



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Υλοποίηση Moodle plugin για διαχείριση βαθμολογιών με χρήση Blockchain

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Αγγελακόπουλος Αθανάσιος

Επιβλέπων : Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2023



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΥΠΟΛΟΓΙΣΤΩΝ

Υλοποίηση Moodle plugin για διαχείριση βαθμολογιών με χρήση Blockchain

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Αγγελακόπουλος Αθανάσιος

Επιβλέπων : Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 7^η Ιουλίου 2023.

(Υπογραφή)

.....
Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....
Παναγιώτης Τσανάκας
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....
Αριστείδης Παγουριτζής
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2023

(Υπογραφή)

.....

ΑΘΑΝΑΣΙΟΣ ΑΓΓΕΛΑΚΟΠΟΥΛΟΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π

© 2023 – All rights reserved

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Τα τελευταία χρόνια το Blockchain γίνεται όλο και πιο δημοφιλές καθώς προσφέρει μια νέα προσέγγιση στην αποθήκευση και τον διαμοιρασμό των δεδομένων. Πρόκειται για ένα δίκτυο κατανεμημένων κόμβων που αλληλοεπιδρούν μεταξύ τους και χειρίζονται τα δεδομένα με τρόπο που εξασφαλίζεται η ακεραιότητα τους χωρίς να απαιτείται εμπιστοσύνη σε μια κεντρική αρχή. Η αποκεντρωμένη φύση αυτής της λύσης εξυπηρετεί στην αυξημένη ασφάλεια των δεδομένων και έχει ανοίξει το δρόμο για πολλές νέες και καινοτόμες εφαρμογές βασισμένες στο Blockchain.

Το Moodle είναι ένα ιδιαίτερα διαδεδομένο ελεύθερο λογισμικό διαχείρισης μάθησης - Learning Management System, για τη διεξαγωγή ηλεκτρονικών μαθημάτων μέσω διαδικτύου, που προσφέρει ολοκληρωμένες υπηρεσίες τηλεκπαίδευσης. Όντας ελεύθερο λογισμικό ανοιχτού κώδικα, δίνει την ευκαιρία στους προγραμματιστές να τον ανακτήσουν και να τον επεκτείνουν αναπτύσσοντας νέες λειτουργίες. Με τον τρόπο αυτό δημιουργείται ένα οικοσύστημα εκπαιδευτικών εφαρμογών πάνω από το Moodle μέσω ενός, αρκετά ώριμου πλέον, μηχανισμού δημιουργίας plugins.

Συνδυάζοντας τα ανωτέρω, αντικείμενο της εργασίας αυτής είναι η αξιοποίηση τεχνολογίας Blockchain στην ανάπτυξη ενός plugin για το Moodle, το οποίο να χρησιμοποιεί το Blockchain για να βελτιώσει την ακεραιότητα και ασφάλεια της διαχείρισης βαθμολογιών μέσω του Moodle. Αρχικά θα πραγματοποιηθεί μελέτη του οικοσυστήματος του Moodle και ιδίως του μηχανισμού ανάπτυξης plugin, ο οποίος θα αξιοποιηθεί στη συνέχεια για την ανάπτυξη ενός νέου plugin που θα συνδέεται σε κατάλληλο ιδιωτικό δίκτυο Blockchain με σκοπό τη διασφάλιση του ότι οι βαθμολογίες που καταχωρούνται για κάθε μάθημα παραμένουν ασφαλείς χωρίς δυνατότητα αλλοίωσης. Η τεκμηρίωση του plugin θα γίνει σύμφωνα με τα πρότυπα της Τεχνολογίας λογισμικού και τα αντίστοιχα UML διαγράμματα, με τρόπο που να αναδεικνύεται η αρχιτεκτονική του συστήματος.

Λέξεις Κλειδιά

Moodle LMS, Moodle plugin development, PHP, JavaScript, Blockchain, Έξυπνα Συμβόλαια, Ιδιωτικό Blockchain, Solidity

Abstract

In recent years, Blockchain has become increasingly popular as it offers a new approach to storing and sharing data. It is a network of distributed nodes that interact with each other and handle data in a way that ensures their integrity without requiring trust in a central authority. This decentralized nature of the solution serves to increase the security of the data and has paved the way for many new and innovative applications based on Blockchain.

Moodle is a widely used free Learning Management System, software for conducting online courses via the Internet, offering integrated e-learning services. Being open-source free software, it gives developers the opportunity to retrieve and extend it by developing new features, it gives developers the opportunity to retrieve and extend it by developing new features. In this way, an ecosystem of educational applications is created on top of Moodle through a now quite mature plugin creating mechanism.

Combining the above, the subject of this thesis is the exploitation of Blockchain technology in the development of a plugin for Moodle, which uses Blockchain to improve the integrity and security of the grade management process through Moodle. Initially, a study of the Moodle ecosystem will be carried out, especially the plugin development mechanism, which will be used to develop a new plugin that will connect to a suitable private Blockchain network with the aim of ensuring that the grades entered for each course remain secure without the possibility of alteration. The documentation of the plugin will be done according to the standards of Software Technology and the corresponding UML diagrams, in a way that highlights the architecture of the system.

Keywords

Moodle LMS, Moodle plugin Development, PHP, Javascript, Blockchain, Smart Contracts, Private Blockchain, Solidity

Ευχαριστίες

Μετά την ολοκλήρωση της διπλωματικής μου εργασίας, ολοκληρώνεται το ταξίδι μου στη Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου (ΕΜΠ). Θα ήθελα να εκφράσω τις ευχαριστίες μου στον καθηγητή και επιβλέποντα μου κ. Βασίλη Βεσκούκη, για την ευκαιρία που μου δόθηκε να εργαστώ και να αποκτήσω πρακτική εμπειρία πάνω σε αυτό το θέμα, αλλά και για την πολύτιμη καθοδήγηση, τα σχόλια και την άποψη συνεργασία του κατά τη διάρκεια της εκπόνησης της διπλωματικής μου. Επιπρόσθετα, θα ήθελα να ευχαριστήσω τη βοήθεια που παρείχε ο Υποψήφιος Διδάκτωρ και ερευνητής Χρίστος Χατζιχριστοφή, ο οποίος με βοήθησε καθ' όλη τη διάρκεια της εκπόνησης της διπλωματικής μου να κατανοήσω το θέμα σε βάθος, όπως και ο Ιωάννης Τζαννέτος. Τέλος, οφείλω να ευχαριστήσω τους γονείς μου Γωγώ και Κώστα, την αδερφή μου Αντωνία καθώς και τους φίλους μου που με στήριξαν σε όλη τη διάρκεια της φοίτησης μου και ιδιαίτερα τον φίλο και συμφοιτητή μου Αντρέα που μαζί μοχθήσαμε από την αρχή μέχρι την απόκτηση του Διπλώματος.

Αθήνα, Ιούλιος 2023

Αθανάσιος Αγγελακόπουλος

Πίνακας Περιεχομένων

Περίληψη	6
Abstract	7
Ευχαριστίες	8
Πίνακας Περιεχομένων	10
Πίνακας Εικόνων	14
Κεφάλαιο 1 Εισαγωγή	16
1.1 Τα κίνητρα και οι στόχοι της εργασίας	16
1.2 Η δομή της εργασίας	17
Κεφάλαιο 2 Η πλατφόρμα του Moodle	18
2.1 Τι είναι το Moodle	18
2.2 Η δομή του Moodle	19
2.3 Χρήστες, ρόλοι και δικαιώματα στο Moodle	21
Κεφάλαιο 3 Εισαγωγή στο Blockchain	24
3.1 Τι είναι το Blockchain	24
3.2 Δομικά μέρη (Node, Block)	25
3.3 Πώς λειτουργεί	25
3.4 Τύποι δικτύων Blockchain	25
3.5 Αλγόριθμοι συναίνεσης (consensus)	27
3.6 Έξυπνα συμβόλαια και συναλλαγές	28
3.7 Οφέλη αλλά και μειονεκτήματα	29
Κεφάλαιο 4 Ανάλυση του προβλήματος	32
4.1 Η διαδικασία βαθμολόγησης	32
4.1.1 Κρίσιμα χαρακτηριστικά	32
4.1.2 Ανάλυση της διαδικασίας	33
4.1.3 Πιθανά σημεία βελτίωσης-Κίνητρα	34
4.2 Περιγραφή της προσέγγισης	35
4.2.1 Γιατί Moodle;	35
4.2.2 Ιδιωτικό αδειοδοτούμενο δίκτυο Blockchain	36
4.2.3 UML διάγραμμα της νέας διαδικασίας	37
4.3 Μοντελοποίηση των περιπτώσεων χρήσης με UML	37
4.4 Τα συστατικά της λύσης	40
	10

4.5 Αλληλεπίδραση με την υπηρεσία Metamask	43
4.6 Περίπτωση χρήσης 1: Εξουσιοδότηση καθηγητή	45
4.6.1 UML Activity Διάγραμμα	46
4.6.2 UML Sequence διάγραμμα	47
4.6.3 Wireflow διαγράμματα εξουσιοδότησης	47
4.7 Περίπτωση χρήσης 2: Οριστικοποίηση βαθμολογιών	48
4.7.1 UML Activity Διάγραμμα	49
4.7.2 UML Sequence διάγραμμα	49
4.7.3 Wireflow διαγράμματα οριστικοποίησης	50
4.8 Περίπτωση χρήσης 3: Επικύρωση βαθμών	51
4.8.1 UML Activity Διάγραμμα	51
4.8.2 UML Sequence διάγραμμα	52
4.8.3 Wireflow διαγράμματα οριστικοποίησης	53
4.9 Συνολικό wireflow διάγραμμα	55
Κεφάλαιο 5 Υλοποίηση	56
5.1 Εγκατάσταση Moodle	56
5.1.1 Δημιουργία μαθήματος	58
5.2 Ανάπτυξη επιπρόσθετου στοιχείου – Moodlechain	62
5.2.1 Τύπος plugin και δομή	62
5.3 Ανάπτυξη έξυπνου συμβολαίου	75
5.3.1 Remix	75
5.3.2 Δομές, μεταβλητές και συναρτήσεις	75
5.3.3 UML Class Diagram	80
5.4 Εγκατάσταση Blockchain Network	80
Κεφάλαιο 6 Demo	84
6.1 Εγκατάσταση plugin και βασική περιήγηση στο Moodle	84
6.2 Καθηγητής που δεν έχει δικαιώματα-Προσπάθεια οριστικοποίησης	87
6.3 Εξουσιοδότηση καθηγητή από διαχειριστή	89
6.4 Καθηγητής που έχει δικαιώματα-Οριστικοποίηση	91
6.5 Επικύρωση βαθμών- δεν βρέθηκαν αλλαγές	92
6.6 Επικύρωση βαθμών-βρέθηκαν αλλαγές	93
6.7 Επικύρωση βαθμών-δεν βρέθηκαν βαθμοί	94
Κεφάλαιο 7 Επίλογος	96

