



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
TECHNICAL UNIVERSITY OF CRETE

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

ΤΙΤΛΟΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Η Τεχνολογία Blockchain Στην Εφοδιαστική Αλυσίδα

Επιμέλεια:

Κυψίδης Ιωάννης

Επιβλέπων καθηγητής:

Μαρινάκης Ιωάννης

Χανιά, 2023

ΠΕΡΙΕΧΟΜΕΝΑ

Πρόλογος.....	4
Abstract	5
Εισαγωγή	6
1. Αποσαφήνιση ορισμών	7
2. Η Εφοδιαστική Αλυσίδα	8
2.1 Η οργάνωση της Εφοδιαστικής Αλυσίδας	9
2.2 Η ιχνηλασιμότητα στην Εφοδιαστική Αλυσίδα	10
3. Η έννοια του Blockchain	11
3.1. Η εξέλιξη του Blockchain	11
3.2 Η λειτουργία του Blockchain	12
3.3 Η ανάλυση του «block/μπλοκ»	13
3.4. Οι τύποι και τα χαρακτηριστικά του Blockchain	13
3.5 Η ταξινόμηση του Blockchain	15
4. Οι χρήσεις του Blockchain	16
4.1 Μελλοντικές χρήσεις	18
5. Τα πλεονεκτήματα του Blockchain	18
6. Τα μειονεκτήματα του Blockchain	20
7. Εφαρμογή του Blockchain στην Εφοδιαστική Αλυσίδα	21
7.1 Πλεονεκτήματα χρήσης του Blockchain στην εφοδιαστική αλυσίδα	23
7.2 Μειονεκτήματα χρήσης του Blockchain στην εφοδιαστική αλυσίδα	25
7.3 Παραδείγματα εφαρμογών Blockchain στην εφοδιαστική αλυσίδα	25
8.Ανάπτυξη Αλγορίθμου	26
8.1 Πρόβλημα Τακτικής Κατανομής Θέσεων Ελλιμενισμού	26
8.2 Μαθηματική Μοντελοποίηση Προβλήματος	27
8.3 Λύση προβλήματος	29

8.4 Που χρησιμοποιείται η μέθοδος Simplex	29
8.5 Προγράμματα και εργαλεία που χρησιμοποιήθηκαν	30
8.6 Ανάλυση κώδικα	31
9.Αποτελέσματα	45
10.Συμπεράσματα	50
11. Κώδικας	51
Βιβλιογραφικές Αναφορές	57

Λέξεις-κλειδιά: Blockchain, Εφοδιαστική Αλυσίδα, Θετικά και αρνητικά χαρακτηριστικά, εφαρμογή

Πρόλογος

Στη σύγχρονη εποχή, η έννοια του Blockchain αναπτύσσεται με ταχύς ρυθμούς και η εφαρμογή της εδραιώνεται από διαφορετικούς τομείς ολοένα και περισσότερο. Ο πρωταρχικός τομέας εφαρμογής του Blockchain είναι κατά κύριο λόγο ο χρηματοοικονομικός τομέας, αλλά αυτό δεν σημαίνει ότι είναι και ο μόνος (Nofer, Gomber & Hinz, 2017). Ιδιαίτερη κατηγορία εφαρμογής του Blockchain, φαίνεται να είναι η εφοδιαστική αλυσίδα. Αν και κεντρίζουν σημαντικά το ενδιαφέρον αυτές οι δύο έννοιες, όπως αναφέρουν έρευνες δεν έχει υπάρξει επαρκή μελέτη που να αποδεικνύει τη συσχέτιση τους (Dujak & Sajter, 2018).

Η παρούσα εργασία θα ασχοληθεί με την προαναφερθείσα σχέση, δηλαδή αυτής του Blockchain με την εφοδιαστική αλυσίδα. Πιο αναλυτικά, αρχικά θα πραγματοποιηθεί η αποσαφήνιση των ορισμών «Blockchain» και «εφοδιαστική αλυσίδα», ενώ μετέπειτα θα παρουσιαστούν γενικά χαρακτηριστικά των δύο εννοιών. Επίσης, θα γίνει αναφορά στα πλεονεκτήματα και μειονεκτήματα που φέρει το Blockchain. Τέλος, θα γίνει ανάλυση της εφαρμογής του Blockchain στην εφοδιαστική αλυσίδα, ενώ δεν θα παραλειφθούν και τα συμπεράσματα.

Abstract

In modern times, the concept of Blockchain is developing rapidly and its application is being increasingly established by different sectors. The primary application area of Blockchain is mainly the financial sector, but this does not mean that it is the only one (Nofer, Gomber & Hinz, 2017). A special category of Blockchain application seems to be the supply chain. Although there is considerable interest in these two concepts, research has reported that there has not been sufficient research to prove their correlation (Dujak & Sajter, 2018).

This thesis will deal with the relationship of Blockchain with the supply chain. Moreover, the definitions of "Blockchain" and "supply chain" will first be clarified, and then general characteristics of the two concepts will be presented. It will also mention the advantages and disadvantages that Blockchain has. Finally, it will present an analysis of the application of Blockchain in the supply chain and the conclusions that derive from it.

Εισαγωγή

Στη σύγχρονη εποχή, οι συναλλαγές χρημάτων πραγματοποιούνται απευθείας από χρήστη σε χρήστη, σε αντίθεση με τα παλαιότερα χρόνια όπου έπρεπε να υπάρχουν πολλοί διαμεσολαβητές (Monrat, Schelen & Andersson, 2019). Το 2008, με την οικονομική κρίση παρουσιάστηκε η ανάγκη των επιχειρήσεων να έχουν ένα σταθερό νόμισμα το οποίο δεν θα επηρεάζεται από εξωτερικούς παράγοντες. Για αυτό το λόγο εμφανίστηκαν τα κρυπτονομίσματα και πιο συγκεκριμένα το Bitcoin το οποίο για να αντέξει στην αγορά, δημιουργήθηκε το Blockchain (Dujak & Sajter, 2018).

Το Blockchain, είναι μία τεχνολογική αλυσίδα μέσω της οποίας γίνονται ψηφιακές καταχωρήσεις. Όσο περισσότερο αυξάνονται οι καταχωρήσεις, τόσο περισσότερο μεγαλώνει η αλυσίδα. Ακόμα, το Blockchain δεν ταυτίζεται μόνο με τα κρυπτονομίσματα, αλλά εφαρμόζεται και σε άλλες κατηγορίες όπως στις χρηματοκοινωνικές υπηρεσίες, στη περίθαλψη, ακόμα και στην εφοδιαστική αλυσίδα (Monrat, Schelen & Andersson, 2019).

Η εφοδιαστική αλυσίδα, παρουσιάζει τη διαδρομή που ακολουθεί ένα προϊόν, από την αρχή μέχρι το τέλος. Δηλαδή, περιλαμβάνει όλη τη διαδικασία παραγωγής ενός προϊόντος, από τις πρώτες ύλες, μέχρι τον αγοραστή/καταναλωτή (Christopher, 2011).

Η σύνδεση του Blockchain με την εφοδιαστική αλυσίδα, άρχισε να κάνει την εμφάνισή της, σε σχέση με τους υπόλοιπους τομείς στους οποίους εφαρμόζεται το Blockchain. Αυτό οφείλεται στο γεγονός ότι η χρήση του Blockchain στην εφοδιαστική αλυσίδα, άρχισε να αναπτύσσεται και να δέχεται περισσότερους χρήστες/πελάτες, τα τελευταία χρόνια (Dujak & Sajter, 2018).

Τέλος, σημαντικό ρόλο στην αργοπορημένη συνεργασία του Blockchain με την εφοδιαστική αλυσίδα, παίζει η μη επαρκής διερεύνηση του πεδίου αυτού. Δηλαδή, γίνονται «πειράματα» για το πως μπορεί να υλοποιηθεί αυτή η σχέση μέχρι να φτάσει στο σημείο που είναι σήμερα (Dujak & Sajter, 2018).

Στη συνέχεια αυτή της εργασίας, θα παρουσιαστούν τα κεφάλαια που αφορούν τόσο την έννοια του Blockchain, όσο και της εφοδιαστικής αλυσίδας, ενώ δεν θα παραλειφθεί να γίνει αναφορά και στην σχέση που έχουν αυτές οι δύο έννοιες.

1. Αποσαφήνιση ορισμών

Η λέξη Blockchain αποτελείται από δύο συνθετικά, τη λέξη block και τη λέξη chain. Πρόκειται για μία αλυσίδα «chain», η οποία δημιουργείται και αυξάνεται μέσω των καταχωρήσεων «block» σε αυτή. Ουσιαστικά το Blockchain είναι ένα υπερσύγχρονο περιβάλλον μέσω του οποίου μπορούν να πραγματοποιηθούν χρηματικές συναλλαγές αξιόπιστα και με ασφαλή τρόπο (Dutta, Choi, Somani & Butala, 2020).

Το Blockchain μπορεί να θεωρηθεί ως ένα διαδικτυακό βιβλίο, όπου οι χρήστες μπορούν να υλοποιήσουν συναλλαγές με εύκολο και γρήγορο τρόπο, δίχως τη χρήση άδειας (Azzi, Chamoun & Sokhn, 2019).

Με τον όρο Εφοδιαστική Αλυσίδα, συχνά θεωρείται ως η διαχείριση της εφοδιαστικής αλυσίδας. Πιο αναλυτικά, πρόκειται για μια ομάδα επιχειρήσεων που βοηθούν στη διαδικασία παραγωγής ενός προϊόντος. Δηλαδή, τη διαδρομή που ακολουθεί το προϊόν, από τις πρώτες ύλες, στους παραγωγούς και από εκεί στην αγορά και μετέπειτα στον καταναλωτή (Mentzer, et.al., 2001).

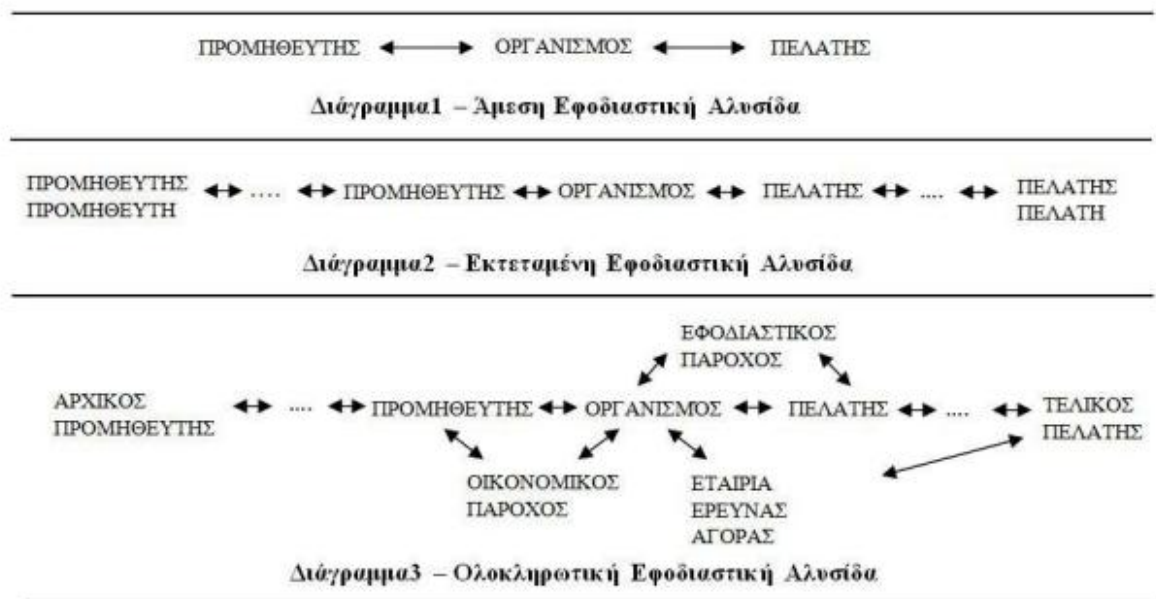
Παρόμοιο ορισμό για την εφοδιαστική αλυσίδα έχει δώσει και ο Christopher (2011), όπου αναφέρεται σε αυτή, ως το σύμπλεγμα των οργανισμών που στο σύνολό τους επιδιώκουν την υλοποίηση παραγωγής ενός προϊόντος. Ακόμη, είναι εκείνες οι δραστηριότητες μέσω των οποίων πραγματοποιείται και κάθε στάδιο παραγωγής προϊόντος. Τέτοιες δραστηριότητες μπορούν να θεωρηθούν η επεξεργασία των πρώτων υλών, ο σχεδιασμός του προϊόντος, ακόμα και η διανομή του προϊόντος στον πελάτη (Λιούμπαρη, 2020).

2. Η Εφοδιαστική Αλυσίδα

Όπως έχει προαναφερθεί, η Εφοδιαστική Αλυσίδα είναι το δίκτυο των επιχειρήσεων που συμβάλλουν στη διαδικασία παραγωγής ενός προϊόντος. Για την πλήρη κατανόηση αυτής της έννοιας, θα πρέπει να πραγματοποιηθεί ανάλυση των κατηγοριών της.

Αναλυτικότερα, υπάρχουν τρεις κατηγορίες Εφοδιαστικής Αλυσίδας: α) η *άμεση*, β) η *εκτεταμένη* και γ) η *τελική*. Η άμεση εφοδιαστική αλυσίδα, είναι αυτή όπου έρχεται σε απευθείας επαφή η εταιρία, ο προμηθευτής και ο πελάτης. Η εκτεταμένη εφοδιαστική αλυσίδα έχει να κάνει με την προσθήκη περισσότερων προμηθευτών και πελατών από την άμεση αλυσίδα, ενώ η τελική εφοδιαστική αλυσίδα, περιλαμβάνει όλα τα μέλη-επιχειρήσεις της διαδικασίας παραγωγής ενός προϊόντος (Mentzer, et.al., 2001).

Στο παρακάτω διάγραμμα απεικονίζονται οι προαναφερθείσες κατηγορίες της εφοδιαστικής αλυσίδας όπως αναφέρονται από τους Mentzer, et al., (2001) στο (Λιούμπαρη, 2020).



2.1 Η οργάνωση της Εφοδιαστικής Αλυσίδας

Η οργάνωση της εφοδιαστικής αλυσίδας, συσχετίζεται από πολλούς ερευνητές, ως η ορθή διαχείριση της αλυσίδας και η εφαρμογή των κατάλληλων στρατηγικών με αποτέλεσμα να φέρουν κέρδη στην εκάστοτε εταιρία. Ακόμα, σημαντικό χαρακτηριστικό στην καλύτερη οργάνωση της εφοδιαστικής αλυσίδας, αποτελούν οι σχέσεις που δημιουργούνται ανάμεσα στα μέλη της αλυσίδας (Lambert, 2014). Οι σχέσεις, ο προμηθευτής και ο πελάτης, αποτελούν τους τρεις βασικούς παράγοντες σωστής διαχείρισης της αλυσίδας (Munir, Jajja, Chatha, & Farooq, 2020).

Σύμφωνα με το Blanchard (2010), μία εταιρία ακολουθώντας κάποιες συμβουλές που αφορούν την εφοδιαστική αλυσίδα, μπορεί να επιφέρει κέρδη. Αυτές οι συμβουλές έχουν να κάνουν τόσο με την εσωτερική όσο και με την εξωτερική διαχείριση της εταιρίας, με τις λειτουργίες της, τις πωλήσεις και την εξυπηρέτηση.

Αυτό που πρέπει να αναφερθεί, είναι το γεγονός ότι υπάρχουν και σημαντικά σημεία σχετικά με την οργάνωση της εφοδιαστικής αλυσίδας. Πιο αναλυτικά, έχουν δημιουργηθεί οκτώ επιμέρους φάσεις της οργάνωσης, οι οποίες όπως φαίνονται στο διάγραμμα έχουν ως εξής (Lambert, 2014).

