020). In a PoS system, a node needs not only to prove ownership of a certain asset but also lock some of it, called a stake, to be able to participate in the block creation process. Thus, this mechanism is based on the idea that the more stake a user has invested into the system, the more likely they want the system to succeed, and less likely to subvert it (Yaga *et al.*, 2018). The number of assets (in the case of cryptocurrencies, a certain number of coins) is the factor to select the node that will add the next block. Consequently, the probability of the selection of a node "is tied to the ratio to their stake to the overall blockchain network amount of staked cryptocurrency" (Yaga *et al.*, 2018). The selected user to validate the transaction is rewarded, either by collecting the transaction fees or receiving a certain amount of assets (coins) (Verma, Jain and Doriya, 2021). On the other hand, it can suffer some type of penalty if caught in trustless transactions (Verma, Jain and Doriya, 2021). Sometimes the selection is not considered justice since it depends on the number of assets a node detent, prioritizing those who have a higher number. The PoS mechanism is presently being used by the cryptocurrencies Neo and Ada (Cardano) (Houben and Snyers, 2018).

2.3 Cryptocurrencies

Cryptocurrency is a digital currency, in other words, a currency that only exists in digital form and is transferred over the internet (CoinDesk, 2022), that uses cryptographic technologies for security (CoinMarketCap Alexandria, 2022b; Dovarganes, 2022). Examples of cryptocurrencies are Bitcoin, Ethereum, Binance Coin and Solana.

Bitcoin, which is the first cryptocurrency, was described in 2008 by Satoshi Nakamoto in response to the financial crises lived around that year (Nakamoto, 2008). The cryptocurrencies that appeared after are entitled to altcoins, in other words, an alternative to Bitcoin (CoinDesk, 2022; CoinMarketCap Alexandria, 2022a).

Another digital currency is stablecoin, however, this has low volatility due to its attachment to a fiat currency or precious metal like gold (CoinDesk, 2022; CoinMarketCap Alexandria, 2022d; Dovarganes, 2022). Tether is an example of a stablecoin backed by the US Dollar.

Most cryptocurrencies use blockchain technology, which enables its transaction between peers without the need for a central authority (Yaga *et al.*, 2018; CoinMarketCap Alexandria, 2022b). Since the cryptocurrencies' transactions are processed within the blockchain network, these are (Khan and Hakami, 2021): (1) Irreversible, meaning that after the transaction is confirmed it cannot be reversed due to the ledger characteristic of blockchain; (2) Anonymous because neither the transaction nor the accounts can be connected to an identity; (3) Global speed which means that the transactions are processed and confirmed almost within a small time and are locally indifferent since the network is global; and (4) Secure due to the same reason that the blockchain is secure, because of the cryptography of the information.

The enumerated characteristics, as well as the characteristics of blockchain, contribute to the advantages of cryptocurrencies (Gupta, 2020; Wang, 2020; Khan and Hakami, 2021; Selimović *et al.*, 2021): Easily accessible to the general public, secure, the payment process is efficient, anonymous, private, unchangeable and the transaction fees are low. Additionally, cryptocurrencies help reduce corruption, by distributing the power among the nodes of the network. They aim to eliminate extreme money printing and give people control of their own money since it is not controlled by central banks or the government (Danial, 2020).

On the other hand, cryptocurrencies have some drawbacks which influence their usage in payments (Wang, 2020; Khan and Hakami, 2021; Selimović *et al.*, 2021): volatility and uncertainty, lack of knowledge and difficulty to understand, lack of regulation and not accepted widely.

3 Value Analysis

This chapter aims to describe and explicit the business and research value that the current project carries. First, the concepts that coexist with the value analysis are defined. Next, the NCD model and the AHP method will be adopted to identify and analyse the opportunity as well as generate and select ideas. Additionally, a functional analysis of the problem will be guided using the FAST technique. A value proposition will also be shaped by taking advantage of the Value Proposition Canvas. Finally, a conclusion will be accomplished.

3.1 Value, Value Proposition and Perceived Value Concept Definition

Value has diverse meanings and can be perceived differently based on the context within one desire to define. In the context of product development, value can be defined as the "quantity, which enhances customer satisfaction or slashes the expense attributable to the product" (Rich and Holweg, 2000), which implies that different customers have different viewpoints toward a product.

Although the value of a product can be interpreted differently by different customers, it may be enhanced by improving three source elements: product use, also known as use-value, esteem value, and market value.

The value proposition can be defined as the value an organization guarantees to deliver to its customers. It provides a statement that should clearly and intuitively identify the product (What is your product?), the product users (Who is your target customer?), the product value (What value you provide?), and the product uniqueness (Why your product is unique?).

In correlation to value proposition, perceived value is the consumer's perception of the product value (Zeithaml, 1988), and not all of them share the same opinion of the product value (Boehm, 2006).

The perceived value is based on a set of characteristics that reflects both benefits and sacrifices for the organization and customer while developing a product, service, and relationship for the customer (Lapierre, 2000).

3.2 Value Analysis Concept Definition

Value Analysis refers to the process of reviewing the product value to compare the functions of the product with the requirements required by the customer. Value Analysis is therefore defined by Nick Rich and Matthias Holweg as "a systematic, formal and organized process of analysis and evaluation", "it is a management activity that requires planning, control, and coordination" (Rich and Holweg, 2000).

This analysis is done by the sum of the performance and capabilities of a product per its cost, which results in the relation of a function by the product cost, as represented in equation 3.1 (NPD Solutions, 2019).

$$Value = \frac{Performance + Capability}{Cost} = \frac{Function}{Cost} \tag{3.1}$$

In the equation, Performance and Capability variables refer to the product use value, therefore the functions that make the product work or sell (also referred to as basic functions). Additionally, supporting functions can be added to the product to boost the basic function to help sell the product, for instance, esteem value properties, which in return increases its value.

3.3 Problem and Opportunity Identification

The problem aimed to be addressed is centred on the process of Bitcoin payment transactions, which arises opportunities. It enables merchants to understand and adopt this payment method. In addition, it helps build trust in this new way of paying for goods and services, by sharing the knowledge acquired, and therefore incentive its usage.

The New Concept Development (NCD) can be used to properly analyse and expand opportunities (Koen, Bertels and E. J. Kleinschmidt, 2014). This model is integrated within the Front End of Innovation (FEI) process, which aims to improve the overall innovation process (Koen *et al.*, 2002).

As illustrated in Figure 2, the Front End of Innovation (FEI) is often envisioned as a linear process of three stages (Koen, Bertels and E. Kleinschmidt, 2014; Koen, 2021): *Discovery* where the market is analysed and therefore generates ideas and new opportunities; *Stage 1*, also called Scoping Stage, where is evaluated the technical and marketing details to have the worth of the project so it can be re-evaluated more thoroughly at next steps; and *Stage 2* which focus on building a detailed business case.

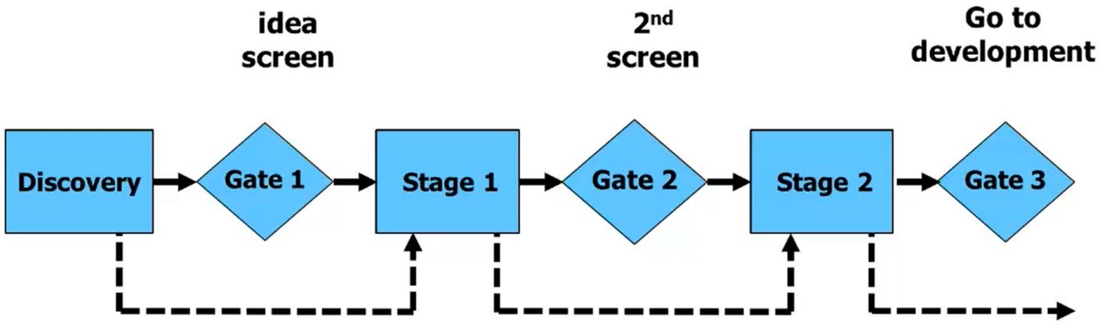


Figure 2 - Flow diagram representing the three stages of the FEI process
(Source: (Koen, 2021))

The New Concept Development (NCD) model "provides a common language and definition of the key components of the Front End of Innovation" (Koen *et al.*, 2001), this is a framework that intends to facilitate the communication between the team members during an innovation or investigation (on parts of the FFE) process. It divides the FEI into three distinct areas: (1) the *Engine*, at the centre of the model, which defines the vision, business strategy, resources, and culture of the organization; (2) The *Wheel*, the inner part of the model, which consists of the *Core Front End Activity Elements* describing the five key elements of the model that are controllable by the corporation and should be fulfilled to properly define the concept; (3) the *Rim*, also mentioned as *Influencing Factors*, consists of relatively uncontrollable environmental aspects, such as competitor threats, customer and worldwide trends, regulatory changes, and others, that influence the engine and the activity elements affecting consequently the entire innovation process (Koen *et al.*, 2001, 2002; Koen, Bertels and E. J. Kleinschmidt, 2014; Koen, Bertels and E. Kleinschmidt, 2014).

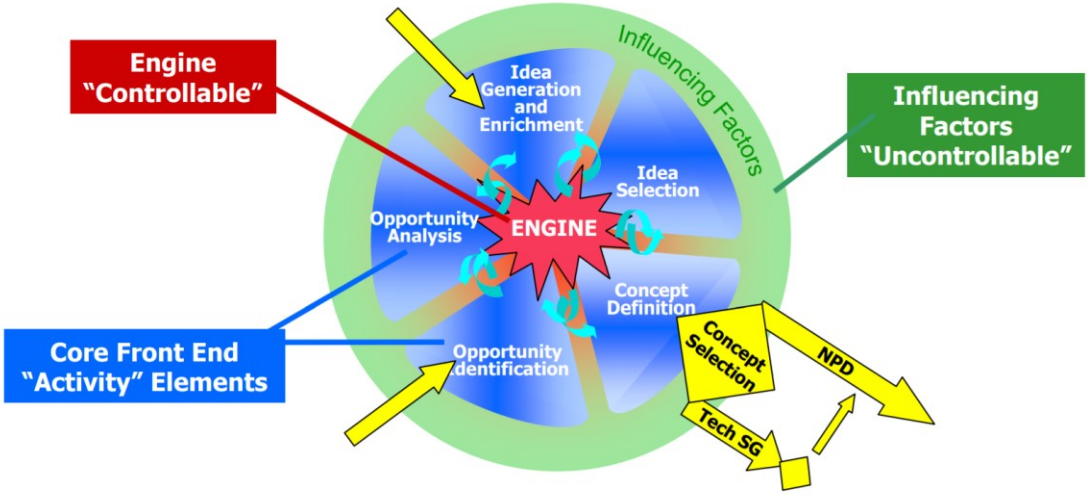


Figure 3 - Representation of the New Concept Development "Wheel" Model
(Source: (Koen, 2004))

As described in Figure 3, the arrows pointing into the model represent the starting points of the project, which means that projects can either start by first identifying opportunities or by first defining ideas. The arrow pointing out from the model indicates the process is finished after the concept is defined and represents how concepts enter the next stages (the New Product Development (NPD) or Technology Stage-Gate (TSG) process). In contrast with the original representation of the FEI process represented in Figure 2, the NCD model has a circular shape suggesting that "ideas are expected to flow, circulate, and iterate between and among all the five elements" (Koen *et al.*, 2002; Koen, Bertels and E. J. Kleinschmidt, 2014; Koen, Bertels and E. Kleinschmidt, 2014).

The main objective of NCD is to define a common language to understand the FEI activities, so it must be defined the terminology used in this model so everyone can understand the used terms. There are three important concepts found in the activity elements of the model: Opportunity, Idea, and Concept. For the analysis of the problem that the current document aims to present, the opportunity can be defined as a need that the company and the author of this thesis realized that exists between the current company's products and envisioned future; the idea can be defined as the emerging views of the solution for the problem identified by the opportunity; and concept can be defined as a well-defined approach that expands the current knowledge on the topic and satisfies the problem identified.

The next subsections detail the five elements of the NCD model for this thesis problem in study. It starts with the opportunity identification and analysis, then idea generation and selection and finishes with the concept definition.

3.3.1 Opportunity Identification

Since the origin of Bitcoin in 2009, thousands of new cryptocurrencies have emerged since then counting now over 17.000 cryptocurrencies, according to data from CoinMarketCap (CoinMarketCap, 2021). Bitcoin, Ethereum, Binance Coin, Tether and Solana are in the top 5 tier list based on their market capitalization (as of February of 2022), as shown by the column chart in Figure 4.

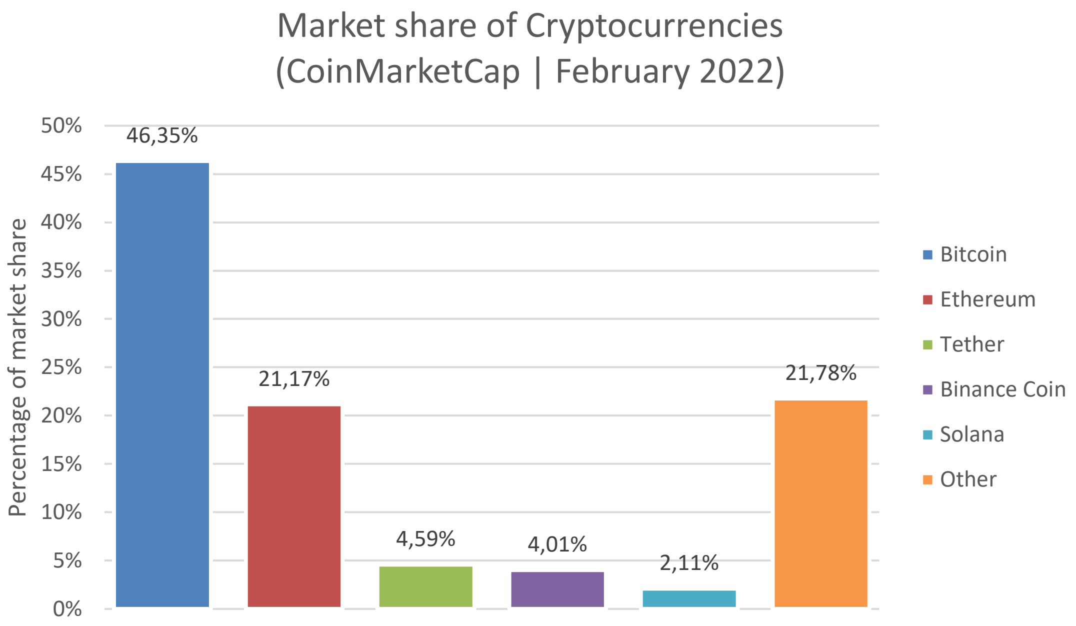


Figure 4 - Percentage of market share of cryptocurrencies
(Adapted from: (CoinMarketCap, 2021))

On the other hand, the Bitcoin network transaction volume had already surpassed American Express and PayPal, as the column chart in Figure 5 illustrates. Bitcoin network processed an average of $475 billion per quarter in 2021 (Blockchain.com, 2022), American Express processed an average of $321 billion per quarter in 2021 (American Express Company, 2020b, 2020a, 2020c, 2021a, 2021c, 2021b, 2021d, 2022), PayPal processed an average of $311 billion per quarter in 2021 (Best, 2022; PayPal, 2022), the Mastercard network processed $1930 billion per quarter in 2021 (Mastercard, 2022), and Visa network processed $3377 billion per quarter in 2021 (Visa, 2021c, 2022). Additionally, the Bitcoin network can do a final settlement of 7 transactions per second, in contrast with 0 transactions per second in Visa or Mastercard network (Wouters, 2021). Although the latter networks can approve thousands of transactions per second, the final settlement of these can take up to months (Wouters, 2021). This topic affects the merchants since it complicates the business cash flows.

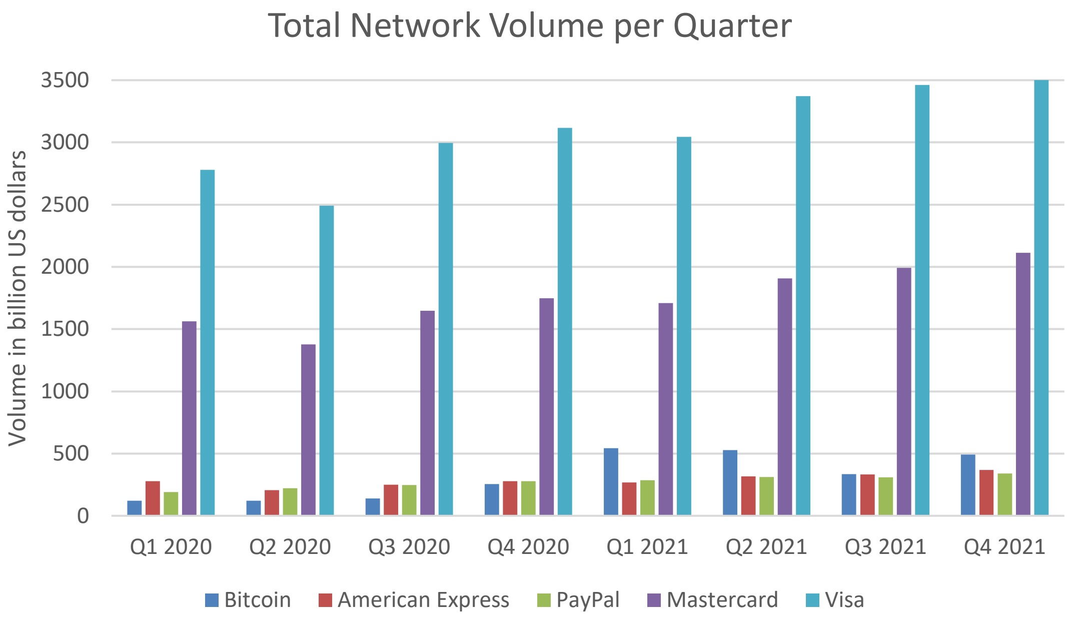


Figure 5 - Total network volume of Bitcoin and other digital payment networks.

The growth and popularity of cryptocurrencies undeniably lead to a more and more massive influx of institutional investors and large payment players accepting them as a form of payment (Choudhury, 2021; Kochkodin, 2021; The Economic Times, 2021). PayPal and Visa along with other leading payments platforms such as Neteller and Robinhood already joined an expanding range of companies that are enabling cryptocurrencies as a payment method (Chan, 2021; Maishera, 2021; Visa, 2021a). Stripe revealed is preparing to re-enter the market after ending the support for Bitcoin payments on its platform in 2018 (after four years in the cryptocurrencies market) (Karlo, 2018; Browne, 2021; Maishera, 2021). Mastercard also announced that any of the banks and merchants on its payment network can soon integrate crypto into their products (Dhamodharan, 2021; Son, 2021). This company is working alongside Wirex, BitPay and LVL to enable the move of cryptocurrencies through their network (Bitcoin, 2020; Wirex Team, 2020; Dhamodharan, 2021; DiCamillo, 2021).

Not only payment platforms and fintech firms have been entering the cryptocurrency market in the past years, but also online retailers (Fortune Business Insights, 2020; Goswami, Borasi and Kumar, 2021; The Economic Times, 2021). Microsoft, Shopify, Magento, and Overstock are some of the large e-commerce platforms that support cryptocurrency payments, mainly Bitcoin (Due.com, 2018; ICOholder, 2018; Beigel, 2021; Lisa, 2021; Tuwiner, 2021). In 2018, WeMakePrice, the largest South Korean e-commerce platform, started to accept cryptocurrencies as a primary payment option for users (Young, 2018; Bezhovski, Davcev and Mitreva, 2021).

Overall, the cryptocurrency market continues to gain momentum and mainstream popularity. Therefore, joining the numerous companies that have already established their position in this market to make a difference in the financial world is a great opportunity.

3.3.2 Opportunity Analysis

The present activity is described as a need to translate the output of the Opportunity Identification into specific business and technology opportunities to prove that the opportunity is worth pursuing (Koen *et al.*, 2001, 2002). Methods like Strategic framing, Market segment assessment, Competitor analysis, and Customer assessment can be used to help analyse the identified opportunity.

In a world where e-commerce has grown impressively over recent years and is expected to continue growing over the next few years (Chevalier, 2021), as the column chart in Figure 6 shows, it proves that online payments are becoming an attractive way to purchase goods and services. With this interest and consumer need also comes a greater expectation that companies provide multiple ways to shop and pay. The Mastercard New Payments Index, a survey conducted by Mastercard's Global Foresights, Insights and Analytics Team and The Harris Poll, ascertain that 71% of inquired expect to use cashless moving forward (Mastercard, 2021).

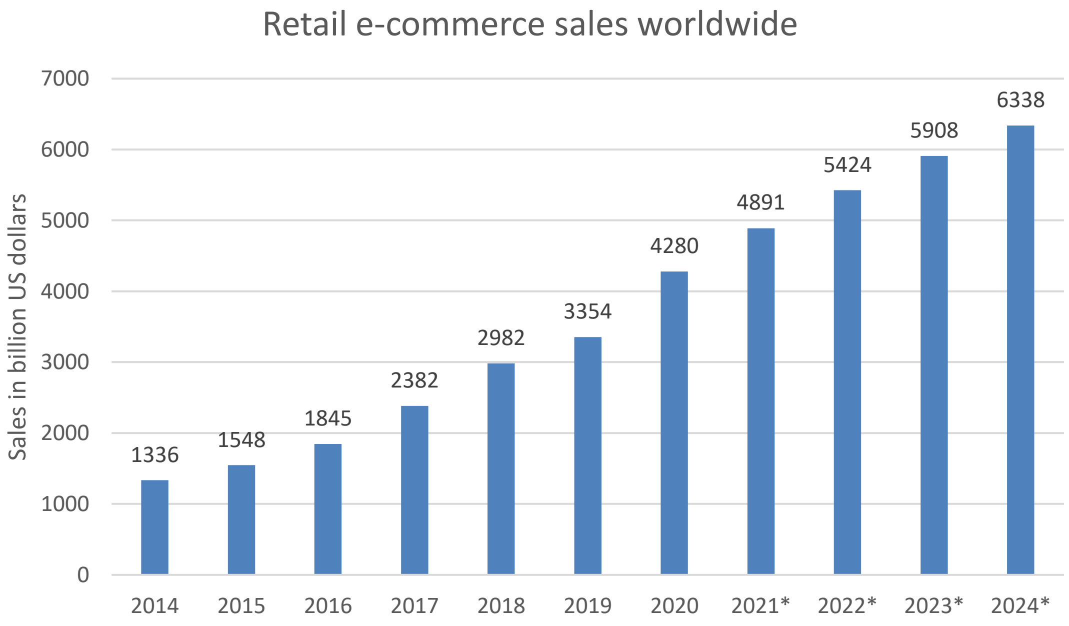


Figure 6 - Retail e-commerce sales worldwide from 2014 to 2024 in billion US Dollars (Adapted from: (Chevalier, 2021))

A report from Allied Market Research states that the global cryptocurrency market size was valued at $1.49 billion in 2020 and is predicted to reach $4.94 billion by 2030 (Goswami, Borasi and Kumar, 2021), which evidences the growth of the adoption of cryptocurrencies. Additionally, there is near-universal awareness of cryptocurrency (94%) among adults that have discretion over their household finances (Visa, 2021b).

Approximately one-third of crypto-aware consumers own cryptocurrency, split between Active Owners (21%) who use it for transactions and Passive Owners (11%) (Visa, 2021b), as displayed in Figure 7.

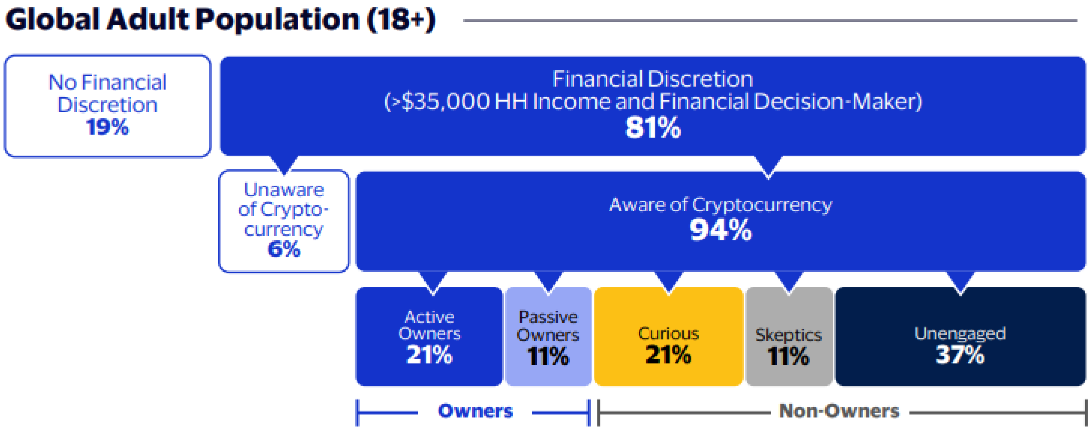

Global Adult Population (18+)
No Financial Discretion 19%
Financial Discretion (>$35,000 HH Income and Financial Decision-Maker) 81%
Unaware of Crypto-currency 6%
Aware of Cryptocurrency 94%
Active Owners 21%
Passive Owners 11%
Curious 21%
Skeptics 11%
Unengaged 37%
Owners
Non-Owners

Figure 7 - Current Market of Cryptocurrency
(Source: (Visa, 2021b)

Moreover, 4 in 10 people (40%) across North America, Latin America and the Caribbean, the Middle East and Africa, and the Asia Pacific state they plan to use cryptocurrency next year (Mastercard, 2021), demonstrating that the adoption is significant and growing fast. In contrast, only 3 in 100 people regret buying cryptocurrencies (Binance Research, 2021). Some findings demonstrate that the perceived ease of use and perceived usefulness affects positively the intention to use Bitcoin (Nadeem *et al.*, 2021). The results show there is a sufficient probability to be adopted if the cryptocurrencies and their technology are easy to use, understandable, flexible, and useful (Nadeem *et al.*, 2021). Additionally, the perceived ease of use was proven to influence the perceived usefulness (Nadeem *et al.*, 2021).

While the interest and usage in cryptocurrencies rise, so does the confidence in them. According to research, there is near-unanimous confidence in cryptocurrencies (97%) amongst users, and this correlates with a lower institutional trust, where less than 17 out of 19 markets represented have less than 50% trust in the local institutions (Binance Research, 2021), as represented by the column chart in Figure 8.

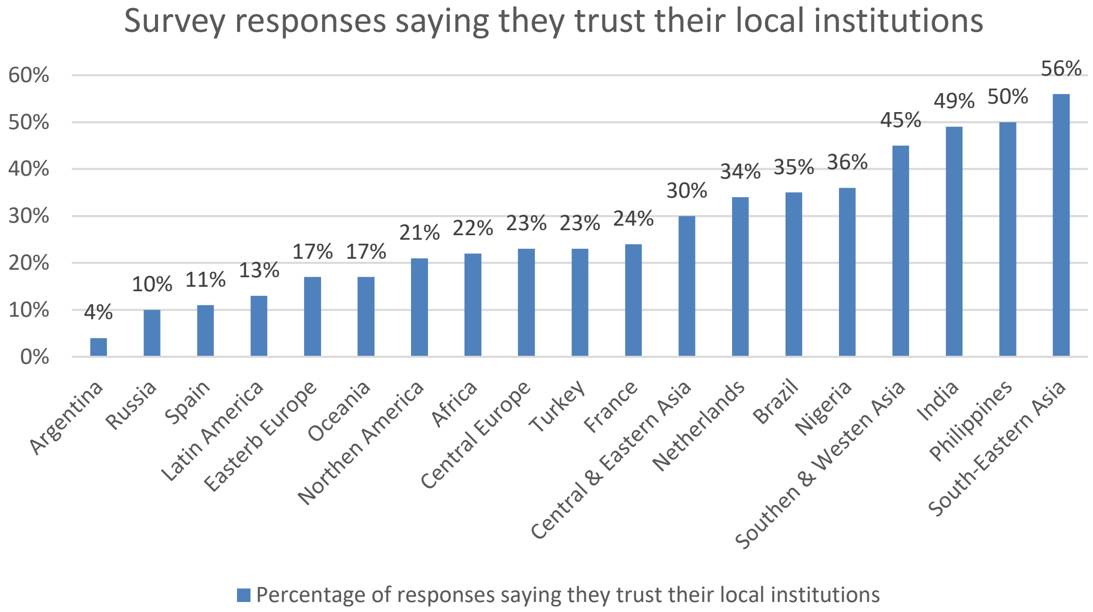


Figure 8 - Percentage of responses saying they trust their local institutions organized by markets
(Adapted from: (Binance Research, 2021))

Regarding the reasons for accepting cryptocurrency as a payment means, 42% of the retailers in The Netherlands state they use this method to attract extra customers, 23% because their customers ask for it, 21% because they are interested in new technology, and 7% because of the low transaction fees (Jonker, 2019), as the column chart in Figure 9 exhibits.

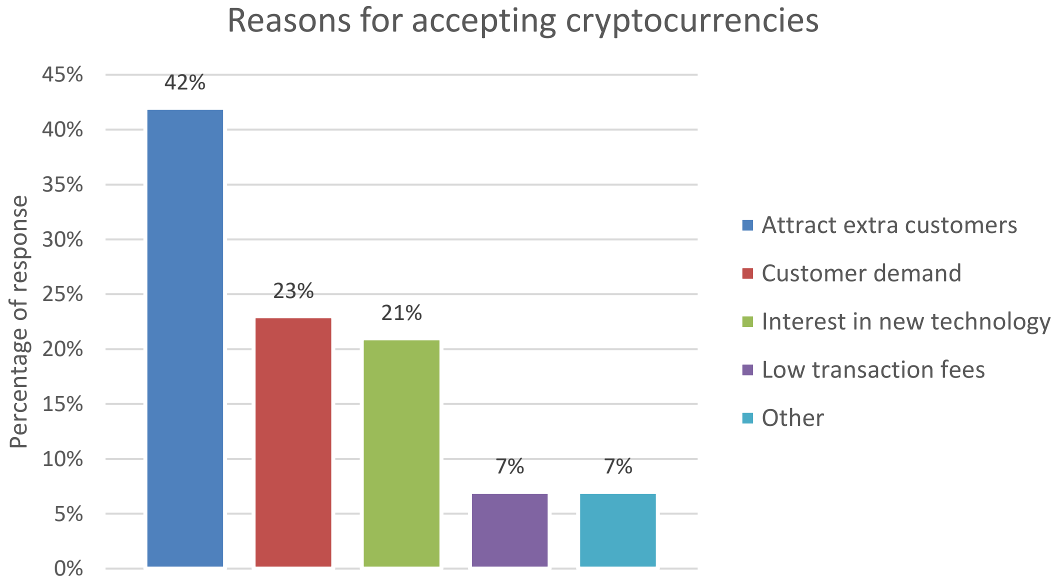


Figure 9 - Reasons given for crypto-acceptance and corresponding percentage of responses
(Adapted from: (Jonker, 2019))

On the other hand, the major reason for not-acceptance is unfamiliarity with cryptocurrencies (58%), followed by lack of trust in crypto (16%), acceptance not being common in their industry (12%), safety concerns (9%) and perceived complexity (5%) (Jonker, 2019). In addition, a study conducted by Forrester Consulting reveals that 40% of customers that pay with cryptocurrencies are new to the merchant, the number of transactions processed with cryptocurrencies on e-commerce sites grows by 12.5% every year, and the purchase amounts are twice that of credit card purchases (Jahosky, 2020; TripleA, 2021). Overall, the acceptance of cryptocurrencies by retailers is influenced by customers' demand and therefore, they can attract more clients with this promising payment method.

All these data about cryptocurrencies owners, markets and retailers suggest that the adoption of cryptocurrencies as a payment method has promising signs of growth with wide-reaching implications for payments, finance, and commerce emphasizing that cryptocurrencies are in the need to be curated by applying these in the real world.

3.3.3 Idea Generation and Enrichment

Once the opportunity has been identified and analysed, it is possible to start yielding ideas with the information obtained through the analysis process. With this activity is desired to generate a group of ideas, rather than one primary idea, to develop and maturate the opportunity.

Previously was depicted that e-commerce transactions and cryptocurrency adoption have increased over the years. Also, the usage of cryptocurrencies on e-commerce systems rises every year.

Ideas have been generated to keep up with this shift to digital payments and currencies, taking into consideration the current state of adoption of cryptocurrencies, the existing work for online payments with cryptocurrencies, and the author's perspectives and ideas. This led to the following approaches:

(1) convert cryptocurrency to fiat currency, in which the transaction is itself proceeds with a fiat currency;
(2) implementing a cryptocurrency exchange. in other words, a service where one can store their cryptocurrencies, and the transfer occurs within this service;
(3) usage of a cryptocurrency payment gateway, which in fact process the transfer between the two actors and can integrate with other application;
(4) usage of multiple cryptocurrency payment gateways.