7.1 Συμπεράσματα	96
7.2 Επόμενα βήματα	97
7.2.1 Διαδικασία διαχείρισης βαθμών	97
7.2.2 Moodle plugin	98
Βιβλιογραφικές αναφορές	102

Πίνακας Εικόνων

ΕΙΚΟΝΑ 1 ΣΤΑΤΙΣΤΙΚΑ ΤΟΥ MOODLE-LMS.....	19
ΕΙΚΟΝΑ 2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ MOODLE-LAMP STACK	23
ΕΙΚΟΝΑ 3 ΙΣΤΟΣ ΜΕ ΚΕΝΤΡΙΚΗ ΑΡΧΗ ΚΑΙ ΑΠΟΚΕΝΤΡΩΜΕΝΟΣ ΙΣΤΟΣ	24
ΕΙΚΟΝΑ 4 ΑΠΛΗ ΔΙΑΔΙΚΑΣΙΑ ΒΑΘΜΟΛΟΓΗΣΗΣ	34
ΕΙΚΟΝΑ 5 Η ΔΙΑΔΙΚΑΣΙΑ ΒΑΘΜΟΛΟΓΗΣΗΣ ΜΕΤΑ ΤΗΝ ΠΡΟΣΘΗΚΗ ΤΟΥ ΣΤΟΙΧΕΙΟΥ.....	37
ΕΙΚΟΝΑ 6 ΔΙΑΓΡΑΜΜΑ ΧΡΗΣΤΩΝ-ΠΕΡΙΠΤΩΣΕΩΝ ΧΡΗΣΗΣ	38
ΕΙΚΟΝΑ 7 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ ΥΨΗΛΟΥ ΕΠΙΠΕΔΟΥ.....	38
ΕΙΚΟΝΑ 8 WIREFLOW ΔΙΑΓΡΑΜΜΑ ΓΙΑ ΤΟΝ ΚΑΘΗΓΗΤΗ ΤΟΥ ΜΑΘΗΜΑΤΟΣ	39
ΕΙΚΟΝΑ 9 WIREFLOW ΔΙΑΓΡΑΜΜΑ ΓΙΑ ΤΟΝ ΔΙΑΧΕΙΡΙΣΤΗ ΤΟΥ MOODLE.....	40
ΕΙΚΟΝΑ 10 ΔΙΑΓΡΑΜΜΑ ΣΥΣΤΑΤΙΚΩΝ ΤΟΥ PLUGIN.....	43
ΕΙΚΟΝΑ 11 ΔΙΑΓΡΑΜΜΑ DEPLOYMENT	43
ΕΙΚΟΝΑ 12 ΓΕΝΙΚΟ ΔΙΑΓΡΑΜΜΑ ΔΙΑΔΙΚΑΣΙΑΣ ΧΡΗΣΗΣ ΤΗΣ ΥΠΗΡΕΣΙΑΣ ΤΟΥ METAMASK.....	45
ΕΙΚΟΝΑ 13 ΔΙΑΓΡΑΜΜΑ ΤΗΣ ΔΙΑΔΙΚΑΣΙΑΣ ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ.....	46
ΕΙΚΟΝΑ 14 ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ.....	47
ΕΙΚΟΝΑ 15 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ ΚΑΘΗΓΗΤΗ : ΕΠΙΛΟΓΗ ΚΑΘΗΓΗΤΗ	47
ΕΙΚΟΝΑ 16 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΞΟΥΣΙΟΔΟΤΗΣΗΣ ΚΑΘΗΓΗΤΗ : ΕΠΙΤΥΧΗΣ ΕΞΟΥΣΙΟΔΟΤΗΣΗ.....	48
ΕΙΚΟΝΑ 17 ΔΙΑΓΡΑΜΜΑ ΔΙΑΔΙΚΑΣΙΑΣ ΟΡΙΣΤΙΚΟΠΟΙΗΣΗΣ ΒΑΘΜΟΛΟΓΙΩΝ	49
ΕΙΚΟΝΑ 18 ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ ΟΡΙΣΤΙΚΟΠΟΙΗΣΗΣ ΒΑΘΜΟΛΟΓΙΩΝ	49
ΕΙΚΟΝΑ 19 ΔΙΑΓΡΑΜΜΑ ΔΙΑΔΙΚΑΣΙΑΣ ΟΡΙΣΤΙΚΟΠΟΙΗΣΗΣ ΒΑΘΜΟΛΟΓΙΩΝ: ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ 50	50
ΕΙΚΟΝΑ 20 ΔΙΑΓΡΑΜΜΑ ΔΙΑΔΙΚΑΣΙΑΣ ΟΡΙΣΤΙΚΟΠΟΙΗΣΗΣ ΒΑΘΜΟΛΟΓΙΩΝ: ΕΠΙΤΥΧΗΣ ΠΡΟΣΘΗΚΗ ΒΑΘΜΩΝ	50
ΕΙΚΟΝΑ 21 ΔΙΑΓΡΑΜΜΑ ΔΙΑΔΙΚΑΣΙΑΣ ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ	51
ΕΙΚΟΝΑ 22 ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ.....	52
ΕΙΚΟΝΑ 23 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ: ΕΠΙΛΕΞΤΕ ΔΡΑΣΤΗΡΙΟΤΗΤΑ ΓΙΑ ΕΠΙΚΥΡΩΣΗ.....	53
ΕΙΚΟΝΑ 24 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ: ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ	53
ΕΙΚΟΝΑ 25 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ: ΠΙΝΑΚΑΣ ΔΙΑΦΟΡΩΝ.....	54
ΕΙΚΟΝΑ 26 ΔΙΑΓΡΑΜΜΑ WIREFLOW ΕΠΙΚΥΡΩΣΗΣ ΒΑΘΜΩΝ: ΔΕΝ ΒΡΕΘΗΚΑΝ ΔΙΑΦΟΡΕΣ	54
ΕΙΚΟΝΑ 27 ΣΥΝΟΛΙΚΟ WIREFLOW ΔΙΑΓΡΑΜΜΑ	55
ΕΙΚΟΝΑ 28 ΕΓΚΑΤΑΣΤΑΣΗ MOODLE, ΑΡΧΙΚΗ ΣΕΛΙΔΑ	58
ΕΙΚΟΝΑ 29 ΕΓΚΑΤΑΣΤΑΣΗ MOODLE, ΡΥΘΜΙΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ.....	58
ΕΙΚΟΝΑ 30 ΔΗΜΙΟΥΡΓΙΑ ΜΑΘΗΜΑΤΟΣ ΣΤΟ MOODLE.....	59
ΕΙΚΟΝΑ 31 ΜΟΡΦΗ ΑΡΧΕΙΟΥ ΧΡΗΣΤΩΝ	60

ΕΙΚΟΝΑ 32 ΑΝΕΒΑΣΜΑ ΑΡΧΕΙΟΥ ΧΡΗΣΤΩΝ	60
ΕΙΚΟΝΑ 33 BULK ENROLMENT PLUGIN, ΑΡΧΙΚΗ ΣΕΛΙΔΑ	61
ΕΙΚΟΝΑ 34 ΒΑΘΜΟΛΟΓΙΑ ΜΑΘΗΜΑΤΟΣ SOFTWARE ENGINEERING.....	61
ΕΙΚΟΝΑ 35 MOODLECHAIN-ΔΟΜΗ PLUGIN	62
ΕΙΚΟΝΑ 36 MOODLECHAIN ΔΙΑΓΡΑΜΜΑ ΣΥΣΤΑΤΙΚΩΝ	63
ΕΙΚΟΝΑ 37 UML CLASS DIAGRAM ΤΟΥ ΕΞΥΠΝΟΥ ΣΥΜΒΟΛΑΙΟΥ	80
ΕΙΚΟΝΑ 38 ΕΓΚΑΤΑΣΤΑΣΗ MOODLECHAIN.....	84
ΕΙΚΟΝΑ 39 ΕΓΚΑΤΑΣΤΑΣΗ MOODLECHAIN, ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΣΤΗ ΒΑΣΗ.....	85
ΕΙΚΟΝΑ 40 ΠΕΡΙΗΓΗΣΗ ΣΤΟ MOODLE, ΑΡΧΙΚΗ ΣΕΛΙΔΑ	85
ΕΙΚΟΝΑ 41 ΑΡΧΙΚΗ ΣΕΛΙΔΑ MOODLECHAIN.....	85
ΕΙΚΟΝΑ 42 ΕΙΣΑΓΩΓΗ ΔΙΚΤΥΟΥ ΣΤΟ ΜΕΤΑΜΑΣΚ.....	86
ΕΙΚΟΝΑ 43 ΕΙΣΑΓΩΓΗ ΛΟΓΑΡΙΑΣΜΟΥ ΣΤΟ ΜΕΤΑΜΑΣΚ ΜΕ ΙΔΙΩΤΙΚΟ ΚΛΕΙΔΙ.....	87
ΕΙΚΟΝΑ 44 ΟΡΙΣΤΙΚΟΠΟΙΗΣΗ ΒΑΘΜΩΝ, ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ	88
ΕΙΚΟΝΑ 45 ΠΑΡΑΘΥΡΟ ΜΕΤΑΜΑΣΚ, ΕΚΤΙΜΗΣΗ ΠΟΡΩΝ ΚΑΙ ΕΠΙΒΕΒΑΙΩΣΗ ΣΥΝΑΛΛΑΓΗΣ	88
ΕΙΚΟΝΑ 46 ΠΑΡΑΘΥΡΟ ΑΠΟΤΥΧΙΑΣ ΣΥΝΑΛΛΑΓΗΣ, Ο ΚΑΘΗΓΗΤΗΣ ΔΕΝ ΕΙΧΕ ΕΞΟΥΣΙΟΔΟΤΗΘΕΙ	89
ΕΙΚΟΝΑ 47 ΕΠΙΛΟΓΗ ΚΑΘΗΓΗΤΗ ΓΙΑ ΕΞΟΥΣΙΟΔΟΤΗΣΗ.....	89
ΕΙΚΟΝΑ 48 ΠΑΡΑΘΥΡΟ ΜΕΤΑΜΑΣΚ-ΑΠΟΔΟΧΗ ΣΥΝΑΛΛΑΓΗΣ	90
ΕΙΚΟΝΑ 49 ΕΞΟΥΣΙΟΔΟΤΗΣΗ ΚΑΘΗΓΗΤΗ-ΜΗΝΥΜΑ ΕΠΙΤΥΧΙΑΣ	90
ΕΙΚΟΝΑ 50 ΟΡΙΣΤΙΚΟΠΟΙΗΣΗ ΒΑΘΜΩΝ-ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ	91
ΕΙΚΟΝΑ 51 ΟΘΟΝΗ MOODLE ΚΑΙ ΠΑΡΑΘΥΡΟ ΜΕΤΑΜΑΣΚ	91
ΕΙΚΟΝΑ 52 ΟΡΙΣΤΙΚΟΠΟΙΗΣΗ ΒΑΘΜΩΝ-ΜΗΝΥΜΑ ΕΠΙΤΥΧΙΑΣ	92
ΕΙΚΟΝΑ 53 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ-ΕΠΙΛΟΓΗ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ ΓΙΑ ΕΠΙΚΥΡΩΣΗ.....	92
ΕΙΚΟΝΑ 54 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ- ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ	93
ΕΙΚΟΝΑ 55 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ-ΔΕΝ ΒΡΕΘΗΚΑΝ ΔΙΑΦΟΡΕΣ	93
ΕΙΚΟΝΑ 56 ΑΛΛΑΓΗ ΒΑΘΜΟΛΟΓΙΑΣ ΦΟΙΤΗΤΗ PRESLEY ANKUNDING.....	94
ΕΙΚΟΝΑ 57 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ, ΒΡΕΘΗΚΑΝ ΔΙΑΦΟΡΕΣ.....	94
ΕΙΚΟΝΑ 58 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ-ΕΠΙΛΟΓΗ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ ΓΙΑ ΕΠΙΚΥΡΩΣΗ.....	95
ΕΙΚΟΝΑ 59 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ-ΣΥΜΠΛΗΡΩΣΗ ΣΤΟΙΧΕΙΩΝ ΜΑΘΗΜΑΤΟΣ	95
ΕΙΚΟΝΑ 60 ΕΠΙΚΥΡΩΣΗ ΒΑΘΜΩΝ-ΔΕΝ ΥΠΑΡΧΟΥΝ ΒΑΘΜΟΙ	95
ΕΙΚΟΝΑ 61 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ ΠΡΟΤΑΣΗ 1	97
ΕΙΚΟΝΑ 62 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ ΠΡΟΤΑΣΗ 2	98
ΕΙΚΟΝΑ 63 ΡΟΗ ΕΡΓΑΣΙΩΝ ΓΙΑ ΔΗΜΟΣΙΕΥΣΗ PLUGIN[33].....	99