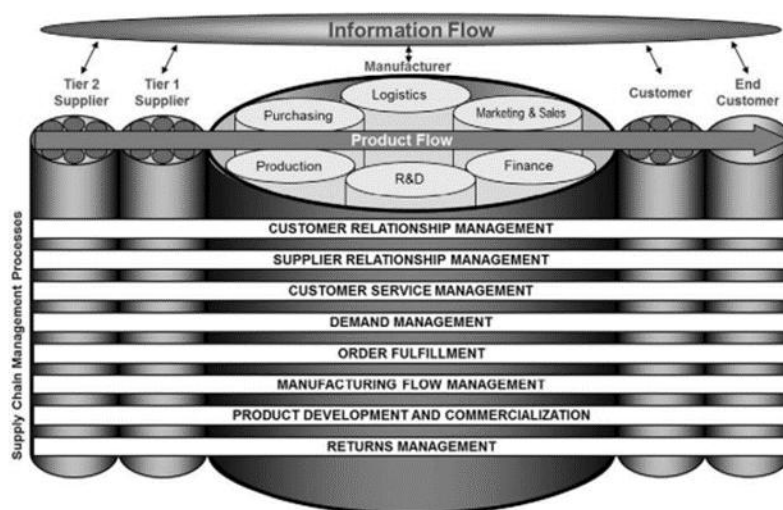
Αρχικά υπάρχει η φάση της οργάνωσης της εφοδιαστικής αλυσίδας που αφορά τη διαχείριση τόσο των πελατειακών σχέσεων, όσο και της εξυπηρέτησης πελατών. Η πρώτη σχέση σχετίζεται μεταξύ της εταιρίας παραγωγής και του καταναλωτή, ενώ η δεύτερη αποτελεί ένα κομμάτι της πρώτης, η οποία ασχολείται με το συγχρονισμό των υπηρεσιών (Lambert, 2014).

Επιπλέον, υπάρχουν οι φάσεις της διαχείρισης των απαιτήσεων και της κατασκευαστικής ροής. Η διαχείριση των απαιτήσεων αφορά την ενασχόληση των απαιτήσεων και των αναγκών που έχει ο κάθε οργανισμός, ενώ η διαχείριση κατασκευαστικής ροής σχετίζεται με ολόκληρη την διαδικασία παραγωγής ενός προϊόντος, όπως για παράδειγμα η διανομή και οι προμήθειες (Lambert & Enz, 2017).

Ακόμα υπάρχουν οι φάσεις που σχετίζονται με τις σχέσεις των προμηθευτών και της εκτέλεσης παραγγελιών. Το πρώτο αφορά, την ορθή επικοινωνία που θα πρέπει να έχουν οι προμηθευτές είτε μεταξύ τους, είτε με την εταιρία που συνεργάζονται, ενώ το δεύτερο

αναφέρεται στις στρατηγικές που είναι απαραίτητες για τη σύνδεση της εταιρίας με τον καταναλωτή (Lambert & Enz, 2017).

Τέλος, υπάρχουν οι φάσεις της *ανάπτυξης των προϊόντων* και της *διαχείρισης επιστροφών*, όπου η ανάπτυξη προϊόντων αναφέρεται κυρίως στην επεξεργασία του προϊόντος για την προώθησή του στην αγορά. Η διαχείριση επιστροφών, ασχολείται με τη σωστή οργάνωση και διαχείριση των επιστρεπτέων προϊόντων (Lambert & Enz, 2017).



2.2 Η ιχνηλασιμότητα στην Εφοδιαστική Αλυσίδα

Με την έννοια της ιχνηλασιμότητας, θεωρείται η δυνατότητα παρακολούθησης ενός προϊόντος, σε όλα τα στάδια παραγωγής του, από τις πρώτες ύλες μέχρι και την παράδοση του προϊόντος στον καταναλωτή (Anon, 2005). Η ιχνηλασιμότητα βοηθάει στην εντόπιση των προϊόντων και κυρίως των τροφίμων (Aung & Chang, 2014), δηλαδή με τον εντοπισμό, την τοποθεσία του προϊόντος και με την ποιότητα του προϊόντος (Zhang & Bhatt, 2014). Σημαντικό

είναι να ειπωθεί ότι δεν υπάρχει μόνο ένα τρόπος ιχνηλασιμότητας του προϊόντος/τροφίμου, αλλά έξι (Opata, 2003).

Πιο αναλυτικά, υπάρχει η ιχνηλασιμότητα του προϊόντος και της διαδικασίας. Η πρώτη αναφέρεται στην τοποθεσία/θέση του προϊόντος, ενώ η δεύτερη σχετίζεται με τους παράγοντες επιρροής του προϊόντος κατά την παραγωγή του (Zhang & Bhatt, 2014).

Ακόμα, υπάρχει η γενική ιχνηλασιμότητα και η ιχνηλασιμότητα εισροών, όπου η γενική ιχνηλασιμότητα αφορά πληροφορίες σχετικά με το προϊόν. Η ιχνηλασιμότητα εισροών, σχετίζεται με τα επιπρόσθετα οργανικά υλικά που εντοπίζονται στο προϊόν (Zhang & Bhatt, 2014).

Τέλος, δύο άλλοι τρόποι ιχνηλασιμότητας είναι αυτός των επιβλαβών οργανισμών και αυτή της μέτρησης. Η ιχνηλασιμότητα των επιβλαβών οργανισμών, έχει να κάνει με τα μικρόβια, τους ιούς και τα βακτήρια που μπορεί να περιέχουν τα προϊόντα, ενώ η ιχνηλασιμότητα της μέτρησης αφορά τη μέτρηση αποδεκτών προτύπων (Zhang & Bhatt, 2014).

3. Η έννοια του Blockchain

Όπως έχει προαναφερθεί το Blockchain είναι μία αλυσίδα, η οποία αυξάνεται και επεκτείνεται ανάλογα με τα «μπλοκ» που πραγματοποιούνται, δηλαδή τις καταχωρήσεις (Zheng, Xie, Dai, Chen & Wang, 2018).

3.1. Η εξέλιξη του Blockchain

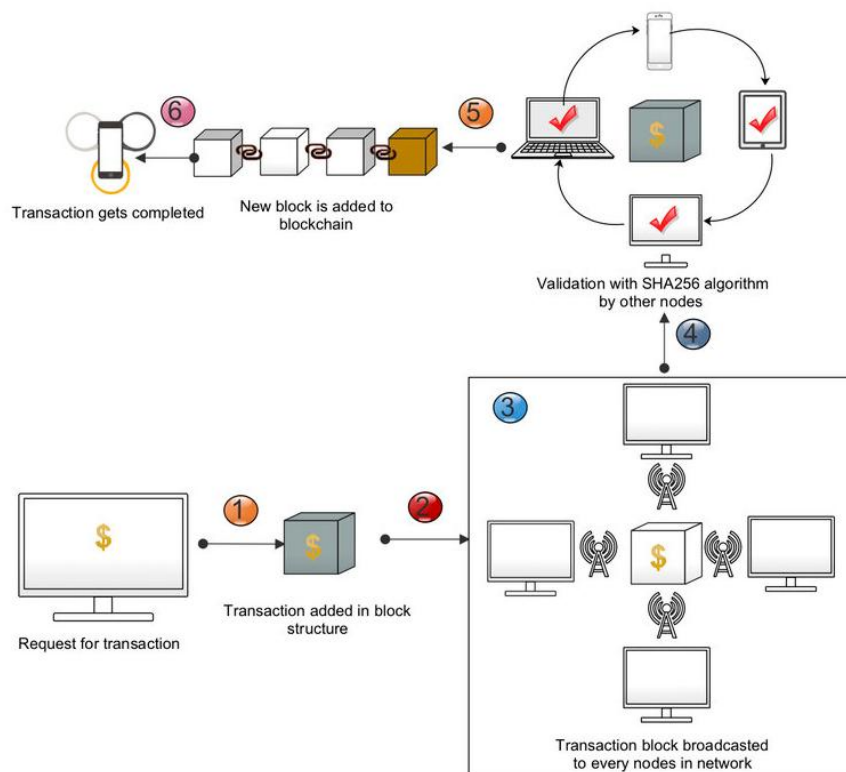
Το Blockchain, είχε πρωτοεμφανιστεί ως έννοια σε ένα βιβλίο του Satoshi Nakamoto, που αφορούσε τη χρήση του κρυπτονομίσματος Bitcoin (Nakamoto, 2008). Το Blockchain ταυτίστηκε με το Bitcoin, και άρχισε να έχει πολύ επιρροή, σε σημείο να αυξάνεται το

ενδιαφέρον των ερευνητών για περαιτέρω μελέτες και ως επέκταση για τη χρήση του Blockchain σε τομείς όπως η υγεία, η εφοδιαστική αλυσίδα, κ.λπ. (Underwood, 2016)

3.2 Η λειτουργία του Blockchain

Η λειτουργία του Blockchain αναλύεται μέσω κάποιων σταδίων. Το πρώτο στάδιο, αναφέρεται στο αίτημα του χρήστη για μία καταχώρηση. Δεύτερο στάδιο, είναι η τροποποίηση της καταχώρησης σε μορφή «μπλοκ» και ακολουθεί η αναζήτηση ενός ιδίου «μπλοκ» και η προσκόλληση σε αυτό. Μετέπειτα, υπάρχει το στάδιο της επικύρωσης των καταχωρήσεων μέσω αλγορίθμου. Κάθε νέο «μπλοκ» προστίθεται στο αρχείο, με σκοπό την επέκταση της αλυσίδας (Bodkhe, et.al., 2020).

Στη παρακάτω εικόνα παρουσιάζεται η βασική λειτουργία του Blockchain (Bodkhe, et.al., 2020).



3.3 Η ανάλυση του «block/μπλοκ»

Το «μπλοκ», δηλαδή οι καταχωρήσεις/συναλλαγές του Blockchain αποτελείται από δύο μέρη, το σώμα και την επικεφαλίδα. Το σώμα αποτελείται από δύο κατηγορίες, (1) τη μέτρηση συναλλαγών και (2) τις συναλλαγές. Η ποσότητα των συναλλαγών εξαρτάται από τη χωρητικότητα του εκάστοτε «μπλοκ» (Zheng, Xie, Dai, Chen & Wang, 2018).

Ακόμα, η επικεφαλίδα αποτελείται από περισσότερες κατηγορίες. Αρχικά, (1) είναι η έκδοση μπλοκ, όπου φανερώνει τους κανόνες που πρέπει να ακολουθήσει το εκάστοτε «μπλοκ». Έπειτα, υπάρχει (2) ο κατακερματισμός του μπλοκ και (3) ο κατακερματισμός του Merkle. Το πρώτο αναφέρεται στον κατακερματισμό των προηγούμενων μπλοκ, ενώ το δεύτερο στο κατακερματισμό των συναλλαγών. Οι τελευταίες κατηγορίες σχετίζονται με (4) τη χρονοφραγίδα, (5) το nBits και με (6) το Nonce (Zheng, Xie, Dai, Chen & Wang, 2018).

3.4. Οι τύποι και τα χαρακτηριστικά του Blockchain

Σημαντικό είναι να γίνει η διατύπωση των τύπων που έχει το Blockchain, καθώς δεν είναι ένα ενιαίο σώμα. Πιο αναλυτικά, υπάρχουν τρεις τύποι που κατέχει το Blockchain, α) ο τύπος χωρίς άδεια, β) ο τύπος με άδεια και γ) ο τύπος των ιδιωτικών αλυσίδων (Dujak & Sajter, 2018).

Ο πρώτος τύπος, αυτός της «χωρίς άδειας», αφορά τις καταχωρήσεις των μπλοκ που πραγματοποιούνται χωρίς έγκριση άδειας από άλλους χρήστες. Ο δεύτερος τύπος αναφέρεται σε μία κοινή απόφαση των χρηστών που θα πρέπει, το κάθε μέλος να δώσει την άδεια για να πραγματοποιηθεί. Τέλος, ο τρίτος σχετίζεται με δικαιώματα εγγραφής τα οποία έχουν μια διαδικασία αναγνώρισης ως Blockchain (Yassein, Shatnawi, Rawashdeh & Mardin, 2019).

Το Blockchain φέρει κάποια σημαντικά χαρακτηριστικά, τα οποία το αναδεικνύουν και το κάνουν να έχει φτάσει στο σημείο που βρίσκεται σήμερα. Αυτά τα χαρακτηριστικά παρουσιάζονται ως εξής:

- Αμετάβλητο, (Dutta, Choi, Somani & Butala, 2020).
- Διαφανές, (Dutta, Choi, Somani & Butala, 2020).
- Μη αναστρέψιμο, (Dutta, Choi, Somani & Butala, 2020).

- Αποκεντρωμένο, (Zheng, Xie, Dai, Chen & Wang, 2018).
- Αυτόνομο, (Zheng, Xie, Dai, Chen & Wang, 2018).
- Ανώνυμο, (Niranjanamurthy, Nithya & Jagannatha, 2018).
- Ανοιχτό σε κώδικες, (Niranjanamurthy, Nithya & Jagannatha, 2018).
- Προέλευση, (Dutta, Choi, Somani & Butala, 2020).
- Ιδιοκτησία και μοναδικότητα, (Dutta, Choi, Somani & Butala, 2020).

Πιο αναλυτικά, το Blockchain χαρακτηρίζεται ως αποκεντρωμένο, καθώς δίνει τη δυνατότητα στους χρήστες να παρακολουθούν και να ενημερώνονται για την πορεία των συναλλαγών από πολλά συστήματα. Επιπλέον χαρακτηριστικό είναι η αυτονομία και η ανωνυμία του Blockchain. Το πρώτο αναφέρεται στην ικανότητα των χρηστών να έχουν πρόσβαση στα μπλοκ, χωρίς παρεμβάσεις και το δεύτερο στην ανωνυμία που κατέχουν τα μπλοκ (Niranjanamurthy, Nithya & Jagannatha, 2018).

Ακόμα, το Blockchain θεωρείται διαφανές και μοναδικό, διότι οι καταχωρήσεις παραμένουν εντοπίσιμες στο περιβάλλον του διαδικτύου και τα μπλοκ φέρουν ένα δικό τους μοναδικό αριθμό καταχώρησης (Dutta, Choi, Somani & Butala, 2020).

Τέλος, υπάρχουν τα χαρακτηριστικά της μη αναστρεψιμότητας, δηλαδή η διατήρηση των συναλλαγών σε κάθε αρχείο και της ιδιοκτησίας, όπου σε κάθε καταχώρηση που πραγματοποιεί ο χρήστης στην αλυσίδα παρέχεται έγγραφο ιδιοκτησίας του (Dutta, Choi, Somani & Butala, 2020).

3.5 Η ταξινόμηση του Blockchain

Υπάρχουν τριών ειδών συστήματα ταξινόμησης του Blockchain, α) το ιδιωτικό, β) το δημόσιο και γ) το κοινό. Η ανάλυση των προαναφερόμενων πραγματοποιείται με βάση κάποιων κατηγοριών (Monrat, Schelen & Andersson, 2019).

Μια από αυτές τις κατηγορίες είναι η συναίνεση. Στο δημόσιο Blockchain η συναίνεση των μπλοκ γίνεται από όλους τους χρήστες, στο κοινό μερικά μέλη είναι υπεύθυνα για τη συναίνεση, ενώ στον ιδιωτικό υπάρχουν ανώτερα μέλη που επιλέγουν ποιος θα είναι υπεύθυνος για τη συναίνεση (Zheng, Xie, Dai, Chen & Wang, 2018).