3.3.4 Idea Selection

Once ideas have been generated, these must be reviewed and analysed to determine those that are worth pursuing to achieve the most business and research value. Selection may be simple as an individual's choice or statistical methods can be applied so the generated ideas are evaluated using mathematical techniques. When a set of criteria must be taken into consideration, one can apply a Multi-Criteria Decision Analysis since this method is based on the crossing of alternatives with the deciding criteria.

For this analysis, the Analytic Hierarchy Process (AHP) Classical method can be used, which focuses on breaking down a complex decision problem into levels organized in a hierarchical order, aiming to ease its comprehension and evaluation (Saaty, 1984).

The decision-making process involves the following phases: (1) structure a problem in objectives (first level), problem criterion (second level) and alternatives (third level), defining, therefore, the hierarchical decision tree; (2) establish a comparison matrix that defines the priorities for each level; (3) obtain the relative priority of each criterion by calculating the priorities vector of the normalized comparison matrix; (4) evaluate the vector values consistency by calculating the consistency ratio based on the priorities vector and compare it with Saaty indices table; (5) define the pairwise alternatives matrix for each criterion; (6) calculate the composed priority vector, based on the multiplication of the alternatives matrix with the priorities vector; and finally, (7) select the best alternative which is indicated by the parcel with the highest value of the composed priority vector (Saaty, 1984).

3.3.4.1 Hierarchical Decision Tree

First is desired to describe the problem objectives, followed by the criterion and alternatives to define the decision tree. The former and latter have been already described in Chapter 1 and Section 3.3.3, respectively, remaining the problem criterion. The needed indicators for the approach are security, facility to support multiple cryptocurrencies, and implementation costs, both monetary and time, to achieve the end solution. Figure 10 illustrates a representative graphic of the hierarchical decision tree.

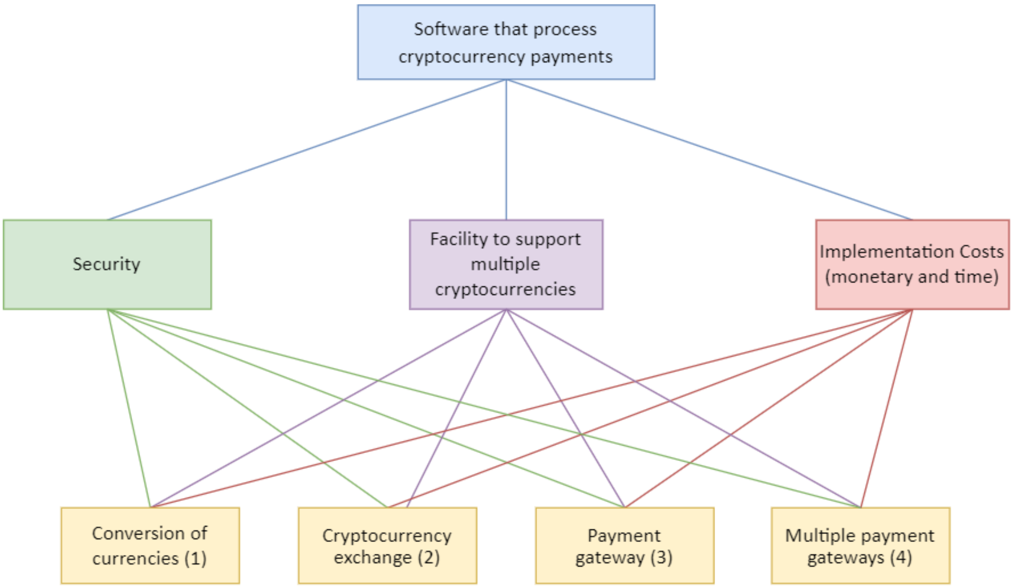

Software that process cryptocurrency payments
Security
Facility to support multiple cryptocurrencies
Implementation Costs (monetary and time)
Conversion of currencies (1)
Cryptocurrency exchange (2)
Payment gateway (3)
Multiple payment gateways (4)

Figure 10 - Hierarchical Decision Tree describing the connections between each criterion and alternative

3.3.4.2 Priorities Comparison Matrix

Being the decision tree defined, it is now possible to prioritize the criterion and alternatives by performing pairwise comparisons. It allows comparing criteria or alternatives to evaluate which one is preferred or has a greater amount of some quantitative property over another. The scale to be used in the pairwise comparison is the fundamental scale, defined by Saaty (Saaty, 1984), which correlates priorities with importance levels, as detailed in Table 2.

Table 2 - Saaty (Saaty, 1984) fundamental scale

Importance Level	**Definition**	**Explanation**
1	Equal importance	Two activities contribute equally to the objective.
3	Low importance	Experience and judgment slightly favour one activity over another.
5	High importance	Experience and judgment strongly favour one activity over another.
7	Very High importance	An activity is strongly favoured, and its dominance demonstrated in practice.
9	Absolute importance	The evidence favouring one activity over another is of the highest possible order of affirmation.
2, 4, 6, 8	Intermediary importance	When compromise is needed.

After defining the scale to be used, the development of the matrix for comparing criteria can be proceeded, as depicted in Table 3.

Table 3 - Pairwise judgements for criterion

	Security	**Facility to support multiple currencies**	**Implementation costs**
Security	1	7	9
Facility to support multiple currencies	1/7	1	3
Implementation costs	1/9	1/3	1
$\sum_{\text{column}}$	79/63	25/3	13

By analysing the comparison matrix is possible to conclude that security is absolutely needed, being strongly important when comparing with the facility to support multiple currencies and even more important when comparing with Implementation costs. On the other hand, the facility to support multiple currencies is slightly more important than implementation costs.

3.3.4.3 Relative Priority Values

This phase involves calculating the relative priority of each criterion. For this to be possible is necessary to normalize the values of the pairwise matrix (NM), which is accomplished by dividing each matrix cell by the sum of the respective column values, resulting in Matrix 3.2.

$$NM = \begin{pmatrix} \frac{63}{79} & \frac{21}{25} & \frac{9}{16} \\ \frac{9}{79} & \frac{3}{25} & \frac{3}{8} \\ \frac{7}{79} & \frac{1}{25} & \frac{1}{16} \end{pmatrix} \tag{3.2}$$

Then, the relative priority values Vector 3.3 (i.e., eigenvector, EV) is determined by calculating the arithmetic average for each normalized matrix row. The values have been rounded with two decimal cases.

$$EV = \begin{pmatrix} 0.78 \\ 0.15 \\ 0.07 \end{pmatrix} \tag{3.3}$$

3.3.4.4 Relative Priority Values Consistencies Analysis

The consistency of the judgements is measured once calculated the priorities eigenvector. For this, the Consistency Ratio (CR) value is required as it permits to know if the criterion judgments are consistent with random samples of judgments. If the CR value is higher than 0.1, then the applied judgments are not trustworthy, as these tend to randomness and are thus not consistent. CR is calculated by Formula 3.4, being CI the Consistency Index and RI the Random Index.

$$CR = \frac{CI}{RI} \tag{3.4}$$

The Random Index is a constant number from the values defined in Table 4 depending on the number of criteria.

Table 4 - Saaty (Saaty, 1984) random judgements indices

Number of criteria	**1**	**2**	**3**	**4**	**5**	**6**	**…**	**15**
RI	0	0	0.58	0.90	1.12	1.24	…	1.59

Since the evaluation takes into consideration four criteria, then RI is 0.90.

To calculate the Consistency Index (CI) is first needed to obtain the λ_{max} which represents the biggest eigenvalue. This is achieved by the product of the calculated criterion priorities eigenvector with the original pairwise comparison matrix, as equations 3.5, 3.6 and 3.7 proof.

$$\begin{pmatrix} 1 & 7 & 9 \\ \frac{1}{7} & 1 & 3 \\ \frac{1}{9} & \frac{1}{3} & 1 \end{pmatrix} \times \begin{pmatrix} 0.78 \\ 0.15 \\ 0.07 \end{pmatrix} = \lambda_{max} \times \begin{pmatrix} 0.78 \\ 0.15 \\ 0.07 \end{pmatrix} \tag{3.5}$$

$$\begin{pmatrix} 2.46 \\ 0.47 \\ 0.21 \end{pmatrix} = \lambda_{max} \times \begin{pmatrix} 0.78 \\ 0.15 \\ 0.07 \end{pmatrix} \tag{3.6}$$

$$\lambda_{max} = \overline{\boldsymbol{X}} \left(\frac{2.46}{0.78}, \frac{0.47}{0.15}, \frac{0.21}{0.07} \right) = 3.10 \tag{3.7}$$

This value can now be applied in the Consistency Index (CI) formula 3.8.

$$CI = \frac{\lambda_{max} - n}{n - 1} = \frac{3.10 - 3}{3 - 1} = 0.05 \tag{3.8}$$

Based on the CI formula output value and RI value, the CR formula 3.9 can be applied, which permits evaluating if the value is lower than the maximum suggested.

$$CR = \frac{CI}{RI} = \frac{0.05}{0.58} = 0.08 \tag{3.9}$$

As demonstrated in equation 3.9, the CR value is lower than 0.1, approving that the judgements realized are in fact consistent and not random.

3.3.4.5 Alternatives Pairwise Comparison Matrix

This phase aims to perform pairwise comparisons by making judgments per alternative while taking into consideration each criterion. This allows finding the best alternative for each defined criterion.

The priority results of this phase will be determined by following a similar process to previous phases, so there is a need to develop a pairwise comparison matrix, in this case, for each criterion and calculate the corresponding priority vector by normalizing the matrix. Since the process of calculating the priority vector and demonstrating its consistency has already been shown in the past sections, it will only be depicted the representation of the comparison matrix and the respective priority vector in Table 5, Table 6 and Table 7.

Table 5 - Alternatives Pairwise Comparison Matrix in the context of Security criteria

Security	Conversion of Cryptocurrencies	Cryptocurrency Exchange	Payment Gateway	Multiple Payment Gateways	Priority Vector
Conversion of Cryptocurrencies	1	2	1/7	1/7	0,07
Cryptocurrency Exchange	1	1	1/8	1/8	0,05
Payment Gateway	7	8	1	1	0,44
Multiple Payment Gateways	7	8	1	1	0,44

Table 6 - Alternatives Pairwise Comparison Matrix in the context of Facility to support multiple currencies criteria

Facility to support multiple currencies	Conversion of Cryptocurrencies	Cryptocurrency Exchange	Payment Gateway	Multiple Payment Gateways	Priority Vector
Conversion of Cryptocurrencies	1	7	1/3	1/3	0,17
Cryptocurrency Exchange	1/7	1	1/9	1/9	0,04
Payment Gateway	3	9	1	1/3	0,29
Multiple Payment Gateways	3	9	3	1	0,50

Table 7 - Alternatives Pairwise Comparison Matrix in the context of Implementation costs criteria

Implementation costs	Conversion of Cryptocurrencies	Cryptocurrency Exchange	Payment Gateway	Multiple Payment Gateways	Priority Vector
Conversion of Cryptocurrencies	1	2	1/9	1/9	0,06
Cryptocurrency Exchange	1/2	1	1/9	1/9	0,04
Payment Gateway	9	9	1	2	0,52
Multiple Payment Gateways	9	9	1/2	1	0,37

3.3.4.6 Composed Priority Vector (CPV) Computation and Alternative Selection

To finalize the AHP method process is needed to calculate the composed priority vector. The vector values are obtained by joining the alternatives priority vectors of each criterion computed in phase five, creating a matrix of priorities, and then applying the product to the criterion relative priority values vector obtained in phase three, as Equation 3.10 shows.

$$CPV = \begin{pmatrix} 0.07 & 0.17 & 0.06 \\ 0.05 & 0.04 & 0.04 \\ 0.44 & 0.29 & 0.52 \\ 0.44 & 0.5 & 0.37 \end{pmatrix} \times \begin{pmatrix} 0.78 \\ 0.15 \\ 0.07 \end{pmatrix} = \begin{pmatrix} 0.08 \\ 0.42 \\ 0.05 \\ 0.44 \end{pmatrix} \tag{3.10}$$

According to the AHP method, the values of this vector indicate the final evaluation for each alternative, meaning that the highest value represents the best alternative to choose. Therefore, looking at Vector 3.10 , the fourth alternative has the highest value, meaning that Payment Gateway is the best alternative to adopt, considering the defined criteria.

3.3.5 Concept Definition

Concept definition is the final element of the NCD model and involves the development of a business case that encompasses the potential market to sell the concept, customer needs, investment requirements, competitor assessments, technology unknowns, and overall project risk (Koen *et al.*, 2001; Koen, Bertels and E. J. Kleinschmidt, 2014).

For this case is proposed that applying the selected idea is possible to create an application that processes cryptocurrency payments. To achieve this is necessary to study the state-of-the-art technologies and approaches that allow these processes, always fulfilling the desired result objectives and considering the features that enrich the solution.

3.4 Value Proposition

The Business Model Canvas (BMC) is a useful tool to “describe how an organization creates, delivers and captures value”, it can therefore be used to better clarify the proposed value of a concept or product. It defines a shared language for describing, visualizing, assessing and changing business models (Osterwalder and Pigneur, 2010). The model proposes nine basic building blocks segmented into two stages: the back and front stage, in which the latter insights into how the customers see the product, and the foremost refers to the internal business processes.

The backstage is composed of the Key Partnerships (who will help develop the business), Key Resources (what assets will be needed), Key Activities (what tasks to do) and Cost Structure (what financial resources are required to operate the business). The front stage includes the Customer Segments (which customers the value focuses on), Value Propositions (which value is being offered to the customers), Customer Relationships (how to acquire and retain the customers), Channels (how to communicate and deliver value to the customers), and Revenue Streams (how to profit from the offered value).

Since this section focus on describing the proposed value of the product being developed, only this building block and Customer Segments will be analysed and detailed.

The Value Proposition Canvas integrates with the BMC, with the former "being like a plug-in" to the latter that allows to detail how to create value for the customers (Osterwalder *et al.*, 2014), as illustrated in Figure 11. The Value Proposition Canvas has two sides:

(1) Value Map that describes the intentions to create value for the customer, breaking it down into products and services, pain relievers, and gains creators (represented in Figure 11 by the blue square);
(2) Customer Profile which describes the motivations to buy the product or service breaking it down into customers' jobs, pains, and gains (represented in Figure 11 by the blue circle).

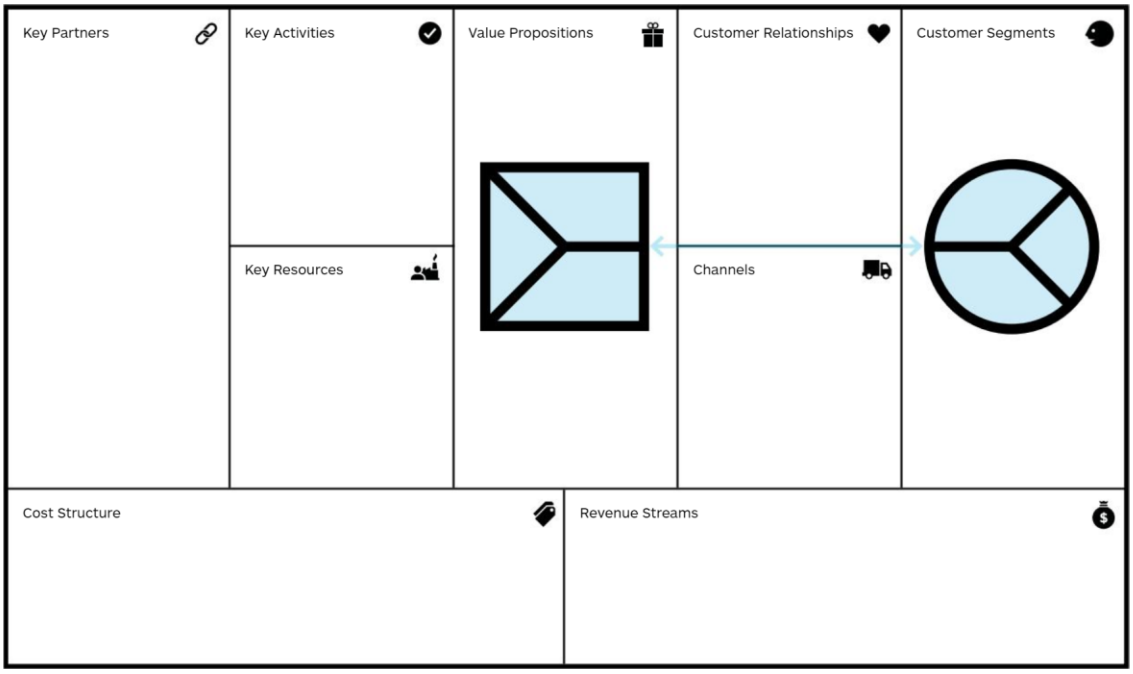


Figure 11 - Value Proposition Canvas integrated with Business Model Canvas
(Source: (Osterwalder *et al.*, 2014))

The Value Proposition Canvas was therefore used to visualise and analyse the customers' needs and pains and how the product being developed can release these pains and create benefits, as depicted in Figure 12. There is a clear connection between the product/services to be developed and what is expected by the customer, thus showing how it simplifies the clients' life and consequently generates benefits.

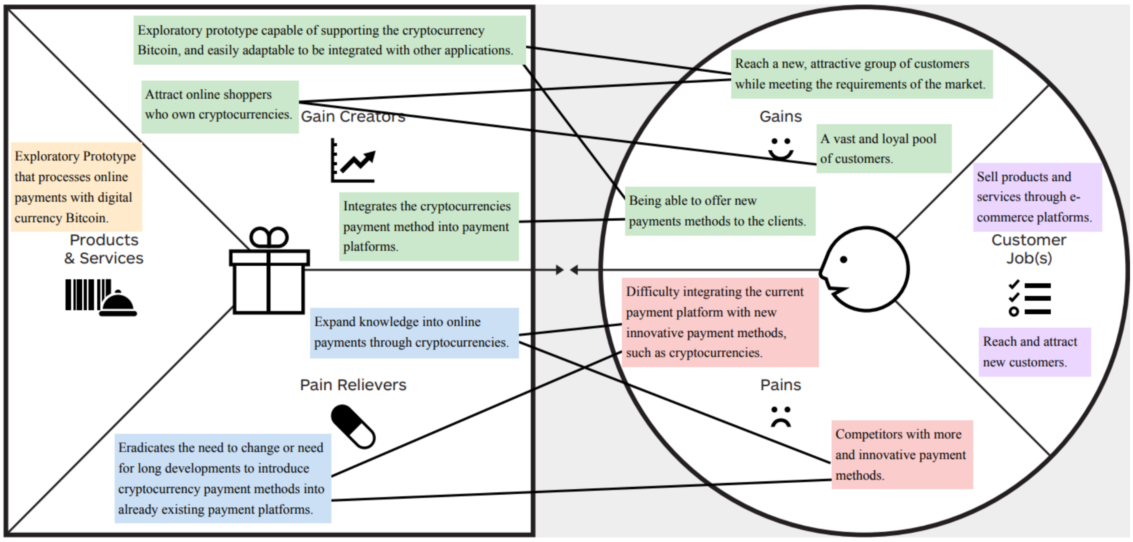

Exploratory prototype capable of supporting the cryptocurrency Bitcoin, and easily adaptable to be integrated with other applications.
Attract online shoppers who own cryptocurrencies.
Gain Creators
Exploratory Prototype that processes online payments with digital currency Bitcoin.
Products & Services
Integrates the cryptocurrencies payment method into payment platforms.
Expand knowledge into online payments through cryptocurrencies.
Pain Relievers
Eradicates the need to change or need for long developments to introduce cryptocurrency payment methods into already existing payment platforms.
Reach a new, attractive group of customers while meeting the requirements of the market.
Gains
A vast and loyal pool of customers.
Being able to offer new payments methods to the clients.
Sell products and services through e-commerce platforms.
Customer Job(s)
Difficulty integrating the current payment platform with new innovative payment methods, such as cryptocurrencies.
Reach and attract new customers.
Pains
Competitors with more and innovative payment methods.

Figure 12 - Value Proposition Canvas of the product being developed
(Adapted from: (Strategyzer AG, 2020))

The value proposition of this project consists of the development of an exploratory prototype capable of supporting the Bitcoin cryptocurrency and easily adaptable to be integrated with other applications and support more cryptocurrencies. It attracts a wide variety of merchants looking to add this innovative and cutting-edge payment method to their online payment platform, creating value not only for merchants but also for their customers.

Value is being generated by expanding the knowledge of cryptocurrency payments and developing an application capable of supporting Bitcoin transactions. The application can be integrated with already existing online applications erasing the need for long developments (Value for the Customer). It means that the final users of the application will easily provide this payment method in their online platform (Perceived Value).

Cryptocurrency payments are gaining a lot of interest over the past few years with the market showing promising values. However, cryptocurrencies are still a topic very new and in need of extended investigation and knowledge on how to properly integrate them, which requires a lot of time researching alternatives on how to implement them and already existing similar projects (Sacrifices). Nevertheless, this investigation and research will increase the overall knowledge of cryptocurrency payments and alternatives to processing them (Benefits).

3.5 Problem Functional Analysis

Functional analysis is a methodology that aims to explain and detail the functions of a given complex system or product.

Function Analysis and System Technique (FAST) can be used to perform this analysis, which organizes in a graphical representation the necessary functions of a product, process, or system in a logical relationship of how?/why?/when? (Borza, 2011). By using this technique is possible to identify not only missing functions but also functions that are not needed or duplicated.

FAST additionally introduces the concepts of higher and lower order functions. The former is the far-most left function and identifies the expected result of the solution (i.e., basic function), and the latter is the far-most right function and identifies the foundations that support the execution of the former. The Higher Order Function and Lower Order Function are connected and arranged using a How?-Why? logic. Moving from Higher Order Function is asked "How?" and the function immediately to the right answers that question. Similarly, moving from Lower Order Function asks "Why?", with the function immediately to the left answering that question. With the question "When?" one identifies functions that occur together with or as a result of each other. In FAST diagrams, functions are named from Active Verb (i.e., action) concatenated with Measurable Noun (i.e., domain entity) notation, as Figure 13 exposes.

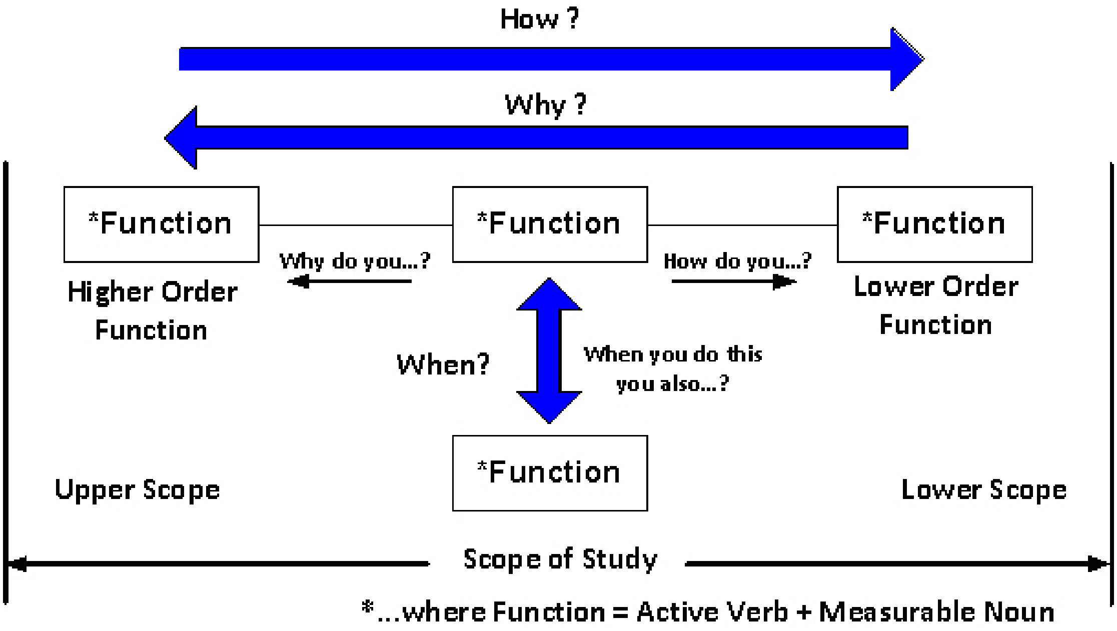


Figure 13 - FAST Diagram illustrating the How, Why and When questions
(Source: (Value Analysis Canada, 2022))

Figure 14 shows the proposed FAST diagram of the problem desired to solve within this thesis based on the existing knowledge and information so far. To execute a Bitcoin payment transaction, the payment method must be provided. The solution provides the payment method to execute a Bitcoin payment transaction. This is possible by converting the fiat to Bitcoin currency, generating the QR Code and processing a transaction. These are enabled by communicating with an API and then communicating with the cryptocurrency payment gateway, which in its turn does the respective action.

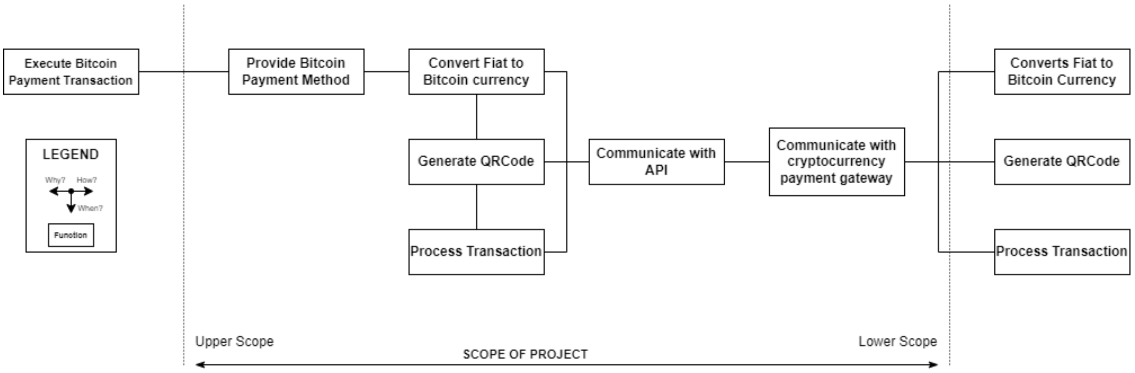

Execute Bitcoin Payment Transaction
LEGEND
Why?
How?
When?
Function
Provide Bitcoin Payment Method
Convert Fiat to Bitcoin currency
Generate QRCode
Process Transaction
Communicate with API
Communicate with cryptocurrency payment gateway
Converts Fiat to Bitcoin Currency
Generate QRCode
Process Transaction
Upper Scope
Lower Scope
SCOPE OF PROJECT

Figure 14 - Functional analysis represented in a FAST diagram

3.6 Value Analysis Conclusion

The adoption of value analysis helps to understand the end-users needs and requirements, what needs to be done and how to fulfil these needs. The cryptocurrency market is growing, as was evidenced, and consequently the need to adopt them in the context for which they were designed, the payment of goods and services, thus leaving the investment niche. To achieve this, opportunities and thus ideas arise, in contrast with three possible criteria. AHP allows to statistically review which idea is superior, and its output reveals that using multiple cryptocurrency payment gateways to process cryptocurrencies transaction is the best approach to follow as it provides more value in accordance with the chosen criteria.

4 State of The Art

This chapter intends to review the major technologies for processing cryptocurrency payments online. It starts by identifying and describing different types of e-commerce platforms. Secondly, cryptocurrency payment gateways are introduced, and some existing systems are described and compared accordingly by some criteria. To finish is classified and given examples of cryptocurrency wallets.

4.1 E-commerce Platforms

E-commerce is the means of selling and buying goods or services over the internet. The different types of commerce include business to business (B2B), business to consumer (B2C), consumer to consumer (C2C), and consumer to business (C2B).

An e-commerce platform is a software that enables the e-commerce process where both parties get involved. It allows the management of sales, inventory, marketing, and operations, such as supply chain planning, shipping logistics and customer service (Schwarz, 2016; Adobe, 2022; SendPulse, 2022).

One can define four types of e-commerce platforms (Hou, 2018): Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), and on-premises. The latter is hosted locally, and the remainders are hosted in the cloud, delivering solutions through the internet.

Infrastructure as a Service covers infrastructure components (storage, networking, servers, and virtualization) (Bernheim, 2017) and usually on-demand and pay-as-you-go services. This platform type is an alternative to on-premises infrastructure, thus helping avoid investing in expensive resources. Magento 1 Enterprise Edition is an example of IaaS for e-commence.

Platform as a Service offers hardware and software tools over the internet, managing hardware tasks, which permits the user to control the application. Therefore is usually preferred by businesses that aim to maintain, customize and adapt the application by the development team

(Bear Group, 2022). Magento Commerce Cloud is an example of this type of e-commerce platform.

Software as a Service is similar to PaaS however, the user does not manage the software. The application is available over the internet from usually any device, and its security, compliance, and updating are handled by the service provider (Wesley Chai, 2021; Sana Commerce, 2022). Shopify, WooCommerce, WordPress and Wix are some of the most common SaaS e-commerce platforms (Folio3, 2020; Singh, 2021).

PaaS e SaaS solutions are typically easy to set up and they offer support and consequently are better options for smaller businesses just emerging in the e-commerce market (Adobe, 2022).

Besides every e-commerce platform being different and satisfying some unique and niche needs, they have some important features in common: the ability to search the store, a shopping cart where the client can store and view the intended purchases before the checkout, and also a payment gateway to facilitate the purchasing (Adobe, 2022). The payment gateway enables the payment to be done online and through different currencies (fiat and cryptocurrency).

4.2 Cryptocurrency Payment Gateways

A cryptocurrency payment gateway, also known as a cryptocurrency payment provider, is a dynamic payment processing system that allows retailers or vendors to accept payments in multiple cryptocurrencies. These systems are built on blockchains ecosystems, thus offering not only immutability and transparency of transactions but also improving transaction speed and reducing payment processing steps (LeewayHertz, 2021).

Some cryptocurrency payment gateways allow the merchant to decide if they prefer to receive the payment in cryptocurrency or a fiat currency. For the latter, the cryptocurrency is stored in an online wallet until the system converts it to fiat currency and then transfers it to the merchant account (Banguis, 2021).

Since these services are based on blockchain technology, then the benefits of this decentralized technology are reflected in it. Some of the benefits are (Gupta, 2020; Banguis, 2021; LeewayHertz, 2021): (a) Security, against tampering, fraud, and cybercrime, which is enabled by the repeated verifications from the nodes of the network; (b) Time savings, the transaction can be completed faster since it does not require verification by a central authority; and (c) Cost savings, which is possible by the less oversight of the network and there is no duplication of effort because all participants have access to the shared ledger.

Additionally, any merchants, even the ones that do not know how cryptocurrencies work, can easily provide their clients with the cryptocurrency payment method when using a cryptocurrency payment gateway. Most cryptocurrency payment gateway simplifies the process of consolidating the funds and setting up the conversions and transfers of a merchant.

They provide more confidence to the vendors because they are a point of contact if any payment issue occurs (Seth, 2022).

The present subsection pertains to the description of some of the cryptocurrency payment gateways present in the market, enumerating some of their features while emphasizing how they can help to solve the present problem in three different criteria: acceptance of cryptocurrency Bitcoin and more, provides to the merchant the ability to choose the currency of the settlement and ways to be integrated with external applications.

The next cryptocurrency payment gateways were obtained by the following method: (1) Searched for the 'top cryptocurrency payment gateways by clients' on Google; (2) Validated all websites shown on the first page and, in this case, was ten (and organized it into Table 18 present in Appendix B); (3) Selected the cryptocurrency payment gateways that occurred in more than half of the websites, which resulted in five cryptocurrency payment gateways; (4) Since none of those provides a free trial, then another five top cryptocurrency payment gateways that provide it were selected.

BitPay

BitPay was founded in 2011 and, two years after, announced that 10000 merchants throughout 164 countries were already using its services to take payments via the decentralized digital cryptocurrency (The BitPay Blog, 2013). It offers support not only for retail payments but also for pay-outs and billings. Additionally, it introduces a differentiated feature, BitPay Card, that enables EMV chip and contactless payments cryptocurrency to fiat currency (The BitPay Blog, 2020).

BitPay pools the merchant's sales each business day and deposits the balance in either fiat or one of the several cryptocurrencies (including Bitcoin) into the merchant's bank account or cryptocurrency wallet, respectively (BitPay, 2022a, 2022b). It also extends other features such as two-factor authentication, compatibility with a long list of cryptocurrency wallets, multi-user facility, open-source plugins for e-commerce platforms, provides an API for easy integration, refunds, and multilingual (supporting more than forty languages) (Bansal, 2021; Herman, 2021; James, 2021; BitPay, 2022c, 2022d).