Κεφάλαιο 1 Εισαγωγή

1.1 Τα κίνητρα και οι στόχοι της εργασίας

Παρατηρώντας τις τεχνολογικές τάσεις τα τελευταία χρόνια είναι φανερό πως το Blockchain είναι μια τεχνολογία που έχει κερδίσει ευρεία δημοτικότητα στη κοινότητα, καθώς δίνει λύσεις σε μια πληθώρα προβλημάτων. Ένας βασικός λόγος είναι η ακεραιότητα και η ασφάλεια που προσφέρει στον διαμοιρασμό της πληροφορίας, είτε πρόκειται για οικονομικές συναλλαγές, είτε ζωντανά δεδομένα σε μια εφοδιαστική αλυσίδα είτε για να αποθηκευτούν ευαίσθητες προσωπικές πληροφορίες. Ο τρόπος που το επιτυγχάνει αυτό είναι παρέχοντας μια αποκεντρωμένη λύση με την οποία τα ενδιαφερόμενα μέρη δεν χρειάζεται να δείξουν εμπιστοσύνη σε μια κεντρική αρχή πιστοποίησης, εξουδετερώνοντας έτσι τα μοναδικά σημεία αποτυχίας - εξουσίας που συνήθως χαρακτηρίζουν παλαιότερες πιο στατικές λύσεις. Ερευνώντας λοιπόν τις λύσεις και τα οφέλη του Blockchain βρήκαμε ένα πιθανό πεδίο εφαρμογής, την διαχείριση των βαθμολογιών σε εκπαιδευτικές δράσεις οι οποίες γίνονται σε περιβάλλον ενός LMS (Learning Management System) σε μια κλίμακα όπως αυτή ενός Πανεπιστημίου. Η σημασία του προβλήματος είναι προφανής, καθώς αφορά την ακεραιότητα των βαθμολογιών η οποία, σε περιβάλλον Blockchain, δεν διασφαλίζεται με την εγγύηση ενός τρίτου μέρους ή του διαχειριστή ενός πληροφοριακού συστήματος, αλλά σύμφωνα με τους αυστηρούς κανόνες του Blockchain.

Υπάρχει μεγάλη πληθώρα συστημάτων LMS που χρησιμοποιούνται για την υποστήριξη της εκπαίδευσης κάθε μορφής. Ένα από τα πιο διαδεδομένα είναι το Moodle, το οποίο αναπτύχθηκε από τον Martin Dougiamas με σκοπό να βοηθήσει τους εκπαιδευτές να δημιουργήσουν online μαθήματα με έντονη αλληλεπίδραση και συνεργατική μάθηση.

Η παρούσα εργασία αποσκοπεί στον σχεδιασμό και την υλοποίηση ενός πρόσθετου (plugin) του Moodle, εργαλείου το οποίο θα υποστηρίξει την διαδικασία διαχείρισης των βαθμολογιών σε εκπαιδευτικές δράσεις που αυτό χρησιμοποιείται, όπως σε ένα πανεπιστήμιο, όπου η ακεραιότητα της βαθμολογίας έχει τη γνωστή κρισιμότητα. Η αρχική υπόθεση είναι πως παράλληλα με τη δια ζώσης διδασκαλία, χρησιμοποιείται και η πλατφόρμα του Moodle, όπως γίνεται και στο Ε.Μ.Πολυτεχνείο (Helios). Υποθέτοντας ότι στο Moodle καταχωρούνται βαθμολογίες που τελικά μεταφέρονται στις επίσημες βαθμολογίες των μαθημάτων, σκοπός μας είναι να αναπτύξουμε ένα επιπρόσθετο στοιχείο (plugin) για το Moodle, το οποίο διασφαλίζει την ακεραιότητα των βαθμολογιών αποθηκεύοντας τες σε

ένα ιδιωτικό δίκτυο Blockchain. Όλες οι σχεδιαστικές επιλογές και τα πιθανά οφέλη από ένα τέτοιο επιπρόσθετο στοιχείο θα μελετηθούν εκτενέστερα στις παρακάτω ενότητες.

1.2 Η δομή της εργασίας

Στα πρώτα δυο κεφάλαια γίνεται μια θεωρητική εισαγωγή στις τεχνολογίες που θα χρησιμοποιηθούν σε αυτή την εργασία. Στο κεφάλαιο 2, ξεκινάμε με μια εισαγωγή στην πλατφόρμα του Moodle και γίνεται αναφορά στην βασική ορολογία του οικοσυστήματος που θα γίνει η ανάπτυξη του επιπρόσθετου στοιχείου. Στο επόμενο κεφάλαιο παρουσιάζουμε τις βασικές αρχές του Blockchain καθώς και διάφορες έννοιες που το περιγράφουν. Γίνεται αναφορά στις ιδιότητες του, στους διάφορους τύπους του καθώς και στα οφέλη που προσφέρει.

Στο κεφάλαιο 4, γίνεται ανάλυση μιας διαδικασίας βαθμολόγησης μαθημάτων σε ένα πανεπιστήμιο. Έτσι εξηγούνται τα σημεία-κίνητρα αυτής της εργασίας και αναλύονται οι τεχνικές λεπτομέρειες, η δομή και αρχιτεκτονική του εργαλείου που θα αναπτύξουμε. Ακόμα, μέσω της μοντελοποίησης των διάφορων περιπτώσεων χρήσης εξάγονται με ακρίβεια οι απαιτήσεις για την ανάπτυξη λογισμικού που θα γίνει στα πλαίσια της εργασίας. Η μοντελοποίηση γίνεται ακολουθώντας αυστηρά τη μεθοδολογία της τεκμηρίωσης κατά την ανάπτυξη λογισμικού.

Στο κεφάλαιο 5, θα συζητηθούν εκτενώς τα βήματα που ακολουθήθηκαν για την υλοποίηση του εργαλείου που παρουσιάστηκε στο κεφάλαιο 4. Αναφέρονται όλες οι ρυθμίσεις και εγκαταστάσεις που έγιναν για την ανάπτυξη, όλες οι τεχνολογίες που χρησιμοποιήθηκαν ενώ υπάρχουν και επεξηγήσεις με κώδικα όπου κρίνεται σημαντικό. Επίσης συζητούνται μερικές σχεδιαστικές επιλογές κατά την ανάπτυξη, ενώ αναλύεται η «σύνδεση» μεταξύ των συμβαλλομένων μερών που οδήγησε στο τελικό παραδοτέο.

Στο κεφάλαιο 6, γίνεται μια παρουσίαση του plugin που αναπτύχθηκε. Υπάρχουν screenshots και επεξηγήσεις για όλες τις πιθανές περιπτώσεις χρήσης. Η παρουσίαση γίνεται χρησιμοποιώντας μια εγκατάσταση του Moodle με δεδομένα φοιτητών και μαθημάτων που συντάχθηκαν μόνο για τους σκοπούς της εργασίας.

Τέλος, στο κεφάλαιο 7 γίνεται αναφορά στα επόμενα βήματα που θα μπορούσαν να ακολουθήσουν αυτή την εργασία για να πλαισιωθεί περαιτέρω η διαδικασία βαθμολόγησης καθώς και αναλύονται τα συμπεράσματα αυτής της εργασίας.

Κεφάλαιο 2 Η πλατφόρμα του Moodle

Το κεφάλαιο αυτό παρουσιάζει τη βασική ορολογία αλλά και την φιλοσοφία του Moodle. Ο σκοπός του κεφαλαίου είναι ο αναγνώστης να έχει κατά νου τα βασικά συστατικά της πλατφόρμας για να μπορεί να κατανοήσει το πλαίσιο ανάπτυξης της εργασίας.