Ακόμα υπάρχει η κατηγορία της άδεια ανάγνωσης, όπου στο δημόσιο Blockchain επιτρέπεται η ανάγνωση δεδομένων από όλους τους χρήστες, ενώ στο κοινό και στον ιδιωτικό είναι περιορισμένη η πρόσβαση. Μία ακόμα κατηγορία είναι αυτή του αμετάβλητου. Στο δημόσιο Blockchain, είναι σχεδόν αδύνατο να τροποποιηθούν οι συναλλαγές, γεγονός που δεν ισχύει στον κοινό και ιδιωτικό, εξαιτίας των ανώτερων μελών που μπορούν να αλλάξουν τα δεδομένα (Monrat, Schelen & Andersson, 2019).

Επιπρόσθετη κατηγορία στην ταξινόμηση των συστημάτων του Blockchain, είναι η αποδοτικότητα. Στον κοινό και στον ιδιωτικό Blockchain υπάρχει καλύτερη απόδοση από ότι στο δημόσιο. Τέλος, υπάρχει η κατηγορία της συγκεντρωτικότητας, η οποία ταυτίζεται με το ιδιωτικό Blockchain, καθώς επιλέγει τους χρήστες της και διοικείται από κάποια ανώτερα μέλη. Το δημόσιο είναι αποκεντρωμένο, καθώς διαθέτει πολλά ανεξάρτητα μέλη και το κοινό έχει ως ένα σημείο συγκεντρωτικότητας (Zheng, Xie, Dai, Chen & Wang, 2018).

4. Οι χρήσεις του Blockchain

Είναι πλέον γνωστό, πως το Blockchain έχει εδραιωθεί και σε άλλους τομείς εκτός του κρυπτονομίσματος (Fahmy, 2018). Όπως φαίνεται και στο παρακάτω διάγραμμα, το Blockchain εφαρμόζεται στην οικονομία, στην υγεία, καθώς και στην εφοδιαστική αλυσίδα, η οποία θα αναλυθεί περαιτέρω στο επόμενο κεφάλαιο (Bodkhe, et.al. 2016).



➤ Διαδίκτυο των Πραγμάτων (Internet of Things, IoT).

Στο Internet of Things (IoT), υπάρχουν οι συσκευές, οι οποίες μπορούν σε αρχική φάση να συνδεθούν στο διαδίκτυο αλλά και μεταξύ τους. Επίσης, μπορούν να παρέχουν δεδομένα όπως οι ενδείξεις της θερμοκρασίας και της υγρασίας ή τη τοποθεσία των πραγμάτων. Με την παροχή

(barcode) σε κάθε συσκευασία, πραγματοποιείται έλεγχος και ασφάλεια για τη σωστή ιχνηλασιμότητα (Wüst & Gervais, 2018).

➤ Smart contracts

Τα Smart contracts ανήκουν στις πιο σύγχρονες εφαρμογές του Blockchain, καθώς πρόκειται για τα ηλεκτρονικά συμβόλαια που μπορούν να υπογράψουν δύο ή περισσότεροι χρήστες. Με το που θα υπογραφεί το συμβόλαιο, εκτελείται αυτόματα η εντολή του (Gerard, 2017).

➤ Υγεία

Στον τομέα της υγείας, η χρήση του Blockchain μπορεί να βοηθήσει στην ιχνηλασιμότητα των φαρμάκων. Αναφορικά με τα φάρμακα, είναι πολύ σημαντική η ιχνηλασιμότητά τους, διότι υπάρχουν αρκετά προβλήματα με τα συστατικά τους και οδηγούν σε προβλήματα υγείας στους χρήστες (Chang & Chen, 2020).

➤ Οικονομία

Ένα από τους πρωταρχικούς και βασικούς τομείς αποτελεί η οικονομία, η οποία με τη βοήθεια του Blockchain, άρχισε να ανακάμπτει και να κάνουν εμφάνιση οι ηλεκτρονικές συναλλαγές. Το bitcoin, ως κρυπτονόμισμα επέφερε πολλές αλλαγές στην αξία και τη μεταφορά χρημάτων (Nakamoto, 2008).

➤ Ασφάλεια

Το Blockchain μπορεί να ενισχύσει και τον τομέα της ασφάλειας, καθώς διαδραματίζει βασικό ρόλο στην ασφάλεια των ηλεκτρονικών συναλλαγών (Chang & Chen, 2020).

➤ Επιχειρήσεις

Οι επιχειρήσεις έχουν αναβαθμιστεί με γοργούς ρυθμούς με τη βοήθεια του Blockchain, λόγω των χρηματικών υπηρεσιών. Οι συναλλαγές πλέον πραγματοποιούνται σε μεγάλο βαθμό ηλεκτρονικά και απαλλάσσουν τις επιχειρήσεις από ασύγχρονες διαδικασίες (Zheng, Xie, Dai, Chen & Wang, 2018).

4.1 Μελλοντικές χρήσεις

Το Blockchain, όπως έχει προαναφερθεί είναι ένα σύγχρονο και διευρυμένο μέσο σε πολλούς τομείς. Υποστηρίζεται ότι στα επόμενα χρόνια η χρήση του Blockchain θα επεκταθεί μέχρι και τις χώρες του τρίτου κόσμου, ενισχύοντάς τις στη λειτουργικότητα της δημόσιας διοίκησης (Atzori, 2015).

Η απόδειξη κτήσης γης ή πλούτου μέσω Blockchain, θα μπορέσει να βοηθήσει τον πληθυσμό με το πρόβλημα της απαλλοτρίωσης στις χώρες του τρίτου κόσμου. Αυτό το γεγονός όμως δεν φαίνεται να είναι τόσο ρεαλιστικό όσο παρουσιάζεται (Glaser, 2017).

Επιπλέον, πιστεύεται ότι τα κρυπτονομίσματα σε βάθος χρόνου, θα έχουν τη δυνατότητα να χρησιμοποιούνται ως κανονικά χρήματα. Μέχρι στιγμής τα κρυπτονομίσματα χρησιμοποιούνται ως κάποιο στοιχείο περιουσιακό και η αξία τους αυξομειώνεται (Luther & White, 2014).

Αυτό είναι πιθανό να αλλάξει και να χρησιμοποιούν οι άνθρωποι τα κρυπτονόμισμα για την πληρωμή των υλικών τους αγαθών, όπως για παράδειγμα τροφή, κατοικία, μεταφορικό μέσο (αμάξι), κ.λπ., (Luther & White, 2014). Ακόμα, θα μπορεί να πραγματοποιείται η εξόφληση χρεών ή η πληρωμή υπηρεσιών (Mishkin, 2004).

5. Τα πλεονεκτήματα του Blockchain

Το Blockchain ως ιδέα μπορεί να επιφέρει πολλά θετικά χαρακτηριστικά και να βοηθήσει με αρκετούς τρόπους τους οργανισμούς που ασχολούνται με την εφοδιαστική αλυσίδα. Αυτό δεν σημαίνει όμως ότι δεν υπάρχουν και κάποια αρνητικά στοιχεία που πρέπει να αναφερθούν.

Όσον αφορά τα θετικά χαρακτηριστικά του Blockchain, αρχικά υπάρχει η μη χρήση διαμεσολαβητή. Οι συναλλαγές πραγματοποιούνται χωρίς κάποιο κύριο διαχειριστή, αλλά αντί αυτού υπάρχουν προγράμματα συναίνεσης για να πραγματοποιούνται οι συναλλαγές ταυτόχρονα. Ακόμα ένα άλλο θετικό του Blockchain είναι οι δυνατότητες που έχει ο εκάστοτε χρήσης, τόσο στον έλεγχο όσο και στις πληροφορίες που έχει πρόσβαση (Niranjanamurthy, Nithya & Jagannatha, 2018).

Επιπλέον, το Blockchain παρέχει στους χρήστες ακεραιότητα και αξιοπιστία. Η ακεραιότητα συμβάλει στην εμπιστοσύνη που μπορούν να έχουν οι χρήστες του Blockchain στις διαδικασίες των συναλλαγών, καθώς πραγματοποιούνται με βάση κανόνων. Η αξιοπιστία συνεισφέρει στην αυθεντικότητα της μακροζωίας (Yassein, Shatnawi, Rawashdeh & Mardin, 2019).

Επίσης σημαντικό πλεονέκτημα του Blockchain είναι η σταθερότητα και η ταξινόμηση που παρέχει. Οι καταχωρήσεις γίνονται με τέτοιο τρόπο που είναι σχεδόν απίθανο να τροποποιηθούν ή να απομακρυνθούν, καθώς και τα δεδομένα είναι κατανεμημένα έτσι ώστε να μπορούν να αποφύγουν τυχόν βλάβες (Buxbaum, 2017).

Ακόμα, βασικό θετικό στοιχείο του Blockchain αποτελεί η έλλειψη της λογοκρισίας και η δυνατότητα επαλήθευσης δεδομένων. Δηλαδή, να μπορούν οι χρήστες να εξετάζουν κατά το πόσο είναι ορθά τα στοιχεία των συναλλαγών μέσω της επαλήθευσης. Επίσης, δεν πρέπει να παραληφθεί το γεγονός ότι το περιβάλλον του Blockchain έχει ελεύθερη πρόσβαση προς όλους τους χρήστες που θέλουν να συμμετέχουν (Erturk, Lopez & Yu, 2020).

Επιπρόσθετα, θετικά χαρακτηριστικά του Blockchain αποτελούν η ποιότητα των δεδομένων, η οποία είναι εξαιρετικά υψηλή, καθώς και οι γρήγορες συναλλαγές, οι οποίες πραγματοποιούνται μέσα σε μόνο λίγα λεπτά, ενώ σε άλλες περιπτώσεις χρειάζονται μέρες. Σημαντικό πλεονέκτημα θεωρείται και το χαμηλό κόστος της συναλλαγής σε συνδυασμό με την ποιότητα και την ταχύτητα (Niranjanamurthy, Nithya & Jagannatha, 2018).

6. Τα μειονεκτήματα του Blockchain

Όπως προαναφέρθηκαν τα πλεονεκτήματα, υπάρχουν και τα αρνητικά χαρακτηριστικά που φέρει το Blockchain. Βασικό μειονέκτημα αποτελεί η διαδικασία της συναλλαγής, καθώς οι βάσεις δεδομένων του Blockchain έχουν πιο αργούς ρυθμούς. Αυτό προκαλείται κυρίως από την επαλήθευση υπογραφής, τους μηχανισμούς συναίνεσης και της επεξεργασίας συναλλαγών (Yassein, Shatnawi, Rawashdeh & Mardin, 2019).

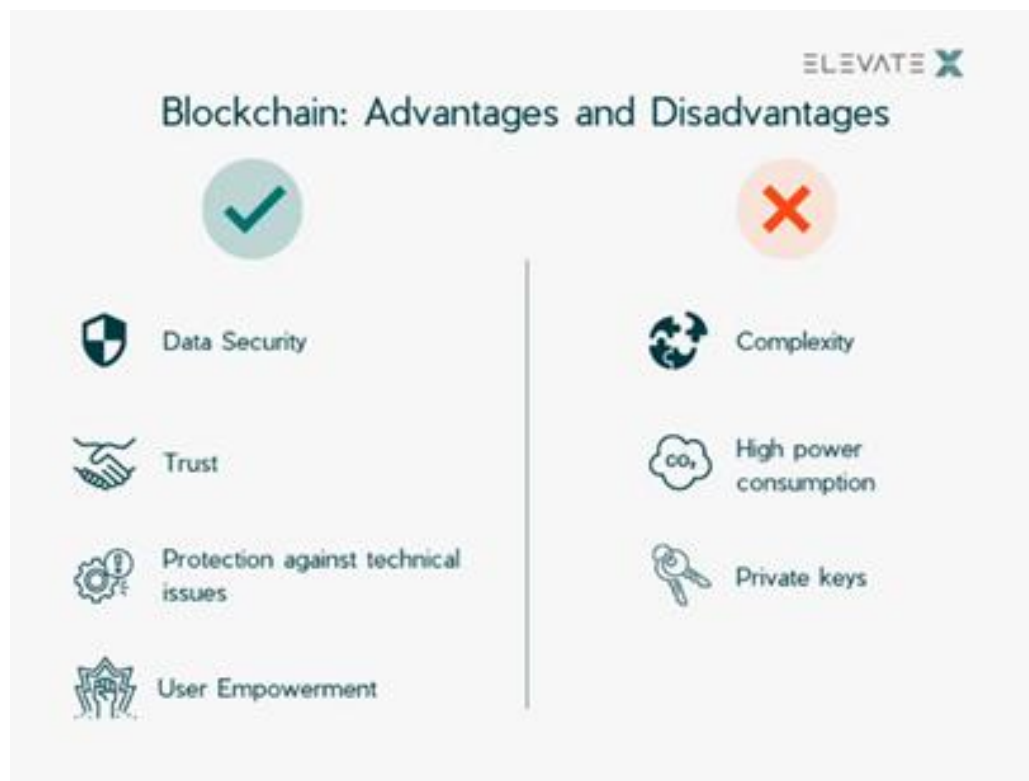
Επιπλέον, ένα άλλο αρνητικό στοιχείο του Blockchain είναι η ασφάλεια της περιήγησης, η οποία είναι μεν σε ένα αρκετά υψηλό επίπεδο, αλλά πάντα θα υπάρχουν κίνδυνοι που δεν θα μπορέσουν να αντιμετωπιστούν. Υπολογίζεται ότι ένα μεγάλο ποσοστό των κινδύνων του Blockchain προέρχεται από επιθέσεις του διαδικτύου για παραβίαση των συναλλαγών (Erturk, Lopez & Yu, 2020).

Ακόμα, σημαντικά μειονεκτήματα του Blockchain είναι το κόστος, δηλαδή η υψηλή χρήση ενέργειας που απαιτείται για τις συναλλαγές, καθώς και το μικρό χρονικό διάστημα που έχει από την ίδρυσή του. Χρειάζεται χρόνο έτσι ώστε το Blockchain να αρχίσει να έχει ολοένα και περισσότερους χρήστες που θα δείχνουν εμπιστοσύνη (Niranjanamurthy, Nithya & Jagannatha, 2018).

Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, η σταθερότητα θεωρείται ως θετικό στοιχείο του Blockchain, αυτό όμως δεν είναι πάντα δεδομένο. Η αλλαγή δεδομένων είναι αρκετά δύσκολη και συχνά απαιτητική και σε πολλές περιπτώσεις εγκαταλείπεται. Επίσης, η χρήση «ιδιωτικών κλειδιών» μπορεί να επιφέρει δυσκολίες σε περιπτώσεις απουσίας. Πιο αναλυτικά, αν χαθεί κάποιο ιδιωτικό κλειδί χάνονται αυτόματα και οι καταχωρήσεις των χρηστών (Erturk, Lopez & Yu, 2020).

Τέλος, οι καταχωρήσεις που μπορούν να πραγματοποιηθούν μέσω του Blockchain είναι περιορισμένου μεγέθους και δεν μπορεί να γίνει κλιμάκωση, όσον αφορά την αποθήκευση πληροφοριών. Το παραπάνω αποτελεί και ένα από τα σημαντικότερα μειονεκτήματα του Blockchain (Niranjanamurthy, Nithya & Jagannatha, 2018).

Στο παρακάτω σχεδιάγραμμα παρουσιάζονται συνοπτικά κάποια από τα πλεονεκτήματα και μειονεκτήματα που έχει το Blockchain (ElevateX, 2021).