CoinGate

Initiated in 2014, CoinGate is considered another strong contender after winning the title of Best Cryptocurrency Payment Gateway in 2021 (Global Brands, 2021).

It permits merchants to accept payments in fiat currencies such as Pounds, US dollars, Euro, and cryptocurrencies such as Bitcoin and altcoins, charging a 1% fee per transaction (CoinGate, 2022d, 2022e, 2022f). Furthermore, the pay-outs can be kept in their original form or be converted to fiat currency (CoinGate Blog, 2019; Modestas, 2021). Some cryptocurrencies can also be converted to other cryptocurrencies.

CoinGate provides different ways to be integrated into other systems, for example, through a plugin or modules for e-commerce applications, CoinGate API and even through payment buttons (Jurgita, 2021; Coinbase Help, 2022; CoinGate, 2022a, 2022b, 2022c).

Coinbase Commerce

Coinbase serves as both a cryptocurrency exchange and cryptocurrency payment gateway, being the latter called Coinbase Commerce (Coinbase Commerce, 2022a). Coinbase Commerce counts more than 8000 merchants and provides two plans, Self Managed and Coinbase Managed. The former permits the merchant to hold and manage the private keys and wallet, and also the merchant should handle the currency conversions. In the latter, the management of the private keys and wallet is done by Coinbase as well as the currency conversions.

In addition, it offers the opportunity for merchants to convert their cryptocurrency payments into a restricted list of fiat currency or stablecoin (Coinbase, 2020), being the withdrawal only possible through the Coinbase Exchange. It also has an API for payment processing, plugins for e-commerce platforms, such as Shopify and WooCommerce, two-factor authentication, refund, and immediate deposit (Coinbase Commerce, 2020, 2022c, 2022b; Coinbase, 2022c). On the other hand, this cryptocurrency payment gateway is currently operating in a limited number of countries worldwide (Coinbase, 2022a).

CoinPayments

CoinPayments was launched in 2013 and became the first payment processor to support altcoins, in other words, alternative cryptocurrencies to Bitcoin (CoinPayments Blog, 2022). It covers a broad variety of cryptocurrencies at only 0.5% transaction fees, being one of the most inexpensive commissions (CoinPayments, 2022b).

Besides accepting payments in almost 175 cryptocurrencies, it additionally supports cryptocurrency to fiat conversion services for all of them and provides not just plugins with nearly 25 popular e-Commerce operating systems, but also an API to access services and information (CoinPayments Blog, 2018a, 2018b, 2021; CoinPayments, 2022a, 2022c, 2022d).

NOWPayments

NOWPayments empowers vendors to not only accept a large variety of cryptocurrencies (more than 50) but also some fiat currencies, choosing to receive in which currency they prefer (NOWPayments, 2020, 2022b, 2022g). This means that NOWPayments converts automatically fiat to cryptocurrency and cryptocurrency to fiat, being the profit of the former transferred to a wallet and the latter transferred to the bank account (NOWPayments, 2022b).

The operations are subjected to different fees, starting from 0.4% but, there is an additional 0.5% if the merchant decides to get paid in a different currency than what the client paid, and also a fixed rate exchange option to protect the merchant from the volatility of cryptocurrencies, which increases the base fee to 1% (NOWPayments, 2022e).

NOWPayments provides an API to be integrated their payment service into different platforms, an invoice feature that generates a QR Code that the client can scan to speed up the payment, plugins to be integrated into major e-commerce solutions and also custom solutions that can be requested (NOWPayments, 2022a, 2022c, 2022d, 2022f).

SpicePay

SpicePay was created in 2009 and it provides services, such as plugins for e-commerce platforms, buttons, hosted page, API, and email invoice, that allows merchants to accept cryptocurrencies (Das, 2022; SpicePay, 2022b; WP Favs, 2022).

Although the funds are deposited by default to the merchant's SpicePay Wallet, the merchant can opt for automatic withdrawals. These automatic withdrawals can be accomplished to another wallet address, or by Single Euro Payments Area (SEPA), Paypal, or prepaid debit card (SpicePay, 2022a). The pay-outs can therefore be kept on the original cryptocurrency or automatically convert to US Dollars, Euro, Pound Sterling or Canadian Dollars.

SpicePay offers new merchants a limited period of free transactions, and once ended starts charging a 1% fee per transaction (Davies, 2021; SourceForge, 2022e; SpicePay, 2022a).

ALFACoins

ALFACoins offers the feature CoinSplit which permits payments to be split between fiat and cryptocurrency (Bansal, 2021; Davies, 2021). Another feature offered is Bitsend which allows to directly transfer the cryptocurrencies to employees (as salaries or bonuses) and clients (as pay-outs) (Bitcoinist, 2016).

ALFACoins charges a 0.99% fee per transaction, and the merchant can choose who pays it (the merchant or the client) (Das, 2022; SourceForge, 2022a). It supports cryptocurrencies such as Bitcoin, Ethereum, Litecoin, Ripple, and more and the withdrawal can be done in US Dollars or Euro (Bansal, 2021).

ALFACoins provides an API, instant payment notification, payment button, refunds, and WordPress plugin to be integrated with e-commerce systems (ALFAcoins, 2022a, 2022b, 2022c). The payments, however, can be processed in any country except for Iran and North Korea (Davies, 2021).

Plisio

Plisio, founded in 2019, provides two different services for merchants to accept cryptocurrencies. The first is called Gateway API and provides an API to be connected to the merchant's system. Although it charges a 0.5% fee per transaction on this service, this can be reduced when doing mass pay-outs (i.e., pool multiple transactions at a time). The other service is White Label which enables the client to be redirected to the Plisio system and, consequently, the merchant can customize the invoice design. The charge in this service is a 1.5% transaction fee (Plisio, 2022c, 2022d, 2022f; SourceForge, 2022d).

Plisio supports the cryptocurrencies Bitcoin, Ethereum, Litecoin, Dash, Dogecoin, Zcash, Bitcoin, Cash, and Monero, and can be integrated with e-commerce solutions, for example, Magento and WooCommerce (Plisio, 2022a, 2022b, 2022d, 2022e).

Blockonomics

Blockonomics was launched in 2015 and provides solutions such as plugins, payment buttons, payment links, refunds, and an API that enables integration with other platforms (Blockonomics, 2022a, 2022b, 2022d). The payment links contain a QR Code and can be sent through email, instant messaging or social networks (Blockonomics, 2022a).

Blockonomics only support Bitcoin and Bitcoin Cash transactions, charging a 1% fee per each (Blockonomics, 2022a). An appealing feature of Blockonomics is Wallet Watcher. It permits merchants to track, monitor and export all the transaction information and get notified by email once a transaction happens (Blockonomics, 2022c).

COINQVEST

COINQVEST originated in 2018 and, in 2021, won the opening round of the Stellar Seed Fund, receiving a prize of 500000 US dollars (Wong, 2021). It provides checkouts in cryptocurrency which are transferred to a cryptocurrency wallet or bank account by instantly converting the cryptocurrency to fiat currency, with charging fees between 0.5% and 1% (Olszowy, 2020b; COINQVEST, 2022d; SourceForge, 2022c).

In addition, it assists the integration with plugins for certain e-commerce platforms, such as WooCommerce, Shopify and Magento, with a payment button and an API (COINQVEST, 2022a, 2022c). It supports multiple languages, enabling the merchant to set the checkout language, provides refund mechanisms, detailed transaction records and accounting, built-in payment exception handling and payment analytics (Olszowy, 2020a; COINQVEST, 2022b).

Comparison of Cryptocurrency Payment Gateways

By exploring the description of the presented system is possible to verify that most of them satisfy the criteria proposed for solving the present problem. In other words, most of the systems support payments of Bitcoin and more cryptocurrencies, enables the merchant to choose the currency of the settlement and can easily be integrated with external applications.

Table 8 and Table 9 present an overview of those and some other features provided by each cryptocurrency payment gateway previously described.

Almost all of them cover a significant list of payment and settlement currencies (cryptocurrency e fiat currency), excluding NOWPayments that currently only support Euro on settlement and Plisio and Blockonomics that do not have this feature. The pay-out time varies between instant and annual, being the instant pay-out time recommended since it helps eliminate currency volatility risks.

Another differing point is the feature for Multiple Wallets (in other words, the merchant can configure multiple wallets managing which currency goes to each wallet). CoinGate, SpicePay, Plisio, Blockonomics and COINQVEST are the ones that do not support it. Furthermore, the transaction fees charged by each cryptocurrency payment gateway vary between 0% and 1.5%, where CoinGate presents the lower and CoinPayments and Plisio the larger.

Finally, all the systems support refunds and provide an API for easy integration. However, SpicePay API is not freely accessible.

Table 8 - Comparison of different cryptocurrency payment gateways regarding some characteristics

	BitPay	Coinbase Commerce	CoinPayments	NOWPayments	CoinGate
Payment Currencies	Bitcoin, Bitcoin Cash, Dogecoin, Ethereum, Gemini US Dollar, Circle USD Coin, Paxos Standard USD, Binance USD, Dai, Wrapped Bitcoin, and Ripple	Bitcoin, Bitcoin Cash, Dai, Litecoin, USD Coin, Ethereum, and Dogecoin	Bitcoin, Bitcoin Cash, Bytecoin, Ethereum, Ether, Electroneum, Litecoin, Qtum, Velas, Apollo, Permission, Dash, Divi, … (more than 175 in total)	All Cryptocurrencies and Euros, US Dollars, Pounds Sterling, Russian ruble, Turkish lira, and Ukrainian hryvnia	Bitcoin, Litecoin, Ethereum, Bitcoin Cash, Ripple, Nano, Dai, Dogecoin, Stellar, Aragon, Civic, OmiseGo, DigiByte, … (more than 80 in total)
Fiat Settlement Currencies	US Dollars, Australian Dollars, Canadian Dollars, Pounds Sterling, Mexican Pesos, New Zealand Dollars, South African Rand, and Euros	US Dollars, Euros, Pounds Sterling, and USD Coin	US Dollars, Chinese yuan renminbi, Brazilian real and Mexican Pesos	Euros	Euros and US Dollars
Crypto Settlement Currencies	Bitcoin, Bitcoin Cash, Dogecoin, Ether, Gemini US Dollar, Circle USD Coin, Paxos Standard USD, Binance USD, Dai, Wrapped Bitcoin, and Ripple	Bitcoin, Bitcoin Cash, Dai, Litecoin, USD Coin, Ethereum, and Dogecoin	Bitcoin, Bitcoin Cash, Bytecoin, Ether, Electroneum, Litecoin, Qtum, Velas, Apollo, Permission, Dash, Divi, … (more than 175 in total)	Bitcoin, Bitcoin Cash, Dogecoin, Ether, Ripple, Tether, Litecoin, Dash, Solana, Cardano, Shiba Inu, Uniswap, Dai, … (more than 50 in total)	Bitcoin, Litecoin, Ethereum, Bitcoin Cash, Ripple, Nano, Dai, Dogecoin, Stellar, Aragon, Civic, OmiseGo, DigiByte, … (more than 80 in total)
Pay-out Time	Daily	Instant	Daily	Instant	In max two days
Refunds	Yes	Yes	Yes	Yes	Yes
Multiple Wallets	Yes	Yes	Yes	Yes	No
Free Trial	No	No	No	No	No
Transaction Fee	1%	1%	0.5 - 1.5%	0.4 - 1%	0 - 1%
Free access to API	Yes	Yes	Yes	Yes	Yes
E-commerce platforms	WooCommerce, Magento 1, Magento 2, Shopify, Wix, etc.	Shopify and WooCommerce	Magento 1, Magento 2, WooCommerce, etc.	Shopify, WooCommerce, Magento 2, etc.	WooCommerce, Magento 2, etc.

Table 9 - Comparison of different cryptocurrency payment gateways (with free trial) regarding some characteristics

	SpicePay	ALFACoins	Plisio	Blockonomics	COINQVEST
Payment Currencies	Bitcoin, Bitcoin Cash, Ethereum, and Litecoin	Bitcoin, Bitcoin Cash, Litecoin, Ethereum, Dash, Tether ERC-20, XRP	Bitcoin, Ethereum, Litecoin, Dash, Dogecoin, Zcash, Bitcoin, Cash, and Monero	Bitcoin and Bitcoin Cash	Bitcoin, Litecoin, Ethereum, Stellar, and any asset on the Stellar Network
Fiat Settlement Currencies	US Dollars, Pounds Sterling, Canadian Dollars, and Euros	US Dollars and Euros	-	-	Argentine pesos, Brazilian real, Euros, Nigerian naira, US Dollars
Crypto Settlement Currencies	Bitcoin, Bitcoin Cash, Ethereum, and Litecoin	Bitcoin, Bitcoin Cash, Litecoin, Ethereum, Dash, Tether ERC-20, XRP	Bitcoin, Ethereum, Litecoin, Dash, Dogecoin, Zcash, Bitcoin, Cash, and Monero	Bitcoin and Bitcoin Cash	Bitcoin, Litecoin, Ethereum, Stellar, USD Coin and Euro Tether
Pay-out Time	Less than 24 hours	daily/monthly/quarterly/ annual	Merchant initiated	Instant	Instant
Refunds	Yes	Yes	Yes	Yes	Yes
Multiple Wallets	No	No	Yes	No	No
Free Trial	Yes	Yes	Yes	Yes	Yes
Transaction Fee	1%	0.99%	0.5% - 1.5%	1%	0.5 - 1%
Free access to API	No	Yes	Yes	Yes	Yes
E-commerce platforms	WooCommerce, Magento 1, Magento 2, etc.	WordPress	WooCommerce, Magento, etc.	Shopify, WordPress, Wix, etc.	Shopify, WooCommerce, Magento and WordPress

4.3 Cryptocurrency Wallet

Cryptocurrency Wallet, contrary to what the name might suggest, does not store cryptocurrencies, instead, it stores the private keys and the address, in other words, the passwords that give access to cryptocurrencies. Cryptocurrencies are on the blockchain network but can only be accessed through the private key, thus proving ownership of the cryptocurrencies (Yaga *et al.*, 2018; Suratkar, Shirole and Bhirud, 2020; Coinbase, 2022b).

Besides storing addresses and private keys, wallets must also create them as well as calculate and display the balance of each address and carry out transactions (Gentilal, Martins and Sousa, 2017; Yaga *et al.*, 2018)

One can define cryptocurrency wallets as Hardware, Paper, Desktop, Web, and Mobile (Khan *et al.*, 2019; Karantias, 2020; Suratkar, Shirole and Bhirud, 2020).

Hardware Wallet generates and stores the keys on a USB stick, Bluetooth device or smartcard (Rezaeighaleh and Zou, 2019a). Although the keys are stored offline, transactions are done online, so connecting the wallet to a computer is sometimes necessary (Suratkar, Shirole and Bhirud, 2020; Coinbase, 2022b). One recommends that these wallets allow direct user interaction (through a screen and buttons), avoiding, therefore, the connection to a computer, as this increases the possibility of attacks (Rezaeighaleh and Zou, 2019b). These are hardware specifically designed to be a wallet, which provides even more increased security features (Bitcoin, 2022c). Examples of Hardware Wallets include BitBox02, Ledger Nano S, KeepKey, and Trezor One (Bitcoin, 2022b).

In Paper Wallets, the keys are printed on paper in the format of strings and QR Codes. Initially, these were very popular and recommended, but this is no longer the case as there are safer alternatives and the paper is fragile and therefore susceptible to water, humidity, fire, etc. (Gemini, 2021b).

Desktop Wallets are software installed on a computer and are considered to be secure, especially when encrypted with a password. Nevertheless, this security can be corrupted by viruses or malware as well as physical damage to the computer (Suratkar, Shirole and Bhirud, 2020; Iredale, 2021). Armony, Bitcoin Core, Bitcoin Knots, Bither, BitPay, and Electrum are some examples of this type of wallet (Bitcoin, 2022a).

Web Wallets are usually software made available in the cloud by a provider, and opting for one protected by two-step encryption is advisable (Coinbase, 2022b). As they are online and controlled by a third party, this type of wallet presents the inherent risks and vulnerabilities of these (Suratkar, Shirole and Bhirud, 2020). Popular Web Wallets involve Coinbase, CoinGate and Guarda (Thompson, 2020; Benzinga, 2022; Guarda, 2022).

Mobile Wallets are software installed on a mobile device that stores the keys (Suratkar, Shirole and Bhirud, 2020). This type of wallet is normally protected with a password generated by the wallet itself, called the seed phrase. (Coinbase, 2022d; CoinMarketCap Alexandria, 2022c). Abra, Coinomi, Jaxx Liberty, Electrum, Exodus and Mycelium are some examples of Mobile Wallets (CoinMarketCap Alexandria, 2022c).

These types of wallets are generally grouped into two types: Hot and Cold Wallets. Hot Wallets are those wallets that have access to the internet, making them more convenient for current use. However, they are more susceptible to a variety of attacks (Gentilal, Martins and Sousa, 2017; Suratkar, Shirole and Bhirud, 2020; Haar, 2021). Web wallets, mobile wallets, and desktop wallets are considered hot wallets (Gemini, 2021a). On the other hand, Cold Wallets are wallets that work in an offline environment, therefore, without an internet connection. For this reason, this type of wallet is considered more secure (Gentilal, Martins and Sousa, 2017; Suratkar, Shirole and Bhirud, 2020; Geroni, 2021; Lielacher, 2021). Hardware and Paper belong to this type of wallet (Gemini, 2021a).

5 Solution Requirements and Analysis

This chapter details the objectives and requirements gathered for the intended solution. It states the constraints and concerns to take into consideration when designing, implementing, and testing the problem's solution. In addition, presents the selection of the cryptocurrency payment gateways to be integrated with the solution. It describes a decision criterion that should be considered by the solution when choosing the cryptocurrency payment gateway to process the transaction. Finally, it demonstrates the domain model in which business concepts are represented.

5.1 Objectives

The main goal of this thesis is the development of a prototype capable of supporting the cryptocurrency Bitcoin, as stated in Chapter 1. After identifying and comparing approaches to cryptocurrency transaction processing, in Chapter 3, the usage of multiple cryptocurrency payment gateways was concluded to be the best approach to follow. Thus, the following activities are planned:

- Analyse the suitability of different cryptocurrency payment gateways and decision criteria that should be considered when selecting a cryptocurrency payment gateway.
- Design and implement a prototype integrating with multiple cryptocurrency payment gateways.
- Validate the solution in terms of some quality attributes.

5.2 Requirements

The main objective of this thesis is the development of a prototype capable of supporting the cryptocurrency Bitcoin, integrating multiple cryptocurrency payment gateways where the most adequate is chosen to process the transaction. The cryptocurrency payment gateway is chosen accordingly to a decision’s criteria. Table 10 enumerates the functional requirements.

Table 10 - System functional requirements

Functional requirement Identification	Description
RQ1	The system should be able to support multiple cryptocurrency payment gateways at the same time.
RQ2	The system should choose the cryptocurrency payment gateway based on decision criteria.
RQ3	The system should be able to authenticate and authorise merchants.
RQ4	The system should be able to perform Bitcoin transactions.
RQ5	The system should be able to perform a payment transaction.

The goals of this thesis are defined to help merchants integrate Bitcoin cryptocurrency payment into their e-commerce websites. Therefore, the main actors of this solution are the merchants themselves, which mainly interact with the system and perform the functional operations.

The UML Use Case Diagram in Figure 15 identifies the functional requirements through the definition of use cases of the system considering the functional analysis performed in Section 3.5. Table 11 describes each identified use case.

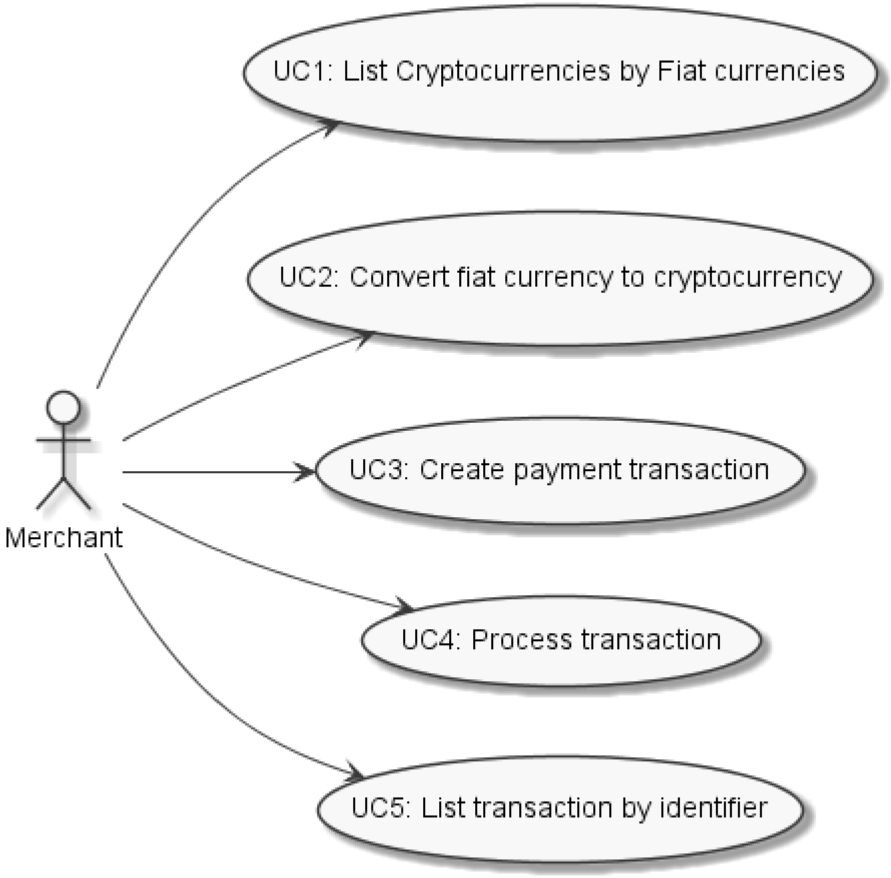

UC1: List Cryptocurrencies by Fiat currencies
UC2: Convert fiat currency to cryptocurrency
UC3: Create payment transaction
UC4: Process transaction
UC5: List transaction by identifier
Merchant

Figure 15 - System functional requirements

Table 11 - Description of use cases

Use Case Identification	Description
UC1	The authenticated and authorised merchant lists the cryptocurrencies that can be converted from a fiat currency.
UC2	The authenticated and authorised merchant converts a fiat currency to a cryptocurrency.
UC3	The authenticated and authorised merchant initiates a cryptocurrency payment transaction.
UC4	The authenticated and authorised merchant process the payment of a previously created transaction.
UC5	The authenticated and authorised merchant lists all transactions.

Table 12 describes the requirements in terms of some quality attributes (Maintainability, Security, Reliability, Traceability, and Interchangeability).

Table 12 - Description of solution’s quality attributes

Identification	Quality Attribute	Description
QA1	Maintainability	• The technical debt ratio of the system should be less than 5%; • There should not be more than 30 code smells; • The system must be well documented.
QA2	Security	• The system must not have any vulnerability; • The system must not have dependencies with vulnerabilities; • The system needs to protect details of transactions and customers from internal and external fraud/criminal usage; • The system must use the HTTPS protocol for data transmissions.
QA3	Reliability	• The system must not have any bugs ensured by an external service; • The system is expected to be available 99.99% of the time.
QA4	Traceability	• The system should trace the transactions and communications.
QA5	Interchangeability	• The system should be able to be changed without the need to change the merchant website; • The system should be able to have different configurations and endpoints of communication.

5.3 Constraints and Concerns

The solution should obey some constraints regarding its design and implementation, enumerated in Table 13.

Table 13 - Constraints of the system

Identification	Description
CON1	Adoption of open-source technologies.
CON2	The system must be accessible from a variety of platforms.
CON3	The communications between components must be secure.
CON4	The system should integrate with cryptocurrency payment gateways that allow the settlement in at least three fiat currencies.

Additionally, Table 14 identifies some concerns to consider.

Table 14 - Concerns

Identification	Description
CRN1	Activity log.
CRN2	Documentation.
CRN3	Authentication.
CRN4	Authorisation.
CRN5	Continuous integration.

5.4 Cryptocurrency Payment Gateways

Section 4.2 presents and describes some cryptocurrency payment gateways, which are the list of possible gateways to be integrated within the intended solution.

The integration with all of them is impossible with the time available to carry out this thesis, so some will be discarded.

SpicePay is the first cryptocurrency payment gateway to be excluded since it does not provide a free access API. It hampers the integration without creating an account, making it impossible to mock their service.

Plisio, Blockonomics, NOWPayments, CoinGate and ALFACoins are also excluded from integration since they do not comply with constraint CON4.

In outline, the solution must integrate with the remaining cryptocurrency payment gateways. These are BitPay, Coinbase Commerce, CoinPayments and COINQVEST.

5.5 Decision Criteria

As explained in Section 5.2, the system should be integrated with multiple cryptocurrency payment gateways (RQ1) and choose the one based on decision criteria (RQ2).

The chosen cryptocurrency payment gateways differentiate essentially on the list of cryptocurrencies supported during payment. For this reason, the decision criterion will be the supported cryptocurrencies for payment.

CoinPayments is the cryptocurrency payment gateway that supports more cryptocurrencies, then BitPay, COINQVEST and finally Coinbase Commerce.

One can consider an ordered list of cryptocurrency payment gateways that support a specific cryptocurrency. In this regard, the application avoids dependency on only one cryptocurrency payment gateway. In case of one cryptocurrency payment gateway is not available, others can be available to perform the transaction. It will help with the reliability (QA3) of the application.

The application should assess the following to obtain an ordered list of cryptocurrency payment gateways:

(1) If CoinPayments supports the transaction's cryptocurrency, then this cryptocurrency payment gateway is considered.
(2) If BitPay supports the transaction's cryptocurrency, then it is taken into consideration.
(3) If COINQVEST supports the transaction's cryptocurrency, then this cryptocurrency payment gateway is considered.
(4) If Coinbase Commerce supports the transaction's cryptocurrency, then this cryptocurrency payment gateway is considered.

The UML Activity Diagram in Figure 16 illustrates this flow.

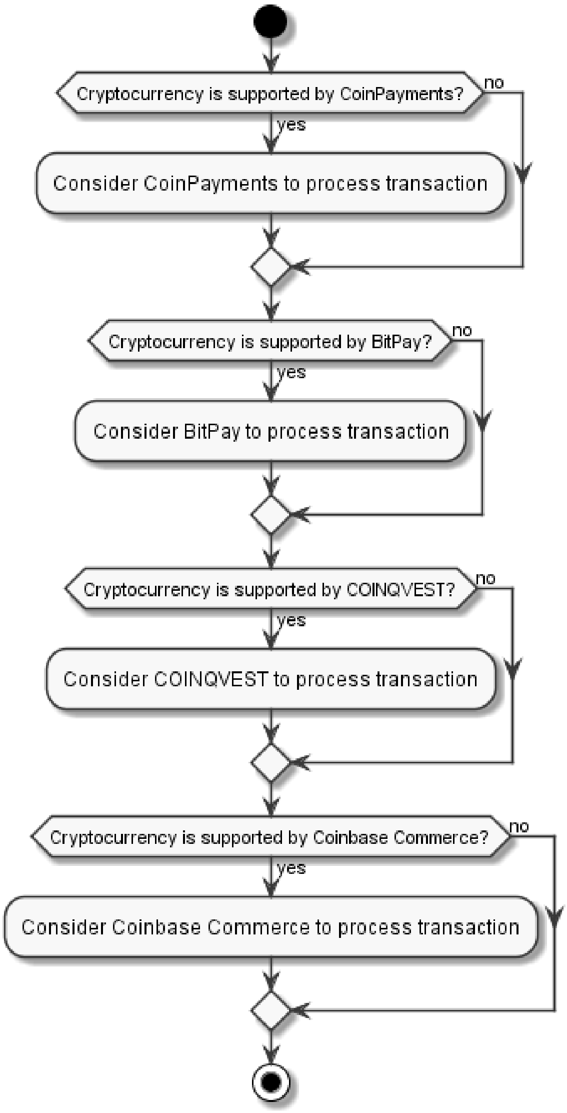

Cryptocurrency is supported by CoinPayments?
no
yes
Consider CoinPayments to process transaction
Cryptocurrency is supported by BitPay?
no
yes
Consider BitPay to process transaction
Cryptocurrency is supported by COINQVEST?
no
yes
Consider COINQVEST to process transaction
Cryptocurrency is supported by Coinbase Commerce?
no
yes
Consider Coinbase Commerce to process transaction

Figure 16 - Decision regarding the cryptocurrencies payment gateway which will process the transaction

5.6 Domain Model

The UML Class Diagram in Figure 17 presents the domain model to enhance the comprehension of the system domain and concepts that impact the business. It represents the classes of domains and their cardinalities.

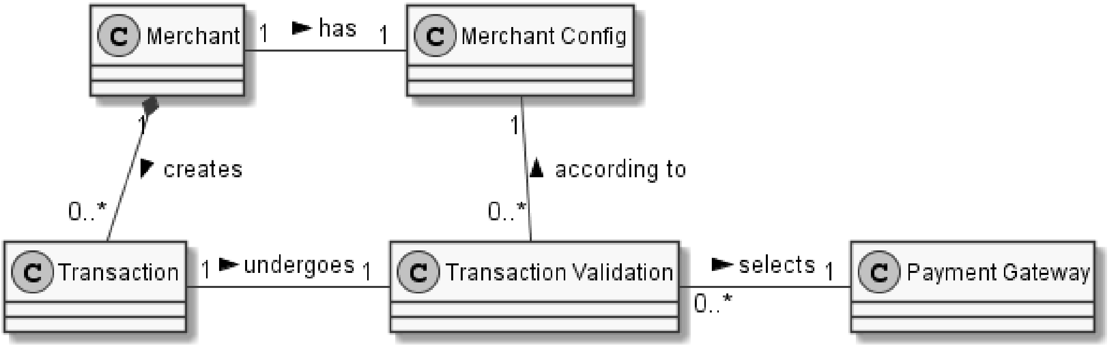


Figure 17 - Domain model diagram

The analysis of Figure 17 allows one to conclude the existence of:

- The concept Merchant represents the entity Merchant that will use the system. It can convert the fiat currency to a cryptocurrency, initiate a transaction and list the transactions.
- Transaction is the entity that represents a financial transaction of cryptocurrencies, which in this case is a payment.
- Payment Gateway is the representative entity of the cryptocurrency payment gateway, and it aggregates some information related to it.
- The concept Merchant Config represents the entity that represents the Merchant configuration.
- Transaction Validation is the entity that intends to represent the validation of the transaction and the decision of the proper cryptocurrency payment gateway to process it.

6 Solution Design

This chapter aids in focusing on the overall concept of the application, its architecture and design choices. It helps identify the main components for the final prototype and layout the relationships between them based on the requirements gathered and the analysis previously conducted.

6.1 Possible Approaches

The expected result (a prototype capable of supporting cryptocurrencies) can be achieved from distinct approaches given the requirements defined in Section 5.2.

Accordingly, two design approaches are presented, analysed, and compared. Both alternatives are introduced from a coarse-grained design perspective, considering that they intend to represent architectural decisions without going into detail about the specifics of each component.

6.1.1 Segregation of responsibilities

The UML Component Diagram in Figure 18 represents the components that composite the system proposed with segregation of responsibilities. It also illustrates how they interact with each other.