2.1 Τι είναι το Moodle

Το Moodle είναι μια πλατφόρμα διαχείρισης μάθησης (LMS – Learning Management System). Παρέχει δυνατότητες εξ' ολοκλήρου διεξαγωγής ασύγχρονων μαθημάτων αλλά χρησιμοποιείται και συμπληρωματικά για την διαχείριση σύγχρονων και δια-ζώσης μαθημάτων. Είναι ένα λογισμικό ανοιχτού κώδικα και προσφέρεται δωρεάν. Είναι πλήρως επεκτάσιμο και παρέχει με αυτόν τον τρόπο στους χρήστες του την δυνατότητα να δημιουργήσουν ένα ισχυρό και ευέλικτο εικονικό περιβάλλον εκπαίδευσης. Επιτρέπει τη διαχείριση πολλών μαθημάτων ταυτόχρονα στα πλαίσια ενός πανεπιστημίου προσφέροντας στους διαχειριστές, καθηγητές και φοιτητές πολλές δυνατότητες παραμετροποίησης. [1]

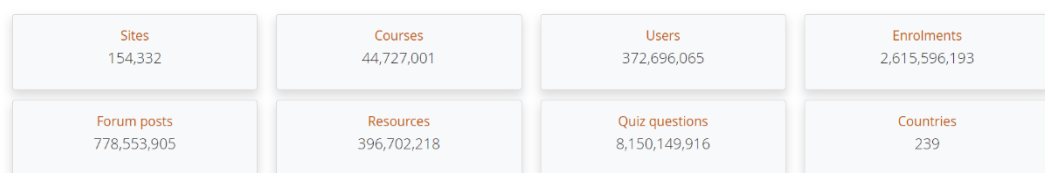
Το όνομά του είναι το ακρώνυμο του Modular Object-Oriented Dynamic Learning Environment (Αρθρωτό Αντικειμενοστραφές Δυναμικό Μαθησιακό Περιβάλλον). Μέσω της ονομασίας, η πλατφόρμα του Moodle μας υποδεικνύει και την γενικότερη φιλοσοφία της. Πιο συγκεκριμένα, η λέξη Modular εκφράζει την δομή της πλατφόρμας, που αποτελείται από αυτοτελή τμήματα κώδικα(modules) και καθένα από αυτά επιτελεί συγκεκριμένες λειτουργίες. Οι λέξεις Object και Oriented υποδεικνύουν ότι πρόκειται για μια πλατφόρμα που έχει χτιστεί με βάση τις αρχές του αντικειμενοστραφούς προγραμματισμού. Παρέχει στο χρήστη μια σειρά από επαναχρησιμοποιήσιμες και αφηρημένες λειτουργίες και του δίνει τη δυνατότητα να δημιουργήσει μια εξατομικευμένη εμπειρία μάθησης υποστηρίζοντας έτσι όχι μόνο την διαχείριση της μάθησης αλλά και της διδασκαλίας. Η λέξη Dynamic εκφράζει την δυναμική δομή της πλατφόρμας που συνεχώς επεκτείνεται και προσφέρει καινούρια λειτουργικότητα στους χρήστες. Αυτό είναι εφικτό μέσω της μεγάλης κοινότητας προγραμματιστών που δουλεύουν πάνω στην πλατφόρμα αναπτύσσοντας συνεχώς νέα plugin.

Η βασική ειδοποιός διαφορά του Moodle από τα περισσότερα αντίστοιχα συστήματα λοιπόν είναι ότι προσφέρει ένα περιβάλλον για τους εκπαιδευτικούς να δημιουργήσουν κύκλους μαθημάτων, με δραστηριότητες που απαιτούν αλληλεπίδραση ενώ δίνει χώρο και στους εκπαιδευόμενους να συμβάλλουν στην διαμόρφωση της εκπαιδευτικής διαδικασίας με πολλούς τρόπους. Ακόμα σημαντική διαφορά αποτελεί ότι πρόκειται για λογισμικό ανοιχτού κώδικα(open source). Αυτό πρακτικά σημαίνει

ότι οι χρήστες έχουν πρόσβαση στον πηγαίο κώδικα της εφαρμογής και ο καθένας μπορεί να μεταφορτώσει και να χρησιμοποιήσει το Moodle δωρεάν. Αυτό ευθυγραμμίζεται με τις ακαδημαϊκές αρχές της ελεύθερης διανομής της γνώσης και της αξιολόγησης.

Έχοντας λοιπόν αναφέρει συνοπτικά, μερικά από τα πλεονεκτήματα που φέρνει το Moodle, γίνεται κατανοητό πως πρόκειται για μια πολύ διαδεδομένη πλατφόρμα που χρησιμοποιείται παγκοσμίως. Ενδεικτικά όπως φαίνεται και στις παρακάτω εικόνες, το Moodle απαριθμεί πάνω από 360 εκατομμύρια χρήστες σε 241 χώρες ενώ υπάρχουν πάνω από 150 χιλιάδες εγγεγραμμένες ιστοσελίδες εκπαιδευτικών ιδρυμάτων στην πλατφόρμα.[2]

Statistics



Εικόνα 1 Στατιστικά του Moodle-LMS

2.2 Η δομή του Moodle

Η αρχική σελίδα του Moodle, η σελίδα που αντικρίζει κανείς όταν συνδέεται στο site για πρώτη φορά, περιέχει εξατομικευμένες πληροφορίες για την τρέχουσα εγκατάσταση και είναι σε μεγάλο βαθμό παραμετροποιήσιμη. Στη συνέχεια, το πως θα συνδέεται ο κάθε χρήστης στο site εξαρτάται και πάλι από την κάθε εγκατάσταση, μπορεί να τους έχουν δοθεί στοιχεία σύνδεσης ή να μπορούν δημιουργήσουν μόνοι τους το λογαριασμό τους ή να συνδεθούν μέσω κάποιου άλλου συστήματος. Π.χ. στο ΕΜΠ, γίνεται χρήση της υπηρεσίας LDAP.[3]

Αφού ο χρήστης εισέλθει στο Moodle μπορεί να περιηγηθεί στα διαθέσιμα μαθήματα της συγκεκριμένης εγκατάστασης. Και πάλι, το ποια μαθήματα είναι ορατά και το πως κάποιος χρήστης μπορεί να εγγραφεί στο μάθημα είναι στην επιλογή του διαχειριστή της συγκεκριμένης εγκατάστασης και τον καθηγητή που έχει οργανώσει το μάθημα. Η δομή του Moodle είναι οργανωμένη γύρω από τα μαθήματα. Στις σελίδες των μαθημάτων εντός της πλατφόρμας οι καθηγητές μπορούν να παρουσιάσουν το εκπαιδευτικό υλικό τους στους μαθητές. Κάθε μάθημα μπορεί να παρουσιάζει την πληροφορία με πολλούς διαφορετικούς τρόπους, χωρίζοντας το μάθημα σε τομείς, προσθέτοντάς κείμενο, εικόνες αλλά και βίντεο. Πιο συγκεκριμένα υπάρχουν τρία βασικά μέρη ενός μαθήματος στο Moodle:

1. Μπλοκ (Block): Τα Μπλοκ είναι αντικείμενα που τοποθετούνται υποστηρικτικά συνήθως στα πλάγια οποιαδήποτε σελίδας του Moodle. Παρέχουν επιπλέον πληροφορία στο χρήστη πέρα από το βασικό περιεχόμενο του μαθήματος. Μερικά παραδείγματα από Μπλοκ είναι το “Online Users” που πληροφορεί για το πόσοι χρήστες είναι συνδεδεμένοι εκείνη τη στιγμή, ή το “Search forums” με το οποίο οι χρήστες μπορούν να κάνουν μια γρήγορη αναζήτηση αποριών στο forum του μαθήματος. Υπάρχουν πολλά μπλοκ που έρχονται ως επιλογή με την βασική έκδοση του πυρήνα του Moodle αλλά και πολλά επιπλέον plugin που εξυπηρετούν ως blocks και έχουν αναπτυχθεί από τρίτους. [4]
2. Resources (πηγές πληροφοριών): Είναι τα αντικείμενα που χρησιμοποιεί ο καθηγητής για να υποστηρίξει τη μάθηση, το εκπαιδευτικό υλικό. Το Moodle υποστηρίζει πολλά διαφορετικά ήδη πηγών όπως βιβλία, αρχεία και φακέλους για καλύτερη οργάνωση της πληροφορίας, πακέτα περιεχομένου IMS [5], σελίδες, κείμενο, πολυμέσα όπως εικόνες ,βίντεο αλλά και υπερσυνδέσμους για άλλες σελίδες.[6]
3. Δραστηριότητες(activities): Οι δραστηριότητες στα μαθήματα του Moodle είναι μια γενική ορολογία που καλύπτει μεγάλο εύρος λειτουργικότητων σε ένα μάθημα. Συνήθως μια δραστηριότητα είναι κάτι μέσω του οποίου οι φοιτητές θα αλληλοεπιδρούν με άλλους φοιτητές ή με τους καθηγητές. Σε αντιπαράθεση με τις πηγές που αναπτύξαμε παραπάνω, οι πηγές είναι κάτι στατικό και ανεξάρτητο από το φοιτητή ορίζονται μόνο από τους καθηγητές του μαθήματος, ενώ οι δραστηριότητες είναι κάτι στο οποίο ο φοιτητής συμμετέχει άμεσα. Υπάρχουν 15 διαφορετικά είδη δραστηριοτήτων στην βασική έκδοση του Moodle, ενώ υπάρχουν και πολλά άλλα που μπορούν να εγκατασταθούν μεμονωμένα ως plugin. Μια κλάση δραστηριοτήτων που θα ξανά αναφέρουμε σε αυτή την εργασία είναι οι βαθμολογούμενες δραστηριότητες(grading activities). Ο καθηγητής μπορεί να δημιουργήσει στο μάθημα του διάφορες δραστηριότητες εξέτασης-βαθμολόγησης όπως ερωτήσεις πολλαπλής επιλογής, ερωτήσεις ανοιχτού τύπου κ.α. Επίσης υπάρχει δυνατότητα στους καθηγητές να αξιολογήσουν την δουλειά των φοιτητών, να αφήσουν σχόλια και εν τέλη να εξάγουν την τελική τους βαθμολογία στο μάθημα.

Τέλος, το Moodle δίνει τη δυνατότητα να δημιουργήσουμε κατηγορίες με σκοπό να διαχωρίσουμε τα μαθήματα ανά θεματολογία ή ακόμα και ανά σχολή. Είναι λοιπόν φανερό ότι το Moodle μπορεί να υποστηρίξει με αυτό τον τρόπο πανεπιστημιακά ιδρύματα, τα οποία αποτελούνται από πολλές σχολές και κάθε μια εξ' αυτών αποτελείται από πολλά διαφορετικά μαθήματα.