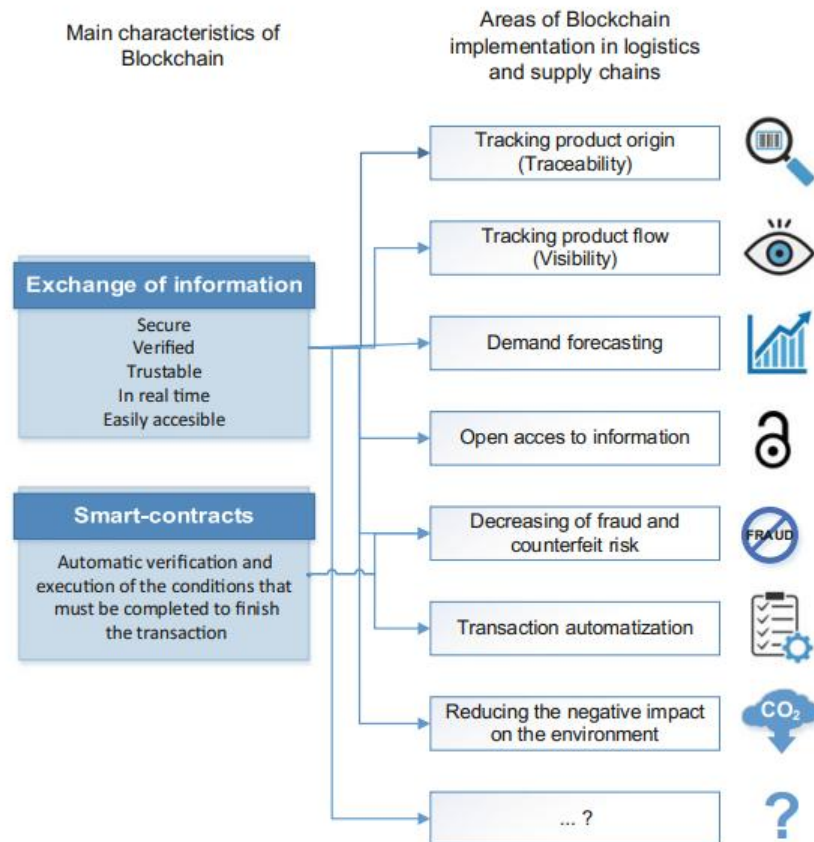


7. Εφαρμογή του Blockchain στην Εφοδιαστική Αλυσίδα

Η εφαρμογή του Blockchain στην εφοδιαστική αλυσίδα είναι ένας αρκετά σύγχρονος συνδυασμός που άρχισε να αναπτύσσεται τα τελευταία χρόνια με μεγάλη επιτυχία. Μέχρι και σήμερα η συγκεκριμένη σχέση βρίσκεται υπό ερευνητική κατάσταση, καθώς υλοποιούνται ακόμα μελέτες ανάλυσης (Tian, 2016).

Δύο είναι τα βασικά στοιχεία του Blockchain, που αποτελούν σημαντικούς παράγοντες για την εφαρμογή του στην εφοδιαστική αλυσίδα: α) οι ασφαλείς συναλλαγές μέσω Blockchain και β) οι αυτόματες και επαληθεύσιμες συναλλαγές (Dujak & Sajter, 2018).

Σύμφωνα με τους παραπάνω παράγοντες, αναπτύσσονται περαιτέρω περιοχές εφαρμογής του Blockchain στην εφοδιαστική αλυσίδα. Αυτές οι περιοχές αναφέρονται στο παρακάτω διάγραμμα (Dujak & Sajter, 2018).



Αρχικά υπάρχει η περιοχή που παρακολουθεί την προέλευση του προϊόντος και η - περιοχή που παρακολουθεί γενικά τη διαδρομή του προϊόντος. Επίσης υπάρχει η περιοχή της ανάλυσης της ζήτησης ή μη και της παροχής πληροφοριών για τις συναλλαγές των προϊόντων. Επιπλέον τομείς αποτελούν η αυτοματοποίηση των συναλλαγών και η μείωση των περιβαλλοντικών επιπτώσεων (Dujak & Sajter, 2018).

Πιο αναλυτικά, η πιο γνωστή εφαρμογή του Blockchain στην εφοδιαστική αλυσίδα αποτελεί η ιχνηλασιμότητα του προϊόντος, δηλαδή η ανάχνευση του προϊόντος καθ' όλη τη διαδικασία παραγωγής του με την παροχή βάσιμων πληροφοριών τοποθεσίας του (Dujak & Sajter, 2018).

Επιπλέον, η ζήτηση αναφέρεται ως βασικός τομέας του Blockchain στην εφοδιαστική αλυσίδα, καθώς αναφέρεται στα στοιχεία που υπάρχουν για τη ζήτηση και το κόστος του εκάστοτε προϊόντος. Συμβάλλει, δηλαδή, άμεσα στην αύξηση ή μείωση των κερδών (Dujak & Sajter, 2018).

Ακόμα, ο τομέας της ανοιχτής πρόσβασης των χρηστών στις αλυσίδες του Blockchain στην εφοδιαστική αλυσίδα, μπορεί να επιφέρει πολλά θετικά χαρακτηριστικά μέσω της διευκόλυνσης των γραφειοκρατικών θεμάτων (Dujak & Sajter, 2018).

7.1 Πλεονεκτήματα χρήσης του Blockchain στην εφοδιαστική αλυσίδα

Υπάρχουν αρκετά πλεονεκτήματα της χρήσης του Blockchain στην εφοδιαστική αλυσίδα, τα οποία παρουσιάζονται ως εξής:

Μέσω της τεχνολογίας του Blockchain, πραγματοποιείται εξοικονόμηση ενέργειας, χρόνου αλλά και κόστους σε σχέση με τους παραδοσιακούς τρόπους των χειροκίνητων εργασιών (Deloitte, 2017). Επίσης, με την εφαρμογή της τεχνολογίας του Blockchain ενισχύεται η ασφάλεια και η τήρηση των κανόνων στον περιβάλλον εργασίας της εφοδιαστικής αλυσίδας.

Επιπλέον, η μη αποτελεσματική επικοινωνία επηρεάζει τη μη ικανοποιητική εργασία. Με την εφαρμογή του Blockchain, τα παραπάνω μειονεκτήματα αρχίζουν να γίνονται πλεονεκτήματα (Petersson & Baur, 2018). Σημαντικό χαρακτηριστικό της χρήσης του του Blockchain στην εφοδιαστική αλυσίδα, αποτελούν οι συναλλαγές οι οποίες πλέον αρχίζουν να γίνονται πιο αποτελεσματικές, οικονομικές και ασφαλείς (Dutta, Choi, Somani & Butala, 2020).

Περισσότερα πλεονεκτήματα θεωρούνται η διαφάνεια και η βελτίωση χρόνου. Η διαφάνεια, ενισχύει την παρακολούθηση του προϊόντος και διακρίνει τα ιεραρχικά δικαιώματα της εταιρίας. Η εφοδιαστική αλυσίδα με τη χρήση του Blockchain, παρουσιάζει βελτιωμένο χρόνο παράδοσης σε σύγκριση με τους παραδοσιακούς τρόπους (Dutta, Choi, Somani & Butala, 2020).

Στην παρακάτω εικόνα παρουσιάζονται τα κύρια πλεονεκτήματα της εφαρμογής του Blockchain στην εφοδιαστική αλυσίδα (Sofocle, 2018).



7.2 Μειονεκτήματα χρήσης του Blockchain στην εφοδιαστική αλυσίδα

Σε αντίθεση με τα πλεονεκτήματα της χρήσης του Blockchain στην εφοδιαστική αλυσίδα, τα μειονεκτήματα είναι ελάχιστα. Το πρώτο αφορά την υπερβολική χρήση της τεχνολογίας. Οι ταχύτητες, εξαιτίας της μέγιστης χρήσης υπολογιστών μειώνονται και το κόστος αυξάνεται, ενώ παράλληλα δημιουργούνται και περιβαλλοντικά προβλήματα (Taylor, 2017).

Το δεύτερο μειονέκτημα σχετίζεται με τη πρόσβαση στα δεδομένα των συναλλαγών. Δηλαδή, υπάρχει πρόβλημα σε ποιο χρήστη που πραγματοποιεί συναλλαγές, θα πρέπει να του ανατεθεί η πλήρη πρόσβαση στα δεδομένα (Taylor, 2017).

7.3 Παραδείγματα εφαρμογών Blockchain στην εφοδιαστική αλυσίδα

Σε αυτό το υποκεφάλαιο, θα παρουσιαστούν κάποια παραδείγματα στα οποία εφαρμόζεται το Blockchain στην εφοδιαστική αλυσίδα. Αυτά τα παραδείγματα έχουν να κάνουν με τη ναυτιλία, τους λιμένες, και τις μεταφορές.

Αρχικά, όσον αφορά τη ναυτιλία το πλήθος των εγγράφων και των μεταφορών που χρησιμοποιείται από αυτή μπορούν να διευκολυνθούν με τη χρήση του Blockchain, καθώς θα υπάρξει απαλλαγή τόσο γραφειοκρατικών θεμάτων όσο και περιττών διαμεσολαβητών (Gregorio & Nustad, 2017). Σχετικά με τους λιμένες, οι οποίοι αποτελούν το διαμεσολαβητή στην μεταφορά φορτίων, η χρήση της τεχνολογίας του Blockchain, βοηθάει στη βελτίωση των επιχειρησιακών δραστηριοτήτων.

Τέλος, αναφορικά με τις μεταφορές, η χρήση του Blockchain για την εντόπιση των προϊόντων αποτελεί βασικό κομμάτι των συναλλαγών. Η ασφαλή μεταφορά δεδομένων συμβάλλει στην εξάλειψη των προβλημάτων στην εφοδιαστική αλυσίδα (Yuan & Wong, 2016).

8.Ανάπτυξη Αλγορίθμου

8.1 Πρόβλημα Τακτικής Κατανομής Θέσεων Ελλιμενισμού

Το πρόβλημα τακτικής κατανομής θέσεων ελλιμενισμού (Tactical Berth Allocation Problem (TBAP)) είναι ένα πρόβλημα που χρησιμοποιείται στον τομέα της ναυτιλίας και της διαχείρισης λιμανιών. Ο στόχος του είναι λαμβάνοντας υπόψη σε δεδομένες παραμέτρους και συνθήκες που σχετίζονται με τα πλοία και τις προβλήτες σε ένα λιμάνι, να αποφασίσει την βέλτιστη κατανομή θέσεων προβλήτας για την αποφόρτωση και φόρτωση των πλοίων. Ας αναλύσουμε τη διαδικασία του αλγορίθμου που αναπτύχθηκε για την επίλυση του συγκεκριμένου προβλήματος:

Αρχικά, εισάγουμε τις απαραίτητες βιβλιοθήκες και διαβάζουμε τα δεδομένα από ένα αρχείο Excel (data.xlsx). Τα δεδομένα περιέχουν πληροφορίες για τον αριθμό των προβλητών, τον αριθμό των πλοίων, τη χωρητικότητα των προβλητών, το κόστος χρήσης της κάθε προβλήτας, τη χωρητικότητα των πλοίων, το ρυθμό άφιξης των πλοίων και την απαίτηση των πλοίων.

Δημιουργούμε ένα πρόβλημα μικτού ακεραίου προγραμματισμού (MIP Problem). Ορίζουμε τις μεταβλητές απόφασης του προβλήματος. οι μεταβλητές αυτές καθορίζουν αν ένα πλοίο θα ανατεθεί σε μία προβλήτα. Η τιμή 1 σημαίνει ότι το πλοίο ανατίθεται στην προβλήτα, ενώ η τιμή 0 σημαίνει ότι δεν ανατίθεται. Ορίζουμε την συνάρτηση στόχου του προβλήματος. Στόχος μας είναι να μεγιστοποιήσουμε τη χρησιμοποίηση των προβλητών. Αυτό επιτυγχάνεται με την προσθήκη των προϊόντων, του ρυθμού άφιξης των πλοίων, της χωρητικότητας των πλοίων και των μεταβλητών απόφασης στη συνάρτηση στόχου.

Ορίζουμε τους περιορισμούς του προβλήματος. Ακολουθούν οι εξής περιορισμοί:

- Κάθε πλοίο πρέπει να ανατίθεται σε μόνο μία προβλήτα.

- Τα πλοία που απαιτούν ψύξη πρέπει να ανατίθενται σε προβλήτες που διαθέτουν συστήματα ψύξης.
- Κάθε προβλήτα μπορεί να ανατεθεί το πολύ σε ένα πλοίο.
- Το άθροισμα των χωρητικοτήτων των πλοίων που ανατίθενται σε έναν λιμένα/προβλήτα δεν πρέπει να υπερβαίνει την συνολική χωρητικότητα του.

Ο αριθμός των προβλητών που χρησιμοποιούνται δεν πρέπει να υπερβαίνει τον αριθμό των διαθέσιμων. Προσθέτουμε έναν περιορισμό για τον συνολικό κόστος των προβλητών. Αυτός ο περιορισμός εξασφαλίζει ότι το συνολικό κόστος της χρήσης των προβλητών δεν υπερβαίνει τον προϋπολογισμό που ορίζεται στη μεταβλητή "budget".

8.2 Μαθηματική Μοντελοποίηση Προβλήματος

Η αποτελεσματική διαχείριση του χώρου σε ένα λιμάνι αποτελεί κρίσιμο παράγοντα για τη βελτίωση της λειτουργίας του. Στην παρούσα εργασία, εξετάζεται ένα πρόβλημα βελτιστοποίησης της ανάθεσης πλοίων σε προβλήτες με σκοπό τη μεγιστοποίηση της συνολικής απόδοσης του λιμανιού. Η βελτιστοποίηση αυτή πραγματοποιείται με τη χρήση μεθόδων μικτού ακεραίου προγραμματισμού.

Αντικειμενική Συνάρτηση

Μεγιστοποίηση της συνολικής απόδοσης του λιμανιού:

$$\max Z = \sum_{i=1}^N \sum_{j=1}^M \text{ship_rate}_i \cdot \text{ship_capacity}_i \cdot x_{ij}$$

Όπου:

- N: Αριθμός πλοίων
- M: Αριθμός προβλητών
- x_{ij} : Μεταβλητή απόφασης, 1 αν το πλοίο i ανατίθεται στην προβλήτα j , 0 διαφορετικά

- y_{ij} : Μεταβλητή απόφασης, 1 αν η προβλήτα j χρησιμοποιείται, 0 διαφορετικά

Περιορισμοί

Κάθε πλοίο ανατίθεται σε μία ακριβώς προβλήτα:

$$\sum_{j=1}^M x_{ij} = 1, \forall i$$

Αν το εμπόρευμα ενός πλοίου απαιτεί ψύξη, τότε ανατίθεται σε προβλήτα με ικανότητα ψύξης του εμπορεύματος:

$$x_{ij} \leq y_j, \forall i, \forall j \text{ ship_requirement}_i = 'refrigerated'$$

Μία προβλήτα μπορεί να εξυπηρετεί το πολύ ένα πλοίο:

$$\sum_{i=1}^N x_{ij} \leq 1, \forall j$$

Η συνολική χωρητικότητα των πλοίων που ανατίθεται σε κάθε προβλήτα δεν πρέπει να υπερβαίνει την χωρητικότητα της προβλήτας:

$$\sum_{i=1}^N \text{ship_capacity}_i \cdot x_{ij} \leq y_j \cdot \text{berth_capacity}_j, \forall j$$

Ο συνολικός αριθμός των προβλητών που χρησιμοποιούνται δεν πρέπει να υπερβαίνει τον συνολικό αριθμό των προβλητών:

$$\sum_{j=1}^M y_j \leq M$$

Το συνολικό κόστος δεν πρέπει να υπερβαίνει τον προϋπολογισμό:

$$\sum_{j=1}^M \text{berth_cost}_j \cdot y_j \leq \text{budget}$$

8.3 Λύση προβλήματος

Λύνουμε το πρόβλημα χρησιμοποιώντας τον αλγόριθμο που περιγράφεται στην συνέχεια. Εκτυπώνουμε την κατάσταση του προβλήματος βελτιστοποίησης (επιτυχής ή μη) και την βέλτιστη τιμή της συνάρτησης στόχου. Εκτυπώνουμε τις τιμές των μεταβλητών απόφασης που είναι ίσες με 1, δηλαδή τη βέλτιστη ανάθεση των πλοίων στις προβλήτες. Δημιουργούμε ένα πίνακα δεδομένων pandas με τα αποτελέσματα της βελτιστοποίησης. Αποθηκεύουμε τον πίνακα δεδομένων σε ένα αρχείο Excel. Στη εφαρμογή που μελετάμε το αρχείο αυτό είναι το PulpOutput.xlsx.