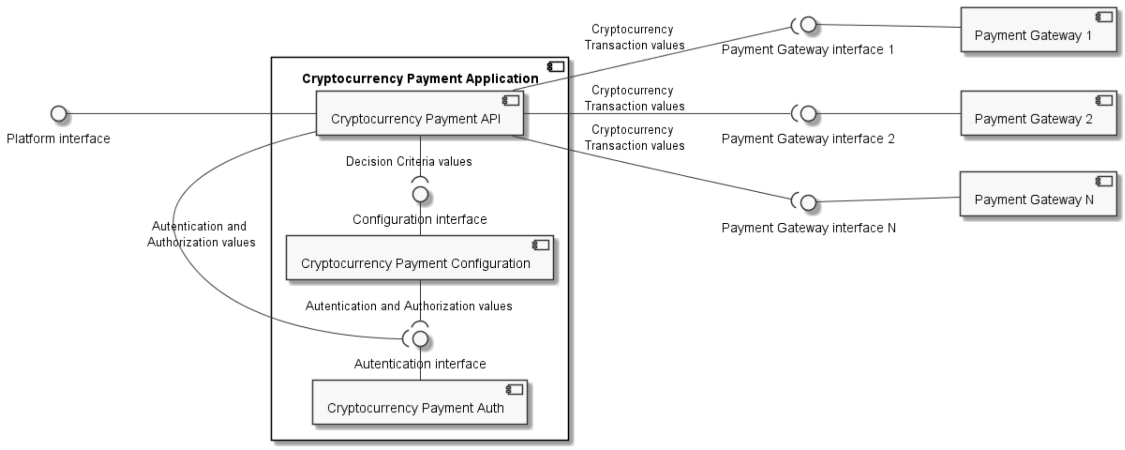

Cryptocurrency Payment Application
Platform interface
Cryptocurrency Payment API
Decision Criteria values
Configuration interface
Cryptocurrency Payment Configuration
Autentication and Authorization values
Autentication interface
Cryptocurrency Payment Auth
Autentication and Authorization values
Cryptocurrency Transaction values
Cryptocurrency Transaction values
Cryptocurrency Transaction values
Payment Gateway interface 1
Payment Gateway interface 2
Payment Gateway interface N
Payment Gateway 1
Payment Gateway 2
Payment Gateway N

Figure 18 - High-level design approach of the system with segregation of responsibilities, presented in a UML Component Diagram

The analysis of Figure 18 allows one to conclude the existence of different components:

- Cryptocurrency Payment API is the component that has the responsibility to process and list cryptocurrency transactions. Thereby, it communicates with the components of the cryptocurrency payment gateways, Cryptocurrency Payment Configuration and Cryptocurrency Payment Auth. It provides an interface that can be consumed by the merchants' services and user interfaces (UIs).
- Cryptocurrency Payment Configuration is the component responsible to manage the decision of the cryptocurrency payment gateway that will process a transaction. It consumes the interface provided by the component Authentication/Authorization API.
- The component Cryptocurrency Payment Auth is responsible for the authentication and authorization of the merchant.
- The components Payment Gateway 1, Payment Gateway 1 and Payment Gateway N represent the cryptocurrency payment gateways that will be integrated into the solution.

In this way, the segregation of responsibilities relies upon the existence of different API abstracting and distributing, therefore, the domain logic.

6.1.2 API Centralization

The UML Component Diagram in Figure 19 depicts the components of the system with API centralization and the interaction with others, in this case, external services.

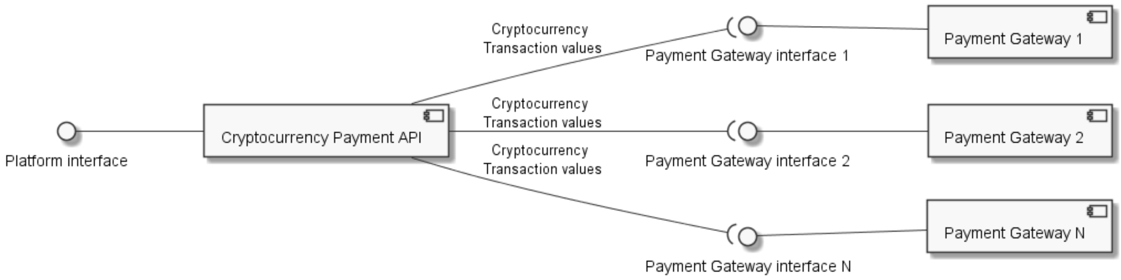

Platform interface
Cryptocurrency Payment API
Cryptocurrency Transaction values
Cryptocurrency Transaction values
Cryptocurrency Transaction values
Payment Gateway interface 1
Payment Gateway interface 2
Payment Gateway interface N
Payment Gateway 1
Payment Gateway 2
Payment Gateway N

Figure 19 - High-level design approach of the system with API centralization

The component Cryptocurrency Payment API centralizes the logic of processing transactions, configuring the decision criteria, and managing the authentication and authorization of the merchants. Consequently, the database will persist all the information regarding transactions, merchants, and decision criteria configuration.

6.1.3 Design Solution Evaluation and Choice

The API centralization solution can be worthwhile as it facilitates the hosting of the services. Nevertheless, centralizing responsibilities implies using the same technology for application development and data persistence. It implicates an added difficulty in the case of future maintenance.

The same does not occur in the segregation of responsibilities approach. Although in this solution the logistic of hosting the services are more complex, the gain accrues from the scalability of the services and this maintenance. On the other hand, the creation of an API whose responsibility is solely to manage the configuration of decision criteria for choosing a cryptocurrency payment gateway for a given transaction facilitates and enhances the evolution of this configuration. For now, this configuration is static and the same for all merchants, however, this approach makes it easier for this configuration to be set up by each merchant. Additionally, the existence of an API only responsible for authentication and authorization helps in the maintenance and extension of the other APIs.

This segregation of responsibilities complies with the Separation of Concerns design concept. It aims to subdivide the complexity of a problem into small layers to be developed and optimized independently, thus reducing development effort and time and improving layer management (Pressman, 2010). Accordingly is possible to conclude that the segregation of responsibilities approach is the most appropriate.

6.2 Detailed Design

Once the following architectural approach is defined, it becomes relevant to detail the structure of the focal component, Transaction API, moving from coarse to fine granularity.

Figure 20 demonstrates the components and their relations that constitute the Cryptocurrency Payment API. It represents a logical view presented in a UML Component Diagram.

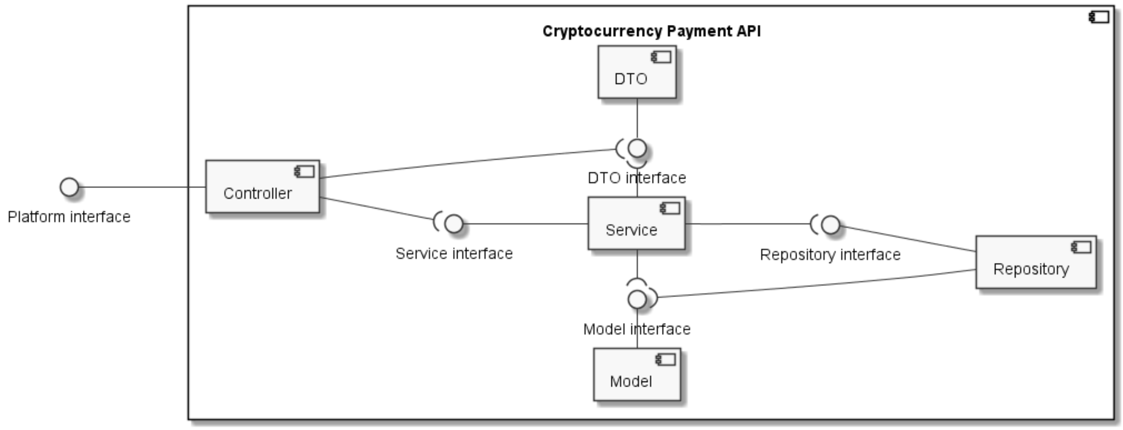


Figure 20 - Logical View of component Cryptocurrency Payment API

The following components are possible to be specified:

- Controller is the entry point of the REST API and has the responsibility to receive and send HTTP requests and responses, respectively. This component is a gateway between the HTTP information and the domain logic.
- Service is responsible for handling the business logic and provides an abstraction to the persistence layer. They are in control of mapping the DTO to domain classes and vice versa. In this way, controllers do not have knowledge about domain classes and, consequently, access to business logic.
- Model is an object that represents some information about the business domain. Manage system data and associated operations.
- DTO represents objects that only contain the information needed by the presentation layer. In this way, irrelevant information and business logic are not exposed.
- Repository is responsible for accessing the database and applying database operations. It contributes to the isolation of the data access layer from the business layer.

Figure 21 exemplifies a process view aiming to represent the relations between the system's components. The UML Sequence Diagram illustrates the functional interaction of UC1, UC2, UC3 and UC4.

The customer starts the payment process by interacting with the Merchant's UI and consequently calls the Cryptocurrency Payment Application to list the cryptocurrencies that can be converted from the fiat currency. The Cryptocurrency Payment Application will authenticate and authorize the merchant and return the list of cryptocurrencies to the Merchant's UI. So the customer can select the preferred cryptocurrency.

Once the client chooses the cryptocurrency, the Merchant's UI will call the Cryptocurrency Payment Application to convert the fiat currency to the cryptocurrency. The Cryptocurrency

Payment Application will authenticate and authorize the merchant, identify the cryptocurrency payment gateway that should be used in the transaction and call it to obtain the converted cryptocurrency.

After the converted cryptocurrency, the customer can then pay with the preferred cryptocurrency. The Cryptocurrency Payment Application is called to create the payment transaction, which will once again authenticate and authorize the merchant, identify the cryptocurrency payment gateway, and call it to create the payment transaction. The system will return to the Merchant's UI the payment information (such as wallet address and/or QR Code), so the customer can proceed with the transfer of the chosen cryptocurrency.

Once the customer transfers the cryptocurrency, the network of the cryptocurrency receives the transaction and broadcasts the payment to the cryptocurrency payment gateway. It will notify the Cryptocurrency Payment Application, which should handle the request respectably. The result of the payment will be shown to the customer.

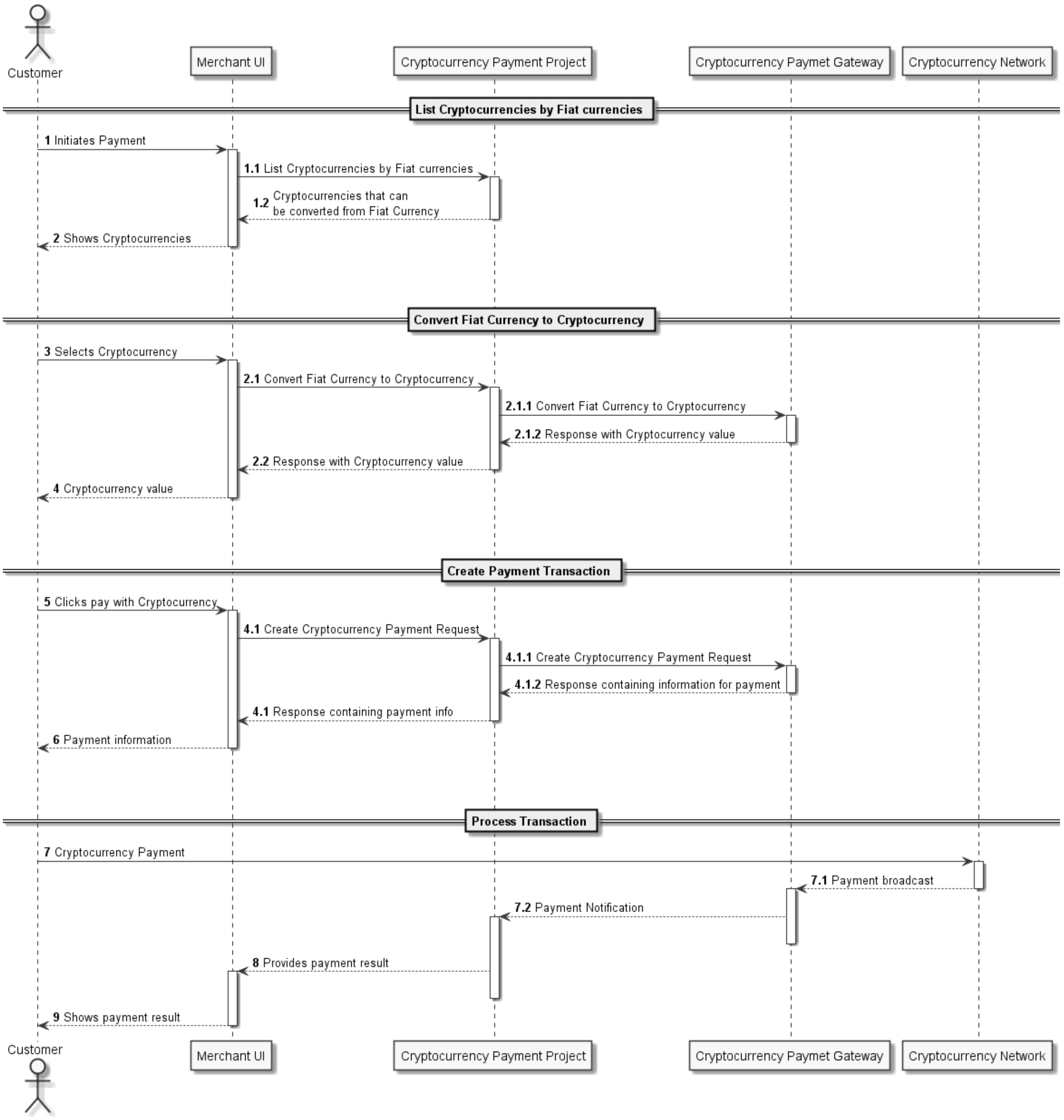

Customer
Merchant UI
Cryptocurrency Payment Project
Cryptocurrency Paymet Gateway
Cryptocurrency Network
List Cryptocurrencies by Fiat currencies
1 Initiates Payment
1.1 List Cryptocurrencies by Fiat currencies
1.2 Cryptocurrencies that can be converted from Fiat Currency
2 Shows Cryptocurrencies
Convert Fiat Currency to Cryptocurrency
3 Selects Cryptocurrency
2.1 Convert Fiat Currency to Cryptocurrency
2.1.1 Convert Fiat Currency to Cryptocurrency
2.1.2 Response with Cryptocurrency value
2.2 Response with Cryptocurrency value
4 Cryptocurrency value
Create Payment Transaction
5 Clicks pay with Cryptocurrency
4.1 Create Cryptocurrency Payment Request
4.1.1 Create Cryptocurrency Payment Request
4.1.2 Response containing information for payment
4.1 Response containing payment info
6 Payment information
Process Transaction
7 Cryptocurrency Payment
7.1 Payment broadcast
7.2 Payment Notification
8 Provides payment result
9 Shows payment result
Customer
Merchant UI
Cryptocurrency Payment Project
Cryptocurrency Paymet Gateway
Cryptocurrency Network

Figure 21 - Process view of use cases UC1, UC2, UC3 and UC4

Figure 22 represents the aggregates of the Cryptocurrency Payment Application, which can be treated as a single unit. There are two aggregates with the entities Merchant and Transaction as their aggregate roots. In this case, the aggregate Transaction has a reference to the aggregate root Merchant to identify the financial transaction initiator, and it is composed of value objects that contain more information about it.

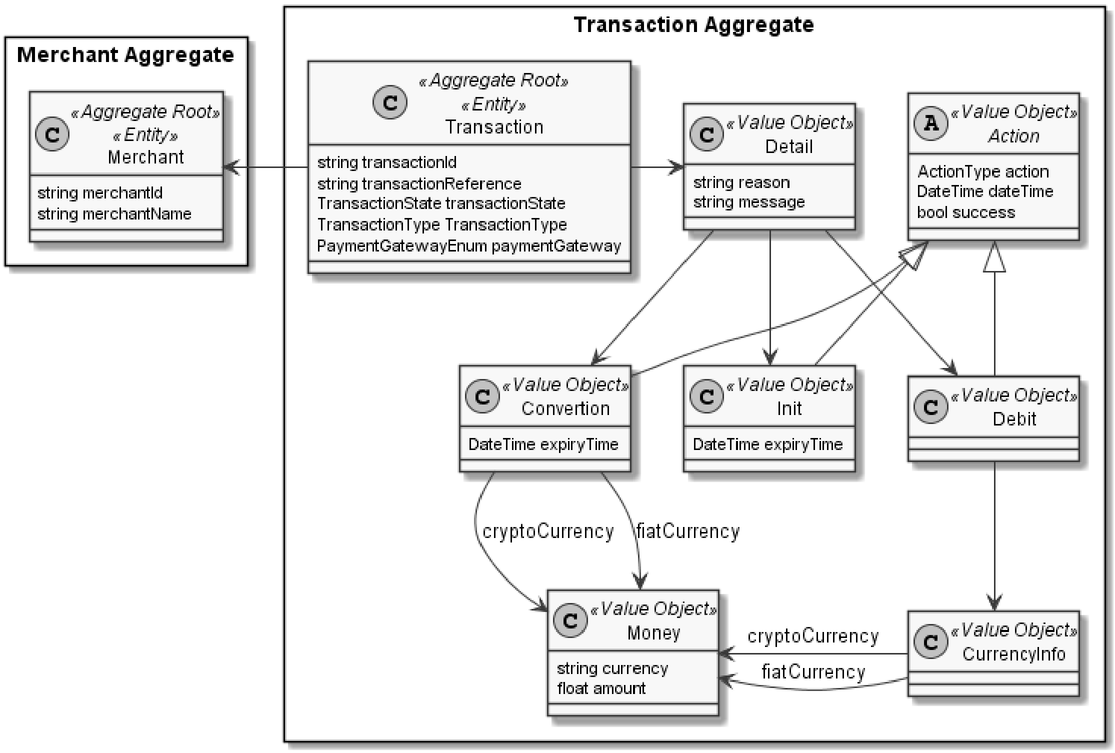


Figure 22 - UML Class Diagram for Transaction Aggregate

7 Solution Implementation

This chapter delves into the implementation details and how the use cases are achieved on a more technical level. It describes the implementation of some requirements that are present in all use cases and, finally, the use cases.

The source code of this prototype and additional documentation are available on GitHub (Santos, 2022).

7.1 Authentication and Authorization

As referred on all use cases and CRN3 and CRN4, only an authenticated and authorized merchant can access the services provided by the Cryptocurrency Payment Application. An application responsible for such functionalities was created to separate the responsibilities, as described in sections 6.1.1 and 6.1.3.

The Basic access authentication was implemented, which means that in each HTTP request, the merchant must specify their username and password. In Basic HTTP authentication, the Authorization header passes to the API a Base64 encoded string representing the username and password joined by a single colon : appended to the text Basic as follows:

$$Basic < Base64(username\colon password) >$$

In the CryptocurrencyPaymentAPI was created a handler (which extends the class `Microsoft.AspNetCore.Authentication.AuthenticationHandler`) responsible to communicate with the component CryptocurrencyPaymentAuth. It sends a request with the Authorization header to know if the merchant is authenticated and authorized.

Code Snippet 1 shows that the presence of the Header Authorization (lines 3-4) is validated, and the request to CryptocurrencyPaymentAuth is done (lines 20-24). In the case of a valid merchant, the Authentication Success is returned to the base Authentication middleware (lines 41-43). Additionally, the answer is saved on the Context (line 33), so it can be accessed during the HTTP request (Code Snippet 2, lines 7-8). On the other hand, if the authenticity of the merchant is not verified, the CryptocurrencyPaymentAuth application will throw an exception. This exception is handled here, and hence `NotAuthorizedException` is thrown, and the HTTP request is not performed (lines 46-50).

```
1.  protected override Task<AuthenticateResult> HandleAuthenticateAsync()
2.  {
3.      bool hasAuthHeader = Request.Headers.
4.          TryGetValue("Authorization", out StringValues authHeaderStringValues);
5.      if (!hasAuthHeader)
6.      {
7.          throw new NotAuthorizedException(MissingAuthorizationHeader.Message);
8.      }
9.
10.     try
11.     {
12.         var authEndPoint = this.configuration.GetSection("AuthEndPoint")?.Value;
13.         if (authEndPoint == null)
14.         {
15.             throw new ServiceUnavailableException(
16.                       ErrorCodes.AuthorizationServiceNotResponding.Message);
17.         }
18.
19.         var authHeader = authHeaderStringValues.ToString();
20.         var authResponse = restClient.Get<MerchantAuthorizationDto>(
21.             authEndPoint,
22.             string.Empty,
23.             out var responseHeaders,
24.             new Dictionary<string, string>() { { "Authorization", authHeader } });
25.
26.
27.         if (authResponse == null)
28.         {
29.             throw new ServiceUnavailableException(
30.                       AuthorizationServiceNotResponding.Message);
31.         }
32.
33.         Context.Items["authorizationRequest"] = authResponse;
34.
35.         var claims = new[] {
36.                     new Claim(ClaimTypes.NameIdentifier, authResponse.MerchantId),
37.                     new Claim(ClaimTypes.Role, "Merchant")
38.                 };
39.         var identity = new ClaimsIdentity(claims, "Basic");
40.         var claimsPrincipal = new ClaimsPrincipal(identity);
41.         return Task.FromResult(
42.             AuthenticateResult.Success(
43.                 new AuthenticationTicket(claimsPrincipal, Scheme.Name)));
44.
45.     }
46.     catch (RestClientException ex)
47.     {
48.         if (ex.Status == (int)System.Net.HttpStatusCode.Unauthorized)
49.         {
50.             throw new NotAuthorizedException(ex.Message);
51.         }
52.         else
53.         {
54.             throw new ServiceUnavailableException(
55.                       AuthorizationServiceNotResponding.Message);
56.         }
57.     }
58. }
```

Code Snippet 1 - Function that handles the authentication/authorization of the merchant (Application: CryptocurrencyPaymentAPI)

```
[HttpPost("{transactionId}")]
public async Task<ActionResult<GetInitTransactionDto>> ConfirmPaymentTransaction(
     [FromRoute] string transactionId)
{
    log.Info($"Confirm Payment transaction '{transactionId}'");
    // Added on BasicAuthenticationHandler
    var authorizationRequestDto =
              HttpContext?.Items["authorizationRequest"] as MerchantAuthorizationDto

    return Ok(await transactionService.
               CreatePaymentTransaction(authorizationRequestDto, transactionId));
}
```

Code Snippet 2 - Example of a function that reads an item from HTTPContext that was added during authentication/authorization of the merchant
(Application: CryptocurrencyPaymentAPI)

7.2 Exception Handling

A global handler middleware was created to catch all exceptions thrown by the application in one place. It removes the need for duplicate code for exception handling throughout the application. Moreover, it disables the possibility of returning to the client information about the source code.

Code Snippet 3 exhibits how this handler was implemented. The error of type `IException` returns its status code and error message (lines 25-29), and all other exceptions yield the 500 Internal Server Error response (lines 37-42).

The `IException` is an interface which contains the specification of the information (status code and error message) that should be available to provide to the end-users. This interface was created to minimize the changes required when creating a custom application exception, therefore applying the Open-Closed Principle. This principle helps guarantee modifiability and, consequently, the application's maintainability (QA1).

The following classes implement this interface:

- `NotAuthorizedException`, which is specific to the authentication/authorization process, returns 401 Unauthorized responses.
- An error of type `ValidationException` is specific to the request validation process and returns 400 Bad Request responses.
- An error of the type `ServiceUnavailableException` is specific to the communication process with the external services that the application depends on and returns 503 Service Unavailable responses.

This information is illustrated in the UML Class Diagram in Figure 25, Appendix C.

```
public class ExceptionHandlingMiddleware
{
    private static readonly ILog log =
            LogManager.GetLogger(MethodBase.GetCurrentMethod()?.DeclaringType);
    private readonly RequestDelegate _next;

    public ExceptionHandlingMiddleware(RequestDelegate next)
    {
        _next = next;
    }

    public async Task Invoke(HttpContext context)
    {
        try
        {
            await _next(context);
        }
        catch (Exception error)
        {
            log.Error(error);
            var response = context.Response;
            response.ContentType = "application/json";
            object message;

            if (error is IException exception)
            {
                // custom application error
                response.StatusCode = exception.StatusCode;
                message = exception.ErrorMessage;

                if (response.StatusCode == (int)HttpStatusCode.Unauthorized)
                {
                    response.Headers.WWWAuthenticate =
                                    "Basic realm=\"dotnetthoughts.net\"";
                }
            }
            else
            {
                // unhandled error
                response.StatusCode = (int)HttpStatusCode.InternalServerError;
                message = error?.Message ?? string.Empty;
            }

            var result = JsonSerializer.Serialize(new ExceptionResult(message));
            await response.WriteAsync(result);
        }
    }
}

public class ExceptionResult
{
    public object Message { get; set; }

    public ExceptionResult() { Message = string.Empty; }

    public ExceptionResult(object message) { Message = message; }
}
```

Code Snippet 3 - Handler class responsible for catching and handling exceptions (Application: CryptocurrencyPaymentAPI)

7.3 Transaction Status

From Figure 21, one can see that a financial transaction goes through different processes and, consequently, states. The UML State Diagram in Figure 23 shows the events and financial transaction states.

When converting fiat currency to cryptocurrency (UC2), the financial transaction is in the "Currency Converted" state. Next, the transaction can only go to the "Initialized" state when the transaction creation request is received (UC3). Finally, the transaction is processed (UC4), where notification may be successful, thus moving to the "Transmitted" state, or unsuccessful, being the "Failed" state.

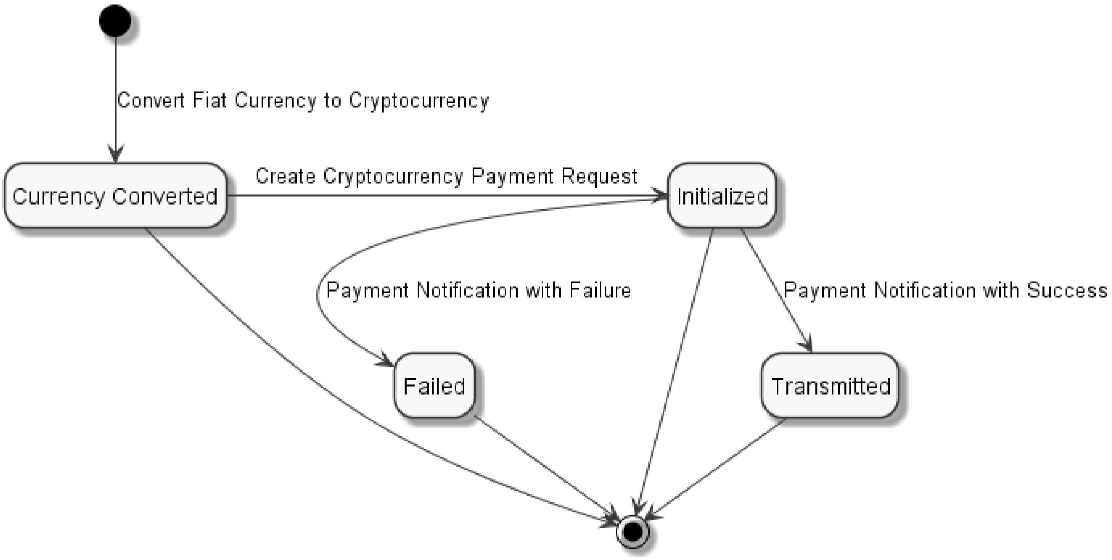

Convert Fiat Currency to Cryptocurrency
Currency Converted
Create Cryptocurrency Payment Request
Initialized
Payment Notification with Failure
Payment Notification with Success
Failed
Transmitted

Figure 23 - Transaction states

7.4 Functional Requirements

As mentioned in Section 6.2, the Controller layer depends on the Service layer, which in turn depends on the Repository layer. For the layers classes and methods to not all be tightly coupled because of direct instantiation, the Dependency Inversion principle was applied.

In that manner, interfaces were created for all services and repositories, which are initiated by the known process of Dependency Injection. In this design pattern, the classes are given other objects that they depend on and the class itself codes against an interface rather than a specific implementation class.

The following sections explain how the different use cases were implemented and introduce some more architectural and design patterns.

7.4.1 UC1: List Cryptocurrencies by Fiat currencies

This use case intends to list the cryptocurrencies that can be converted from a fiat currency, as described in section 5.2. Since this information is handled by the application CryptocurrencyPaymentConfiguration, the CryptocurrencyPaymentAPI makes it an HTTP request to obtain it. The list is then returned to the merchant, as represented in Code Snippet 4.

```
public async Task<GetCryptoFromFiatCurrencyDto> GetCryptoFromFiatCurrency(
    MerchantAuthorizationDto authorizationRequestDto,
    string currency)
{
    var response = await Task.Run(() => restClient
       .Get<GetCryptoFromFiatCurrencyDto>(
       configurationEndPoint,
       $"{fiatcurrenciesPath}{currency}/{authorizationRequestDto.MerchantId}",
       out _,
       new Dictionary<string, string>() {
           { "Authorization", authorizationRequestDto.AuthorizationHeader }
       }));

    if (response == null)
    {
        return new();
    }

    return response;
}
```

Code Snippet 4 - Function that gets the list of cryptocurrencies
(Application: CryptocurrencyPaymentAPI)

7.4.2 UC2: Convert fiat currency to cryptocurrency

The choice of payment gateway is made in this use case. Consequently, this is where the decision criteria presented in section 5.5 apply.

After authentication and authorization of the merchant and validation of the request, an HTTP request is made to CryptocurrencyPaymentConfiguration to get the ordered list of payment gateways that can process the transaction in the specified currencies, as verified by Code Snippet 5 (lines 9-12). As introduced in section 5.5 was decided to get a list of possible payment gateways instead of just one for cases where the services are unavailable. The next step is to validate the availability of the payment gateway services (lines 22-25) with a Ping request.

For the creation of the payment gateway service class was used the creational design pattern, Factory. An interface (`ICryptoGatewayFactory)`, with the list of the creation methods (lines 3-5) and its concrete class (`CryptoGatewayFactory`) were created. An interface (`ICryptoGatewayService`) for each payment gateway service and its concrete classes were also created. The maintainability (QA1) and Interchangeability (QA5) are benefited from this approach since it facilitates the integration with more cryptocurrency payment gateways without the need to change the existing source code.

The UML Class Diagram in Figure 28, Appendix C, provides a visual view of these interfaces, classes and other services created to implement this use case.

```
public class CryptoGatewayFactory : ICryptoGatewayFactory {
    …
    public List<ICryptoGatewayService> GetCryptoGatewayServices(
        MerchantAuthorizationDto authorizationRequestDto,
        CreatePaymentTransactionDto createPaymentTransactionDto)
    {
        try
        {
            var listPossiblePaymentGateways =
                decisionConfigurationService.GetPossiblePaymentGateway(
                    authorizationRequestDto,
                    createPaymentTransactionDto);

            if (listPossiblePaymentGateways == null
                || listPossiblePaymentGateways.Count == 0)
            {
                var validationResult = new ValidationResult();
                validationResult.AddMessages(ErrorCodes.InvalidCryptoCurrency);
                throw new ValidationException(validationResult);
            }

            var listAvailablePaymentGateways = listPossiblePaymentGateways
                .Select(e => GetCryptoGatewayService(e))
                .Where(e => e.ServiceWorking())
                .ToList();

            if (listAvailablePaymentGateways.Count == 0)
            {
                throw new ServiceUnavailableException("available");
            }
            return listAvailablePaymentGateways;
        }
        catch (Exception ex)
        {
            if (ex is ValidationException || ex is ServiceUnavailableException)
            {
                log.Error(ex.Message);
            }
            else
            {
                log.Error($"Unexpected exception {ex.Message}");
            }
            throw;
        }
    …
}
```

Code Snippet 5 - Function responsible for providing the list of valid and accessible cryptocurrency payment gateways
(Application: CryptocurrencyPaymentAPI)

As shown in Code Snippet 6, after getting the list of payment gateways that are accessible and can supposedly convert currencies, the first payment gateway with a valid response to the conversion request is found (lines 5-18).

```
public CurrencyConvertedDto GetCurrencyRates(
    MerchantAuthorizationDto authorizationRequestDto,
    CreatePaymentTransactionDto createPaymentTransaction)
{
    var listAvailablePaymentGateways =cryptoGatewayFactory.GetCryptoGatewayServices(
            authorizationRequestDto, createPaymentTransaction);

    foreach (var cryptoGatewayService in listAvailablePaymentGateways)
    {
        var rates = cryptoGatewayService.
            GetCurrencyRates(createPaymentTransaction);

        if (rates != null)
        {
            return rates;
        }
    }

    throw new ServiceUnavailableException("able");
}
```

Code Snippet 6 - Function responsible for finding the first valid response from a list of cryptocurrency payment gateways
(Application: CryptocurrencyPaymentAPI)

Finally, the Transaction entity (Code Snippet 7, lines 13-20) is built and stored on DB, with the payment gateway identification and the status "Currency Converted".

```
1.  public async Task<GetRatesDto> ConvertFiatToCryptocurrency(
2.       MerchantAuthorizationDto authorizationRequestDto,
3.       CreatePaymentTransactionDto createPaymentTransaction)
4.   {
5.       log.Info("Validating request");
6.       paymentValidation.ValidatePaymentTransactionCreation(createPaymentTransaction);
7.
8.       log.Info($"Getting Rate");
9.       var rates = transactionService.
10.          GetCurrencyRates(authorizationRequestDto, createPaymentTransaction);
11.      log.Info($"Got Rate '{rates.CurrencyRate}'");
12.
13.      log.Info($"Building Transaction");
14.      var transaction = createPaymentTransaction.ToEntity(
15.          rates, authorizationRequestDto.MerchantId);
16.      log.Info($"Built Transaction");
17.
18.      log.Info($"Adding Transaction '{transaction.DomainIdentifier}' to DB");
19.      transaction = await transactionRepository.Add(transaction);
20.      log.Info($"Added Transaction '{transaction.DomainIdentifier}' to DB");
21.
22.      var result = transaction.ToDtoRates();
23.      return result;
24.  }
```

Code Snippet 7 - Function responsible for validating the request, converting the currency, building the transaction, storing the transaction on the database, and building the response
(Application: CryptocurrencyPaymentAPI)

The UML Sequence Diagram in Figure 24 depicts, in general, the flow described herein. In Appendix C, Figure 26 and Figure 27 show in more detail the steps taking place in the CryptocurrencyPaymentAPI application.