2.3 Χρήστες, ρόλοι και δικαιώματα στο Moodle

Το Moodle παρέχει μηχανισμό ανάληψης ρόλου ανά χρήστη ώστε να καθοριστεί τι μπορεί να κάνει ο κάθε χρήστης μέσα στο σύστημα και να είναι πιο σαφές και εύκολο να οριστούν τα δικαιώματα. Οι ρόλοι αυτοί μπορούν να καθοριστούν σε διάφορα επίπεδα. Υπάρχει για παράδειγμα ο admin ρόλος που αφορά όλο το site και ο χρήστης μπορεί να διαχειριστεί όλα τα μαθήματα και τους χρήστες. Υπάρχουν όμως και πιο εξειδικευμένοι ρόλοι όπως αυτός του course teacher οι οποίοι καθορίζονται σε επίπεδο μαθήματος και έτσι ένας τέτοιος μπορεί να διαχειριστεί μόνο το συγκεκριμένο μάθημα. Ο teacher έχει συνήθως όλα τα δικαιώματα πάνω στο μάθημα και διαχειρίζεται όλα τα μέρη του που αναφέραμε παραπάνω. Ένας ακόμα πολύ συνηθισμένος ρόλος είναι αυτός του student.

Η έννοια του ρόλου στο Moodle δεν θα έβγαζε πολύ νόημα χωρίς την έννοια του capability. Στο Moodle τα capabilities αντιστοιχούν σε μια πιθανή συμπεριφορά κάποιου χρήστη. Τα capabilities ανατίθενται στους χρήστες του Moodle μέσω των ρόλων. Για παράδειγμα το capability "moodle/grade:viewall" ορίζει ποιος χρήστης μπορεί να δει τους βαθμούς ενός συγκεκριμένου μαθήματος. Ουσιαστικά ένας ρόλος στο Moodle είναι ένα σύνολο από capabilities, που ορίζουν τι ακριβώς μπορεί να κάνουν οι χρήστες που έχουν αυτό το ρόλο.

Μια ακόμα σημαντική έννοια που ορίζει ποιος χρήστης έχει ποιους ρόλους είναι το context. Το context είναι ένας «χώρος» στο Moodle όπου γίνεται η ανάθεση ρόλων. Είναι οργανωμένο ιεραρχικά, σε διάφορα επίπεδα, όπως για παράδειγμα system-context, course category context, course context. Έτσι ένας χρήστης μπορεί να λάβει τον ρόλο "teacher" σε system-context που θα σημαίνει ότι έχει το ρόλο αυτό σε όλα τα μαθήματα μέσα στο site, ή από την άλλη να το λάβει σε ένα συγκεκριμένο course-context, οπότε έχει τα ανάλογα δικαιώματα μόνο σε αυτό το μάθημα και δεν σημαίνει ότι έχει τον ίδιο ρόλο στα άλλα. Ορίζοντας αυτές τις 3 βασικές έννοιες (role, capability, context) το Moodle μας δίνει τη δυνατότητα να έχουμε μεγαλύτερο έλεγχο πάνω στο τι μπορεί να κάνει και ποιος χρήστης. [7][8]

2.4 Διαδικασία ανάπτυξης στο Moodle

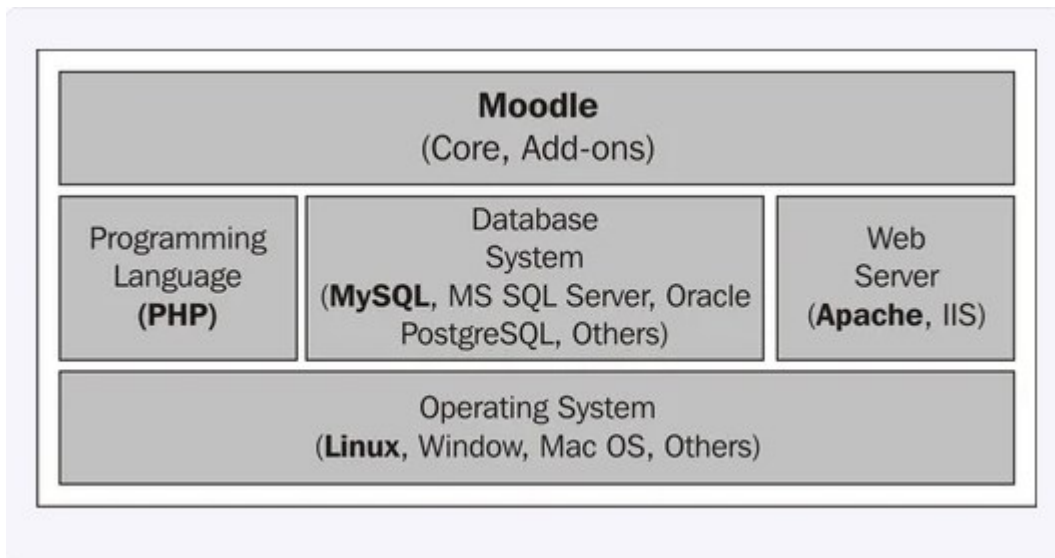
2.4.1 Ελάχιστες απαιτήσεις για την εγκατάσταση του Moodle

Τα προτεινόμενα πακέτα λογισμικού για την εγκατάσταση του Moodle ορίζονται από το γνωστό ως "LAMP" stack. Αυτό από τα αρχικά του αναφέρεται σε χρήση Linux λειτουργικού συστήματος, έναν Apache server, μια βάση δεδομένων MySQL ή MariaDB και php ως γλώσσα προγραμματισμού. Αυτός είναι ο πιο συνηθισμένος συνδυασμός για την εγκατάσταση του Moodle

αλλά ανάλογα με την έκδοση μπορούν να υπάρχουν και άλλες επιλογές. Από άποψη Hardware οι ελάχιστες απαιτήσεις αναφέρουν ότι για να τρέξει η ιστοσελίδα χρειάζεται περίπου 200 MB για την αποθήκευση του κώδικα επιπλέον στο υλικό που θα κουβαλά κάθε ιστοσελίδα. Χρειάζεται ένας επεξεργαστής τουλάχιστον στο 1 Ghz και 512MB-1GB μνήμη RAM. Ανάλογα με το μέγεθος της εγκατάστασης οι διαχειριστές είναι καλό να συμβουλευτούν την τεκμηρίωση του Moodle για περισσότερες πληροφορίες. [9]

Για να συντηρηθεί και να εξελιχθεί μια τέτοια πλατφόρμα τέτοιου εύρους αλλά διατηρώντας και εξασφαλίζοντας την ποιότητα των υπηρεσιών κρίθηκε πολύ σημαντικό να σχεδιαστεί μια διαδικασία ανάπτυξης με ξεκάθαρα βήματα για τους προγραμματιστές. Η πλατφόρμα παρέχει εκτεταμένη τεκμηρίωση(documentation) και οδηγίες για να μπορούν οι προγραμματιστές να αναπτύξουν επιπρόσθετα στοιχεία υψηλής ποιότητας που να διασφαλίζουν τα πρότυπα ποιότητας και να είναι εύκολο να συντηρηθούν και αναπτυχθούν από άλλους στο μέλλον.

Η διαδικασία ανάπτυξης περιέχει διάφορα βήματα. Αρχικά ο προγραμματιστής πρέπει να διαλέξει τι τύπου στοιχείο θέλει να αναπτύξει από τις διάφορες επιλογές που δίνει το Moodle. Ο τύπος καθορίζει το που θα τοποθετηθούν τα αρχεία του πηγαίου κώδικα του plugin έτσι ώστε να διατηρείται η λογική συνέπεια ανάμεσα στα αρχεία που βρίσκονται σε διπλανούς φακέλους. Η επιλογή τύπου εξαρτάται από την λειτουργικότητα που θέλουμε να προσφέρουμε το νέο στοιχείο. Αφού είναι ξεκάθαρες οι λειτουργίες ο προγραμματιστής πρέπει να κάνει την ανάπτυξη τους χρησιμοποιώντας PHP για τις server-side λειτουργίες του στοιχείου. Σε αυτό το σημείο είναι σημαντικό να μελετηθούν εκτενώς όλες οι προ υπάρχουσες λειτουργίες και να χρησιμοποιηθούν για να αποφύγουμε τον επαναλαμβανόμενο κώδικα. Όπου αυτό δεν είναι δυνατό πρέπει ο προγραμματιστής που γράφει εξολοκλήρου νέο κώδικα να έχει πάντα υπόψιν του τις βασικές αρχές του αντικειμενοστραφή προγραμματισμού που είναι και οι βασικές αρχές ανάπτυξης στο Moodle. Στις νεότερες εκδόσεις του Moodle παρέχεται η δυνατότητα ανάπτυξης με JavaScript για τις client-side λειτουργίες με σκοπό την δημιουργία πιο διαδραστικών και δυναμικών επιπρόσθετων στοιχείων. Επίσης παρέχεται και μια διεπαφή προγραμματισμού(Javascript API) για να μπορεί να επικοινωνεί το επιπρόσθετο στοιχείο με τον πυρήνα του Moodle.



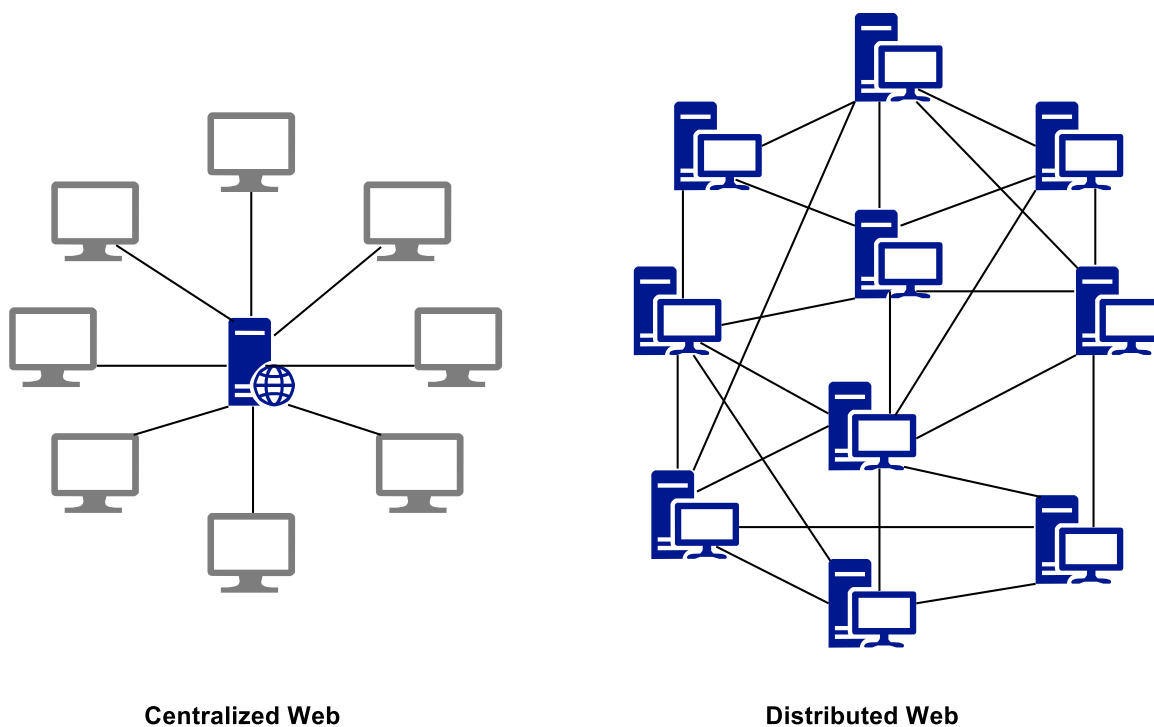
Εικόνα 2 Αρχιτεκτονική Moodle-LAMP stack

Κεφάλαιο 3 Εισαγωγή στο Blockchain

Σκοπός του κεφαλαίου είναι ο αναγνώστης να καταλάβει την βασική ορολογία γύρω από το Blockchain αλλά και τα βασικά οφέλη του ώστε να είναι σε θέση να αντιληφθεί τα κίνητρα και την αξία αυτής της εργασίας.