Αυτή είναι η γενική διαδικασία του αλγορίθμου που παρουσιάζεται στον κώδικα (βλ.Σελ.53) Εφαρμόζοντας αυτόν τον αλγόριθμο, μπορούμε να βρούμε τη βέλτιστη ανάθεση των πλοίων στους λιμένες που θα μεγιστοποιήσει την χρησιμοποίηση των προβλητών, με την τήρηση των περιορισμών που έχουν οριστεί, όπως η ανάθεση ενός πλοίου σε μόνο έναν χώρο, η ανάθεση πλοίων που απαιτούν ψύξη σε χώρους με ψύξη, η μη υπέρβαση των χωρητικότητων των χώρων και άλλοι περιορισμοί που προκύπτουν από τα δεδομένα.

Οι τελικές απαντήσεις του προβλήματος βελτιστοποίησης περιέχονται στο αρχείο Excel με το όνομα "Pulp Output.xlsx". Αυτό το αρχείο περιέχει τα αποτελέσματα των μεταβλητών απόφασης καθώς και τη βέλτιστη τιμή της αντικειμενικής συνάρτησης.

8.4 Που χρησιμοποιείται η μέθοδος επίλυσης

Η μέθοδος που χρησιμοποιείται για την επίλυση του προβλήματος ακέραιου προγραμματισμού που διατυπώθηκε προηγούμενως παρουσιάζεται αναλυτικά στη συνέχεια. Πιο συγκεκριμένα, η συνάρτηση prob.solve() χρησιμοποιείται για να επιλύσει το πρόβλημα βελτιστοποίησης που ορίζεται από τα δεδομένα και τους περιορισμούς που έχουν οριστεί.

8.5 Προγράμματα και εργαλεία που χρησιμοποιήθηκαν

Στην ανάπτυξη του αλγορίθμου για την εφαρμογή του blockchain στην εφοδιαστική αλυσίδα, χρησιμοποιήθηκε ένα ευρύ φάσμα εργαλείων και τεχνολογιών που συνδυάζονται για την επίτευξη των στόχων μας.

Εργαλεία και Τεχνολογίες:

1. Visual Studio και Python: Το Visual Studio, ένα ισχυρό περιβάλλον ανάπτυξης, επέτρεψε τη δημιουργία και τον προγραμματισμό του αλγορίθμου μας στη γλώσσα προγραμματισμού Python. Αυτή η επιλογή έγινε για την ευκολία της σύνταξης και την ευελιξία της γλώσσας για την ανάπτυξη εφαρμογών.
2. Βιβλιοθήκη pandas: Η βιβλιοθήκη pandas αποτελεί ένα ισχυρό εργαλείο για τη διαχείριση και την ανάλυση δεδομένων. Χρησιμοποιήθηκε για την επεξεργασία και τον χειρισμό των δεδομένων που συνδέονται με την εφοδιαστική αλυσίδα, βοηθώντας μας να λάβουμε αποφάσεις βάσει των δεδομένων.
3. Τεχνολογία Blockchain: Για την ενσωμάτωση της τεχνολογίας blockchain στην εφοδιαστική αλυσίδα, χρησιμοποιήθηκαν διάφορα εργαλεία και πλατφόρμες:
 - i. Ganache: Το Ganache είναι ένα εργαλείο προσομοίωσης blockchain που δημιουργεί ένα τοπικό περιβάλλον για τον προγραμματισμό και τον δοκιμαστικό έλεγχο των έξυπνων συμβολαίων μας.
 - ii. Truffle: Το Truffle είναι ένα πλαίσιο ανάπτυξης που διευκολύνει τη δημιουργία, τη διαχείριση και τον προγραμματισμό των έξυπνων συμβολαίων σε Solidity, τη γλώσσα προγραμματισμού του Ethereum.
 - iii. Solidity: Η Solidity είναι η γλώσσα προγραμματισμού που χρησιμοποιείται για την ανάπτυξη έξυπνων συμβολαίων στο Ethereum. Επιτρέπει την εκφραστική υλοποίηση της λογικής των συμβολαίων.

- iv. Web3.js: Το Web3.js είναι μια βιβλιοθήκη JavaScript που επιτρέπει τη διασύνδεση με το Ethereum blockchain από εφαρμογές JavaScript, εκτελώντας συναλλαγές και ανακτώντας δεδομένα.

Με αυτήν την ενσωμάτωση των παραπάνω εργαλείων και τεχνολογιών, δημιουργούμε ένα περιβάλλον που επιτρέπει την αποτελεσματική επίλυση προβλημάτων στην εφοδιαστική αλυσίδα με την αξιοποίηση της τεχνολογίας blockchain. Αυτό θα παρέχει τη διαφάνεια, την ασφάλεια και την αποτελεσματικότητα που απαιτούνται για μια σύγχρονη εφοδιαστική αλυσίδα.

8.6 Ανάλυση κώδικα

Ξεκινώντας τον κώδικα, εισάγουμε τα απαραίτητα πακέτα για την εκτέλεση ενός προβλήματος ακεραίου προγραμματισμού χρησιμοποιώντας τη βιβλιοθήκη PuLP και την βιβλιοθήκη pandas για την επεξεργασία και ανάλυση δεδομένων στην Python.

```
# Εισαγωγή απαιτούμενων πακέτων
```

```
from pulp import *
```

```
import pandas as pd
```

Στον παραπάνω τμήμα του κώδικα, οι ενέργειες που εκτελούνται είναι:

- **from pulp import *:** Η δήλωση αυτή εισάγει όλες τις συναρτήσεις και τα αντικείμενα που χρειάζονται από τη βιβλιοθήκη PuLP για τον γραμμικό προγραμματισμό.
- **import pandas as pd:** Αυτή η εντολή εισάγει τη βιβλιοθήκη pandas με όνομα προσανατολισμένο στην προσπέλαση των συναρτήσεων και αντικειμένων της βιβλιοθήκης. Η pandas χρησιμοποιείται κυρίως για την επεξεργασία, ανάλυση και διαχείριση δεδομένων στην Python.

Με αυτόν τον τρόπο, ο κώδικας είναι έτοιμος να χρησιμοποιήσει τις συναρτήσεις και τα αντικείμενα της PuLP και της pandas για την εκτέλεση ακεραίου προγραμματισμού και την επεξεργασία των δεδομένων αντίστοιχα.

Στη συνέχεια ο κώδικας αναγνωρίζει και διαβάζει δεδομένα από ένα αρχείο Excel και τα αποθηκεύει σε ένα πίνακα δεδομένων (DataFrame) στην Python.

```
# Ανάγνωση δεδομένων από αρχείο Excel
```

```
df = pd.read_excel('data.xlsx')
```

Στον κώδικα, οι κύριες ενέργειες που εκτελούνται είναι:

- **pd**: Αναφέρεται στη βιβλιοθήκη pandas, η οποία παρέχει εργαλεία για την επεξεργασία και ανάλυση δεδομένων στην Python.
- **read_excel()**: Είναι μια συνάρτηση της βιβλιοθήκης pandas που χρησιμοποιείται για να διαβάσει δεδομένα από ένα αρχείο Excel. Παίρνει ως είσοδο τη διαδρομή του αρχείου Excel και επιστρέφει ένα πίνακα δεδομένων (DataFrame) που περιέχει τα δεδομένα από το αρχείο.
- **'data.xlsx'**: Είναι η διαδρομή (path) προς το αρχείο Excel που περιέχει τα δεδομένα που θέλουμε να διαβάσουμε. Σε αυτό το παράδειγμα, υποθέτουμε ότι το αρχείο ονομάζεται "data.xlsx".

Έτσι, ο κώδικας διαβάζει τα δεδομένα από το αρχείο Excel με τη βοήθεια της βιβλιοθήκης pandas και αποθηκεύει αυτά τα δεδομένα στη μεταβλητή **df**, η οποία είναι ένας πίνακας δεδομένων που μπορούμε να χρησιμοποιήσουμε για επεξεργασία και ανάλυση των δεδομένων.

Το παρακάτω κομμάτι κώδικα αντλεί δεδομένα από ένα DataFrame και τα αποθηκεύει σε μεταβλητές στην Python.

```
# Ανάκτηση δεδομένων σε μεταβλητές
```

```
num_berths = len(df['Number of berths'])
```

```
num_ships = len(df['Number of ships'])
```

```
berth_capacity = df['Berth Capacity'].tolist()
```

```
berth_cost = df['Berth Cost'].tolist()
```

```
ship_capacity = df['Ship Capacity'].tolist()
```

```
ship_rate = df['Ship Rate'].tolist()
```



```
ship_requirement = df['Ship Requirement'].tolist()
```

Οι μεταβλητές που χρησιμοποιούνται είναι:

- **num_berths**: Αποθηκεύει τον αριθμό των προβλητών μετρώντας το πλήθος των στοιχείων στην στήλη "Number of berths" του DataFrame.
- **num_ships**: Αποθηκεύει τον αριθμό των πλοίων μετρώντας το πλήθος των στοιχείων στην στήλη "Number of ships" του DataFrame.
- **berth_capacity**: Αποθηκεύει τις δυνατότητες προβλητών σε μια λίστα, αντλώντας τα δεδομένα από την στήλη "Berth Capacity" του DataFrame.
- **berth_cost**: Αποθηκεύει το κόστος προβλητών σε μια λίστα, αντλώντας τα δεδομένα από την στήλη "Berth Cost" του DataFrame.
- **ship_capacity**: Αποθηκεύει τις χωρητικότητες των πλοίων σε μια λίστα, αντλώντας τα δεδομένα από την στήλη "Ship Capacity" του DataFrame.
- **ship_rate**: Αποθηκεύει τον ρυθμό εργασίας / απόδοσης των πλοίων σε μία λίστα, που χρησιμοποιείται για τον υπολογισμό της αξίας της αντικειμενικής συνάρτησης του προβλήματος για τη βέλτιστη ανάθεση πλοίων σε προβλήτες, αντλώντας τα δεδομένα από την στήλη "Ship Rate" του DataFrame.
- **ship_requirement**: Αποθηκεύει τις απαιτήσεις ή ειδικές ανάγκες των πλοίων σε μία λίστα, όπως αν το φορτίο του εκάστοτε πλοίου χρειάζεται ψύξη, αντλώντας τα δεδομένα από την στήλη "Ship Requirement".

Στη συνέχεια δημιουργούμε ένα πρόβλημα ακέραιου προγραμματισμού και καθορίζουμε τις μεταβλητές απόφασης και την αντικειμενική συνάρτηση.

```
# Δημιουργία αντικειμένου προβλήματος LP
```

```
prob = LpProblem("Tactical Berth Allocation", LpMaximize)
```

```
# ορισμός μεταβλητών απόφασης
```

```

x = LpVariable.dicts("Το πλοίο i ανατίθεται στην προβλήτα j (1 αν ανατίθεται, 0 αν δεν
ανατίθεται)", ((i,j) for i in range(num_ships) for j in range(num_berths)), cat='Binary')

y = LpVariable.dicts("η αποβάθρα j χρησιμοποιείται αν είναι 1, αλλιώς 0", range(num_berths),
cat='Binary')

# ορισμός αντικειμενικής συνάρτησης

prob += lpSum(ship_rate[i] * ship_capacity[i] * x[(i,j)] for i in range(num_ships) for j in
range(num_berths))

```

Στον παραπάνω κώδικα, οι ενέργειες που εκτελούνται είναι:

- **LpProblem("Tactical Berth Allocation", LpMaximize):** Δημιουργεί ένα νέο αντικείμενο προβλήματος LP με το όνομα "Tactical Berth Allocation" και με σκοπό τη μεγιστοποίηση (LpMaximize) της αντικειμενικής συνάρτησης.
- **LpVariable.dicts(...):** ορίζει τις μεταβλητές απόφασης **x** και **y**. οι μεταβλητές **x** αναφέρονται στο αν ένα πλοίο **i** ανατίθεται στην αποβάθρα **j**, ενώ οι μεταβλητές **y** αναφέρονται στο αν η αποβάθρα **j** χρησιμοποιείται ή όχι. Ορίζονται ως δυαδικές μεταβλητές (cat='Binary').
- **prob += lpSum(...):** ορίζει τη συνάρτηση στόχου που πρέπει να μεγιστοποιηθεί. Η συνάρτηση στόχου υπολογίζει το άθροισμα των γινομένων των ρυθμών **ship_rate[i]**, των δυνατοτήτων **ship_capacity[i]** και των μεταβλητών **x[(i,j)]** για όλα τα **i** και **j**.

Με αυτόν τον τρόπο, ο κώδικας ορίζει ένα πρόβλημα γραμμικού προγραμματισμού με το όνομα "Tactical Berth Allocation" και ορίζει τις μεταβλητές απόφασης και τη συνάρτηση στόχου που πρέπει να μεγιστοποιηθεί.

Έπειτα ορίζουμε τους περιορισμούς για το πρόβλημα ακέραιου προγραμματισμού.

```

# ορισμός περιορισμών

# Κάθε πλοίο ανατίθεται σε μία μόνο αποβάθρα

```

```

for i in range(num_ships):

    prob += lpSum(x[(i,j)] for j in range(num_berths)) == 1

# Τα πλοία που το φορτίο τους χρειάζεται ψύξη ανατίθενται σε αποβάθρες με ψύξη

for i in range(num_ships):

    if ship_requirement[i] == 'refrigerated':

        for j in range(num_berths):

            if berth_capacity[j] >= ship_capacity[i]:

                prob += x[(i,j)] <= y[j]

# Κάθε αποβάθρα ανατίθεται τουλάχιστον σε ένα πλοίο

for j in range(num_berths):

    prob += lpSum(x[(i,j)] for i in range(num_ships)) <= 1

# Το άθροισμα της χωρητικότητας των πλοίων που ανατίθενται στην αποβάθρα δεν υπερβαίνει
τη δυνατότητα εξυπηρέτησής της

for j in range(num_berths):

    prob += lpSum(ship_capacity[i] * x[(i,j)] for i in range(num_ships) if (ship_requirement[i] ==
        None or ship_requirement[i] != 'refrigerated')) <= y[j] * berth_capacity[j]

```

Στον παραπάνω κώδικα, οι ενέργειες που εκτελούνται είναι:

- ο περιορισμός **prob += lpSum(x[(i,j)] for j in range(num_berths)) == 1** βεβαιώνει ότι κάθε πλοίο ανατίθεται σε μία μόνο αποβάθρα.
- ο περιορισμός **prob += x[(i,j)] <= y[j]** βεβαιώνει ότι τα ψυγεία των πλοίων ανατίθενται σε αποβάθρες με ψύξη.