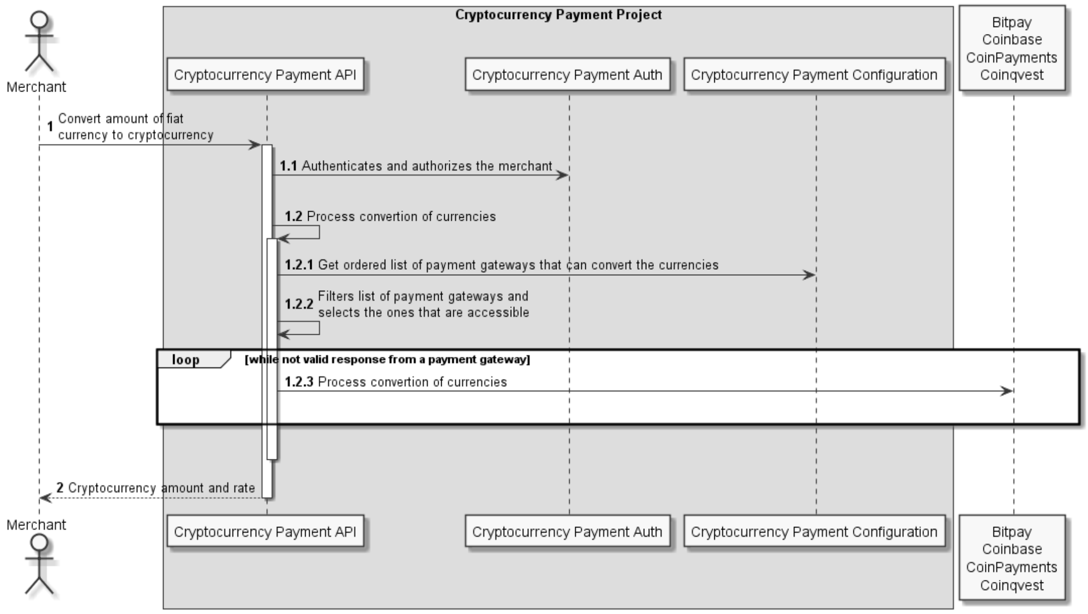

Cryptocurrency Payment Project
Merchant
Cryptocurrency Payment API
Cryptocurrency Payment Auth
Cryptocurrency Payment Configuration
Bitpay
Coinbase
CoinPayments
Coinqvest
1 Convert amount of fiat currency to cryptocurrency
1.1 Authenticates and authorizes the merchant
1.2 Process convertion of currencies
1.2.1 Get ordered list of payment gateways that can convert the currencies
1.2.2 Filters list of payment gateways and selects the ones that are accessible
loop
[while not valid response from a payment gateway]
1.2.3 Process convertion of currencies
2 Cryptocurrency amount and rate
Merchant
Cryptocurrency Payment API
Cryptocurrency Payment Auth
Cryptocurrency Payment Configuration
Bitpay
Coinbase
CoinPayments
Coinqvest

Figure 24 - High-level flow of converting fiat currency to cryptocurrency

7.4.3 UC3: Create payment transaction

This use case is simpler since one already knows which cryptocurrency payment gateway to use to process the transaction. The merchant authentication and authorization are performed. Next, the transaction previously stored in the database is retrieved, using the transaction identifier provided in the HTTP request.

As Code Snippet 8 demonstrates is checked whether this transaction was initiated by the merchant placing the request (lines 5-7). Also is validated that the Transaction status is "Currency Converted" and that the conversion rate did not expire (lines 11-15). If at least one of those validations is not verified, the request ends herein with an error.

```
1.  public void ValidateTransactionConfirm(Transaction? transaction, string merchantId)
2.  {
3.      var validationResult = new ValidationResult();
4.
5.      if (transaction is null || !transaction.MerchantId.Equals(merchantId))
6.      {
7.          validationResult.AddMessages(ErrorCodes.InvalidTransaction);
8.      }
9.      else
10.     {
11.         if (!transaction.TransactionState
12.                         .Equals(TransactionState.CurrencyConverted))
13.             validationResult.AddMessages(ErrorCodes.TransactionStateConverted);
14.         else if (transaction.Details.Conversion.ExpiryDate < DateTime.Today)
15.             validationResult.AddMessages(ErrorCodes.ConversionRateExpired);
16.     }
17.
18.     validationResult.ShouldThrowValidationException();
19. }
```

Code Snippet 8 - Method that validates the request to create payment transaction use case (Application: CryptocurrencyPaymentAPI)

The cryptocurrency payment gateway is called to provide the payment data, which can be either the wallet address or the QR Code URL, depending on the payment gateway (Code Snippet 9). Finally, the transaction is updated in the database with the new data received and the status “Initialized”. Finally, the response is returned to the client.

```
public PaymentCreatedDto CreateTransaction(
    ConfirmPaymentTransactionDto confirmTransaction)
{
    var cryptoGatewayService = cryptoGatewayFactory
        .GetCryptoGatewayService(confirmTransaction.PaymentGateway);
    log.Info($"Creating transaction");

    var response = cryptoGatewayService.
        CreateTransaction(confirmTransaction);

    if (response == null)
    {
        var validationResult = new ValidationResult();
        validationResult.AddMessages(ErrorCodes.OperationInvalid);
        throw new ValidationException(validationResult);
    }
    return response;
}
```

Code Snippet 9 - Function that gets the cryptocurrency payment gateway and calls it to create a transaction
(Application: CryptocurrencyPaymentAPI)

7.4.4 UC4: Process transaction

Figure 21 exhibits that the cryptocurrency payment gateways are responsible for notifying the Cryptocurrency Payment Application of the transaction completion. Here the concept of Webhook is introduced since the communication is initiated by the application providing the information. The CryptocurrencyPaymentAPI will receive an HTTP POST request from the cryptocurrency payment gateways once they have data about the transaction.

CryptocurrencyPaymentAPI must give the cryptocurrency payment gateway the URL to which it must deliver the requests. Depending on the cryptocurrency payment gateway was accomplished on UC2 or UC3.

O Code Snippet 10 exemplifies the URL specification (lines 8-9) in the cryptocurrency conversion (UC2).

```
public override CurrencyConvertedDto? GetCurrencyRates(
    CreatePaymentTransactionDto createPaymentTransaction)
{
    try
    {
        var request = new CoinqvestRequest()
        {
            Webhook =
           $"{NotificationEndPoint}{createPaymentTransaction.TransactionReference}",
            Charge = new CoinqvestCharge()
            {
                Currency = createPaymentTransaction.FiatCurrency ?? string.Empty,
            },
        };
        var currencyRates = RestClient?.Post<CoinqvestRequest, CoinqvestResponse>(
           ConvertCurrencyEndPoint,
           string.Empty,
           request,
           out var responseHeaders);
        …
    }
    catch (Exception ex)
    {
        log.Error($"Unexpected exception {ex.Message}");
        return null;
    }
}
```

Code Snippet 10 - Part of the function that calls COINQVEST to convert cryptocurrency (Application: CryptocurrencyPaymentAPI)

On the other hand, Code Snippet 11 displays the URL specification (lines 10-11) when creating the transaction (UC3).

```
1.  public override PaymentCreatedDto? CreateTransaction(
2.      ConfirmPaymentTransactionDto confirmTransactionDto)
3.  {
4.      try
5.      {
6.          var request = new InvoiceRequest()
7.          {
8.              Currency = confirmTransactionDto.FiatCurrency,
9.              Price = confirmTransactionDto.Amount,
10.             NotificationURL =
11.                  $”{NotificationEndPoint}{confirmTransactionDto.TransactionId}”,
12.         };
13.
14.         var response = RestClient?.Post<InvoiceRequest, InvoiceResponse>(
15.             CreateTransactionEndPoint,
16.             string.Empty,
17.             request,
18.             out var responseHeaders);
19.         …
20.     }
21.     catch (Exception ex)
22.     {
23.         log.Error($"Unexpected exception {ex.Message}");
24.         return null;
25.     }
26. }
```

Code Snippet 11 - Part of the function responsible for calling BitPay to create a transaction (Application: CryptocurrencyPaymentAPI)

Also was necessary to implement the controller responsible for receiving these POST requests. Consequently, was defined an endpoint for each cryptocurrency payment gateway that the CryptocurrencyPaymentAPI is integrating, as displayed in Code Snippet 12.

```
[ApiController]
[Route("[controller]")]
public class NotificationController : ControllerBase
{
    private static readonly ILog log =
                LogManager.GetLogger(MethodBase.GetCurrentMethod()?.DeclaringType);
    private readonly INotificationService notificationService;

    public NotificationController(INotificationService notificationService)
    {
        this.notificationService = notificationService;
    }

    [HttpPost("bitpay/{transactionId}")]
    public async Task<ActionResult> BitPayPaymentTransactionNotification(
        [FromRoute] string transactionId,
        [FromBody] InvoiceResponseData bitpayNotification)
    {
        log.Info("Bitpay Payment transaction Notification");
        await notificationService.ProcessBitPayTransaction(
                        transactionId, bitpayNotification);
        return Ok();
    }

    [HttpPost("coinbase/{transactionId}")]
    public async Task<ActionResult> CoinbasePaymentTransactionNotification(
        [FromRoute] string transactionId,
        [FromBody] CoinbaseChargeResponse coinbaseNotification)
    {
        log.Info("Coinbase Payment transaction Notification");
        await notificationService.ProcessCoinbaseTransaction(
                        transactionId, coinbaseNotification);
        return Ok();
    }

    [HttpPost("coinpayments/{transactionId}")]
    public async Task<ActionResult> CoinPaymentsPaymentTransactionNotification(
        [FromRoute] string transactionId,
        [FromBody] CoinPaymentNotification coinPaymentsNotification)
    {
        log.Info("CoinPayments Payment transaction Notification");
        await notificationService.ProcessCoinPaymentsTransaction(
                        transactionId, coinPaymentsNotification);
        return Ok();
    }

    [HttpPost("coinqvest/{transactionId}")]
    public async Task<ActionResult> CoinqvestPaymentTransactionNotification(
        [FromRoute] string transactionId,
        [FromBody] CoinqvestNotification coinqvestNotification)
    {
        log.Info("Coinqvest Payment transaction Notification");
        await notificationService.ProcessCoinqvestTransaction(
                        transactionId, coinqvestNotification);
        return Ok();
    }
}
```

Code Snippet 12 - Controller with the Webhooks for each cryptocurrency payment gateway (Application: CryptocurrencyPaymentAPI)

Each cryptocurrency payment gateway provides a different structure for the Webhook, so its handling is distinct. Nevertheless, the data is interpreted and mapped to mutual fields and values. For this reason, the process is as seamless as possible for the merchant.

The transaction is updated in the database with the new data and the status "Transmitted" or "Failed". The latter happens whenever the transaction is not completed successfully. For example, a transaction timeout can ensue when the client exceeds the time to pay the transaction.

7.4.5 UC5: List transaction by identifier

After authentication and authorization of the merchant, the transaction is retrieved from the database using the identifier provided by the merchant (Code Snippet 13, lines 8-9). The transaction is then validated (Code Snippet 13, lines 13-15).

```
1.  public async Task<GetTransactionDto> GetTransaction(
2.      MerchantAuthorizationDto authorizationRequestDto,
3.      string transactionId)
4.  {
5.      log.Info($"Get transaction '{transactionId}'");
6.
7.      log.Info($"Getting transaction '{transactionId}' from DB");
8.      var transaction = await transactionRepository
9.          .GetByDomainIdentifier(transactionId);
10.     log.Info($"Got transaction '{transactionId}' from DB");
11.
12.     log.Info("Validating request");
13.     paymentValidation.ValidateTransactionGet(
14.         transaction,
15.         authorizationRequestDto.MerchantId);
16.
17.     var result = transaction.ToDto();
18.
19.     return result;
20. }
```

Code Snippet 13 - Function that gets the transaction from the database, validates it and creates the response
(Application: CryptocurrencyPaymentAPI)

Particularly is verified that the transaction exists and was initiated by the merchant (Code Snippet 14, lines 5-7).

```
1.  public void ValidateTransactionGet(Transaction? transaction, string merchantId)
2.  {
3.      var validationResult = new ValidationResult();
4.
5.      if (transaction is null || !transaction.MerchantId.Equals(merchantId))
6.      {
7.          validationResult.AddMessages(ErrorCodes.InvalidTransaction);
8.      }
9.
10.     validationResult.ShouldThrowValidationException();
11. }
```

Code Snippet 14 - Function that validates the request to list a transaction by identifier
(Application: CryptocurrencyPaymentAPI)

8 Solution Assessment

This chapter exposes the evaluation of the software solution, following the Goal/Question/Metric (GQM) approach. This method aims to measure a product/process/resource by defining its goals and questions to be answered with the collected data (Basili, Caldiera and Rombach, 1994) to denote the rationale and need for collecting measurements. The method organizes in the phases of Planning, Definition, Data Collection, and Interpretation (Solingen and Berghout, 1999), which are addressed herein in this chapter.

8.1 Planning

The planning phase fulfils all basic requirements to succeed in the GQM measurement programme, resulting in a project plan. It includes selecting, defining, and planning a project (Solingen and Berghout, 1999).

Chapter 1 provides an overview of the selected project, and Appendix D details its plan.

8.2 Definition

The definition phase intends to provide a formal definition of measurement, including the goals, questions, metrics and expectations (hypotheses) (Solingen and Berghout, 1999).

GQM defines a measurement model on three levels (Basili, Caldiera and Rombach, 1994):

(1) Conceptual level (Goal) defines the goal for a product/process/resource for several reasons concerning different models of quality within a particular environment;
(2) Operational level (Question) introduces a set of questions that provides an understanding of how the assessment of a specific goal is going to be accomplished;
(3) Quantitative level (Metric) associates with each question a set of objective or subjective metrics to answer those questions and conclude how the goals are met.

Table 15 depicts the goals, questions, and metrics, and in this case, the aim is defined accordingly to some quality attributes with which the solution must comply.

Table 15 - Goal/Question/Metric for each Quality Attribute

Quality attribute	Goal	Question	Metric
QA1 Maintainability	**G1** Analyse the solution to understand the maintainability and its causes.	**Q1** What is the technical debt ratio value at the end of the project?	**M1** Technical Debt Ratio (technical debt divided by the estimated cost to rewrite the application)
		Q2 How many code smells exist at the end of the project?	**M2** Number of code smells
QA2 Security	**G3** Analyse the solution to understand the security and its causes.	**Q3** How many vulnerabilities exist at the end of the project?	**M3** Number of vulnerabilities
		Q4 What is the distribution of vulnerabilities over security categories?	**M4** Number of informational, low, medium, and high vulnerabilities
QA3 Reliability	**G4** Analyse the solution to understand the reliability and its causes.	**Q5** How many bugs exist at the end of the project?	**M5** Number of bugs
		Q6 What is the distribution of bugs over reliability categories?	**M6** Number of minor, major, critical and blocker bugs

With the goals, questions and metrics defined, one can specify the expected results of the assessment:

(1) The technical debt ratio is less than 5.0% (M1);
(2) There are no more than 30 code smells (M2);
(3) The system does not have any vulnerability (M3 and M4);
(4) The system does not have any bugs (M5 and M6).

8.3 Data Collection

Once the definition phase is complete, one can start the actual measurements, resulting in the collected data. Automatic and digitized tools, such as Sonarqube, Snyk and Acunetix, were used during the development of the solution and in this phase to acquire accurate measurements.

The Security quality attribute was analysed in two different ways (Techopedia, 2014; OWASP, 2022b):

- Dynamic Application Security Testing (DAST) tests the application in an operation state, finding problems that happen during the application's use. Acunetix is the tool used to perform this type of analysis.
- Static Application Security Testing (SAST), also known as source code analysis, tests the application by scanning the source code to find design and construction flaws with potential for vulnerability. Sonarqube and Snyk were the selected tools to perform this analysis.

SonarQube is an open-source platform developed to perform automatic reviews with static analysis of the code to detect bugs, code smells and security vulnerabilities (SonarQube, 2022).

This tool was used to evaluate the quality attributes of Maintainability, Security, and Reliability:

(1) The technical debt ratio is 0.0% (M1);
(2) The number of code smells is 23 (M2);
(3) Number of vulnerabilities is 0 (M3);
(4) Number of bugs is 0 (M5)

Figure 30, Figure 31, Figure 32, and Figure 33 in Appendix E display the complete SonarQube result's analysis regarding the quality attributes.

Snyk is a developer security platform to find and automatically fix vulnerabilities. This tool was used to detect disclosed vulnerabilities in the source code and both the direct and in-direct (transitive) open-source dependencies (Snyk, 2022). Figure 34 in Appendix E reveals that this tool did not find any vulnerability (M3).

Acunetix is a security scanner tool designed to replicate techniques used by attackers. It performs automated and manual penetration tests to find publicly disclosed vulnerabilities, such as SQL or command injections, OWASP Top 10, Sensitive Data Exposure, and Cross-Site Scripting (XSS) (Acunetix, 2022).

During the development, this tool found some vulnerabilities (M3) that were fixed. Figure 35, and Figure 36, in Appendix E, exhibit the scan report result with the vulnerabilities found, and Figure 37 presents the scan report result performed after fixing them.

Table 16 describes the vulnerabilities and how they were fixed.

Table 16 - Vulnerabilities found on the Acunetix scan

Vulnerability	Description
TLS 1.0 enabled **TLS 1.1 enabled**	TLS 1.0 and 1.1 were formally deprecated in March 2021 because of inherent security issues (Moriarty and Farrell, 2021). Consequently, these versions of TLS were disabled, by specifying the versions supported (TLS 1.2 and 1.3) on the code.

TLS/SSL Sweet32 attack **TLS/SSL Weak Cipher Suites**	The Sweet32 attack is an SSL/TLS vulnerability that allows attackers to compromise HTTPS connections using 64-bit block cyphers (Sweet32, 2022). The remote host supports TLS/SSL cypher suites with weak or insecure properties (SSL Labs, 2020). `TLS_RSA_WITH_3DES_EDE_CBC_SHA` was the cypher suite with weak or insecure properties. This cypher suite had to be disabled at the windows server level since this vulnerability is related to the machine where the code is running.
Clickjacking: X-Frame-Options header	Clickjacking is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information, or taking control of their computer while clicking on seemingly innocuous web pages (Goodin, 2008; OWASP, 2022a). This vulnerability was found on the Swagger page (The web page with the API documentation), and the solution was to return the header `X-Frame-Options`.
HTTP Strict Transport Security (HSTS) not implemented	HTTP Strict Transport Security (HSTS) tells a browser that a website is only accessible using HTTPS (MDN Web Docs, 2022b). The URL where HSTS is not enabled was the Swagger page. Adding the header `Strict-Transport-Security` fixed this vulnerability.
Sensitive pages could be cached	The Swagger pages may contain sensitive information that could be potentially cached. Adding the headers `Cache-Control` and `Pragma` fixed the vulnerability.
Content Security Policy (CSP) not implemented	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross-Site Scripting (XSS) and data injection attacks (MDN Web Docs, 2022a). The Swagger path did not have the header `Content-Security-Policy`, so adding it fixed this vulnerability.

8.4 Data Analysis and Interpretation

The results of the measurement are used to answer the stated questions and validate if the stated goals have been attained. If expected results and conclusions are consistent, it implies that the goal has been achieved (Basili, Caldiera and Rombach, 1994; Solingen and Berghout, 1999). Table 17 presents the questions' answers and the conclusions made in terms of metrics results.

Table 17 - Answers and conclusion of the assessment goals

Goal	Conclusions
G1 Analyse the solution to understand the maintainability and its causes.	**Q1 Q2** The final prototype has a technical debt ratio of 0.0%, which is less than the expected result (5.0%), and 23 code smells, which is less than the limit established (30). In summary, the solution fulfils the requirements of the maintainability quality attribute, even though there is scope for improvement to minimize the number of code smells.
G3 Analyse the solution to understand the security and its causes.	**Q3 Q4** All the vulnerabilities found were fixed during the development of the solution, finishing with an application free of security weaknesses. This goal is a crucial one since the solution is a payment application. A lot of effort was put into the security analysis of the application, where various tools were used to scan the application's security. Besides was possible to find, understand, and fix some vulnerabilities.
G4 Analyse the solution to understand the reliability and its causes.	**Q5 Q6** The application does not have any bugs. The solution can be considered reliable because there are no bugs that can lead to an error or unexpected behaviour at runtime.

Although the final solution attained the expected result, some factors were identified which may have an undue influence on the development or distort the data collected. These raise questions about the validity of the solution in other environments or operation systems:

- All applications were running on the same machine during testing and validation. Although it should work the same way in different machines and operating systems, this project did not validate it.
- This project focused on using four cryptocurrency payment gateways, which were mocked up. So, this project has not tested or validated the integration with the cryptocurrency payment gateways, and this research cannot be used to claim this.
- At the time of validation, no disclosed vulnerabilities were found in the open-source dependencies, which does not mean that vulnerabilities are not found in them later.

9 Conclusion

This chapter means to wrap up what was accomplished, the project limitations and the changes this project can suffer to potentially improve it. A final appreciation of the application and work will also be conducted.

9.1 Achieved Objectives

The project began with the definition of the problem, the study of basic concepts and a technology overview. Value analysis was carried out from which the validity of the idea and the best way to achieve the objective was inferred based on different criteria. Additionally was conducted a review of the principal technologies for processing online cryptocurrency payments. Then, based on the accumulated information, the objectives and requirements of the solution were identified. It resulted in the architecture modelling that was later detailed and implemented. Ultimately, the solution was evaluated following a method based on the identification goals, questions, and metrics to qualify the conclusion of the requirements,

This Section intends to conclude the final state of the objectives defined in Section 1.3. Thus, the three objectives defined are as follows:

- Analyse the suitability of different cryptocurrency payment gateways and decision criteria that should be considered when selecting a cryptocurrency payment gateway.
- Design and implement a prototype integrating with multiple cryptocurrency payment gateways.
- Validate the solution in terms of some quality attributes.

The investigation and exploration that shaped the State of The Art chapter allowed the identification of cryptocurrency payment gateways that were later analysed. This analysis resulted in a list of four cryptocurrency payment gateways considered suitable to be integrated into the solution (5.4) as they meet the requirements (5.2) and constraints (5.3).

Decision criteria were defined that should be considered by the solution when choosing the cryptocurrency payment gateway to process the transaction (5.5). This definition was based on the cryptocurrencies supported by these cryptocurrency payment gateways. Thus the first objective is successfully achieved, given the analysis and the selection of cryptocurrency payment gateways and decision criteria.

After defining the requirements and identifying cryptocurrency payment gateways and decision criteria, an analysis was performed in which all business entities were identified in a domain model diagram (5.6).

Then the architecture to be developed was modelled in detail, identifying all the existing components and their interactions (6). With all this, it became possible to implement the prototype capable of supporting the Bitcoin cryptocurrency, fulfilling all functional and non-functional requirements (7).

During the design and implementation phase, some patterns were considered to make the prototype easily adaptable to be integrated with other applications, such as the Open-Closed Principle, Dependency Injection, and the architectural style RESTful API.

At last, the third objective aims to evaluate the system based on some quality attributes. Some metrics were specified (8.2) for this objective to become measurable. This objective was also successful, as the evaluation achieved the expected results, and some evaluation threats were raised (8.4).

9.2 Limitations and Future Work

Despite the successful implementation and positive evaluation of the developed system, some limitations and tasks that aim to improve the presented solution have been identified.

The developed solution processes cryptocurrency transactions through integration with cryptocurrency payment gateways. The major limitation of the project is that these services were mocked, and consequently, the integration was simulated. The choice of cryptocurrency payment gateway with a free trial was not a selection factor since the trial time would not be enough to complete the project. In the future will be essential to do and tests the integration with the right services.

It would be interesting for the decision of the cryptocurrency payment gateway to process the transaction to be configurable, although it was not the project's primary focus. At the moment, this choice is the same for all traders, the deciding factor being the currencies of the transaction. A possible future work would be to make this choice configurable by the merchant based on more criteria that might make sense. The existence of an application already dedicated to this effect makes its evolution easier without changing or jeopardizing the existing behaviour of the CryptocurrencyPaymentAPI application.

Another point of improvement would be the activity logs, which were considered during the implementation of the application. However, store these logs in a database would be an asset. This change would make the application more advantageous for merchants, as they could track transactions and, perhaps, generate reports. This topic would need more research to respect the security quality attribute.

A solution was implemented to avoid errors still exceptions were created to control the flow of orders. It would be interesting to change this part of the code to use monads. Result/error monads would create a wrapper of the function result defining the success and failure path removing the need to define exceptions. This work was not executed due to a lack of experience with this concept, which would have compromised the implementation's completion.

Lastly, an attractive perspective for evaluating the solution would be to perform performance tests. The performance quality attribute was not a requirement raised and therefore was not evaluated. However, transaction time is a valuable factor in financial transactions.

9.3 Final Appreciation

On a final note, this project was successfully completed, fulfilling all established requirements.

An investigation was performed on concepts and technologies adhering to cryptocurrencies and an extensive value analysis, which mathematically compared different approaches to processing cryptocurrencies transactions.

Moreover, the entire project held and the description of the developed prototype allow sharing of knowledge about this type of financial transaction, which is a good starting point for a future project.

It would be a rewarding prize with the completion of this master's project, that this vast work contributes to the state of the art.

Bibliographic References

Acunetix (2022) *Acunetix | Web Application Security Scanner*. Available at: https://www.acunetix.com/ (Accessed: 7 June 2022).

Adewole, K., Saxena, N. and Bhadauria, S. (2020) ‘Application of Cryptocurrencies Using Blockchain for E-Commerce Online Payment’, in Maleh, Y. et al. (eds) *Blockchain for Cybersecurity and Privacy*. 1st edn. First edition. | Boca Raton, FL : CRC Press, 2020. | Series: Internal audit and it audit: CRC Press, pp. 263–305. doi:10.1201/9780429324932-16.

Adobe (2022) *What is an e-commerce platform? | Adobe Glossary*. Available at: https://business.adobe.com/au/glossary/ecommerce-platforms.html (Accessed: 2 February 2022).

Agrawal, H. (2021) ‘9 Best Bitcoin Payment Gateways For Merchant Account & Services’, 10 March. Available at: https://coinsutra.com/bitcoin-payment-gateways-merchants/ (Accessed: 11 February 2022).

ALFAcoins (2022a) *ALFAcoins | Fee schedule*, *ALFAcoins*. Available at: https://www.alfacoins.com/fees (Accessed: 13 February 2022).

ALFAcoins (2022b) *ALFAcoins | Start accepting Bitcoin & Cryptocurrency payments on your website*, *ALFAcoins*. Available at: https://www.alfacoins.com/merchant (Accessed: 12 February 2022).

ALFAcoins (2022c) *ALFAcoins API | ALFAcoins*, *ALFAcoins*. Available at: https://www.alfacoins.com/developers/api-documentation (Accessed: 13 February 2022).

American Express Company (2020a) *Quarterly Report Pursuant for the Quarterly Period Ended June 30, 2020*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2020/q2/7fa0299e-6c48-4a0b-9a89-57bd70f5d24b.pdf (Accessed: 29 March 2022).

American Express Company (2020b) *Quarterly Report Pursuant for the Quarterly Period Ended March 31, 2020*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2020/q1/178715b5-8762-4cb7-9cf4-6a3bd4778ba1.pdf (Accessed: 29 March 2022).

American Express Company (2020c) *Quarterly Report Pursuant for the Quarterly Period Ended September 30, 2020*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2020/q3/aec24047-7ba7-4e21-aa10-1c38ff5b4707.pdf (Accessed: 29 March 2022).

American Express Company (2021a) *Annual Report Pursuant for the fiscal year ended December 31, 2020*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2020/q4/0dc0103f-bd50-40bc-83fd-8a741f073c15.pdf (Accessed: 29 March 2022).

American Express Company (2021b) *Quarterly Report Pursuant for the Quarterly Period Ended June 30, 2021*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2021/q2/3f1cda3a-bca0-4b81-8bb1-d8313b0a8dc3.pdf (Accessed: 29 March 2022).

American Express Company (2021c) *Quarterly Report Pursuant for the Quarterly Period Ended March 31, 2021*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2021/q1/c8b3b960-bfb1-4adb-9afd-cea5d6c55661.pdf (Accessed: 29 March 2022).

American Express Company (2021d) *Quarterly Report Pursuant for the Quarterly Period Ended September 30, 2021*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2021/q3/7a17b8fd-8465-4af6-a2ad-85a0dd6e23fc.pdf (Accessed: 29 March 2022).

American Express Company (2022) *Annual Report Pursuant for the fiscal year ended December 31, 2021*. Available at: https://s26.q4cdn.com/747928648/files/doc_financials/2021/q4/d13acb37-2f7e-411d-8ed1-6516668bf861.pdf (Accessed: 29 March 2022).

Banguis, M.D. (2021) *What Is The Difference Between Cryptocurrency Wallet and Cryptocurrency Payment Gateway*. Available at: https://www.coinqvest.com/en/blog/what-is-the-difference-between-cryptocurrency-wallet-and-cryptocurrency-payment-gateway-e0f6bfc1a05f (Accessed: 25 January 2022).

Bank for International Settlements (2015) 'Glossary'. Available at: https://www.bis.org/cpmi/publ/d00b.htm (Accessed: 30 March 2022).

Bank of England (2021) *New forms of digital money*. Available at: https://www.bankofengland.co.uk/paper/2021/new-forms-of-digital-money (Accessed: 23 January 2022).

Bansal, L. (2021) *Top 10 Payment Gateways For Cryptocurrency In 2021*. Available at: https://www.c-sharpcorner.com/article/top-10-payment-gateways-for-cryptocurrency-in-2021/ (Accessed: 26 January 2022).

Baran, P. (1964) *On Distributed Communications: I. Introduction to Distributed Communications Networks*. RAND Corporation. Available at: https://www.rand.org/pubs/research_memoranda/RM3420.html (Accessed: 26 March 2022).

Basili, V.R., Caldiera, G. and Rombach, H.D. (1994) 'THE GOAL QUESTION METRIC APPROACH', p. 10.

Bear Group (2022) *Ecommerce Hosting Comparison: PaaS, SaaS, Modular, and Self-Hosted Open Source | Bear Group*, *Bear Group*. Available at: https://www.beargroup.com/ideas/ecommerce-hosting-comparison-paas-saas-modular-and-self-hosted-open-source (Accessed: 2 February 2022).

Beigel, O. (2021) *Who Accepts Bitcoins in 2021? List of 20+ Major Companies*, *99 Bitcoins*. Available at: https://99bitcoins.com/bitcoin/who-accepts/ (Accessed: 18 December 2021).

Benzinga (2022) *Best Crypto Wallets • Hardware & Software • Benzinga*, *Benzinga*. Available at: https://www.benzinga.com/money/best-crypto-wallet/ (Accessed: 5 February 2022).

Bernheim, L. (2017) *IaaS vs. PaaS vs. SaaS Cloud Models (Differences & Examples)*, *HostingAdvice.com*. Available at: https://www.hostingadvice.com/how-to/iaas-vs-paas-vs-saas/ (Accessed: 2 February 2022).

Best, R. de (2022) *PayPal: transaction volume 2021*, *Statista*. Available at: https://www.statista.com/statistics/277841/paypals-total-payment-volume/ (Accessed: 29 March 2022).

Bezhovski, Z., Davcev, L. and Mitreva, M. (2021) 'Current adoption state of cryptocurrencies as an electronic payment method', *Management Research and Practice*, 13(1), pp. 44–50.

Binance Research (2021) 'Global Crypto User Index - 2021', *Binance Research*, p. 69.

Bitcoin (2020) 'US Consumers Flock To the First Mastercard Branded BitPay Card – Sponsored Bitcoin News', *Bitcoin News*, 15 September. Available at: https://news.bitcoin.com/us-consumers-flock-to-the-first-mastercard-branded-bitpay-card/ (Accessed: 18 December 2021).

Bitcoin (2022a) *Desktop - Choose your wallet - Bitcoin*, *Bitcoin*. Available at: https://bitcoin.org/en/wallets/desktop/linux/?step=5&platform=linux (Accessed: 3 February 2022).

Bitcoin (2022b) *Hardware - Choose your wallet - Bitcoin*, *Bitcoin*. Available at: https://bitcoin.org/en/wallets/hardware/?step=5&platform=hardware (Accessed: 3 February 2022).