3.1 Τι είναι το Blockchain

Ονομάζουμε Blockchain ένα κατανεμημένο σύστημα που αποτελείται από κατάλληλα συνδεδεμένους κόμβους. Η βασική αξία του Blockchain είναι ότι μπορεί να αποθηκευτεί στους κόμβους του πληροφορία η οποία είναι σχεδόν αδύνατο να αλλάξει. Η βασική διαφορά του από τα άλλα, μέχρι τώρα γνωστά, συστήματα αποθήκευσης και διαχείρισης δεδομένων είναι ότι σε ένα τέτοιο σύστημα δεν υπάρχει μια κεντρική αρχή που έχει εξουσία πάνω από το σύστημα, αλλά αυτό χρησιμοποιεί μηχανισμούς συναίνεσης ανάμεσα στους κόμβους του με σκοπό να επαληθεύονται οι πληροφορίες που υπάρχουν στο δίκτυο και να εξασφαλίζεται η ακεραιότητά τους. [10]



Εικόνα 3 Ιστός με κεντρική αρχή και αποκεντρωμένος ιστός

3.2 Δομικά μέρη (Node, Block)

Το Blockchain όπως λέει και το όνομα του είναι ουσιαστικά μια αλυσίδα από Blocks. Τα Blocks «κουβαλούν» όλη την πληροφορία. Γενικά η ακριβής δομή τους διαφέρει ανάμεσα στις διάφορες υλοποιήσεις του Blockchain αλλά εδώ θα αναφέρουμε τις βασικά στοιχεία ενός Block. Πέρα από τα δεδομένα των συναλλαγών το block περιέχει το block header. Αυτό είναι αποτέλεσμα μιας συνάρτησης κατακερματισμού (hash function) η οποία παίρνει ως είσοδο διάφορα δεδομένα όπως το header του προηγούμενου block στην αλυσίδα, τον αριθμό έκδοσης του block, τα δεδομένα συναλλαγών αλλά και ένα τυχαίο string που ονομάζεται nonce. [11]

Ένας άλλος σημαντικός όρος που αξίζει να συζητηθεί είναι ο κόμβος. Όπως αναφέρθηκε πρόκειται για ένα κατακερματισμένο σύστημα συνεπώς αποτελείται από διάφορους κόμβους. Ουσιαστικά κάθε συμμετέχων στο δίκτυο, κάθε ηλεκτρονική συσκευή που συνδέεται, αποτελεί έναν κόμβο του δικτύου και διατηρεί ένα αντίγραφο του κοινόχρηστου αρχείου καταγραφής(public ledger) όπου είναι αποθηκευμένες όλες οι συναλλαγές. Ο αριθμός των κόμβων που συμμετέχουν είναι ανάλογος της ακεραιότητας και της ασφάλειας ενός τέτοιου δικτύου.

3.3 Πώς λειτουργεί

Όταν ένας χρήστης θέλει να προσθέσει δεδομένα στο δίκτυο ουσιαστικά ζητά μια συναλλαγή από το υπόλοιπο δίκτυο, μέσω του κόμβου που του αντιστοιχεί. Αυτή η συναλλαγή μεταδίδεται σε όλους τους κόμβους του κατακερματισμένου δικτύου και αυτοί με τη σειρά τους είναι υπεύθυνοι να επαληθεύσουν την συναλλαγή ακολουθώντας τους προκαθορισμένους κανόνες του δικτύου. Αφού ένας κόμβος λάβει μια συναλλαγή, θα προσπαθήσει να δημιουργήσει ένα καινούριο block σύμφωνα με τον αλγόριθμο συναίνεσης που έχει συμφωνηθεί. Μόλις δημιουργηθεί ένα έγκυρο block θα μεταδοθεί εκ νέου και θα προστεθεί στην αλυσίδα των block. Από εκεί και μετά, αν θεωρητικά κανείς είχε την εξουσία να αλλάξει τα blocks ενός κόμβου, ο αλγόριθμος συναίνεσης θα απέτρεπε σε αυτό τον κόμβο να επιβάλει την αλυσίδα του στους άλλους κόμβους, καθιστώντας το σύστημα πλήρως αποκεντρωμένο και την πληροφορία αμετάβλητη.

3.4 Τύποι δικτύων Blockchain

Παρόλο που όπως αναφέρθηκε, το Blockchain βασίζεται τα οφέλη του στην αποκεντρωμένη του αρχιτεκτονική, στον πραγματικό κόσμο υπάρχουν υλοποιήσεις του που δεν είναι πλήρως

αποκεντρωμένες. Πιο συγκεκριμένα υπάρχουν 4 τύποι δικτύων Blockchain ο κάθε ένας με διαφορετικά πλεονεκτήματα και μειονεκτήματα. [12]

Public Blockchain

Είναι ο πρώτος ιστορικά τύπος Blockchain που βγήκε στην αγορά. Πρόκειται για την πλήρως αποκεντρωμένη λύση, όπου οποιοσδήποτε χρήστης μπορεί να συμμετάσχει στο δίκτυο, να γίνει δηλαδή κόμβος και να επιβεβαιώσει συναλλαγές. Αυτό επιτρέπει μεγάλη συμμετοχή στο δίκτυο και ως συνέπεια της μεγάλης ασφάλεια και ακεραιότητα των συναλλαγών. Αυτός ο τύπος Blockchain είναι permissionless. Έτσι λοιπόν πιο συγκεκριμένα τα πλεονεκτήματα αυτού του δημόσιου Blockchain είναι ότι είναι πλήρως αποκεντρωμένο και ασφαλές και είναι τελείως ανεξάρτητο από τους οργανισμούς που το υποστηρίζουν και το χρησιμοποιούν, αφού αυτό θα μπορεί να λειτουργεί αρκεί να υπάρχουν συνδεδεμένοι κόμβοι οπουδήποτε στο διαδίκτυο. Στα μειονεκτήματα του συγκαταλέγεται το υψηλό κόστος συναλλαγών, σε χρόνο και σε ενέργεια αφού αυτά είναι ανάλογα του πλήθους των κόμβων. Επίσης, μιας και κανείς δεν μπορεί να απαγορεύσει η να περιορίσει την είσοδο κακόβουλων χρηστών, αν αυτοί κατάφεραν να έχουν πάνω από το 50% της υπολογιστικής δύναμης του δικτύου τότε θα μπορούσαν να μεταβάλλουν τα περιεχόμενα του.

Private Blockchain

Σε αυτό τον τύπο Blockchain μόνο συγκεκριμένοι κόμβοι μπορούν να συμμετάσχουν στο δίκτυο. Οι κόμβοι που έχουν δικαίωμα να συμμετάσχουν καθορίζονται από μια οντότητα. Κατά τα υπόλοιπα ο τρόπος λειτουργίας δεν διαφέρει πολύ από το public Blockchain. Αποκαλούνται και permissioned blockchains. Στα πλεονεκτήματα είναι ότι η κεντρική αρχή μπορεί να ρυθμίσει τα δικαιώματα των συμμετεχόντων στο δίκτυο και να απορρίψει εξωτερικούς, μη επιθυμητούς συμμετέχοντες. Ακόμη, λόγω του μικρότερου αριθμού συμμετεχόντων, οι συναλλαγές είναι πολύ γρήγορες. Στα μειονεκτήματα είναι ότι αλλοιώνεται η κύρια φιλοσοφία του Blockchain που είναι η αποκέντρωση, αφού υπάρχει μια οντότητα που έχει την εξουσία. Επίσης συνήθως ο κώδικας του είναι και αυτός μυστικός με συνέπεια λιγότερη διαφάνεια.

Hybrid Blockchain

Αυτός ο τύπος blockchain συνδυάζει χαρακτηριστικά από τους δύο προηγούμενους τύπους. Δίνει τη δυνατότητα, να στήσει κανείς ένα private blockchain δίπλα σε ένα public ώστε να υπάρχει εποπτεία και ρύθμιση δικαιωμάτων σχετικά με το ποια δεδομένα θα είναι δημόσια και ποια θα είναι μυστικά. Ένα από τα πλεονεκτήματα του είναι ότι κακόβουλοι χρήστες δεν μπορούν να καταλάβουν με κάποιο τρόπο πάνω από 50% της υπολογιστικής δύναμης του συστήματος, κάνοντας το πιο ασφαλές από ένα public, χωρίς όμως να απαγορεύει εξολοκλήρου την συμμετοχή εξωτερικών χρηστών. Λόγω

του, σε κάποιο βαθμό ελεγχόμενου, αριθμού συμμετεχόντων οι συναλλαγές είναι πιο γρήγορες και φτηνές. Στα αρνητικά και πάλι, όπως και στο private αυτός ο τύπος δεν διασφαλίζει την διαφάνεια, εφόσον υπάρχει και πάλι μια κεντρική αρχή.