- ο περιορισμός **prob += lpSum(x[(i,j)] for i in range(num_ships)) <= 1** βεβαιώνει ότι κάθε αποβάθρα ανατίθεται τουλάχιστον σε ένα πλοίο.
- ο περιορισμός **prob += lpSum(ship_capacity[i] * x[(i,j)] for i in range(num_ships) if (ship_requirement[i] == None or ship_requirement[i] != 'refrigerated')) <= y[j] * berth_capacity[j]** βεβαιώνει ότι το άθροισμα της χωρητικότητας των πλοίων που ανατίθενται στην αποβάθρα δεν υπερβαίνει τη δυνατότητα εξυπηρέτησής της.

Στην συνέχεια περνάμε στην επίλυση του προβλήματος βελτιστοποίησης που έχει διατυπωθεί χρησιμοποιώντας το εργαλείο PuLP.

```
# Επίλυση του προβλήματος

prob.solve()

# Εκτύπωση της κατάστασης του προβλήματος βελτιστοποίησης

print('Status:', LpStatus[prob.status])

# Εκτύπωση της βέλτιστης τιμής της αντικειμενικής συνάρτησης

print('Optimal Value:', value(prob.objective))

# Εκτύπωση των τιμών των μεταβλητών που είναι ίσες με 1 (δηλαδή η βέλτιστη ανάθεση των
# πλοίων στις αποβάθρες)

for v in prob.variables():

    if v.varValue == 1: print(v.name, '=', v.varValue)
```

Στον παραπάνω κώδικα, οι ενέργειες που εκτελούνται είναι:

- **prob.solve()**: Επιλύει το πρόβλημα βελτιστοποίησης που έχει διατυπωθεί με τη χρήση του PuLP.

- Εκτυπώνει την κατάσταση του προβλήματος βελτιστοποίησης με την εντολή **print('Status:', LpStatus[prob.status])**.
- Εκτυπώνει τη βέλτιστη τιμή της συνάρτησης στόχου με την εντολή **print('Optimal Value:', value(prob.objective))**.
- Εκτυπώνει τις τιμές των μεταβλητών που έχουν τιμή 1, που αντιστοιχούν στη βέλτιστη ανάθεση των πλοίων στις προβλήτες, με την εντολή **if v.varValue == 1: print(v.name, '=', v.varValue)**.

Με αυτόν τον τρόπο, ο κώδικας επιλύει το πρόβλημα βελτιστοποίησης και εκτυπώνει την κατάσταση, τη βέλτιστη τιμή της συνάρτησης στόχου και την ανάθεση πλοίων στις προβλήτες που αντιστοιχεί στη βέλτιστη λύση.

Στη συνέχεια, ο κώδικας δημιουργεί ένα πίνακα δεδομένων χρησιμοποιώντας την βιβλιοθήκη pandas με τα αποτελέσματα της βελτιστοποίησης:

```
# Δημιουργία ενός πίνακα δεδομένων pandas με τα αποτελέσματα

results = []

for v in prob.variables():

    results.append([v.name, v.varValue])

results_df = pd.DataFrame(results, columns=['Variable Name', 'Value'])

obj_val = prob.objective.value()

results_df = results_df.append({'Variable Name': 'Objective Function Value', 'Value': obj_val},
                               ignore_index=True)
```

Στο τελευταίο μέρος του κώδικα, τα αποτελέσματα που έχουν αποθηκευτεί στο πίνακα δεδομένων **results_df** καταγράφονται σε ένα αρχείο Excel:

```
# Εγγραφή του πίνακα δεδομένων σε ένα αρχείο Excel

writer = pd.ExcelWriter('Pulp Output.xlsx', engine='xlsxwriter')

results_df.to_excel(writer, sheet_name='Sheet1', index=False)
```

```
# Αποθήκευση του αρχείου Excel  
writer.save()
```

Στο πλαίσιο της ανάπτυξης της εφαρμογής, εκτελούμε ενέργειες για την αλληλεπίδραση με το δίκτυο Ethereum και τη δημιουργία Έξυπνων Συμβολαίων (Smart Contracts). Η χρήση των βιβλιοθηκών web3 και json είναι καίρια για τη σύνδεση μας με το Ethereum blockchain.

Η βιβλιοθήκη web3 παρέχει τα απαραίτητα εργαλεία για τη διαχείριση λειτουργιών στο Ethereum, ενώ η json βοηθά στη διαχείριση των μορφών JSON, ουσιαστικές για την ανάγνωση του Contract ABI. Πριν εμβαθύνουμε σε τεχνικές λεπτομέρειες, είναι ουσιαστικό να κατανοήσουμε τον λόγο για τον οποίο υλοποιούμε αυτές τις ενέργειες και πώς σχετίζονται με τους στόχους και τη λειτουργία της εφαρμογής. Τα Smart Contracts μας επιτρέπουν να υλοποιήσουμε λειτουργίες στο blockchain, και η επιλογή αυτών των βιβλιοθηκών ενισχύει τη συνοχή και την αποτελεσματικότητα της εφαρμογής στο περιβάλλον του Ethereum blockchain.

Εισαγωγή των βιβλιοθηκών web3 και json.

- Βιβλιοθήκη Web3 για τη διαχείριση των λειτουργιών του Ethereum blockchain.
- Βιβλιοθήκη json για τη διαχείριση των μορφών JSON (JavaScript Object Notation).

```
from web3 import Web3  
  
import json
```

Δημιουργία σύνδεσης με ένα τοπικό δίκτυο ανάπτυξης που είναι προσβάσιμο στη διεύθυνση <http://localhost:7545>. Αυτό απαιτεί την ύπαρξη ενός κοντινού δικτύου blockchain, όπως το Ganache, που είναι ενεργό στην καθορισμένη θύρα.

```
w3 = Web3(Web3.HTTPProvider('http://localhost:7545')).
```

Σε αυτήν την ενότητα δημιουργείται ένας λογαριασμός Ethereum χρησιμοποιώντας το ιδιωτικό κλειδί που παρέχεται στη μέθοδο `from_key()`. Πρώτα, η μέθοδος `from_key()` παίρνει το ιδιωτικό κλειδί ως είσοδο και δημιουργεί έναν λογαριασμό Ethereum που συσχετίζεται με αυτό το κλειδί. Το ιδιωτικό κλειδί είναι ένας μυστικός κωδικός που επιτρέπει την πρόσβαση στον λογαριασμό.

```
account=w3.eth.account.from_key('1ca84b340ea2b8e9e19f5999cd6f10c1fa90146333cd56c856e1cdf7705ebe02').
```

Στη συνέχεια, η μεταβλητή διεύθυνσης (`address`) ορίζεται σε μια συγκεκριμένη διεύθυνση Ethereum. Μια διεύθυνση Ethereum είναι ένας μοναδικός αναγνωριστικός για έναν λογαριασμό στον blockchain του Ethereum.

```
address = "0x6cc097AfCb8874089c743CF7323542a73eD7De48"
```

Στη συνέχεια στη μεταβλητή `balance` ανατίθεται η τιμή που επιστρέφεται από τη συνάρτηση `get_balance()`. Αυτή η συνάρτηση ανακτά το τρέχον υπόλοιπο της καθορισμένης διεύθυνσης Ethereum. Το υπόλοιπο αντιπροσωπεύει το ποσό του Ether (το κρυπτονόμισμα του δικτύου Ethereum) που υπάρχει στον λογαριασμό.

```
balance = w3.eth.get_balance(address)
```

Η διεύθυνση Ethereum στην οποία έχει αναπτυχθεί το συμβόλαιο ανατίθεται στη μεταβλητή `contract_address`. Ένα συμβόλαιο στο Ethereum είναι ένα αυτοεκτελούμενο πρόγραμμα βασισμένο στο blockchain. Μπορεί να θεωρηθεί ως μια συλλογή δεδομένων και κώδικα που μπορεί να επικοινωνεί με άλλους λογαριασμούς και άλλα συμβόλαια στο δίκτυο.

```
Contract_address = "0x9d536ebc92E3c1B5824f1ddF76Ffa8AfCc89A7e7"
```

Το JSON αρχείο "ExcelData.json" χρησιμοποιείται για το φόρτωμα του contract_abi. Η Application Binary Interface (ABI), ορίζει τη διεπαφή του έξυπνου συμβολαίου. Καθορίζει τις συναρτήσεις, τα γεγονότα και τις δομές δεδομένων που μπορούν να προσπελαστούν και να αλληλεπιδράσουν από εξωτερικά στοιχεία. Η ABI είναι απαραίτητη για την αλληλεπίδραση με το συμβόλαιο, καθώς παρέχει έναν τυποποιημένο τρόπο κωδικοποίησης και αποκωδικοποίησης των κλήσεων συναρτήσεων και των δεδομένων.

```
with open('ExcelData.json') as f:  
    contract_abi = json.load(f)['abi']
```

Με την βοήθεια της βιβλιοθήκης pandas, ο το πρόγραμμα διαβάζει τα δεδομένα από ένα αρχείο Excel με όνομα 'data.xlsx.' και αποθηκεύει τα δεδομένα σε ένα πίνακα με το όνομα 'df'.

Έπειτα υπολογίζει τον αριθμό των προβλητών και των πλοίων από τις αντίστοιχες στήλες, με την χρήση της εντολής len() και αποθηκεύει τις τιμές απο τις στήλες 'Berth Cost', 'Ship Capacity', 'Ship Rate', και 'Ship Requirement' στις αντίστοιχες λίστες.

```
df = pd.read_excel('data.xlsx')  
  
num_berths = len(df['Number of berths'])  
  
num_ships = len(df['Number of ships'])  
  
berth_capacity = len(df['Berth Capacity'])  
  
berth_cost = df['Berth Cost'].tolist()  
  
ship_capacity = df['Ship Capacity'].tolist()  
  
ship_rate = df['Ship Rate'].tolist()  
  
ship_requirement = df['Ship Requirement'].tolist()
```


Ορίζουμε τον δείκτη `index=1`, ώστε να μπορούμε να καθορίσουμε από ποια σειρά δεδομένων θέλουμε να ανακτήσουμε δεδομένα. Για να έχουμε πρόσβαση στα δεδομένα που είναι αποθηκευμένα σε αυτήν την συγκεκριμένη γραμμή, χρησιμοποιούμε την ενολή `getRow()` από το έξυπνο συμβόλαιο και περνάμε τον δείκτη ως όρισμα. Τα δεδομένα από το blockchain μπορούν να ζητηθούν χρησιμοποιώντας τη συνάρτηση `call()`. Η μεταβλητή `rows` κρατά τα ανακτηθέντα δεδομένα, τα οποία στη συνέχεια εκτυπώνονται.

```
index = 1

rows = contract.functions.getRow(index).call()

for i in range(len(rows)):

    print(rows[i])
```

Για να προχωρήσουμε πρέπει να δημιουργήσουμε ένα φάκελο με την ονομασία “contracts” με την βοήθεια του Truffle. Σε αυτό τον φάκελο, θα τοποθετήσουμε το αρχείο “task.sol” που θα περιέχει τον κώδικα του έξυπνου συμβολαίου.

Παρακάτω παρουσιάζεται ο κώδικας για τα smart contracts από το Solidity:

```
#Task.sol
// SPDX-License-Identifier: MIT
pragma experimental ABIEncoderV2;
pragma solidity >=0.4.22 <0.9.0;

contract ExcelData {

    struct Row { uint256 num_berths;

        uint256 num_ships;

        uint256 berth_capacity;

        uint256[] berth_cost;
```

```
uint256[] ship_capacity;  
  
uint256[] ship_rate;  
  
string[] ship_requirement;}  
Row[] private rows;
```

Το σχόλιο "// SPDX-License-Identifier: MIT" αναφέρεται στην άδεια χρήσης του κώδικα. Το "MIT" αναφέρεται στην άδεια MIT, μια ανοιχτή άδεια χρήσης που επιτρέπει την ελεύθερη χρήση, αναδιανομή και τροποποίηση του κώδικα.

Ορίζουμε ένα συμβόλαιο με το όνομα ExcelData. Μέσα στο συμβόλαιο, δηλώνεται μια δομή (struct) με το όνομα Row. Αυτή η δομή επιτρέπει την αποθήκευση και την πρόσβαση σε πολλές γραμμές δεδομένων με δομημένο τρόπο.

Στη συνέχεια, ορίζεται η μεταβλητή index και εκτελείται η γραμμή κώδικα rows = contract.functions.getRow(index).call(). Αυτή η γραμμή κώδικα αλληλεπιδρά με την έξυπνη σύμβαση Ethereum και καλεί τη συνάρτηση getRow με τον δείκτη index ως όρισμα. Η έξυπνη σύμβαση επιστρέφει μια δομή δεδομένων Row που αποθηκεύεται στη μεταβλητή rows.

Δημιουργούμε μία συνάρτηση με το όνομα addRow, η οποία δέχεται ορίσματα που αντιπροσωπεύουν τις τιμές κάθε πεδίου στη δομή Row. Τα ορίσματα περιλαμβάνουν τα εξής: _num_berths (αντιπροσωπεύει τον αριθμό των προβλητών), _num_ships (αντιπροσωπεύει τον αριθμό των πλοίων), _berth_capacity (που αντιπροσωπεύει τη χωρητικότητα των προβλητών), _berth_cost (ένας πίνακας τιμών uint256 που αντιπροσωπεύει το κόστος χρήσης κάθε προβλήτας), _ship_capacity (ένας πίνακας τιμών uint256 που αντιπροσωπεύει τη χωρητικότητα κάθε πλοίου), _ship_rate (ένας πίνακας τιμών που αντιπροσωπεύει τον ρυθμό εργασίας / απόδοσης κάθε πλοίου) και _ship_requirement (ένας πίνακας τιμών string που αντιπροσωπεύει την απαίτηση κάθε πλοίου). Το uint256 στη Solidity αναφέρεται σε έναν μη αρνητικό ακέραιο, με εύρος από 0 έως $2^{256}-1$. Συχνά χρησιμοποιείται για να αντιπροσωπεύσει ποσότητες χρημάτων, αριθμούς ευρυθμίας και άλλες περιπτώσεις όπου η ακρίβεια και η μη αρνητικότητα των τιμών είναι σημαντικές. Στον κώδικα το uint256 χρησιμοποιείται για τα πεδία δεδομένων στη δομή Row του έξυπνου συμβολαίου Solidity. Για παράδειγμα, οι μεταβλητές num_berths,

num_ships έχουν τύπο uint256, καθιστώντας τις κατάλληλες για την αναπαράσταση μη αρνητικών ακεραίων τιμών όπως ο αριθμός των λιμένων, των πλοίων, και άλλων παραμέτρων σχετικών με το έργο. Η λέξη-κλειδί memory χρησιμοποιείται για τις παραμέτρους του πίνακα για να καθορίσει ότι τα δεδομένα θα πρέπει να αποθηκεύονται στη μνήμη (memory) προσωρινά. Αυτό συμβαίνει διότι τα δεδομένα του πίνακα δεν χρειάζεται να διατηρηθούν πέρα από την εκτέλεση της συνάρτησης. Στο τμήμα κώδικα της έξυπνης σύμβασης Solidity, παρουσιάζεται η δήλωση της δομής Row, η οποία περιέχει τα ίδια δεδομένα με τις μεταβλητές που ορίζονται στο αρχείο Excel. Αυτή η δομή χρησιμοποιείται για να αποθηκεύσει μια εγγραφή με πληροφορίες για τα πλοία.

```
function addRow(uint256 _num_berths, uint256 _num_ships,
    uint256 _berth_capacity,
    uint256[] memory _berth_cost,
    uint256[] memory _ship_capacity,
    uint256[] memory _ship_rate,
    string[] memory _ship_requirement) public {
    rows.push(Row(_num_berths,
        _num_ships,
        _berth_capacity,
        _berth_cost,
        _ship_capacity,
        _ship_rate,
        _ship_requirement));}
function getRow(uint256 index) public view returns (Row memory) {
    require(index < rows.length, "Index out of bounds");
    return rows[index];}
```

Public view returns (Row memory): Η λέξη-κλειδί view εξασφαλίζει ότι καμία μεταβλητή κατάστασης δεν τροποποιείται από τη συνάρτηση. Η συνάρτηση επιστρέφει έναν μόνο

δομημένο τύπο Row που διατηρείται στη μνήμη, όπως υποδεικνύεται από την παράμετρο “returns (Row memory).”