Bitcoin (2022c) *Securing your wallet - Bitcoin*, *Bitcoin*. Available at: https://bitcoin.org/en/secure-your-wallet#offlinetx (Accessed: 3 February 2022).

Bitcoinist (2016) 'ALFAcoins Introduces Freshly Designed Website with New Features', 16 December. Available at: https://bitcoinist.com/alfacoins-website-new-features/ (Accessed: 12 February 2022).

BitPay (2022a) *BitPay Banking & Settlements*, *BitPay*. Available at: https://bitpay.com/docs/settlement#all (Accessed: 13 February 2022).

BitPay (2022b) *BitPay: Buy Crypto Without Fees | Store, Swap & Spend Bitcoin*, *BitPay*. Available at: https://bitpay.com/pricing/ (Accessed: 13 February 2022).

BitPay (2022c) *BitPay Integrations*, *BitPay*. Available at: https://bitpay.com/integrations/ (Accessed: 3 February 2022).

BitPay (2022d) *REST API – API Reference*, *BitPay REST API*. Available at: https://bitpay.com/api/#rest-api (Accessed: 13 February 2022).

Blockchain.com (2022) *Estimated Transaction Volume in USD*, *Blockchain.com*. Available at: https://www.blockchain.com/pt/charts/estimated-transaction-volume-usd (Accessed: 29 March 2022).

Blockonomics (2022a) *Accept bitcoin payments - Blockonomics*, *Blockonomics*. Available at: https://www.blockonomics.co/merchants (Accessed: 12 February 2022).

Blockonomics (2022b) *Blockchain API | Bitcoin API - Blockonomics*, *Blockonomics*. Available at: https://www.blockonomics.co/views/api.html#basicinfo (Accessed: 13 February 2022).

Blockonomics (2022c) *Track Bitcoin - Blockonomics*, *Blockonomics*. Available at: https://www.blockonomics.co/views/wallet-watcher.html?next=%2Fblockonomics (Accessed: 12 February 2022).

Blockonomics (2022d) 'WordPress Bitcoin Payments – Blockonomics – Changelog - WordPress Plugin | 2022', *WPSocket*. Available at: https://wpsocket.com/plugin/blockonomics-bitcoin-payments/changelog/ (Accessed: 13 February 2022).

Boehm, B.W. (2006) 'Value-Based Software Engineering: Overview and Agenda', in Biffl, S. et al. (eds) *Value-Based Software Engineering*. Berlin, Heidelberg: Springer, pp. 3–14. doi:10.1007/3-540-29263-2_1.

Borza, J.S. (2011) 'FAST Diagrams: The Foundation for Creating Effective Function Models', p. 10.

Brown, R.G. (2020) 'The Internet Is A Public Permissioned Network. Should Blockchains Be The Same?', *R3*, 6 May. Available at: https://www.forbes.com/sites/richardgendalbrown/2020/05/06/the-internet-is-a-public-permissioned-network-should-blockchains-be-the-same/#4c0711a47326 (Accessed: 6 January 2022).

Browne, R. (2021) *Stripe says it's open to accepting crypto for payments, three years after ending bitcoin support*, *CNBC*. Available at: https://www.cnbc.com/2021/11/24/stripe-open-to-accepting-crypto-for-payments-again.html (Accessed: 18 December 2021).

Chan, J. (2021) *Visa, PayPal to Allow Customers to Pay With Cryptocurrency*, *Entrepreneur*. Available at: https://www.entrepreneur.com/article/368256 (Accessed: 18 December 2021).

Chevalier, S. (2021) *Global retail e-commerce market size 2014-2023*, *Statista*. Available at: https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/ (Accessed: 12 October 2021).

Choudhury, S.R. (2021) *Bitcoin is breaking records because bigger investors are buying it now, says PwC*, *CNBC*. Available at: https://www.cnbc.com/2021/01/04/bitcoin-btc-rally-partly-driven-by-more-institutional-investors-pwc-says.html (Accessed: 5 February 2022).

Christine, L. (2019) *Winds of Change: The Case for New Digital Currency*, *Winds of Change*. International Monetary Fund. Available at: https://www.elibrary.imf.org/view/books/076/25701-9781484389171-en/25701-9781484389171-en-book.xml (Accessed: 23 January 2022).

Coinbase (2020) *Easily Convert Crypto on Coinbase Commerce using Coinbase.com*, *Medium*. Available at: https://blog.coinbase.com/easily-convert-crypto-on-coinbase-commerce-using-coinbase-com-89aed4cb4a65 (Accessed: 31 January 2022).

Coinbase (2022a) *Coinbase Supported Countries | Coinbase*. Available at: https://www.coinbase.com/places (Accessed: 31 January 2022).

Coinbase (2022b) *Crypto basics - What is a crypto wallet?* Available at: https://www.coinbase.com/learn/crypto-basics/what-is-a-crypto-wallet (Accessed: 2 February 2022).

Coinbase (2022c) *Instant Cashouts*, *Coinbase Help*. Available at: https://help.coinbase.com/en/coinbase/getting-started/add-a-payment-method/instant-withdrawal (Accessed: 13 February 2022).

Coinbase (2022d) *What is a seed phrase?* Available at: https://www.coinbase.com/pt/learn/crypto-basics/what-is-a-seed-phrase (Accessed: 4 February 2022).

Coinbase Commerce (2020) 'Announcing Easier Refund Support for Coinbase Commerce', *Medium*, 18 March. Available at: https://medium.com/@coinbasecommerce/announcing-easier-refund-support-for-coinbase-commerce-a891979e9aaa (Accessed: 13 February 2022).

Coinbase Commerce (2022a) *Accept crypto within minutes*, *Coinbase Commerce*. Available at: https://commerce.coinbase.com/ (Accessed: 13 February 2022).

Coinbase Commerce (2022b) *Commerce API Documentation | Coinbase Commerce*, *Coinbase Commerce*. Available at: https://commerce.coinbase.com/docs/api/ (Accessed: 13 February 2022).

Coinbase Commerce (2022c) *Integrate bitcoin payments into your business in minutes | Coinbase Commerce*, *Coinbase Commerce*. Available at: https://commerce.coinbase.com/integrate (Accessed: 13 February 2022).

Coinbase Help (2022) *Vaults*, *Coinbase Help*. Available at: https://help.coinbase.com/en/coinbase/getting-started/other/vaults-faq (Accessed: 13 February 2022).

CoinDesk (2022) *CoinDesk: Crypto Explainer+ Crypto Terms*. Available at: https://www.coindesk.com/learn/glossary/ (Accessed: 6 February 2022).

CoinGate (2022a) *API Integration Information | Merchant Payment Tools - CoinGate*, *CoinGate*. Available at: https://coingate.com/integration (Accessed: 1 February 2022).

CoinGate (2022b) *Crypto Payment Plugins for eCommerce Platforms - CoinGate*, *CoinGate*. Available at: https://coingate.com/plugins (Accessed: 13 February 2022).

CoinGate (2022c) *Merchant Payment Gateway · CoinGate Cryptocurrency Payment API*, *CoinGate Cryptocurrency Payment API*. Available at: https://developer.coingate.com/ (Accessed: 13 February 2022).

CoinGate (2022d) *Pricing & Service Fees - CoinGate*. Available at: https://coingate.com/pricing (Accessed: 13 February 2022).

CoinGate (2022e) *Supported Cryptocurrencies For Each Service - CoinGate*, *CoinGate*. Available at: https://coingate.com/supported-currencies (Accessed: 13 February 2022).

CoinGate (2022f) *What currencies can I get paid in (for my sales)?*, *CoinGate*. Available at: https://support.coingate.com/hc/en-us/articles/4402506061970-What-currencies-can-I-get-paid-in-for-my-sales- (Accessed: 13 February 2022).

CoinGate Blog (2019) 'CoinGate payout and settlement options - what is possible?', *CoinGate*, 8 August. Available at: https://blog.coingate.com/2019/08/payouts-fiat-settlements/ (Accessed: 13 February 2022).

CoinMarketCap (2021) *All Cryptocurrencies*, *CoinMarketCap*. Available at: https://coinmarketcap.com/all/views/all/ (Accessed: 8 December 2021).

CoinMarketCap Alexandria (2022a) *Altcoin | CoinMarketCap*, *CoinMarketCap Alexandria*. Available at: https://coinmarketcap.com/alexandria/glossary/altcoin (Accessed: 6 February 2022).

CoinMarketCap Alexandria (2022b) *Cryptocurrency | CoinMarketCap*, *CoinMarketCap Alexandria*. Available at: https://coinmarketcap.com/alexandria/glossary/cryptocurrency (Accessed: 6 February 2022).

CoinMarketCap Alexandria (2022c) *Mobile Wallet | CoinMarketCap*, *CoinMarketCap Alexandria*. Available at: https://coinmarketcap.com/alexandria/glossary/mobile-wallet (Accessed: 4 February 2022).

CoinMarketCap Alexandria (2022d) *Stablecoin | CoinMarketCap*, *CoinMarketCap Alexandria*. Available at: https://coinmarketcap.com/alexandria/glossary/stablecoin (Accessed: 6 February 2022).

CoinPayments (2022a) *CoinPayments API Services - Merchant Tool*, *CoinPayments*. Available at: https://www.coinpayments.net/apidoc (Accessed: 13 February 2022).

CoinPayments (2022b) *CoinPayments Fees*, *CoinPayments*. Available at: https://www.coinpayments.net/help-fees (Accessed: 13 February 2022).

CoinPayments (2022c) *Cryptocurrency Shopping Cart Plugins | CoinPayments*. Available at: https://www.coinpayments.net/merchant-tools-plugins (Accessed: 31 January 2022).

CoinPayments (2022d) *Multicoin Wallet - List of Supported Cryptocurrencies | CoinPayments*, *CoinPayments*. Available at: https://www.coinpayments.net/supported-coins (Accessed: 13 February 2022).

CoinPayments Blog (2018a) 'CoinPayments + Wyre Partnership', *CoinPayments Blog*, 29 June. Available at: https://blog.coinpayments.net/announcements/coinpayments-wyre-partnership (Accessed: 13 February 2022).

CoinPayments Blog (2018b) 'CoinPayments Integration: Step 3 Additional Features', *CoinPayments Blog*, 11 December. Available at: https://blog.coinpayments.net/tutorials/integrating-coinpayments-step-3-additional-features (Accessed: 13 February 2022).

CoinPayments Blog (2021) ‘Experiencing Payment Errors? We Can Help!’, *CoinPayments Blog*, 3 June. Available at: https://blog.coinpayments.net/resources/experiencing-payment-errors-we-can-help (Accessed: 13 February 2022).

CoinPayments Blog (2022) ‘About Us’, *CoinPayments Blog*. Available at: https://blog.coinpayments.net/about-us (Accessed: 31 January 2022).

COINQVEST (2022a) *E-Commerce Integrations · COINQVEST · Enterprise Cryptocurrency Payment Processing*, *COINQVEST*. Available at: https://www.coinqvest.com/en/integrations (Accessed: 13 February 2022).

COINQVEST (2022b) *Features · COINQVEST · Enterprise Cryptocurrency Payment Processing*, *COINQVEST*. Available at: https://www.coinqvest.com/en/features (Accessed: 1 February 2022).

COINQVEST (2022c) *Merchant API Docs · COINQVEST · Enterprise Cryptocurrency Payment Processing*, *COINQVEST*. Available at: https://www.coinqvest.com/en/api-docs (Accessed: 13 February 2022).

COINQVEST (2022d) *Pricing · COINQVEST · Enterprise Cryptocurrency Payment Processing*, *COINQVEST*. Available at: https://www.coinqvest.com/en/pricing (Accessed: 13 February 2022).

Crosby, M. (2016) ‘BlockChain Technology: Beyond Bitcoin’, (2), p. 16.

Crypto Listy (2021) ‘12 Best Crypto Payment Gateways in 2022’, *Crypto Listy*, 13 August. Available at: https://cryptolisty.com/payment-gateways/best-crypto-payment-gateways/ (Accessed: 11 February 2022).

Daley, S. (2021) *34 Top Blockchain Applications to Know for 2022 | Built In*. Available at: https://builtin.com/blockchain/blockchain-applications (Accessed: 23 January 2022).

Danial, K. (2020) *What Is Cryptocurrency?*, *dummies*. Available at: https://www.dummies.com/article/business-careers-money/personal-finance/cryptocurrency/what-is-cryptocurrency-237561 (Accessed: 6 February 2022).

Das, T. (2022) *The 8 Best Cryptocurrency Payment Gateways*, *MUO*. Available at: https://www.makeuseof.com/best-cryptocurrency-payment-gateways/ (Accessed: 11 February 2022).

Davies, A. (2021) ‘10 Best Bitcoin Payment Gateways for 2022’, *DevTeam.Space*, 30 November. Available at: https://www.devteam.space/blog/10-best-bitcoin-payment-gateways/ (Accessed: 11 February 2022).

Dhamodharan, R. (2021) *Why Mastercard is bringing crypto onto its network*. Available at: https://www.mastercard.com/news/perspectives/2021/why-mastercard-is-bringing-crypto-onto-our-network/ (Accessed: 18 December 2021).

DiCamillo, N. (2021) *Bitcoin Exchange LVL Launches Mastercard Debit Card*. Available at: https://www.coindesk.com/business/2021/01/22/bitcoin-exchange-lvl-launches-mastercard-debit-card/ (Accessed: 18 December 2021).

Dovarganes, D. (2022) ‘Blockchain And Cryptocurrencies’, *Harvard Model Congress 2012*, p. 13.

Due.com (2018) *7 Major Companies That Accept Cryptocurrency*. Available at: https://www.nasdaq.com/articles/7-major-companies-that-accept-cryptocurrency-2018-01-31 (Accessed: 18 December 2021).

Eikmanns, B.C. and Sandner, P.G. (2015) *Bitcoin: The Next Revolution in International Payment Processing? An Empirical Analysis of Potential Use Cases*. SSRN Scholarly Paper ID 2619759. Rochester, NY: Social Science Research Network. doi:10.2139/ssrn.2619759.

El Ioini, N. and Pahl, C. (2018) ‘A Review of Distributed Ledger Technologies: Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018, Valletta, Malta, October 22-26, 2018, Proceedings, Part II’, in, pp. 277–288. doi:10.1007/978-3-030-02671-4_16.

Federal Reserve Bank Of New York (2022) *Payments Glossary - FEDERAL RESERVE BANK of NEW YORK*. Available at: https://www.newyorkfed.org/banking/payment_glossary.html#s (Accessed: 30 March 2022).

Ferdous, M.S. *et al.* (2020) ‘Blockchain Consensus Algorithms: A Survey’, *arXiv:2001.07091 [cs]* [Preprint]. Available at: http://arxiv.org/abs/2001.07091 (Accessed: 7 January 2022).

Financial Conduct Authority (2022) *Payment Transaction - FCA Handbook*. Available at: https://www.handbook.fca.org.uk/handbook/glossary/G3490p.html (Accessed: 10 April 2022).

Folio3 (2020) ‘What are the best 5 eCommerce SaaS platforms in 2021 - Folio3’, *Blog | Folio3 Ecommerce*, 3 June. Available at: https://ecommerce.folio3.com/blog/ecommerce-saas-platforms/ (Accessed: 3 February 2022).

Fortune Business Insights (2020) *Cryptocurrency Market Size, Growth & Trends | Forecast [2028]*. Available at: https://www.fortunebusinessinsights.com/industry-reports/cryptocurrency-market-100149 (Accessed: 5 February 2022).

Fowler, M. (2013) *Domain-Driven Design Aggregate*, *martinfowler.com*. Available at: https://martinfowler.com/bliki/DDD_Aggregate.html (Accessed: 24 May 2022).

Gemini (2021a) *Crypto Wallets: Hot vs. Cold Wallets*, *Gemini*. Available at: https://www.gemini.com/cryptopedia/crypto-wallets-hot-cold (Accessed: 3 February 2022).

Gemini (2021b) *Paper Wallets: How Do They Work?*, *Gemini*. Available at: https://www.gemini.com/cryptopedia/paper-wallet-crypto-cold-storage (Accessed: 3 February 2022).

Gentilal, M., Martins, P. and Sousa, L. (2017) ‘TrustZone-backed bitcoin wallet’, in *Proceedings of the Fourth Workshop on Cryptography and Security in Computing Systems*. *CS2 ’17: Cryptography and Security in Computing Systems*, Stockholm Sweden: ACM, pp. 25–28. doi:10.1145/3031836.3031841.

Geroni, D. (2021) ‘Hot Wallet vs Cold Wallet Comparison’, *101 Blockchains*, 28 October. Available at: https://101blockchains.com/hot-wallet-vs-cold-wallet/ (Accessed: 3 February 2022).

Gerth, S. and Heim, L. (2020) *Trust through Digital Technologies: Blockchain in Online Consultancy Services*.

Global Brands (2021) 'BRAND AWARDS WINNERS 2021 - Global Brands Magazine', 8 February. Available at: https://www.globalbrandsmagazine.com/award-winners-2021/ (Accessed: 1 February 2022).

GoCardless (2021) *Difference Between Refund and Reversal Transaction*. Available at: https://gocardless.com/en-us/guides/posts/difference-between-refund-and-reversal-transaction/ (Accessed: 10 April 2022).

Goodin, D. (2008) *Net game turns PC into undercover surveillance zombie*. Available at: https://www.theregister.com/2008/10/07/clickjacking_surveillance_zombie/ (Accessed: 9 June 2022).

Goswami, A., Borasi, P. and Kumar, V. (2021) *Cryptocurrency Market By Offering Process, Type, and End User: Global Opportunity Analysis and Industry Forecast, 2021–2030*. Allied Market Research, p. 354. Available at: https://www.alliedmarketresearch.com/crypto-currency-market (Accessed: 8 December 2021).

GSMA (2018) 'Distributed Ledger Technology Blockchains and Identity'. GSM Association. Available at: https://www.gsma.com/identity/wp-content/uploads/2018/09/Distributed-Ledger-Technology-Blockchains-and-Identity-20180907ii.pdf (Accessed: 5 January 2022).

Guarda (2022) *Multi-currency Online Crypto Wallet for BTC, ETH, DOGE and more | Guarda, Guarda Wallet | Secure Crypto Wallet – Multiplatform, Non-Custodial*. Available at: https://guarda.com (Accessed: 5 February 2022).

Gupta, M. (2020) 'Blockchain For Dummies®, 3rd IBM Limited Edition', p. 50.

Haar, R. (2021) 'How to Choose a Hot Wallet or Cold Wallet for Your Crypto, and Whether You Need One at All', *Time*, 23 September. Available at: https://time.com/nextadvisor/investing/cryptocurrency/hot-wallet-vs-cold-wallet/ (Accessed: 3 February 2022).

Hancock, M. and Vaizey, E. (2016) 'Distributed Ledger Technology: beyond block chain', p. 88.

Herman, J. (2021) *Best Crypto Payment Gateways*, *Finextra Research*. Available at: https://www.finextra.com/blogposting/21303/best-crypto-payment-gateways (Accessed: 3 February 2022).

Hou, T. (2018) *IaaS vs PaaS vs SaaS: What You Need to Know, Examples & More*, *The BigCommerce Blog*. Available at: https://www.bigcommerce.com/blog/saas-vs-paas-vs-iaas/ (Accessed: 2 February 2022).

Houben, D.R. and Snyers, A. (2018) 'Cryptocurrencies and blockchain', *European Parliament*, p. 103.

Iansiti, M. and Lakhani, K.R. (2017) 'The Truth About Blockchain', *Harvard Business Review*, 1 January. Available at: https://hbr.org/2017/01/the-truth-about-blockchain (Accessed: 5 January 2022).

ICOholder (2018) *250+ Places That Accept Bitcoin Payment (Online & Physical Companies)*, *ICOholder Blog*. Available at: https://icoholder.com/blog/places-accept-bitcoin/ (Accessed: 18 December 2021).

Inzirillo, H. and Mat, B. (2021) 'Dimensionality reduction for prediction: Application to Bitcoin and Ethereum', *arXiv:2112.15036 [cs, q-fin]* [Preprint]. Available at: http://arxiv.org/abs/2112.15036 (Accessed: 23 January 2022).

Iredale, G. (2021) 'What is Software Wallet?', *101 Blockchains*, 10 November. Available at: https://101blockchains.com/software-wallet/ (Accessed: 3 February 2022).

Jaag, C. and Bach, C. (2015) 'Cryptocurrencies: New Opportunities for Postal Financial Services', p. 15.

Jahosky, J. (2020) *Study Shows Merchants That Accept Bitcoin Attract New Customers and Sales*, *Businesswire*. Available at: https://www.businesswire.com/news/home/20200929005406/en/Study-Shows-Merchants-That-Accept-Bitcoin-Attract-New-Customers-and-Sales (Accessed: 26 December 2021).

James (2021) *Can BitPay refund my order?*, *BitPay Support*. Available at: https://support.bitpay.com/hc/en-us/articles/203411523-Can-BitPay-refund-my-order- (Accessed: 13 February 2022).

Jonker, N. (2019) 'What drives the adoption of crypto-payments by online retailers?', *Electronic Commerce Research and Applications*, 35, p. 100848. doi:10.1016/j.elerap.2019.100848.

Jurgita (2021) *How can I receive a refund for a fully paid order?*, *CoinGate*. Available at: https://support.coingate.com/hc/en-us/articles/4402452097298-How-can-I-receive-a-refund-for-a-fully-paid-order- (Accessed: 13 February 2022).

Karantias, K. (2020) *SoK: A Taxonomy of Cryptocurrency Wallets*. 868. Available at: http://eprint.iacr.org/2020/868 (Accessed: 3 February 2022).

Karlo, T. (2018) *Ending Bitcoin support*. Available at: https://stripe.com/blog/ending-bitcoin-support (Accessed: 18 December 2021).

Khan, A.G. *et al.* (2019) 'Security Of Cryptocurrency Using Hardware Wallet And QR Code', in *2019 International Conference on Innovative Computing (ICIC)*. *2019 International Conference on Innovative Computing (ICIC)*, pp. 1–10. doi:10.1109/ICIC48496.2019.8966739.

Khan, R. and Hakami, T.A. (2021) 'Cryptocurrency: usability perspective versus volatility threat', *Journal of Money and Business*, ahead-of-print(ahead-of-print). doi:10.1108/JMB-11-2021-0051.

Kochkodin, B. (2021) *Venture Capital Makes a Record $17 Billion Bet on Crypto World - Bloomberg*. Available at: https://www.bloomberg.com/news/articles/2021-06-18/venture-capital-makes-a-record-17-billion-bet-on-crypto-world?sref=3REHEaVI (Accessed: 5 February 2022).

Koen, P. *et al.* (2001) 'Providing clarity and a common language to the "fuzzy front end"', *Research-Technology Management*, 44(2), pp. 46–55. doi:10.1080/08956308.2001.11671418.

Koen, P. *et al.* (2002) *1 Fuzzy Front End : Effective Methods , Tools , and Techniques*. Available at: https://www.semanticscholar.org/paper/1-Fuzzy-Front-End-%3A-Effective-Methods-%2C-Tools-%2C-and-Koen-Ajamian/b6731a73075c82622ad9babe296f853fce62bf71 (Accessed: 28 November 2021).

Koen, P.A. (2004) 'Understanding the Front End: A Common Language and Structured Picture'.

Koen, P.A. (2021) *Front End Innovation - FEI*. Available at: http://frontendinnovation.com/fei (Accessed: 28 November 2021).

Koen, P.A., Bertels, H.M.J. and Kleinschmidt, E. (2014) 'Managing the Front End of Innovation—Part I', *Research-Technology Management*, pp. 34–43.

Koen, P.A., Bertels, H.M.J. and Kleinschmidt, E.J. (2014) 'Managing the Front End of Innovation—Part II', *Research-Technology Management*, pp. 25–35.

Krause, S.K., Natarajan, H. and Gradstein, H.L. (2017) 'Distributed Ledger Technology (DLT) and blockchain'. Available at: https://documents1.worldbank.org/curated/en/177911513714062215/pdf/122140-WP-PUBLIC-Distributed-Ledger-Technology-and-Blockchain-Fintech-Notes.pdf (Accessed: 7 January 2022).

Lapierre, J. (2000) 'Customer-perceived value in industrial contexts', *Journal of Business & Industrial Marketing*, 15(2/3), pp. 122–145. doi:10.1108/08858620010316831.

LeewayHertz (2021) *All you Need to Know About Crypto Payment Gateways*, *LeewayHertz - Software Development Company*. Available at: https://www.leewayhertz.com/crypto-payment-gateway/ (Accessed: 25 January 2022).

Lenz, R. (2019) 'Managing Distributed Ledgers: Blockchain and Beyond', *SSRN Electronic Journal* [Preprint]. doi:10.2139/ssrn.3360655.

Lielacher, A. (2021) *Hot Wallets vs Cold Wallets: What's the Difference? | CoinMarketCap*, *CoinMarketCap Alexandria*. Available at: https://coinmarketcap.com/alexandria/article/hot-wallets-vs-cold-wallets-whats-the-difference (Accessed: 3 February 2022).

Lifshits, K. (2021) *15 Best Cryptocurrency Payment Gateways*, *Finextra Research*. Available at: https://www.finextra.com/blogposting/20906/15-best-cryptocurrency-payment-gateways (Accessed: 11 February 2022).

Lisa, A. (2021) *10 Major Companies That Accept Bitcoin*. Available at: https://finance.yahoo.com/news/10-major-companies-accept-bitcoin-190340692.html (Accessed: 18 December 2021).

Maishera, H. (2021) *Stripe is Open to Accepting Crypto Payments Again*. Available at: https://www.fxempire.com/forecasts/article/stripe-is-open-to-accepting-crypto-payment-again-820651 (Accessed: 18 December 2021).

Mamonova, Y. (2022) *Top 7 Bitcoin Payment Gateways for Merchants in 2022*, *Ikajo - global payment provider*. Available at: https://ikajo.com/blog/top-7-bitcoin-payment-gateways-merchants (Accessed: 11 February 2022).

Mastercard (2021) *Mastercard New Payments Index: Consumer Appetite for Digital Payments Takes Off*, *Mastercard*. Available at: https://www.mastercard.com/news/press/2021/april/mastercard-new-payments-index-consumer-appetite-for-digital-payments-takes-off/ (Accessed: 21 December 2021).

Mastercard (2022) *Mastercard Supplemental Operational Performance Data*. Available at: https://s25.q4cdn.com/479285134/files/doc_financials/2021/q4/4Q21-Mastercard-Supplemental-Operational-Performance-Data.pdf (Accessed: 29 March 2022).

MDN Web Docs (2022a) *Content Security Policy (CSP) - HTTP | MDN*. Available at: https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP (Accessed: 9 June 2022).

MDN Web Docs (2022b) *Strict-Transport-Security - HTTP | MDN*. Available at: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security (Accessed: 9 June 2022).

Meijer, D.B. (2017) 'Consequences of the implementation of blockchain technology'. Available at: https://repository.tudelft.nl/islandora/object/uuid%3Ada0b8d80-d19e-4149-bfbd-64b0ca79042a (Accessed: 5 January 2022).

Modestas (2021) *How long do I have to wait for my cryptocurrency payout after a successful phone payment?*, *CoinGate*. Available at: https://support.coingate.com/hc/en-us/articles/4402499476242-How-long-do-I-have-to-wait-for-my-cryptocurrency-payout-after-a-successful-phone-payment- (Accessed: 13 February 2022).

Mohamed, A., Almasri, F. and Lasheen, I. (2018) 'The effect of Bitcoin on E-Commerce', 1, pp. 9–17.

Moriarty, K. and Farrell, S. (2021) *Deprecating TLS 1.0 and TLS 1.1*. Request for Comments RFC 8996. Internet Engineering Task Force. doi:10.17487/RFC8996.

Motola, C. (2021) *The 5 Most Popular Crypto Payment Gateway & Processors*, *Merchant Maverick*. Available at: https://www.merchantmaverick.com/best-cryptocurrency-payment-gateway/ (Accessed: 11 February 2022).

Nadeem, M.A. *et al.* (2021) 'Investigating the Adoption Factors of Cryptocurrencies—A Case of Bitcoin: Empirical Evidence From China', *SAGE Open*, 11(1), p. 2158244021998704. doi:10.1177/2158244021998704.

Nakamoto, S. (2008) 'Bitcoin: A Peer-to-Peer Electronic Cash System', p. 9.

Neureuter, J. (2021) 'THE INSTITUTIONAL INVESTOR DIGITAL ASSETS STUDY', p. 40.

NOWPayments (2020) *NOWPayments: Unlimited number of crypto wallets for your payments*. Available at: https://nowpayments.io/blog/new-update-add-as-much-outcome-wallets-as-you-want (Accessed: 13 February 2022).

NOWPayments (2022a) *Accept payments in Bitcoin and 60+ altcoins*, *NOWPayments*. Available at: https://nowpayments.io/payment-tools (Accessed: 13 February 2022).

NOWPayments (2022b) *Fiat Processing: Cash In or Cash Out in Crypto or Fiat*, *NOWPayments*. Available at: https://nowpayments.io/fiat (Accessed: 1 February 2022).

NOWPayments (2022c) *Instant Payouts in Cryptocurrency: Payout your Bitcoins*, *NOWPayments*. Available at: https://nowpayments.io/instant-payouts (Accessed: 13 February 2022).

NOWPayments (2022d) *NOWPayments API*, *NOWPayments API*. Available at: https://documenter.getpostman.com/view/7907941/S1a32n38 (Accessed: 13 February 2022).

NOWPayments (2022e) *Pricing on crypto payments | Transaction & Exchange fees*, *NOWPayments*. Available at: https://nowpayments.io/pricing (Accessed: 1 February 2022).

NOWPayments (2022f) *Refund policy*, *NOWPayments*. Available at: https://nowpayments.io/help/payments/common/refund-policy (Accessed: 13 February 2022).

NOWPayments (2022g) *What cryptocurrencies are supported for payments?*, *NOWPayments*. Available at: https://nowpayments.io/supported-coins (Accessed: 13 February 2022).

NPD Solutions (2019) ‘Value Analysis and Function Analysis System Technique’. Available at: https://www.npd-solutions.com/va.html (Accessed: 21 November 2021).

Olszowy, M. (2020a) *Introduction to Cryptocurrency Payment Refunds with COINQVEST*, *COINQVEST Blog*. Available at: https://www.coinqvest.com/en/blog/introduction-to-cryptocurrency-payment-refunds-with-coinqvest-1464ac3a5097 (Accessed: 13 February 2022).

Olszowy, M. (2020b) *Merchant Payouts with COINQVEST - Withdrawing Funds to Bank Accounts and Cryptocurrency Wallets*, *COINQVEST Blog*. Available at: https://www.coinqvest.com/en/blog/merchant-payouts-with-coinqvest-withdrawing-funds-to-bank-accounts-and-cryptocurrency-wallets-54316dd8edd6 (Accessed: 13 February 2022).

Osterwalder, A. *et al.* (2014) ‘Value Proposition Design: How to Create Products and Services Customers Want | Wiley’.

Osterwalder, A. and Pigneur, Y. (2010) ‘Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers’.

OWASP (2022a) *Clickjacking Defense - OWASP Cheat Sheet Series*. Available at: https://cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html (Accessed: 9 June 2022).

OWASP (2022b) *Free for Open Source Application Security Tools | OWASP Foundation*. Available at: https://owasp.org/www-community/Free_for_Open_Source_Application_Security_Tools (Accessed: 9 June 2022).

OWASP (2022c) *OWASP Top Ten Web Application Security Risks | OWASP*. Available at: https://owasp.org/www-project-top-ten/ (Accessed: 7 June 2022).

PayPal (2022) ‘Fourth Quarter and Full Year 2021 Results’, p. 16.

Peffers, K. *et al.* (2006) 'The Design Science Research Process: A Model For Producing And Presenting Information Systems Research', *SYSTEMS RESEARCH*, p. 24.

Peffers, K. *et al.* (2007) 'A Design Science Research Methodology for Information Systems Research', *Journal of Management Information Systems*, 24(3), pp. 45–77. doi:10.2753/MIS0742-1222240302.

Plisio (2022a) *Accept Cryptocurrencies with Plisio*, *WordPress.org España*. Available at: https://es.wordpress.org/plugins/plisio-payment-gateway-for-woocommerce/ (Accessed: 13 February 2022).

Plisio (2022b) *Cryptocurrency Payment Gateway*, *Plisio*. Available at: https://plisio.net/ (Accessed: 12 February 2022).

Plisio (2022c) *Documentation*, *Plisio*. Available at: https://plisio.net/documentation (Accessed: 13 February 2022).

Plisio (2022d) *Pricing and Fees - Cryptocurrency Payment Gateway*, *Plisio*. Available at: https://plisio.net/pricing (Accessed: 12 February 2022).

Plisio (2022e) *Supported cryptocurrencies*, *Plisio*. Available at: https://plisio.net/documentation/appendices/supported-cryptocurrencies (Accessed: 13 February 2022).