Consortium Blockchain

Ο τέταρτος τύπος blockchain, ονομάζεται αλλιώς federated blockchain. Μοιάζει με το hybrid blockchain στο ότι έχει χαρακτηριστικά και των δύο βασικών τύπων blockchain (public, private). Ουσιαστικά ένα consortium blockchain είναι ένα private blockchain στο οποίο η κεντρική αρχή δεν είναι μόνο μια οντότητα, αλλά ένα group συμμετεχόντων. Είναι πιο αποκεντρωμένα από ένα private blockchain και κατά συνέπεια πιο ασφαλή. Παράλληλα εφόσον πάλι υπάρχει έλεγχος ως προς τους συμμετέχοντες, οι συναλλαγές είναι πιο γρήγορες από ένα public. Όπως και στις δυο παραπάνω περιπτώσεις, εφόσον υπάρχει κάποια μορφή εξουσίας, χάνεται η διαφάνεια ενώ υπάρχει κίνδυνος εφόσον κάποιος κόμβος μέλος παραβιαστεί.

3.5 Αλγόριθμοι συναίνεσης (consensus)

Οι αλγόριθμοι συναίνεσης στην επιστήμη υπολογιστών, χρησιμοποιούνται για να επιτευχθεί «συμφωνία» ανάμεσα σε κατανεμημένους κόμβους σε ένα σύστημα, σε ότι αφορά τα δεδομένα που κουβαλούν. Τέτοιοι αλγόριθμοι έχουν σχεδιαστεί για να προσφέρουν αξιοπιστία και συνέπεια σε συστήματα με πολλούς συνεργαζόμενους κόμβους. Είναι ζωτικής σημασίας γιατί ουσιαστικά εννοχηστρώνουν την συνεργασία ανάμεσα σε πολλούς κατανεμημένους κόμβους με σκοπό να δουλεύουν σαν μια συνδεδεμένη ομάδα. Τέτοιοι αλγόριθμοι υποστηρίζουν τα συστήματα Blockchain που αναφέρθηκαν παραπάνω. Είναι αναγκαίος ένας τέτοιος αλγόριθμος για κάθε δίκτυο Blockchain ώστε να μπορούν οι κόμβοι να έρθουν σε συμφωνία σχετικά με τα δεδομένα που εισέρχονται στο δίκτυο χωρίς να χρειάζεται να επέμβει μια κεντρική αρχή. Παρακάτω θα αναλύσουμε τους πιο γνωστούς τέτοιους αλγόριθμους στον κόσμο του Blockchain. [13]

Proof of Work

Ο αλγόριθμος αυτός βασίζεται στους miners. Οι miners είναι ουσιαστικά κόμβοι που συμμετέχουν στο σύστημα. Είναι στην επιλογή του κάθε χρήστη-κόμβου αν αυτός επιθυμεί να είναι μόνο χρήστης που στέλνει συναλλαγές ή και miner. Σκοπός των miner είναι να δημιουργήσουν πριν από όλους τους υπόλοιπους το καινούριο block. Για να γίνει αυτό πρέπει να λύσουν ένα πολύ σύνθετο υπολογιστικό πρόβλημα για την εύρεση του nonce. Μόλις κάποιος miner λύσει αυτό το πρόβλημα και μεταδώσει το καινούριο block στους άλλους κόμβους αυτοί θα πρέπει να το επικυρώσουν και να το

βάλουν στην αλυσίδα τους. Θεωρητικά αυτό το υπολογιστικό πρόβλημα είναι αρκετά τυχαίο με αποτέλεσμα να είναι απίθανο να το λύσει πολλές φορές συνεχόμενα ο ίδιος κόμβος και έτσι είναι αδύνατο ακόμα και αν κάποιος κόμβος παραβιαστεί, το σύστημα να χάσει την αξιοπιστία του. Η αδυναμία αυτού του αλγόριθμου προκύπτει σε περίπτωση που κάποιος αποκτήσει πάνω από το 50% της υπολογιστικής δύναμης του δικτύου, τότε θα έχει τη δυνατότητα να επιβάλλει στο σύστημα το «μολυσμένο» block. Επίσης σοβαρό μειονέκτημα αυτού του αλγορίθμου είναι η υψηλή κατανάλωση ενέργειας που απαιτείται για την υλοποίησή του σε ένα δίκτυο μεγάλης κλίμακας.

Proof of Stake

Σε αυτό τον αλγόριθμο τον ρόλο των miners αναλαμβάνουν οι validators. Δεν απαιτείται να λυθεί κάποιο σύνθετο υπολογιστικό πρόβλημα αλλά απαιτείται από τους validators να κάνουν stake τα tokens τους. Είναι σαν να στοιχηματίζουν τα tokens με σκοπό να κερδίσουν την εμπιστοσύνη των άλλων κόμβων και να δημιουργήσουν αυτοί το νέο block. Όσο περισσότερα τα tokens που γίνονται stake τόσο πιο πολλές οι πιθανότητες να επιλεγεί ο κόμβος για να δημιουργήσει το νέο block. Σε αυτό τον αλγόριθμο δεν έχουμε υψηλή κατανάλωση ενέργειας αλλά υπάρχει κίνδυνος σε περίπτωση που κάποιος χρήστης αποκτήσει πάνω από 50% των tokens να παραβιαστεί η ακεραιότητα του συστήματος.

Proof of Authority

Σύμφωνα με τον παραπάνω αλγόριθμο, υπάρχουν προεπιλεγμένοι κόμβοι οι οποίοι είναι υπεύθυνοι για την επικύρωση των συναλλαγών και την δημιουργία νέων block. Σε αντιστοιχία με τους άλλους αλγόριθμους ο κόμβος που θέλει να δημιουργήσει το νέο block παρουσιάζει την «ταυτότητα» του και όχι τα tokens που έχει κάνει stake, η την υπολογιστική του δύναμη. Αυτός ο αλγόριθμος είναι πολύ αποδοτικός αφού μόνο συγκεκριμένοι κόμβοι μπορούν δημιουργήσουν block και οι συναλλαγές γίνονται πολύ γρήγορα. Ακόμη εξουδετερώνει τον κίνδυνο της «51%» επίθεσης, αφού θα έπρεπε ο κακόβουλος χρήστης να είχε στον έλεγχο του πάνω από 50% των κόμβων που έχουν προεπιλεγεί ως συμμετέχοντες. Αυτός ο αλγόριθμος βρίσκει εφαρμογή κυρίως σε private ή permissioned Blockchains.

3.6 Έξυπνα συμβόλαια και συναλλαγές

Μια σημαντική επέκταση στην λειτουργικότητα που παρέχει ένα δίκτυο Blockchain, έρχεται με τα έξυπνα συμβόλαια. Ένα έξυπνο συμβόλαιο είναι ουσιαστικά ένα πρόγραμμα που είναι αποθηκευμένο σε ένα δίκτυο Blockchain και τρέχει όταν συντρέξουν κάποιες προκαθορισμένες

συνθήκες. Χρησιμοποιούνται για την εκτέλεση συγκεκριμένων συναλλαγών και την διεκπεραίωση συμφωνιών χωρίς την ανάγκη ύπαρξης κάποιας κεντρικής αρχής. Ουσιαστικά εξασφαλίζουν στα συνεργαζόμενα μέρη ότι μπορούν να επικοινωνούν σύμφωνα με τους αρχικά συμφωνημένους όρους, που έχουν αποτυπωθεί σε κώδικα και κανείς δεν έχει την εξουσία να τους αλλάξει αφού το συμβόλαιο είναι αποθηκευμένο και εκτελείται μέσα στο Blockchain. Σαν παράδειγμα μια περίπτωση χρήσης που θα μπορούσαμε να χρησιμοποιήσουμε έξυπνα συμβόλαια είναι για την αγορά προϊόντων από ένα ηλεκτρονικό κατάστημα. Ο χρήστης επιλέγει τα προϊόντα που θέλει να αγοράσει και πληρώνει ηλεκτρονικά. Το έξυπνο συμβόλαιο εκτελεί την συναλλαγή και αφού ολοκληρωθεί η πληρωμή το συμβόλαιο δρομολογεί την διαδικασία αποστολής των προϊόντων. Έτσι εξουδετερώνει την πιθανότητα εξαπάτησης, αφού δεν εμπλέκεται κάποιος ενδιάμεσος σε αντίθεση με ένα παραδοσιακό σύστημα που μια τέτοια διαδικασία θα βασιζόταν στην διεκπεραίωση από τον διαχειριστή της σελίδας.

Τα έξυπνα συμβόλαια εμφανίστηκαν για πρώτη φορά στο main δίκτυο του Ethereum. Στη συνέχεια υιοθετήθηκαν και από άλλα δίκτυα Blockchain και μπορούν πλέον να γραφτούν σε πολλές γλώσσες προγραμματισμού. Στο Ethereum η κύρια γλώσσα είναι η Solidity που είναι γλώσσα καθαρά για την ανάπτυξη έξυπνων συμβολαίων. Πλέον η ανάπτυξη συμβολαίων γίνεται και με γλώσσες όπως Rust, Vyper, JavaScript, Python, Golang. [14][15]

3.7 Οφέλη αλλά και μειονεκτήματα

Η τεχνολογία του Blockchain βρίσκει όλο και περισσότερες εφαρμογές στις μέρες μας, χάρη στα οφέλη που συνδέονται με τη χρήση τους σε κάποιο σύστημα. Ο βασικός πυλώνας του είναι η αποκέντρωση. Ένα σύστημα Blockchain μπορεί να λειτουργήσει χωρίς την κεντρική εποπτεία και διαχείριση από έναν εξωτερικό χρήστη, ενώ οι συμμετέχοντες στο δίκτυο επωφελούνται από φθηνότερες συναλλαγές αφού δεν χρειάζεται προμήθεια για τον μεσάζοντα. Κατά συνέπεια, υπάρχει πλήρης διαφάνεια σχετικά με τα δεδομένα που καταγράφονται στην αλυσίδα. Ακόμα δεδομένου ότι είναι εξαιρετικά δύσκολο για έναν κακόβουλο χρήστη να μεταβάλλει με κάποιο τρόπο τα δεδομένα που είναι αποθηκευμένα είναι ένα απόλυτα ασφαλές δίκτυο για αποθήκευση δεδομένων. Επίσης εφόσον υποστηρίζεται από κατάλληλα συνδεδεμένους καταναμημένους κόμβους είναι εγγυημένη η απρόσκοπτη λειτουργία αφού η αλυσίδα φιλοξενείται σε διαφορετικούς κόμβους που συνδέονται στο δίκτυο.