Require(index < rows.length, "Index out of bounds"): Αυτή η γραμμή προσθέτει έναν έλεγχο για να εξασφαλίσει ότι το index βρίσκεται εντός των ορίων του πίνακα rows. Η δήλωση require επαληθεύει τη συνθήκη που βρίσκεται μέσα στις παρενθέσεις, και αν αξιολογηθεί ως ψευδής, προκαλεί ένα σφάλμα και διακόπτει την εκτέλεση της συνάρτησης. Σε αυτήν την περίπτωση, αν ο δείκτης είναι μεγαλύτερος ή ίσος με το μήκος του πίνακα rows, θα προκληθεί ένα σφάλμα με το μήνυμα σφάλματος "Index out of bounds".

Return rows[index]: Αυτή η δήλωση επιστρέφει το δομημένο τύπο Row που βρίσκεται στον καθορισμένο δείκτη (index) στον πίνακα rows. Ανακτά το δομημένο τύπο Row χρησιμοποιώντας την παράμετρο index και τον επιστρέφει στον χρήστη που την καλεί.

Έτσι ο πρώτος κώδικας πραγματοποιεί τον υπολογισμό της βέλτιστης κατανομής των πλοίων στις αποβάθρες του λιμένα. ο κώδικας χρησιμοποιεί τη βιβλιοθήκη **pulp** για να διαμορφώσει και να λύσει το πρόβλημα γραμμικού προγραμματισμού που αναλύσαμε. Ο δεύτερος κώδικας, από την άλλη πλευρά, χρησιμοποιεί τη βιβλιοθήκη **web3** για να αλληλεπιδράσει με μια έξυπνη σύμβαση Ethereum. Αυτός ο κώδικας επιτρέπει τη σύνδεση και την αλληλεπίδραση με την τοπική blockchain και την έξυπνη σύμβαση που έχει αναπτυχθεί πάνω της.

Συνοψίζοντας, οι δύο κώδικες λειτουργούν ανεξάρτητα μεταξύ τους. Ο πρώτος κώδικας αναλαμβάνει τη λύση του προβλήματος διαμοιρασμού των πλοίων, ενώ ο δεύτερος κώδικας είναι υπεύθυνος για τη σύνδεση και την αλληλεπίδραση με την έξυπνη σύμβαση Ethereum.

9.Αποτελέσματα

Εδώ βλέπουμε το αποτέλεσμα της ανάπτυξης του συμβολαίου που δημιουργήσαμε. Για την φόρτωση των δεδομένων στο blockchain, χρησιμοποιήθηκαν 3 blocks και το κόστος της διαδικασίας αναγράφεται στην γραμμή total cost.

```
Replacing 'ExcelData'
-----
> transaction hash: 0x3828a2ddb734143f7e73b13b79364c658c5575673ebc4bdb061fa7b9d2aed628
> Blocks: 0
> contract address: 0x3aFE1008D9F69949Fd34E555f5B3a2Ff9778b4cb
> block number: 3
> block timestamp: 1682114482
> account: 0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1
> balance: 99.9798121
> gas used: 775114 (0xbd3ca)
> gas price: 20 gwei
> value sent: 0 ETH
> total cost: 0.01550228 ETH

> Saving migration to chain.
> Saving artifacts
-----
> Total cost: 0.01550228 ETH
```

Προηγουμένως με την συνάρτηση getRow, δώσαμε στον χρήστη την δυνατότητα να μπορεί να λάβει τα δεδομένα μέσω του Blockchain. Αυτό είναι χρήσιμο για πολλούς λόγους. Για παράδειγμα οι ναυτιλιακές εταιρείες είναι σε θέση να ανεβάζουν τα δεδομένα για τα πλοία και τα φορτία τους και η διοίκηση του λιμανιού να μπορεί να βλέπει, χωρίς καθυστερήσεις, τα δεδομένα. Αντίστοιχα όλοι οι φορείς είναι σε θέση να μπορούν να βλέπουν με πλήρη διαφάνεια τα δεδομένα.

```
web.py x
src > web.py > ...
46
47
48 index = 0
49 rows = contract.functions.getRow(index).call()
50 for i in range(len(rows)):
51     print(rows[i])
52
53
54
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL**

```
(env) PS C:\Users\Lenovo T470s\Downloads\data\Fiver\2nd\env\src> python web.py
10
10
10
[1000, 800, 600, 1200, 1000, 1400, 800, 1000, 900, 456]
[100, 200, 150, 300, 250, 400, 150, 100, 200, 232]
[5, 6, 7, 4, 5, 6, 3, 2, 1, 323]
['None', 'refrigerated', 'None', 'refrigerated', 'None', 'None', 'refrigerated', 'refrigerated', 'refrigerated', 'None']
(env) PS C:\Users\Lenovo T470s\Downloads\data\Fiver\2nd\env\src>
```

Το Ganache GUI εμφανίζει όλα τα διαθέσιμα blocks στο blockchain. Κάθε τμήμα περιέχει συναλλαγές και πληροφορίες για την κατάσταση του δικτύου σε ένα συγκεκριμένο χρονικό σημείο. Το Ganache GUI εμφανίζει αυτά τα τμήματα με τρόπο που επιτρέπει στους χρήστες να παρακολουθούν τις δραστηριότητες του δικτύου, να ελέγχουν τις συναλλαγές και να αναλύουν την κατάσταση του blockchain τους.

CURRENT BLOCK	GAS PRICE	GAS LIMIT	HARDFORK	NETWORK ID	RPC SERVER	MINING STATUS	WORKSPACE	QUICKSTART	SAVE	SWITCH	⚙
5	20000000000	6721975	MUIRGLACIER	5777	HTTP://127.0.0.1:7545	AUTOMINING					
BLOCK 5	MINED ON 2023-04-22 03:08:27					GAS USED 1027105				1 TRANSACTION	
BLOCK 4	MINED ON 2023-04-22 03:01:23					GAS USED 27338				1 TRANSACTION	
BLOCK 3	MINED ON 2023-04-22 03:01:22					GAS USED 775114				1 TRANSACTION	
BLOCK 2	MINED ON 2023-04-22 03:01:20					GAS USED 42338				1 TRANSACTION	
BLOCK 1	MINED ON 2023-04-22 03:01:19					GAS USED 191943				1 TRANSACTION	
BLOCK 0	MINED ON 2023-04-22 02:55:54					GAS USED 0				NO TRANSACTIONS	

Όλες οι συναλλαγές που πραγματοποιήσαμε από την Python (με τη βιβλιοθήκη web3.py) προς το blockchain του Ganache.

Ganache

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK

5

GAS PRICE

20000000000

GAS LIMIT

6721975

HARDWARE

MURGLACHER

NETWORK ID

5777

RPC SERVER

HTTP://127.0.0.1:7545

MINING STATUS

AUTOMINING

WORKSPACE

QUICKSTART

SAVE

SWITCH

TX HASH

0x0f1730d9734e2f00afa327e5f69de830a2667a08c5a96cecad48fba55108bf25

CONTRACT CALL

FROM ADDRESS

0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1

TO CONTRACT ADDRESS

0x3aFE108D9F69949Fd34E555f5B3a2Ff9778b4cb

GAS USED

1027105

VALUE

0

TX HASH

0x8c27316d71150e6d007df01859e9b58475979faa78cce734c4b9d6e9b25ae887

CONTRACT CALL

FROM ADDRESS

0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1

TO CONTRACT ADDRESS

0x91bf1655963D8749AA8162C472b52B947F8b18f1

GAS USED

27338

VALUE

0

TX HASH

0x3828a2ddb734143f7e73b13b79364c658c5575673ebc4bdb061fa7b9d2aed628

CONTRACT CREATION

FROM ADDRESS

0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1

CREATED CONTRACT ADDRESS

0x3aFE108D9F69949Fd34E555f5B3a2Ff9778b4cb

GAS USED

775114

VALUE

0

TX HASH

0x7a9a0631dee2932eaa8ab4a8431412d7486a4f02a2d609c7a1a40cf41279b87

CONTRACT CALL

FROM ADDRESS

0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1

TO CONTRACT ADDRESS

0x91bf1655963D8749AA8162C472b52B947F8b18f1

GAS USED

42338

VALUE

0

← BACK

BLOCK 5

GAS USED

1027105

GAS LIMIT

6721975

MINED ON

2023-04-22 03:08:27

BLOCK HASH

0x7bf804f1503c8e5ec662c8879cea8850cf7ff1964147624b8702a7819ba59e03

TX HASH

0x0f1730d9734e2f00afa327e5f69de830a2667a08c5a96cecad48fba55108bf25

FROM ADDRESS

0x7958eF91fc401a8Ea4bE274662Fe18709eDf81a1

TO CONTRACT ADDRESS

0x3aFE108D9F69949Fd34E555f5B3a2Ff9778b4cb

GAS USED

1027105

VALUE

0

CONTRACT CALL

Συνοψίζοντας, δημιουργήσαμε μια σύνδεση με το δίκτυο Ethereum χρησιμοποιώντας τη βιβλιοθήκη web3 στην Python. Συνδεθήκαμε σε ένα τοπικό αναπτυξιακό δίκτυο μέσω του καθορισμένου HTTP παροχέα. Παρέχοντας ένα ιδιωτικό κλειδί, δημιουργήσαμε ένα λογαριασμό Ethereum και αποκτήσαμε τη διεύθυνση του λογαριασμού. Επίσης, ανακτήσαμε το υπόλοιπο του λογαριασμού, το οποίο αντιπροσωπεύει το ποσό του Ether που κρατείται στον λογαριασμό.

Επιπλέον, αλληλεπιδράσαμε με ένα smart contract που είχε αναπτυχθεί στο δίκτυο Ethereum. Ανακτήσαμε το ABI του συμβολαίου από ένα αρχείο JSON και δημιουργήσαμε μια έκδοση του συμβολαίου χρησιμοποιώντας τη συνάρτηση `web3.eth.contract()`. Αυτό μας επέτρεψε να διαβάσουμε τα δεδομένα του συμβολαίου και να καλέσουμε τις λειτουργίες του.

Καλέσαμε τη συνάρτηση `getRow()` με παράμετρο το δείκτη χρησιμοποιώντας το αντίστοιχο αντικείμενο συμβολαίου για να ανακτήσουμε μια σειρά δεδομένων από την αποθήκη του συμβολαίου. Οι πληροφορίες εκτυπώθηκαν για επιπρόσθετη εξέταση μετά την επιστροφή τους ως δομημένο αντικείμενο.

Επίσης, εξετάσαμε τη δημιουργία ενός έξυπνου συμβολαίου Solidity. Δημιουργήσαμε ένα συμβόλαιο με το όνομα `ExcelData` που χρησιμοποίησε ένα struct με το όνομα `Row` για να αποθηκεύει σειρές δεδομένων.

Συνολικά, συνδεθήκαμε με το blockchain, αλληλεπιδράσαμε με ένα έξυπνο συμβόλαιο, ανακτήσαμε δεδομένα και διενεργήσαμε διάφορες δραστηριότητες στο blockchain χρησιμοποιώντας τη βιβλιοθήκη `web3` και συνδεθήκαμε με το δίκτυο Ethereum.

Number of ships	Berth Capacity	Berth Cost	Ship Capacity	Ship Rate	Ship Requirement
15	500	1000	100	5	None
14	300	800	200	6	refrigerated
	200	600	150	7	None
	700	1200	300	4	refrigerated
	600	1000	250	5	None

	800	1400	400	6	None
	400	800	150	3	refrigerated
	500	1000	100	2	refrigerated
	400	900	200	1	refrigerated
	123	456	232	323	None

Δεδομένα σε μορφή Excel (αρχείο data.xlsx).

Variable Name	Value
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(0,_2)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(1,_6)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(2,_1)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(3,_7)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(4,_3)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(5,_8)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(6,_5)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(7,_9)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(8,_0)	1
Ship_i_is_assigned_to_berth_j(1_if_assigned,_0_not_assigned)_(9,_4)	1
berth_j_is_used_if_1,_else_0_0	1
berth_j_is_used_if_1,_else_0_1	1
berth_j_is_used_if_1,_else_0_2	1
berth_j_is_used_if_1,_else_0_3	1
berth_j_is_used_if_1,_else_0_4	1
berth_j_is_used_if_1,_else_0_5	1
berth_j_is_used_if_1,_else_0_6	1

berth_j_is_used_if_1,_else_0_7	1
berth_j_is_used_if_1,_else_0_8	1
berth_j_is_used_if_1,_else_0_9	1
Objective Function Value	83386

Αποτελέσματα σε μορφή Excel (αρχείο output.xlsx).

10.Συμπεράσματα

Το Blockchain αποτελεί ένα σύγχρονο τεχνολογικό σύστημα, το οποίο με το χρόνο ολοένα και περισσότερο εδραιώνεται σε διάφορες τομείς. Κάποιοι από αυτούς τους τομείς είναι η υγεία, η οικονομία ακόμα και η εφοδιαστική αλυσίδα. Η σχέση του Blockchain με την εφοδιαστική αλυσίδα εφαρμόζεται σε οργανισμούς όπως η ναυτιλιακή βιομηχανία, οι κοινές μεταφορές, τα αεροδρόμια κ.λπ. και μπορεί να έχει είτε θετικό είτε αρνητικό αντίκτυπο.

Η παρούσα εργασία σκόπευε στην συσχέτιση του Blockchain και της εφοδιαστικής αλυσίδας. Αρχικά, πραγματοποιήθηκε αποσαφήνιση των παραπάνω εννοιών και παρουσιάστηκαν οι χρήσεις και οι λειτουργίες, τόσο του Blockchain, όσο και της εφοδιαστικής αλυσίδας. Μετέπειτα, ειπώθηκαν τα παραδείγματα εφαρμογής του Blockchain στην εφοδιαστική αλυσίδα και τέλος αναφέρθηκαν τα θετικά και αρνητικά στοιχεία των εννοιών.

Το Blockchain αποτελεί έναν κρίσιμο παράγοντα στην επίλυση προβλημάτων στην εφοδιαστική αλυσίδα. Στο συγκεκριμένο πρόβλημα, Tactical Berth Allocation, το Blockchain συνδέεται στον κώδικα με διάφορους τρόπους για να βελτιώσει την απόδοση των λιμανιών και να επιλύσει τα προβλήματα κατανομής θέσεων για τα πλοία.

Καταρχάς, τα έξυπνα συμβόλαια (smart contracts) χρησιμοποιούνται για την αυτοματοποίηση της διαδικασίας λήψης αποφάσεων. Ορίζεται ένα πρόβλημα ακεραίου προγραμματισμού με στόχο τη μεγιστοποίηση της απόδοσης του λιμένα, και το Blockchain είναι υπεύθυνο για την εκτέλεση αυτού του προβλήματος με διαφάνεια και αξιοπιστία.

Συγκεκριμένα, το Blockchain επιλύει το πρόβλημα της διαφάνειας και της ακρίβειας των δεδομένων. Οι πληροφορίες σχετικά με τις θέσεις των πλοίων, τις δυνατότητες των λιμανιών, και τις απαιτήσεις των πλοίων συγκεντρώνονται σε ένα αποκεντρωμένο και ασφαλές μέσο. Αυτό εξασφαλίζει ότι όλα τα εμπλεκόμενα μέρη έχουν πρόσβαση στα ίδια ενημερωμένα δεδομένα, αποφεύγοντας την πιθανότητα σφαλμάτων ή αντιφάσεων.

Επιπλέον, το Blockchain χρησιμοποιείται για τη διασφάλιση της ασφάλειας και της ακεραιότητας των δεδομένων. Η τεχνολογία αυτή προσφέρει ασφαλείς μεθόδους κρυπτογράφησης και αποθήκευσης των δεδομένων, προστατεύοντάς τα από ανεπιθύμητες τροποποιήσεις ή αλλοιώσεις.

Συνοψίζοντας, το Blockchain ενσωματώνεται στο πρόβλημα Tactical Berth Allocation ως ισχυρό εργαλείο για την αυτοματοποίηση, τη διαφάνεια, και την ασφάλεια της διαδικασίας λήψης αποφάσεων, βελτιώνοντας έτσι την αποδοτικότητα και τη συνολική λειτουργία της εφοδιαστικής αλυσίδας.

11. Κώδικας