Plisio (2022f) *Withdrawal / Mass withdrawal*, *Plisio*. Available at: https://plisio.net/documentation/endpoints/withdrawal-mass-withdrawal (Accessed: 13 February 2022).

Pressman, R.S. (2010) *Software engineering: a practitioner's approach*. 7th ed. New York: McGraw-Hill Higher Education.

Rezaeighaleh, H. and Zou, C.C. (2019a) 'Deterministic Sub-Wallet for Cryptocurrencies', in *2019 IEEE International Conference on Blockchain (Blockchain)*. *2019 IEEE International Conference on Blockchain (Blockchain)*, pp. 419–424. doi:10.1109/Blockchain.2019.00064.

Rezaeighaleh, H. and Zou, C.C. (2019b) 'New Secure Approach to Backup Cryptocurrency Wallets', in *2019 IEEE Global Communications Conference (GLOBECOM)*. *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9014007.

Rich, N. and Holweg, M. (2000) *Value engineering: Innoregio: dissemination of innovation and knowledge management techniques, report produced for the EC funded project*. United Kingdom: Lean Enterprise Research Centre Cardiff, p. 32.

Ryan, S. (2021) *APIs vs. Webhooks: What's the difference?* Available at: https://www.mparticle.com/blog/apis-vs-webhooks (Accessed: 28 May 2022).

Saaty, T.L. (1984) 'The Analytic Hierarchy Process: Decision Making in Complex Environments', in Avenhaus, R. and Huber, R.K. (eds) *Quantitative Assessment in Arms Control: Mathematical Modeling and Simulation in the Analysis of Arms Control Problems*. Boston, MA: Springer US, pp. 285–308. doi:10.1007/978-1-4613-2805-6_12.

Sana Commerce (2022) *SaaS e-commerce*, *Sana Commerce*. Available at: https://www.sana-commerce.com/e-commerce-terms/what-is-saas-e-commerce/ (Accessed: 3 February 2022).

Santos, R. (2022) *rutesantos4/tmdei-21-22: Master Dissertation Project, Specialisation Area of Software Engineering at ISEP*, *GitHub*. Available at: https://github.com/rutesantos4/tmdei-21-22 (Accessed: 22 June 2022).

Schwarz, L. (2016) *What is an Ecommerce Platform?*, *NetSuite.com*. Available at: https://www.netsuite.com/portal/resource/articles/ecommerce/what-is-an-ecommerce-platform.shtml (Accessed: 3 February 2022).

Selimović, A. *et al.* (2021) 'Cryptocurrency - Advantages, Disadvantages, Determinants: Case of Bitcoin', *Sarajevo Business and Economics Review*, 39, pp. 77–98.

SendGrid (2014) *What's a Webhook?*, *SendGrid*. Available at: https://sendgrid.com/blog/whats-webhook/ (Accessed: 28 May 2022).

SendPulse (2022) *What is an eCommerce Platform: Features, Best Platforms - Definition*, *SendPulse*. Available at: https://sendpulse.com/support/glossary/ecommerce-platform (Accessed: 3 February 2022).

Seth, S. (2022) *What Is a Cryptocurrency Payment Gateway?*, *Investopedia*. Available at: https://www.investopedia.com/tech/bitcoin-payment-services-introduction/ (Accessed: 26 January 2022).

Sheth, H. and Dattani, J. (2019) 'Overview of Blockchain Technology', *Asian Journal For Convergence In Technology (AJCT) ISSN -2350-1146* [Preprint]. Available at: https://asianssr.org/index.php/ajct/article/view/728 (Accessed: 5 January 2022).

Singh, A. (2021) 'Top 13 SAAS eCommerce Platform Software for Your Online Business', *SoftwareSuggest Blog*, 23 June. Available at: http://trak.in/tags/business/2014/06/30/top-10-saas-ecommerce-platforms-india/ (Accessed: 3 February 2022).

Snyk (2022) *Snyk Vulnerability Database | Snyk*, *Snyk Vulnerability Database*. Available at: https://snyk.io/vuln (Accessed: 7 June 2022).

Solingen, R. and Berghout, E. (1999) 'The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development'.

Son, H. (2021) *Mastercard says any bank or merchant on its vast network can soon offer crypto services*, *CNBC*. Available at: https://www.cnbc.com/2021/10/25/mastercard-says-any-bank-or-merchant-on-its-vast-network-can-soon-offer-crypto-services.html (Accessed: 18 December 2021).

SonarQube (2022) *Code Quality and Code Security | SonarQube*. Available at: https://www.sonarqube.org/ (Accessed: 7 June 2022).

SourceForge (2022a) *ALFAcoins*, *SourceForge*. Available at: https://sourceforge.net/software/product/ALFAcoins/ (Accessed: 13 February 2022).

SourceForge (2022b) *Blockonomics*, *SourceForge*. Available at: https://sourceforge.net/software/product/Blockonomics/ (Accessed: 13 February 2022).

SourceForge (2022c) *COINQVEST*, *SourceForge*. Available at: https://sourceforge.net/software/product/COINQVEST/ (Accessed: 13 February 2022).

SourceForge (2022d) *Plisio*, *SourceForge*. Available at: https://sourceforge.net/software/product/Plisio/ (Accessed: 13 February 2022).

SourceForge (2022e) *SpicePay*, *SourceForge*. Available at: https://sourceforge.net/software/product/SpicePay/ (Accessed: 13 February 2022).

SpicePay (2022a) 'Spicepay - Accept Payment', *Accept cryptocurrencies, payment gateway for bitcoin merchant accounts | SpicePay*. Available at: https://www.spicepay.com/accept-payment-2/ (Accessed: 12 February 2022).

SpicePay (2022b) *SpicePay - Repositories*, *GitHub*. Available at: https://github.com/spicepay (Accessed: 13 February 2022).

SSL Labs (2020) *SSL and TLS Deployment Best Practices · ssllabs/research Wiki*, *GitHub*. Available at: https://github.com/ssllabs/research (Accessed: 9 June 2022).

Strategyzer AG (2020) *Value Proposition Canvas – Download the Official Template*. Available at: https://www.strategyzer.com/canvas/value-proposition-canvas (Accessed: 16 December 2021).

Suratkar, S., Shirole, M. and Bhirud, S. (2020) 'Cryptocurrency Wallet: A Review', in *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP)*. *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP)*, pp. 1–7. doi:10.1109/ICCCSP49186.2020.9315193.

Sweet32 (2022) *Sweet32: Birthday attacks on 64-bit block ciphers in TLS and OpenVPN*. Available at: https://sweet32.info/ (Accessed: 9 June 2022).

Techopedia (2014) *What is Dynamic Application Security Testing (DAST)? - Definition from Techopedia*, *Techopedia.com*. Available at: http://www.techopedia.com/definition/30958/dynamic-application-security-testing-dast (Accessed: 9 June 2022).

The BitPay Blog (2013) *BitPay Surpasses 10,000 Merchants*, *The BitPay Blog*. Available at: https://bitpay.com/blog/bitpay-surpasses-10000-merchants/ (Accessed: 30 January 2022).

The BitPay Blog (2020) *Introducing The New BitPay Card*, *The BitPay Blog*. Available at: https://bitpay.com/blog/introducing-new-bitpay-card/ (Accessed: 31 January 2022).

The Economic Times (2021) *Cryptocurrency is gaining worldwide acceptance, here are 5 reasons why - The Economic Times*. Available at: https://economictimes.indiatimes.com/markets/cryptocurrency/cryptocurrency-is-gaining-worldwide-acceptance-here-are-5-reasons-why/articleshow/87209465.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cppst (Accessed: 5 February 2022).

Thompson, B. (2020) *BEST Crypto Wallets: Top 20 Bitcoin Wallets App for 2022*. Available at: https://www.guru99.com/best-bitcoin-cryptocurrency-wallets.html (Accessed: 5 February 2022).

Treiblmaier, H. and Sillaber, C. (2021) 'The impact of blockchain on e-commerce: A framework for salient research topics', *Electronic Commerce Research and Applications*, 48, p. 101054. doi:10.1016/j.elerap.2021.101054.

TripleA (2021) *Global Cryptocurrency Ownership Data 2021*, *TripleA*. Available at: https://triple-a.io/crypto-ownership/ (Accessed: 26 December 2021).

Tuwiner, J. (2021) *9 Major Companies Who Accept Bitcoin [Spend Crypto 2022]*, *Buy Bitcoin Worldwide*. Available at: https://www.buybitcoinworldwide.com/who-accepts-bitcoin/ (Accessed: 18 December 2021).

University of Minnesota (2016) '24.1 What Is Money?', *Principles of Economics* [Preprint]. Available at: https://open.lib.umn.edu/principleseconomics/chapter/24-1-what-is-money/ (Accessed: 27 January 2022).

Value Analysis Canada (2022) *Function Analysis system Technique (FAST) - Canadian Society of Value Analysis*, *Value Analysis Canada*. Available at: https://www.valueanalysis.ca/fast.php (Accessed: 30 January 2022).

Verma, N., Jain, S. and Doriya, R. (2021) 'Review on Consensus Protocols for Blockchain', in *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*. *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pp. 281–286. doi:10.1109/ICCCIS51004.2021.9397089.

Visa (2021a) *PayPal Launches New Service Enabling Users to Buy, Hold and Sell Cryptocurrency*, *PayPal Newsroom*. Available at: https://newsroom.paypal-corp.com/2020-10-21-PayPal-Launches-New-Service-Enabling-Users-to-Buy-Hold-and-Sell-Cryptocurrency (Accessed: 18 December 2021).

Visa (2021b) 'The Crypto Phenomenon: Consumer Attitudes & Usage', *LRW, a Material Company*, p. 30.

Visa (2021c) *Visa Inc. Q1 2021 Operational Performance Data*. Available at: https://s1.q4cdn.com/050606653/files/doc_financials/2021/q1/Visa-Inc.-Q1-2021-Operational-Performance-Data.pdf (Accessed: 29 March 2022).

Visa (2022) *Visa Inc. Q1 2022 Operational Performance Data*. Available at: https://s1.q4cdn.com/050606653/files/doc_financials/2022/q1/Q1FY22-Visa-Operational-Performance-Data-FINAL.pdf (Accessed: 29 March 2022).

Wang, M. (2020) 'Bitcoin and its impact on the economy', p. 9.

Wegrzyn, K.E. and Wang, E. (2021) *Types of Blockchain: Public, Private, or Something in Between | Foley & Lardner LLP*, *Foley & Lardner LLP*. Available at: https://www.foley.com/en/insights/publications/2021/08/types-of-blockchain-public-private-between (Accessed: 5 January 2022).

Wesley Chai (2021) *What is SaaS (Software as a Service)? Everything You Need to Know*, *SearchCloudComputing*. Available at: https://www.techtarget.com/searchcloudcomputing/definition/Software-as-a-Service (Accessed: 2 February 2022).

Wirex Team (2020) *5 Reasons the New Mastercard Card is the Ideal Travel Companion*, *Crypto-Friendly Currency Accounts*. Available at: https://wirexapp.com/blog/post/5-reasons-the-new-mastercard-card-is-the-ideal-travel-companion-0247 (Accessed: 18 December 2021).

Wong, R. (2021) *COINQVEST Receives 0.5M USD as Equity-Free Grant through the Stellar Seed Fund*. Available at: https://www.coinqvest.com/en/blog/coinqvest-receives-0-5m-usd-as-equity-free-grant-through-the-stellar-seed-fund-5ce86d303280 (Accessed: 1 February 2022).

Wouters, S. (2021) *When might the Bitcoin network process volumes like Mastercard & Visa?*, *Blockdata*. Available at: https://www.blockdata.tech/blog/general/bitcoin-volume-mastercard-visa (Accessed: 27 March 2022).

WP Favs (2022) *Spicepay WooCommerce Plugin*, *WP Favs*. Available at: https://wpfavs.com/plugins/spicepay (Accessed: 13 February 2022).

Yaga, D. *et al.* (2018) *Blockchain technology overview*. NIST IR 8202. Gaithersburg, MD: National Institute of Standards and Technology, p. NIST IR 8202. doi:10.6028/NIST.IR.8202.

Young, J. (2018) *South Korea's Largest ECommerce Platform is Integrating Bitcoin*, *CCN.com*. Available at: https://www.ccn.com/south-koreas-largest-e-commerce-platform-integrating-12-cryptocurrencies-including-bitcoin/ (Accessed: 26 December 2021).

Zeithaml, V. (1988) 'Consumer Perceptions of Price, Quality and Value: A Means-End Model and Synthesis of Evidence', *Journal of Marketing*, 52, pp. 2–22. doi:10.1177/002224298805200302.

Appendix A
Glossary

Aggregate Is “a pattern in Domain-Driven Design. A Domain-Driven Design aggregate is a cluster of domain objects that can be treated as a single unit.

An aggregate will have one of its component objects be the aggregate root. Any references from outside the aggregate should only go to the aggregate root. The root can thus ensure the integrity of the aggregate as a whole.” (Fowler, 2013)

Final Settlement Is “the final step in the transfer of ownership involving the physical exchange of securities or payment” (Federal Reserve Bank Of New York, 2022).

After the settlement, the obligation is irrevocably discharged by the Financial Market Infrastructures or its participants following the terms of the underlying contract (Bank for International Settlements, 2015). The transaction is considered complete at this moment.

Payment Is “the action of transferring funds, initiated by the payer or on its behalf or by the payee, irrespective of any underlying obligations between the payer and the payee” (Financial Conduct Authority, 2022).

Refund Is “the process of reimbursing somebody for a transaction which has already been completed” (GoCardless, 2021).

Webhook Is “a way for an app to provide other applications with real-time information. A webhook delivers data to other applications as it happens, meaning you get data immediately. Unlike typical APIs where you would need to poll for data very frequently to get it real-time” (SendGrid, 2014).

“Webhooks are sometimes referred to as “reverse APIs,” because communication is initiated by the application sending the data rather than the one receiving it” (Ryan, 2021).

OWASP Top 10 Is a “standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications” (OWASP, 2022c) .

Appendix B
Analysis of top cryptocurrency payment gateways

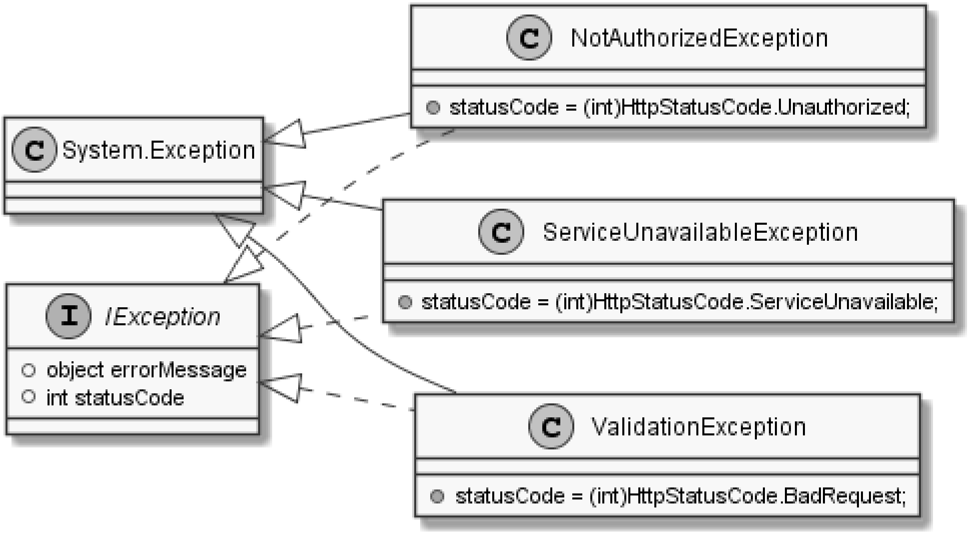
Table 18 - Top websites that list the top cryptocurrency payment gateways by clients

(Davies, 2021)	(Herman, 2021)	(Motola, 2021)	(Das, 2022)	(Mamonova, 2022)	(Crypto Listy, 2021)	(Bansal, 2021)	(Agrawal, 2021)	(Banguis, 2021)	(Lifshits, 2021)
Coinbase Coingate CoinsBank ALFACoins Shopify BitPay GoCoin BitcoinPay GoURL SpicePay	Coinbase CoinsPaid Bitpay CoinPayments Coingate Blockonomics Blockchain.info Spectrocoin Plisio payWALA OpenNode CoinsBank B2BinPAY TripleA	Coinbase Bitpay Coingate NOWPayments ALFACoins	Coingate Coinbase BitPay ALFACoins GoURL Shopify Gateway NOWPayments SpicePay	Bitpay Ikajo International CoinPayments PaySpacelv CoinGate Coinbase GoCoin	COINQVEST CoinGate NOWPayments Payid19 CoinPayments CoinRemitter BTCPay Server BitPay Crypto.com GoCrypto Plisio CryptAPI	Coingate CoinPayments NOWPayments Bitpay Shopify Gateway GoURL Coinbase CoinsBank ALFACoins SpicePay	CoinPayments Binance Pay BitPay CoinBase NOWPayments CoinGate Blockchain.info Spectrocoin GoURL	BitPay COINQVEST Coinbase Coingate CoinsBank GoCoin BitcoinPay	Coinbase NOWPayments Blockonomics Coingate Blockchain.info Spectrocoin CoinPayments Bitpay Plisio payWALA OpenNode CoinsBank B2BinPAY TripleA SpicePay

Table 19 - Top cryptocurrency payment gateways by clients and their respective occurrence
(Blue - top 5 cryptocurrency payment gateways that do not provide a free trial
Green - top 5 cryptocurrency payment gateways that provide a free trial)

Cryptocurrency payment gateway	Occurrences	Free trial
BitPay	10	No
CoinGate	10	No
Coinbase Commerce	9	No
CoinPayments	6	No
NOWPayments	6	No
CoinsBank	5	No
SpicePay	5	Yes (SourceForge, 2022e)
ALFACoins	4	Yes (SourceForge, 2022a)
GoURL	4	No
Blockchain.info	3	No
GoCoin	3	No
Plisio	3	Yes (SourceForge, 2022d)
Shopify	3	No
Spectrocoin	3	No
B2BinPAY	2	No
BitcoinPay	2	No
Blockonomics	2	Yes (Blockonomics, 2022a; SourceForge, 2022b)
COINQVEST	2	Yes (SourceForge, 2022c)
OpenNode	2	No
payWALA	2	No
TripleA	2	No
Binance Pay	1	No
BTCPay Server	1	No
CoinRemitter	1	No
CoinsPaid	1	No
CryptAPI	1	No
Crypto.com	1	No
Ikajo International	1	No
Payid19	1	No
PaySpacelv	1	No

Appendix C
Exception Handling



C NotAuthorizedException
statusCode = (int)HttpStatusCode.Unauthorized;
C System.Exception
C ServiceUnavailableException
statusCode = (int)HttpStatusCode.ServiceUnavailable;
I IException
object errorMessage
int statusCode
C ValidationException
statusCode = (int)HttpStatusCode.BadRequest;

Figure 25 - Interface IException and its implementation classes), presented in a UML Class Diagram

UC2: Convert fiat currency to cryptocurrency

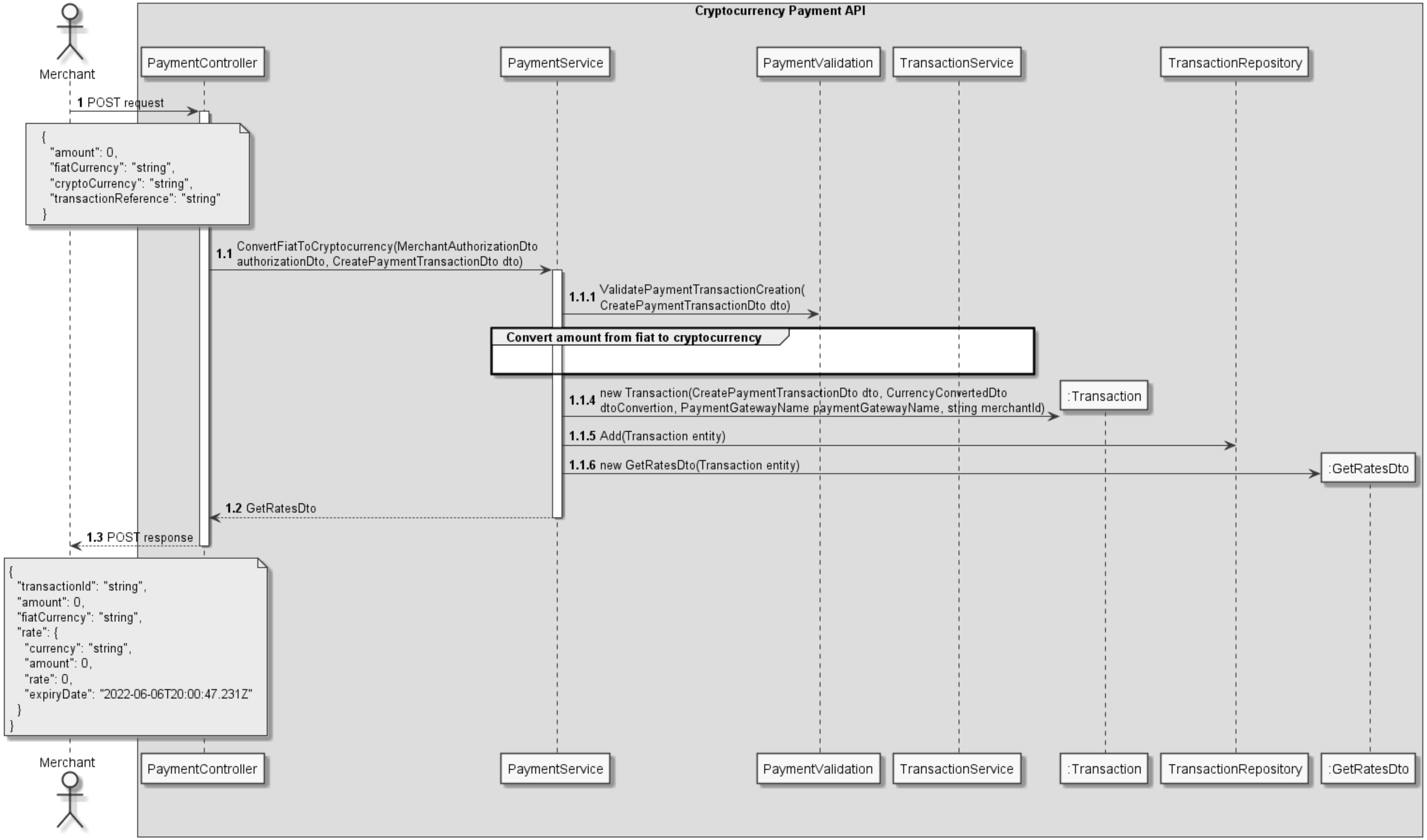

Cryptocurrency Payment API
Merchant
PaymentController
PaymentService
PaymentValidation
TransactionService
TransactionRepository
1 POST request
{
"amount": 0,
"fiatCurrency": "string",
"cryptoCurrency": "string",
"transactionReference": "string"
}
1.1 ConvertFiatToCryptocurrency(MerchantAuthorizationDto authorizationDto, CreatePaymentTransactionDto dto)
1.1.1 ValidatePaymentTransactionCreation(CreatePaymentTransactionDto dto)
Convert amount from fiat to cryptocurrency
1.1.4 new Transaction(CreatePaymentTransactionDto dto, CurrencyConvertedDto dtoConvertion, PaymentGatewayName paymentGatewayName, string merchantId)
:Transaction
1.1.5 Add(Transaction entity)
1.1.6 new GetRatesDto(Transaction entity)
:GetRatesDto
1.2 GetRatesDto
1.3 POST response
{
"transactionId": "string",
"amount": 0,
"fiatCurrency": "string",
"rate": {
"currency": "string",
"amount": 0,
"rate": 0,
"expiryDate": "2022-06-06T20:00:47.231Z"
}
}
Merchant
PaymentController
PaymentService
PaymentValidation
TransactionService
:Transaction
TransactionRepository
:GetRatesDto

Figure 26 - Flow of converting fiat currency to cryptocurrency (with Interaction Use), presented in a UML Sequence Diagram
(Application: CryptocurrencyPaymentAPI)

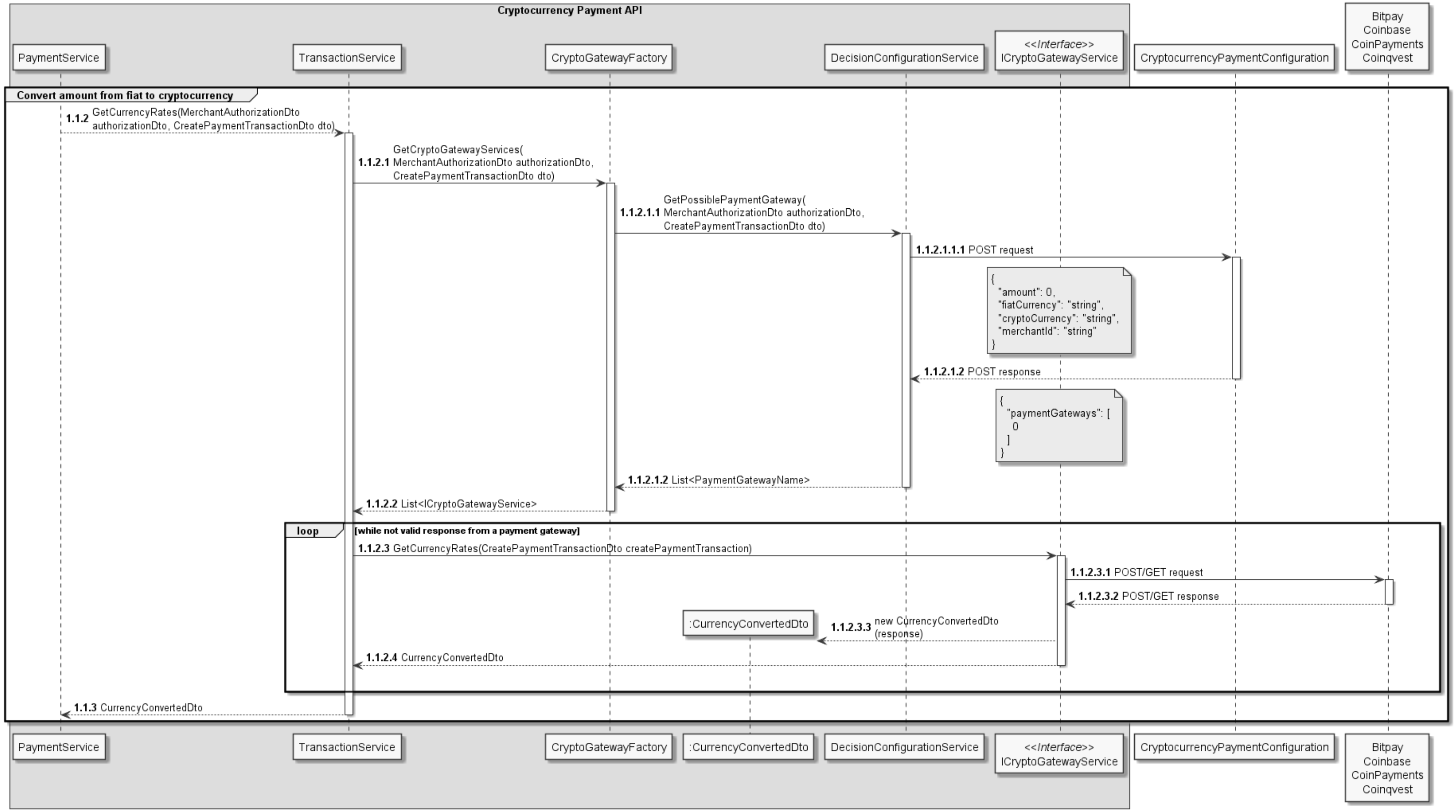
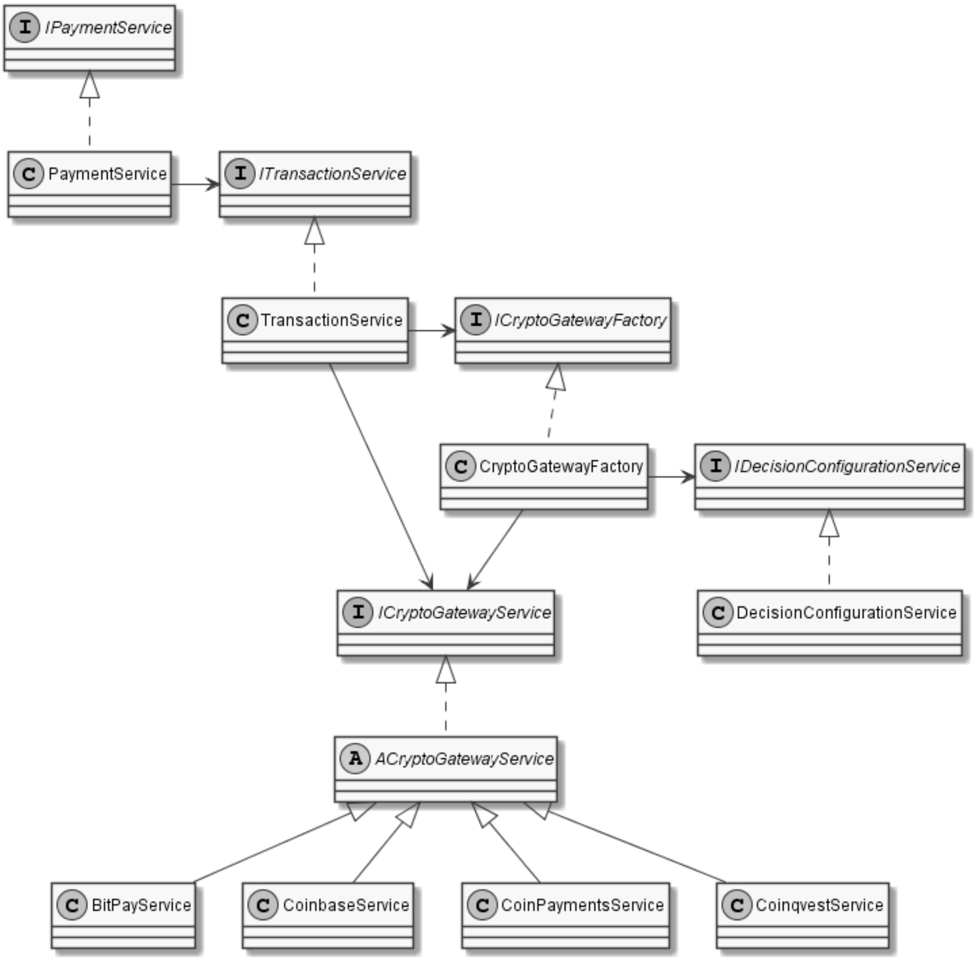

Cryptocurrency Payment API
PaymentService
TransactionService
CryptoGatewayFactory
DecisionConfigurationService
<<Interface>>
ICryptoGatewayService
CryptocurrencyPaymentConfiguration
Bitpay
Coinbase
CoinPayments
Coinqvest
Convert amount from fiat to cryptocurrency
1.1.2 GetCurrencyRates(MerchantAuthorizationDto authorizationDto, CreatePaymentTransactionDto dto)
1.1.2.1 GetCryptoGatewayServices(MerchantAuthorizationDto authorizationDto, CreatePaymentTransactionDto dto)
1.1.2.1.1 GetPossiblePaymentGateway(MerchantAuthorizationDto authorizationDto, CreatePaymentTransactionDto dto)
1.1.2.1.1.1 POST request
{ "amount": 0, "fiatCurrency": "string", "cryptoCurrency": "string", "merchantId": "string" }
1.1.2.1.2 POST response
{ "paymentGateways": [0] }
1.1.2.1.2 List<PaymentGatewayName>
1.1.2.2 List<ICryptoGatewayService>
loop
[while not valid response from a payment gateway]
1.1.2.3 GetCurrencyRates(CreatePaymentTransactionDto createPaymentTransaction)
1.1.2.3.1 POST/GET request
1.1.2.3.2 POST/GET response
:CurrencyConvertedDto
1.1.2.3.3 new CurrencyConvertedDto (response)
1.1.2.4 CurrencyConvertedDto
1.1.3 CurrencyConvertedDto
PaymentService
TransactionService
CryptoGatewayFactory
:CurrencyConvertedDto
DecisionConfigurationService
<<Interface>>
ICryptoGatewayService
CryptocurrencyPaymentConfiguration
Bitpay
Coinbase
CoinPayments
Coinqvest

Figure 27 - Flow of converting fiat currency to cryptocurrency (Interaction Use part), presented in a UML Sequence Diagram
(Application: CryptocurrencyPaymentAPI)



Interface	Description
IPaymentService	Interface that specifies the functions that are responsible to validate the request, store or update the transaction and create the response.
ITransactionService	Interface that specifies the functions that are responsible to convert the currencies and create the transaction on the selected payment gateway. For that it calls the functions of ICryptoGatewayFactory (to find the payment gateway) and ICryptoGatewayService (to convert currencies and create the transaction).
ICryptoGatewayFactory	Interface that specifies the functions that are responsible to create the ICryptoGatewayService.
ICryptoGatewayService	Interface that specifies the functions that are in common to all the payment gateways.
IDecisionConfigurationService	Interface that specifies the functions responsible to communicate with the project CryptocurrencyPaymentConfiguration to obtain the payment gateway(s) and cryptocurrencies supported by the merchant.

Figure 28 - Services created to handle the conversion of fiat currency to cryptocurrency, presented in a UML Class Diagram
(Application: CryptocurrencyPaymentAPI)

Appendix D
Project Planning

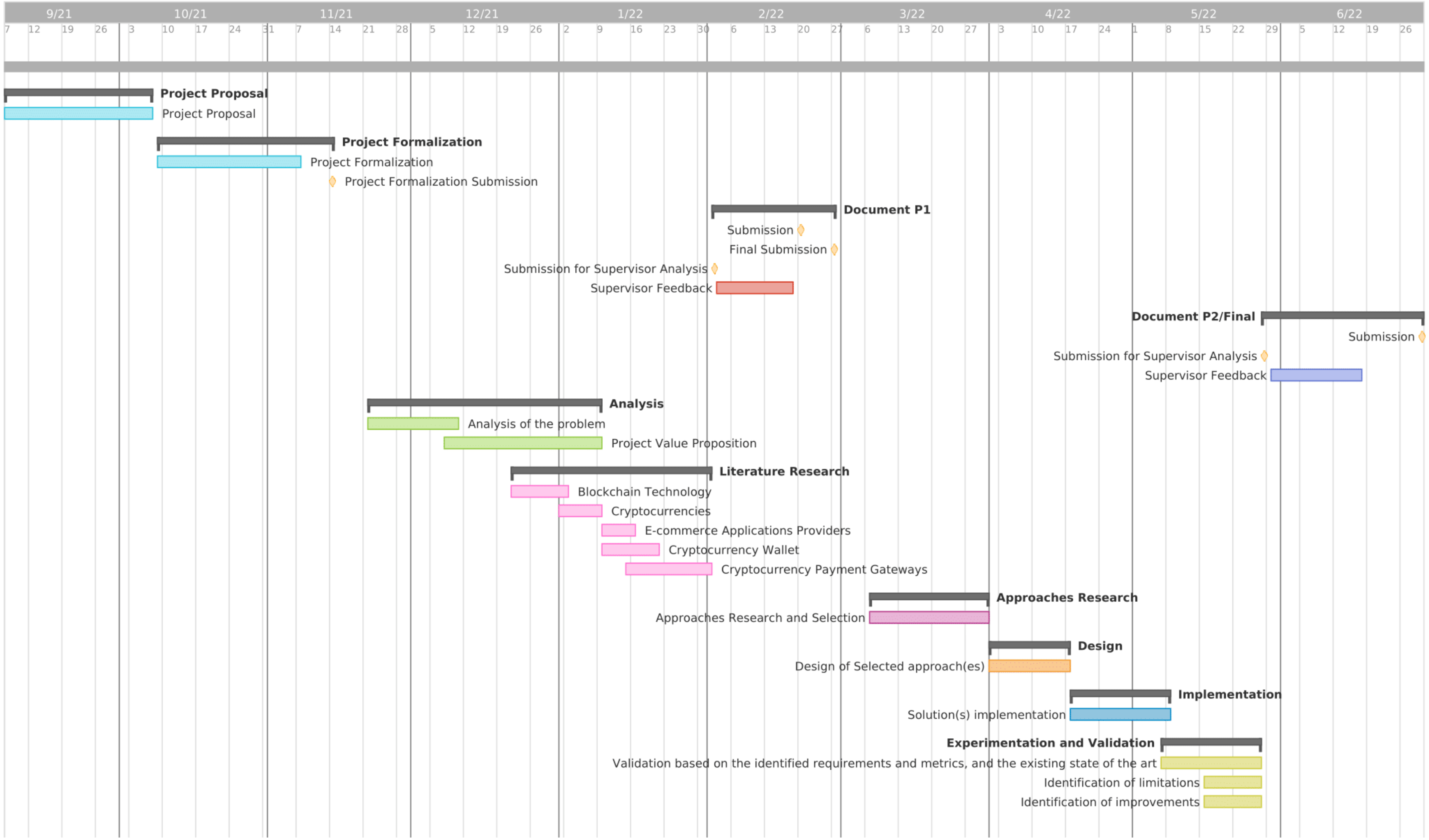
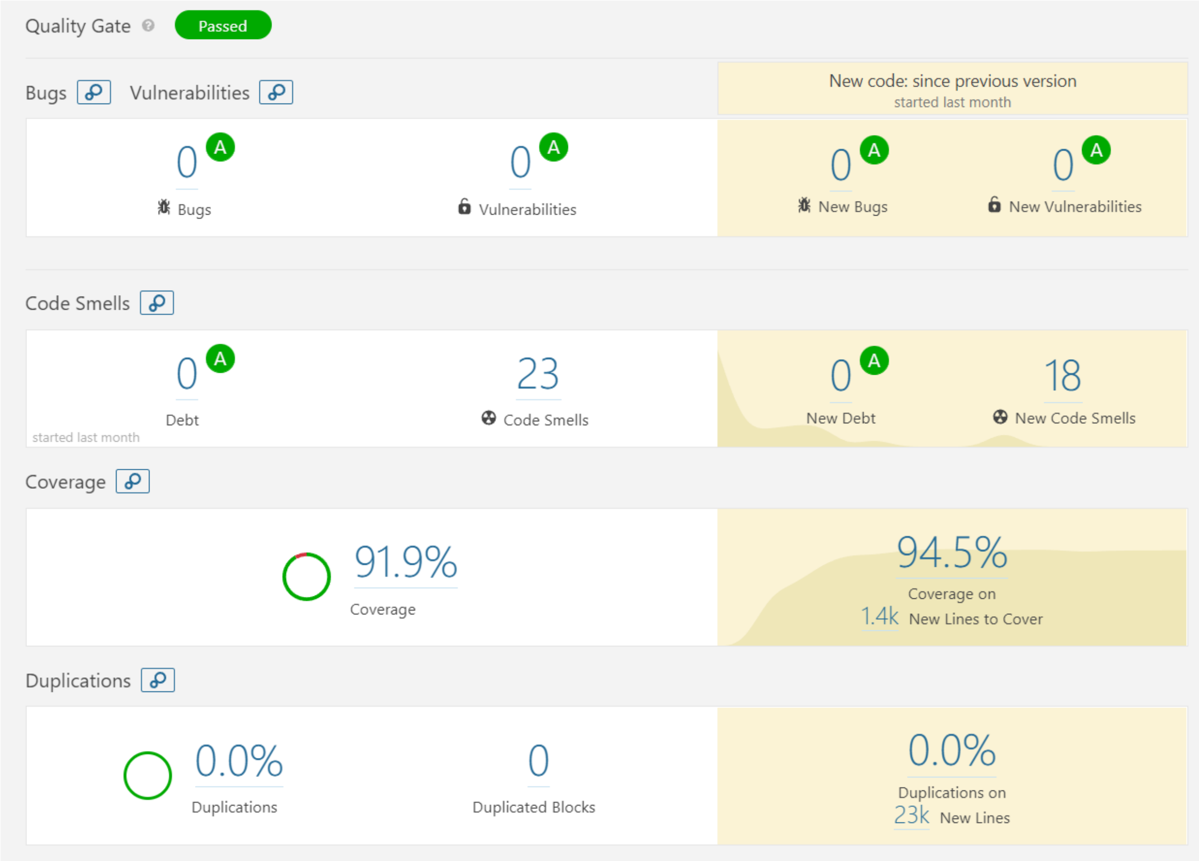


Figure 29 - Gantt Chart diagram providing an illustrative representation of the project planning phases

Appendix E
SonarQube Analysis



Quality Gate Passed

Bugs Vulnerabilities

New code: since previous version
started last month

0 A Bugs

0 A Vulnerabilities

0 A New Bugs

0 A New Vulnerabilities

Code Smells

0 A Debt
started last month

23 Code Smells

0 A New Debt

18 New Code Smells

Coverage

91.9% Coverage

94.5%
Coverage on
1.4k New Lines to Cover

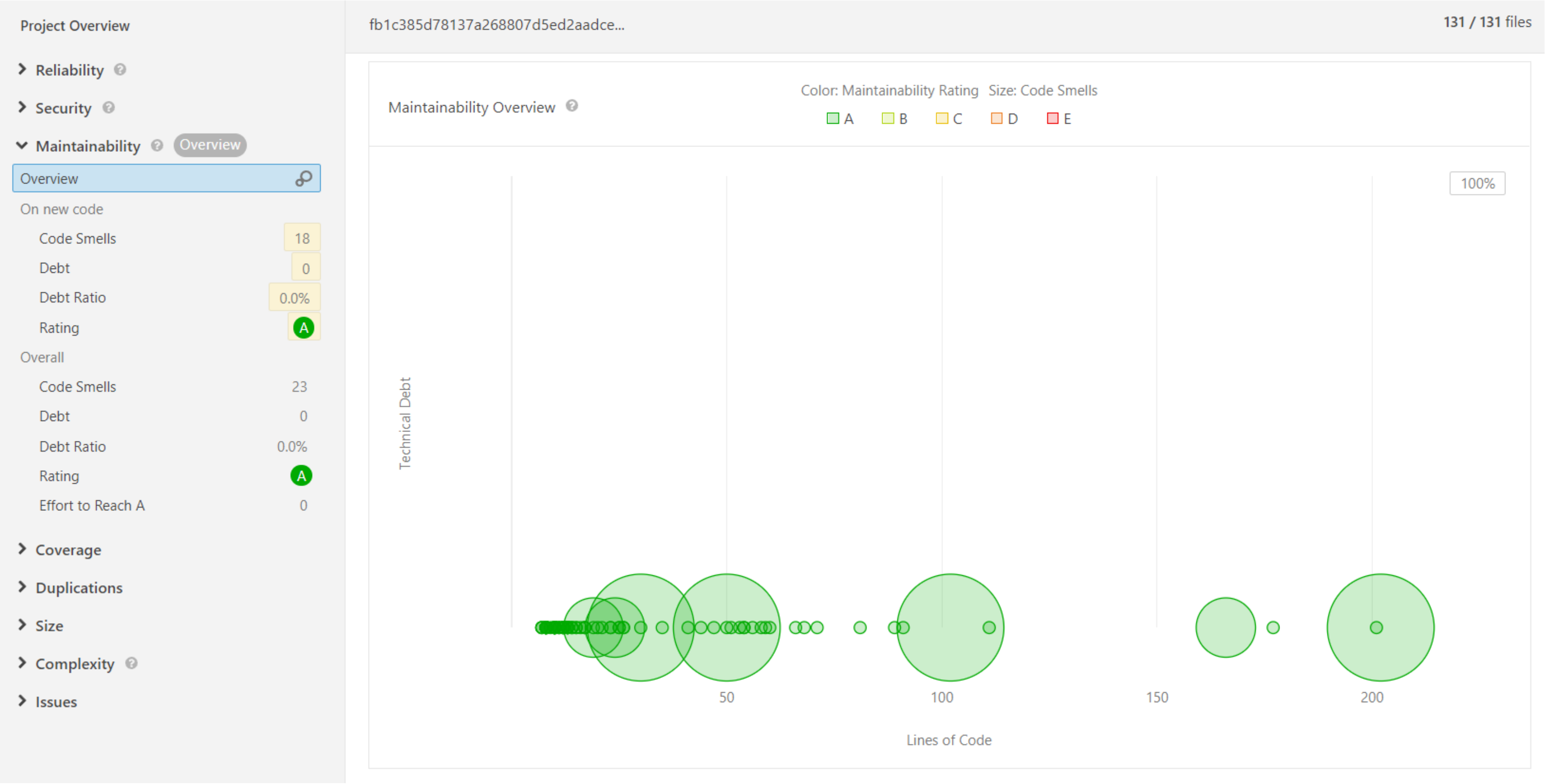
Duplications

0.0% Duplications

0 Duplicated Blocks

0.0%
Duplications on
23k New Lines

Figure 30 - Overview of SonarQube analysis results



131 / 131 files

Overview

On new code

Code Smells 18

Debt 0

Debt Ratio 0.0%

Rating A

Overall

Code Smells 23

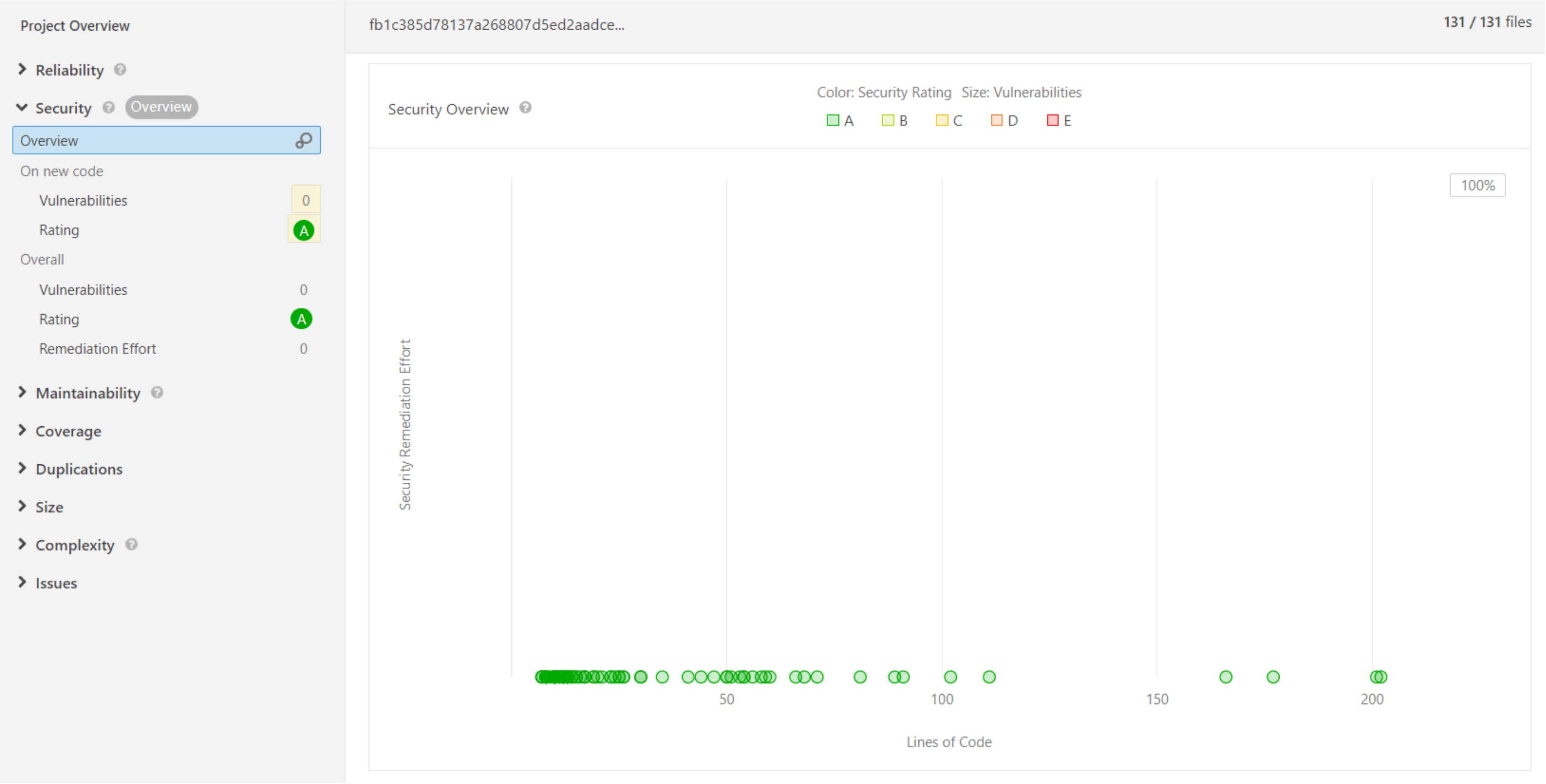
Debt 0

Debt Ratio 0.0%

Rating A

Effort to Reach A 0

Figure 31 - Summary of maintainability rating of SonarQube analysis results



Project Overview

- Reliability
- Security — Overview
  - Overview
  - On new code
    - Vulnerabilities 0
    - Rating A
  - Overall
    - Vulnerabilities 0
    - Rating A
    - Remediation Effort 0
- Maintainability
- Coverage
- Duplications
- Size
- Complexity
- Issues

fb1c385d78137a268807d5ed2aadce...

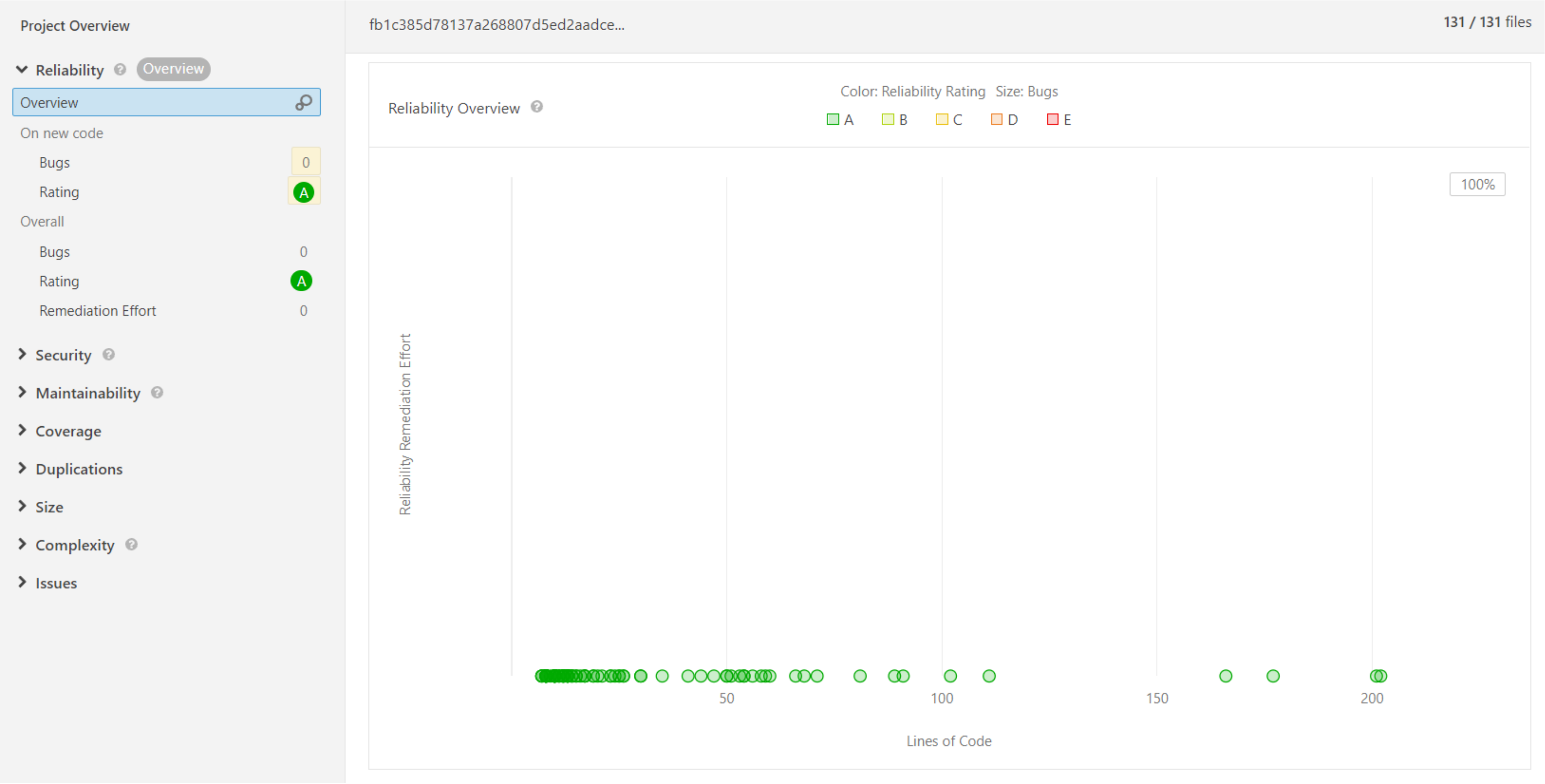
131 / 131 files

Security Overview

Color: Security Rating Size: Vulnerabilities

🟩 A 🟩 B 🟨 C 🟧 D 🟥 E

Figure 32 - Summary of security rating of SonarQube analysis results



Project Overview

fb1c385d78137a268807d5ed2aadce...

131 / 131 files

Reliability Overview

Overview

On new code

Bugs 0

Rating A

Overall

Bugs 0

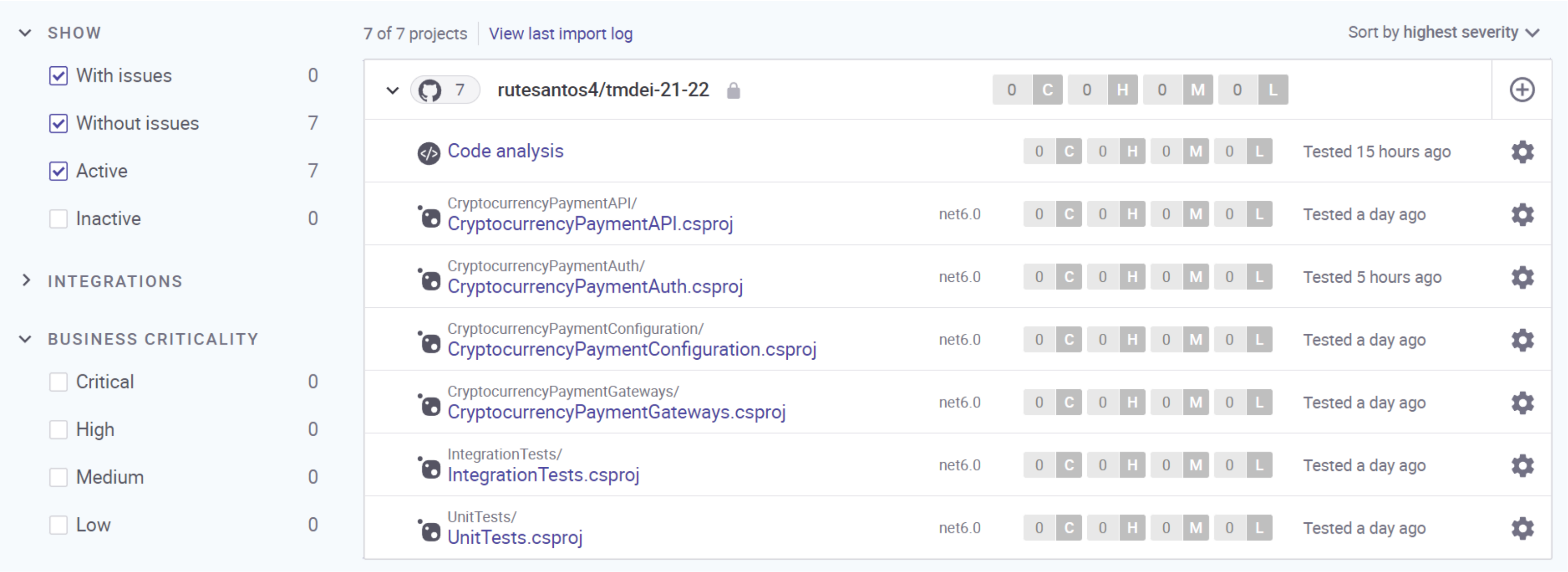
Rating A

Remediation Effort 0

Color: Reliability Rating Size: Bugs

Figure 33 - Summary of reliability rating of SonarQube analysis results

Snyk Analysis



SHOW

☑ With issues	0
☑ Without issues	7
☑ Active	7
□ Inactive	0

INTEGRATIONS

BUSINESS CRITICALITY

□ Critical	0
□ High	0
□ Medium	0
□ Low	0

7 of 7 projects | View last import log

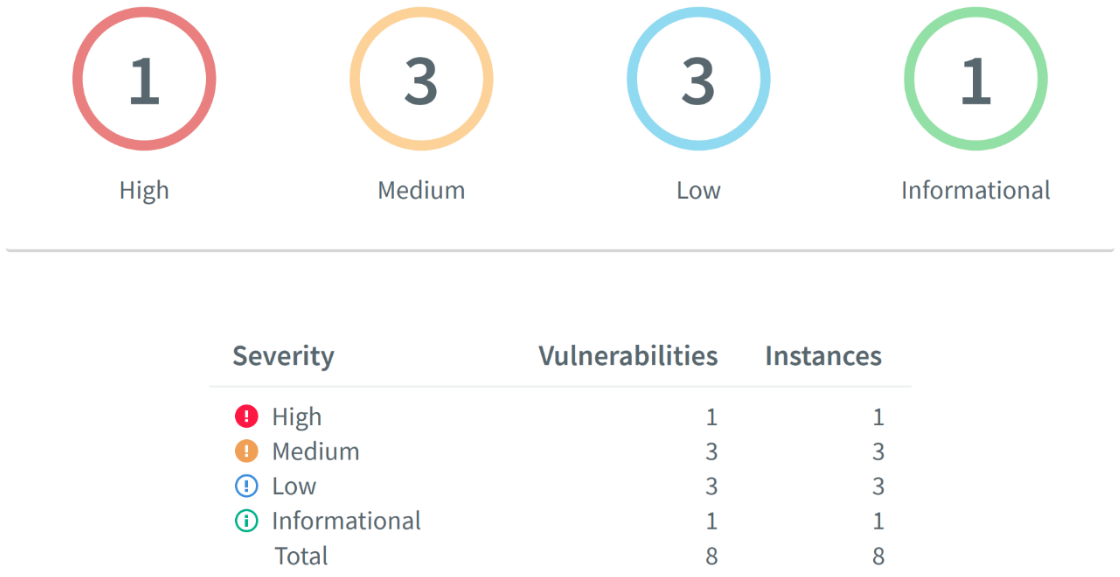
Sort by highest severity

rutesantos4/tmdei-21-22 (7) — 0 C 0 H 0 M 0 L

Project	Target	C	H	M	L	Tested
Code analysis		0	0	0	0	Tested 15 hours ago
CryptocurrencyPaymentAPI/ CryptocurrencyPaymentAPI.csproj	net6.0	0	0	0	0	Tested a day ago
CryptocurrencyPaymentAuth/ CryptocurrencyPaymentAuth.csproj	net6.0	0	0	0	0	Tested 5 hours ago
CryptocurrencyPaymentConfiguration/ CryptocurrencyPaymentConfiguration.csproj	net6.0	0	0	0	0	Tested a day ago
CryptocurrencyPaymentGateways/ CryptocurrencyPaymentGateways.csproj	net6.0	0	0	0	0	Tested a day ago
IntegrationTests/ IntegrationTests.csproj	net6.0	0	0	0	0	Tested a day ago
UnitTests/ UnitTests.csproj	net6.0	0	0	0	0	Tested a day ago

Figure 34 - Overview of Snyk analysis results

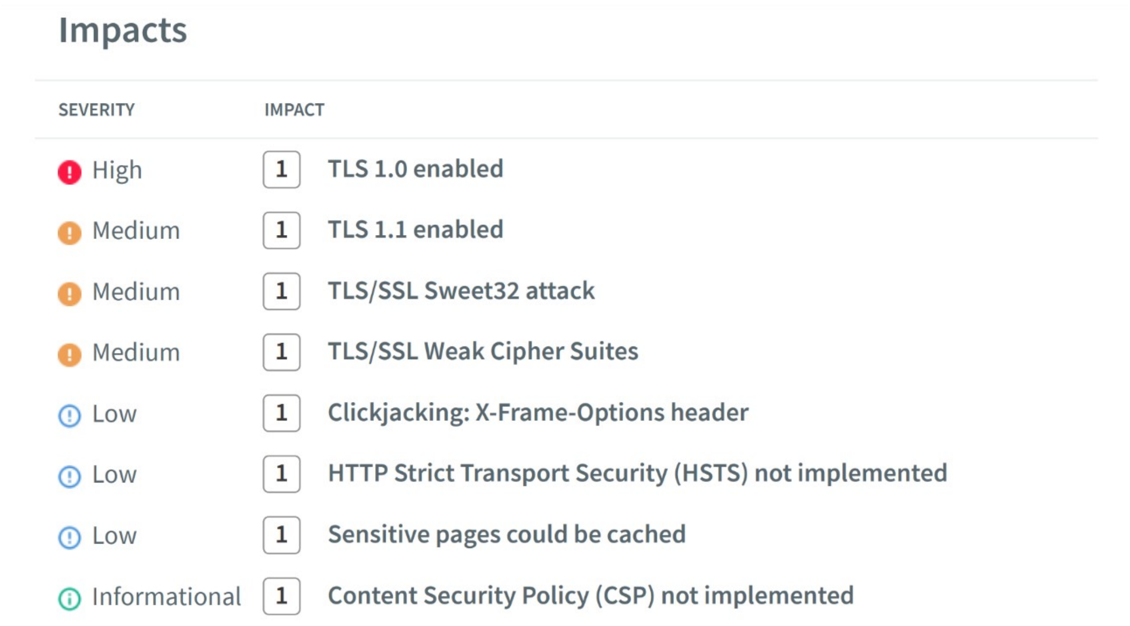
Acunetix Analysis



1	3	3	1
High	Medium	Low	Informational

Severity	Vulnerabilities	Instances
High	1	1
Medium	3	3
Low	3	3
Informational	1	1
Total	8	8

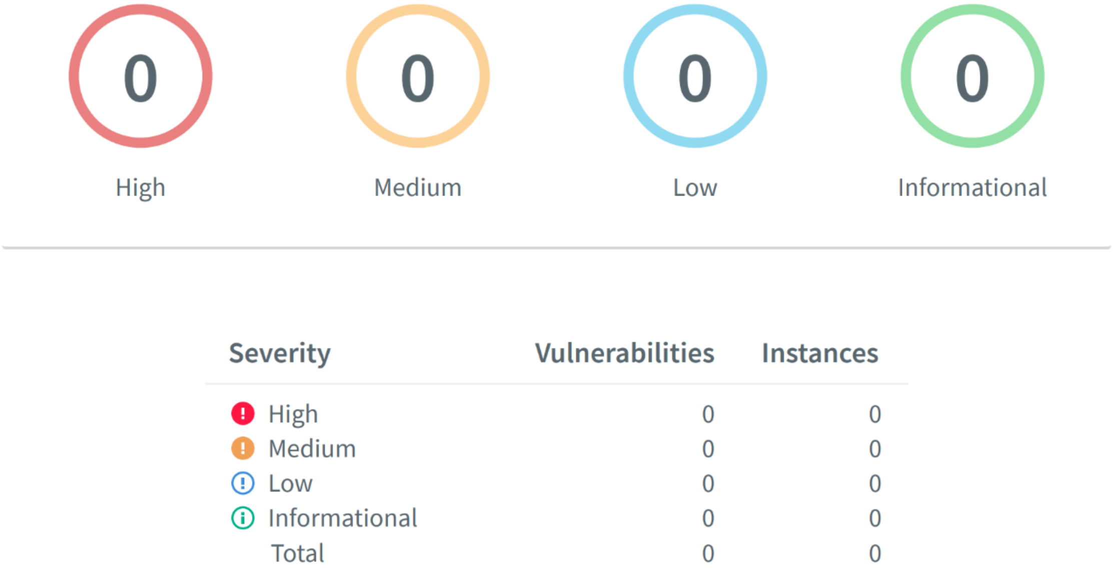
Figure 35 - Acunetix report with vulnerabilities



Impacts

SEVERITY	IMPACT	
High	1	TLS 1.0 enabled
Medium	1	TLS 1.1 enabled
Medium	1	TLS/SSL Sweet32 attack
Medium	1	TLS/SSL Weak Cipher Suites
Low	1	Clickjacking: X-Frame-Options header
Low	1	HTTP Strict Transport Security (HSTS) not implemented
Low	1	Sensitive pages could be cached
Informational	1	Content Security Policy (CSP) not implemented

Figure 36 - Acunetix report with vulnerabilities specification



0 High | 0 Medium | 0 Low | 0 Informational

Severity	Vulnerabilities	Instances
High	0	0
Medium	0	0
Low	0	0
Informational	0	0
Total	0	0

Figure 37 - Acunetix report without vulnerabilities