```
# Εισαγωγή απαιτούμενων πακέτων
from pulp import *
import pandas as pd

# Ανάγνωση δεδομένων από αρχείο Excel
df = pd.read_excel('data.xlsx')

# Ανάκτηση δεδομένων σε μεταβλητές
num_berths = len(df['Number of berths'])
```

```

num_ships = len(df['Number of ships'])
berth_capacity = df['Berth Capacity'].tolist()
berth_cost = df['Berth Cost'].tolist()
ship_capacity = df['Ship Capacity'].tolist()
ship_rate = df['Ship Rate'].tolist()
ship_requirement = df['Ship Requirement'].tolist()

# Δημιουργία αντικειμένου προβλήματος LP
prob = LpProblem("Tactical Berth Allocation", LpMaximize)

# ορισμός μεταβλητών απόφασης
x = LpVariable.dicts("Το πλοίο i ανατίθεται στην προβλήτα j (1 αν ανατίθεται, 0 αν δεν ανατίθεται)", ((i,j)
for i in range(num_ships) for j in range(num_berths)), cat='Binary')

y = LpVariable.dicts("η αποβάθρα j χρησιμοποιείται αν είναι 1, αλλιώς 0", range(num_berths),
cat='Binary')

# ορισμός αντικειμενικής συνάρτησης
prob += lpSum(ship_rate[i] * ship_capacity[i] * x[(i,j)] for i in range(num_ships) for j in
range(num_berths))

# ορισμός περιορισμών
# Κάθε πλοίο ανατίθεται σε μία μόνο αποβάθρα
for i in range(num_ships):
prob += lpSum(x[(i,j)] for j in range(num_berths)) == 1

# Τα πλοία που το φορτίο τους χρειάζεται ψύξη ανατίθενται σε αποβάθρες με ψύξη
for i in range(num_ships):
if ship_requirement[i] == 'refrigerated':
for j in range(num_berths):
if berth_capacity[j] >= ship_capacity[i]:

```

```

prob += x[(i,j)] <= y[j]

# Κάθε αποβάθρα ανατίθεται τουλάχιστον σε ένα πλοίο
for j in range(num_berths):
    prob += lpSum(x[(i,j)] for i in range(num_ships)) <= 1

# Το άθροισμα της χωρητικότητας των πλοίων που ανατίθενται στην αποβάθρα δεν υπερβαίνει τη
# δυνατότητα εξυπηρέτησής της
for j in range(num_berths):
    prob += lpSum(ship_capacity[i] * x[(i,j)] for i in range(num_ships) if (ship_requirement[i] == None or
    ship_requirement[i] != 'refrigerated')) <= y[j] * berth_capacity[j]

# Επίλυση του προβλήματος
prob.solve()

# Εκτύπωση της κατάστασης του προβλήματος βελτιστοποίησης
print('Status:', LpStatus[prob.status])

# Εκτύπωση της βέλτιστης τιμής της αντικειμενικής συνάρτησης
print('Optimal Value:', value(prob.objective))

# Εκτύπωση των τιμών των μεταβλητών που είναι ίσες με 1 (δηλαδή η βέλτιστη ανάθεση των πλοίων
# στις αποβάθρες)
for v in prob.variables():
    if v.varValue == 1: print(v.name, '=', v.varValue)

# Δημιουργία ενός πίνακα δεδομένων pandas με τα αποτελέσματα
results = []
for v in prob.variables():
    results.append([v.name, v.varValue])

```

```

results_df = pd.DataFrame(results, columns=['Variable Name', 'Value'])

obj_val = prob.objective.value()

results_df = results_df.append({'Variable Name': 'Objective Function Value', 'Value': obj_val},
ignore_index=True)

# Εγγραφή του πίνακα δεδομένων σε ένα αρχείο Excel
writer = pd.ExcelWriter('Pulp Output.xlsx', engine='xlsxwriter')
results_df.to_excel(writer, sheet_name='Sheet1', index=False)

# Αποθήκευση του αρχείου Excel
writer.save()

from web3 import Web3

import json

w3 = Web3(Web3.HTTPProvider('http://localhost:7545')).

account=w3.eth.account.from_key('1ca84b340ea2b8e9e19f5999cd6f10c1fa90146333cd56c856e1cdf77
05ebe02').

address = "0x6cc097AfCb8874089c743CF7323542a73eD7De48"

balance = w3.eth.get_balance(address)

Contract_address = "0x9d536ebc92E3c1B5824f1ddF76Ffa8AfCc89A7e7"

with open('ExcelData.json') as f:
contract_abi = json.load(f)['abi']

#Υπολογισμός προβλητών και πλοίων απο αρχείο Excel
df = pd.read_excel('data.xlsx')

num_berths = len(df['Number of berths'])

num_ships = len(df['Number of ships'])

berth_capacity = len(df['Berth Capacity'])

berth_cost = df['Berth Cost'].tolist()

ship_capacity = df['Ship Capacity'].tolist()

```

```
ship_rate = df['Ship Rate'].tolist()
ship_requirement = df['Ship Requirement'].tolist()
```

```
index = 1
rows = contract.functions.getRow(index).call()
for i in range(len(rows)):
    print(rows[i])
```

```
#Smart Contract
```

```
#Task.sol
```

```
// SPDX-License-Identifier: MIT
```

```
pragma experimental ABIEncoderV2;
```

```
pragma solidity >=0.4.22 <0.9.0;
```

```
contract ExcelData {
```

```
    struct Row { uint256 num_berths;
```

```
        uint256 num_ships;
```

```
        uint256 berth_capacity;
```

```
        uint256[] berth_cost;
```

```
        uint256[] ship_capacity;
```

```
        uint256[] ship_rate;
```

```
        string[] ship_requirement;}
```

```
    Row[] private rows;
```

```
#Αποθήκευση πληροφοριών
```

```
function addRow(uint256 _num_berths, uint256 _num_ships,
```

```
    uint256 _berth_capacity,
```

```
    uint256[] memory _berth_cost,
```

```
    uint256[] memory _ship_capacity,
```

```

        uint256[] memory _ship_rate,
        string[] memory _ship_requirement) public {
    rows.push(Row(_num_berths,
        _num_ships,
        _berth_capacity,
        _berth_cost,
        _ship_capacity,
        _ship_rate,
        _ship_requirement));}

    function getRow(uint256 index) public view returns (Row memory) {
        require(index < rows.length, "Index out of bounds");
    return rows[index];}}

```


Βιβλιογραφικές Αναφορές

- Anon., (2005). International Standards Organization 9000. *s.l.: s.n.*
- Atzori, M. (2015). Blockchain technology and decentralized governance: Is the state still necessary? *Work Pap*
- Aung, M. M. & Chang, Y. S., (2014). Traceability in a food supply chain: *Safety and quality perspectives*. 39, p. 172-184.
- Azzi, R., Chamoun, R. K. & Sokhn, M. (2019). The power of a blockchain-based supply chain. *Computers & Industrial Engineering*, 135, p. 582-592.
<https://doi.org/10.1016/j.cie.2019.06.042>
- Bodkhe, U., Tanwar, S., Parekh, K., Khanpara, P., Tyagi, S., Kumar, N. & Alazab, M. (2020). Blockchain for Industry 4.0: A Comprehensive Review. *IEEE Access*, 8.
<https://doi.org/10.1109/ACCESS.2020.2988579>
- Blanchard, D. (2010). *Supply Chain Management - Best Practices* (2 εκδ.). Canada: John Wiley & Sons, Inc.
- Buxbaum, P. (2017). Chinese logistics industry enters blockchain era. Ανακτήθηκε από:
<http://www.globaltrademag.com/global-logistics/chinese-logistics-industry-entersblockchain-era>.
- Chang, S. E. & Chen, Y. (2020). When Blockchain Meets Supply Chain: Systematic Literature Review on Current Development and Potential Applications. *IEEE Access*, 8. DOI: [10.1109/ACCESS.2020.2983601](https://doi.org/10.1109/ACCESS.2020.2983601)
- Christopher, M. (2011). *Logistics & Supply Chain Management* (4η εκδ.). Great Britain: Pearson Education Limited.
- Glaser F (2017) Pervasive decentralization of digital infrastructures: a framework for blockchain enabled system and use case analysis. In: *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS 2017)*, Waikoloa Village, Hawaii
- Deloitte (2017). When two chains combine. Supply chain meets blockchain.

- Dujak, D & Sajter, D. (2018). Blockchain Applications in Supply Chain. https://doi.org/10.1007/978-3-319-91668-2_2
- Dutta, P., Choi, T., Somani, S. & Butala, R. (2020). Blockchain technology in supply chain operations: Applications, challenges and research opportunities. *Transportation Research Part E: Logistics and Transportation Review*, 142. <https://doi.org/10.1016/j.tre.2020.102067>
- ElevateX (2021). Beginner's Guide: A Simple Introduction to Blockchain. Ανάκτηση από: <https://elevatex.de/blog/web3/beginners-guide-a-simple-introduction-to-blockchain/>
- Erturk, E., Lopez, D., & Yu, W. Y. (2020). Benefits and Risks of Using Blockchain in Smart Energy: A Literature Review. *Contemporary Management Research*, 15(3), p. 205-225. <https://doi.org/10.7903/cmr.19650>
- Fahmy, S. F. (2018). Blockchain and its uses. Ανακτήθηκε από: <https://www.sheriffadelfahmy.org/wp-content/uploads/2018/01/doc-1.pdf>
- Gerard, D. (2017). Attack of the 50-foot blockchain: Bitcoin, blockchain, Ethereum & smart contracts. CreateSpace Independent Publishing Platform.
- Gregorio, D., and Nustad, R.S. (2017), "Blockchain adoption in the shipping industry: a study of adoption likelihood and scenario-based opportunities and risks for IT service providers", *Doctoral dissertation, Master Thesis in International Business*, doi: [10.13140/RG.2.2.21839.38561](https://doi.org/10.13140/RG.2.2.21839.38561).
- Lambert, D. M. (2014). *Supply chain management: Processes, partnerships, performance* (4 εκδ.). Ponte Vedra Beach: FL: Supply Chain Management Institute.
- Lambert, D. M., & Enz, M. G. (2017). Issues in Supply Chain Management: Progress and potential. *Industrial Marketing Management*, 62.
- Λιούμπαρη, Π. Α. (2020). Η εφοδιαστική αλυσίδα στα κατασκευαστικά έργα: θεωρητικό υπόβαθρο, μοντέλα διαχείρισης και μελέτη εφαρμογής. (Μεταπτυχιακή Διπλωματική

Εργασία, Πανεπιστήμιο Θεσσαλίας). Ανακτήθηκε από:
<https://ir.lib.uth.gr/xmlui/bitstream/handle/11615/53418/21145.pdf?sequence=1>

Luther, W.J. & White, L.H. (2014). Can bitcoin become a major currency? *Working Paper*.

Mentzer, J., DeWitt, W., Keebler, J., Min, S., Nix, N., Smith, C., και συν. (2001). Defining Supply Chain Management. *Journal Of Business Logistics*, 22 (2).

Mishkin, F. S. (2004). The economics of money and financial markets, 7th edn. Pearson, Boston

Monrat, A. A., Schelen, O. & Andersson, K. (2019). Survey of Blockchain from the Perspectives of Applications, Challenges and Opportunities. DOI: [10.1109/ACCESS.2019.2936094](https://doi.org/10.1109/ACCESS.2019.2936094)

Munir, M., Jajja, M. S., Chatha, K. A., & Farooq, S. (2020). Supply chain risk management and operational performance: The enabling role of supply chain integration. *International Journal of Production Economics* (227).

Nakamoto, S. (2008). Bitcoin: A peer -to – peer electronic cash system. Ανακτήθηκε από:
<https://bitcoin.org/bitcoin.pdf>.

Niranjanamurthy, M., Nithya, B. N. & Jagannatha, S. (2018). Analysis of Blockchain technology: pros, cons and SWOT. *Cluster Computing*. <https://doi.org/10.1007/s10586-018-2387-5>

Nofer, M., Gomber, P. & Hinz, O. (2017). *Blockchain. Business & Information Systems Engineering*, 59(3), p. 183-187. DOI: [10.1007/s12599-017-0467-3](https://doi.org/10.1007/s12599-017-0467-3)

Peters, G., & Panayi, E. (2015, 11 18). Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. SSRN. Doi: <https://dx.doi.org/10.2139/ssrn.2692487>.

Petersson, E. & Baur, K. (2018). Impact of Blockchain Technology on Supply Chain Collaboration. <http://www.diva-portal.org/smash/get/diva2:1215210/FULLTEXT01.pdf>

Sofocle, (2016). Benefits of Blockchain in Food Supply Chain. Ανακτήθηκε από:
<https://medium.com/sofocle-technologies/benefits-of-blockchain-in-food-supply-chain-35b1f24eecfb>

- Taylor, G. (2017). Three benefits of using blockchain in your supply chain (and three downsides). ChainPoint. <http://blog.chainpoint.com/blog/three-benefits-of-using-blockchain-in-your-supply-chain-and-three-downsides>
- Tian, F. (2016). An agri-food supply chain traceability system for China based on RFID & blockchain technology. *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*. IEEE. <https://doi.org/10.1109/icsssm.2016.7538424>
- Underwood, S. (2016). Blockchain beyond bitcoin. *Communications of the ACM*, 59 (11), p.15-17.
- Wüst, K & Gervais, A. (2018). Do you need a Blockchain? *Crypto Valley Conference on Blockchain Technology*. DOI: [10.1109/CVCBT.2018.00011](https://doi.org/10.1109/CVCBT.2018.00011)
- Yassein, M. B., Shatnawi, F., Rawashdeh, S. & Mardin, W. (2019). Blockchain Technology: Characteristics, Security and Privacy, Issues and Solutions. *IEEE*. Doi:[10.1109/AICCSA47632.2019.9035216](https://doi.org/10.1109/AICCSA47632.2019.9035216)
- Yuan, Y. & Wong, F.Y. (2016). Towards blockchain-based intelligent transportation systems.
- Zhang, J. & Bhatt, T., (2014). A Guidance Document on the Best Practices in Food Traceability. *Food Science and Food Safety*, 13(5).
- Zheng, Z., Xie, S., Dai, H-N., Chen, X. and Wang, H. (2018) ‘Blockchain challenges and opportunities: a survey. *International Journal of Web and Grid Services*, 14(4), pp. 352–375. DOI: [10.1504/IJWGS.2018.10016848](https://doi.org/10.1504/IJWGS.2018.10016848)