Προφανώς και αυτή η τεχνολογία έρχεται μαζί με τα αρνητικά της. Δεδομένου ότι οι συναλλαγές καταγράφονται σε όλους τους κόμβους, καταλαβαίνουμε ότι όσο αυξάνεται ο όγκος των συναλλαγών που είναι αποθηκευμένες, η αλυσίδα μεγαλώνει με μη γραμμικό τρόπο γεγονός που

περιορίζει σημαντικά τον όγκο δεδομένων που μπορούν να αποθηκευτούν. Επίσης είναι μια τεχνολογία που κοστίζει και χρονικά, αφού για κάθε μια εγγραφή γίνονται πολύ περισσότερες διαδικασίες σε σχέση με τα παραδοσιακά συστήματα. Τέλος, σε σχέση πάντα και με την συγκεκριμένη υλοποίηση μπορεί το σύστημα να είναι πολύ απαιτητικό σε ενέργεια όπως για παράδειγμα στα proof-of-work δίκτυα που εξηγήσαμε παραπάνω.

Κεφάλαιο 4 Ανάλυση του προβλήματος

4.1 Η διαδικασία βαθμολόγησης

Παρακάτω θα αναφερθούμε στη διαδικασία διαχείρισης των βαθμών για τα μαθήματα ενός πανεπιστημίου. Η ανάλυση αυτή αφορά ιδρύματα τα οποία λειτουργούν δια ζώσης και παράλληλα υποστηρίζεται η διεξαγωγή της διδασκαλίας στο Moodle όπου αναρτάται εκπαιδευτικό υλικό ενώ διεξάγονται στην πλατφόρμα μερικές βαθμολογούμενες δραστηριότητες όπως για παράδειγμα υποβολή σειρών ασκήσεων ή μερικά κουίζ πολλαπλής επιλογής. Θα θεωρήσουμε ότι όλες οι βαθμολογούμενες δραστηριότητες είτε διεξάγονται στο Moodle, είτε καταγράφονται εκεί από τους καθηγητές του μαθήματος, για την διαμόρφωση της τελικής βαθμολογίας μέσω της πλατφόρμας. Η διαδικασία που θα αναλύσουμε παρακάτω είναι εμπνευσμένη από την διαδικασία που υφίσταται σε αρκετά μαθήματα του Ε.Μ.Πολυτεχνείου, αλλά είναι αρκετά γενική ώστε να μπορεί να εφαρμοστεί σε οποιαδήποτε εκπαιδευτική δράσει που χρησιμοποιεί το Moodle.

4.1.1 Κρίσιμα χαρακτηριστικά

Αρχικά πριν αναλύσουμε τα στάδια της διαδικασίας κρίνεται σημαντικό να σταθούμε στα χαρακτηριστικά που συμβάλλουν στην ποιότητα και την εγκυρότητα μιας τέτοιας διαδικασίας:

Ιδιωτικότητα: Η ιδιωτικότητα είναι απαραίτητο χαρακτηριστικό μιας τέτοιας διαδικασίας καθώς οι βαθμοί αποτελούν προσωπικά και ευαίσθητα δεδομένα των φοιτητών. Συνεπώς μόνο συγκεκριμένοι και εξουσιοδοτημένοι χρήστες θα πρέπει να έχουν την δυνατότητα πρόσβασης για προβολή και επεξεργασία σε τέτοια δεδομένα. Είναι σημαντικό λοιπόν τα πανεπιστήμια να επιστρατεύουν τα όλο και πιο ανεπτυγμένα τεχνολογικά μέσα για να διασφαλίσουν την ιδιωτικότητα των βαθμών και γενικότερα των δεδομένων των φοιτητών.

Ακεραιότητα: Οι βαθμοί αντικατοπτρίζουν την ποιότητα της δουλειάς του κάθε φοιτητή και το επίπεδο γνώσεων του σε κάθε μάθημα ή δραστηριότητα. Καταλαβαίνουμε λοιπόν ότι η ακεραιότητα των βαθμών είναι πολύ σημαντική σε προσωπικό επίπεδο για τον κάθε φοιτητή, σαν μέσο ανατροφοδότησης, για να ξέρει ότι βαθμολογείται δίκαια η δουλειά του και ανταμείβεται ή από την άλλη ότι θα πρέπει να δουλέψει περισσότερο. Επίσης οι βαθμοί ενός μαθήματος χαρακτηρίζουν και την ποιότητα της διδασκαλίας, την συμβατότητα της με τους φοιτητές, την συμμετοχή των φοιτητών στην διαδικασία. Συνεπώς καταλήγουμε ότι η ακεραιότητα των βαθμών είναι πολύ σημαντική και σε

επίπεδο ιδρύματος, καθώς εξασφαλίζει την εγκυρότητα μετρικών και δεικτών που αξιολογούν την ποιότητα και το επίπεδο της διδασκαλίας που παρέχεται.

Διαφάνεια: Η διαφάνεια στην διαδικασία διαχείρισης βαθμών διαμορφώνει ένα περιβάλλον αξιοκρατίας για τους φοιτητές, ώστε και αυτοί να νιώθουν ότι οι κόπιοι τους αποδίδουν και ότι όλοι παίρνουν αυτό που τους αξίζει. Παράλληλα η διαφάνεια, ενισχύει την εγκυρότητα της διαδικασίας και κατ' επέκταση την ακαδημαϊκή φήμη του εκπαιδευτικού ιδρύματος.

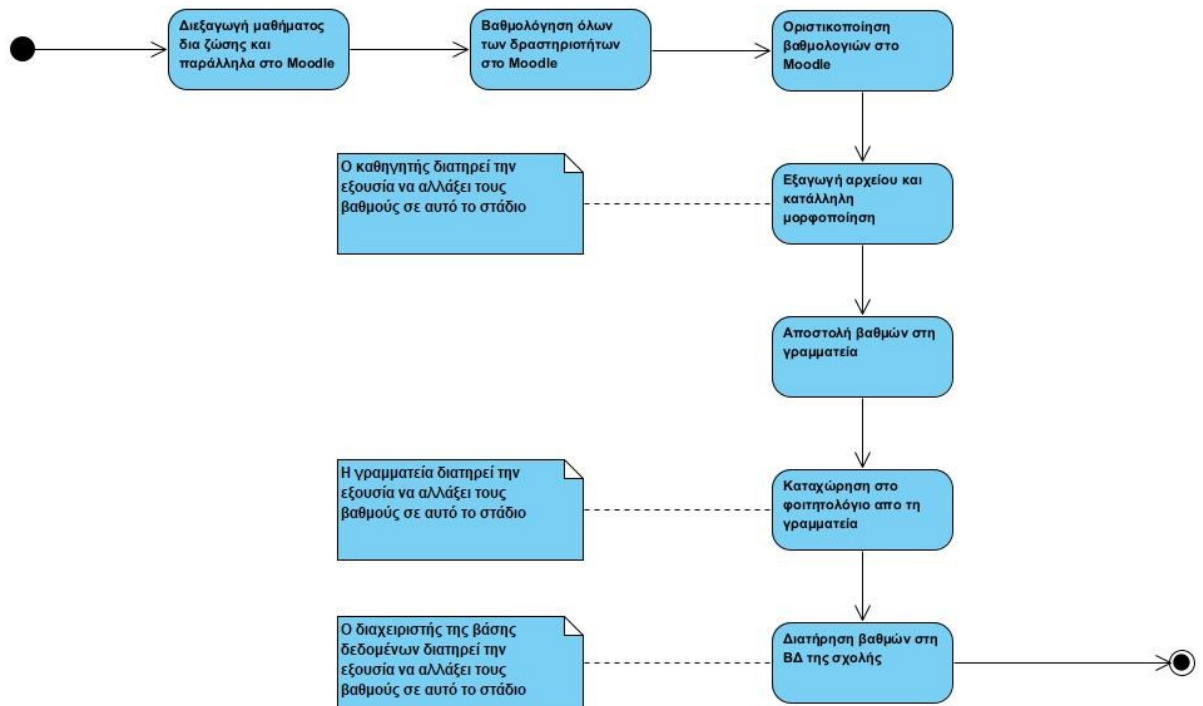
Ευκολία στη χρήση: Η διαδικασία διαχείρισης βαθμών είναι μια διαδικασία στην οποία συμμετέχει σχεδόν ολόκληρη η πανεπιστημιακή κοινότητα. Από καθηγητές και φοιτητές έως τους υπάλληλους της γραμματείας και τους διαχειριστές των υπολογιστικών συστημάτων κάθε σχολής. Είναι συνεπώς απαραίτητο για τα πανεπιστήμια να εκμεταλλεύονται συνεχώς τα όλο και περισσότερα αναπτυσσόμενα τεχνολογικά μέσα, για να υποστηρίξουν αυτή την διαδικασία. Η αυτοματοποίηση διαδικασιών για να εξαλειφθεί ο ανθρώπινος παράγοντας και να μειωθούν τα λάθη, καθώς και η αφαίρεση απαρχαιωμένων δραστηριοτήτων μπορεί να αυξήσει την παραγωγικότητα σε ένα πανεπιστήμιο κάνοντας την διαδικασία πιο γρήγορη και αξιόπιστη ενώ διευκολύνει την καθημερινότητα για όλα τα ενδιαφερόμενα μέρη.

Η παρακάτω ανάλυση λοιπόν καθώς και κάθε απόφαση κατά τον σχεδιασμό της λύσης αυτής της εργασίας γίνεται, έχοντας πάντα κατά νου αυτά τα βασικά χαρακτηριστικά και τη βελτίωσή τους. Αναλύοντας πάντως τα παραπάνω χαρακτηριστικά μπορούμε να δούμε εύκολα πως μια τέτοια διαδικασία επωφελείται από μια λύση που χρησιμοποιεί Blockchain. Μια τέτοια λύση κληρονομεί όλα τα χαρακτηριστικά που αναλύσαμε στο κεφάλαιο 3, όπως η ακεραιότητα των δεδομένων, η διαφάνεια των διαδικασιών και η αδυναμία να παραβιαστεί το σύστημα ενισχύοντας την ασφάλεια του.

4.1.2 Ανάλυση της διαδικασίας

Η διαδικασία διαχείρισης βαθμολογιών ξεκινά μόλις όλοι οι φοιτητές έχουν βαθμολογηθεί σε όλες τις δραστηριότητες του μαθήματος(π.χ. σειρές ασκήσεων, online αξιολόγηση πολλαπλής επιλογής στο Moodle, τελική εξέταση) και έχει υπολογιστεί στο Moodle η τελική βαθμολογία, σύμφωνα με τους κανόνες του μαθήματος. Ο καθηγητής τώρα θα πρέπει να εξάγει το αρχείο των βαθμολογιών από την πλατφόρμα και να το φέρει στην απαιτούμενη μορφή ώστε οι βαθμολογίες να υποβληθούν στην γραμματεία. Αφού το αρχείο βαθμών σταλεί στη γραμματεία, αυτή με τη σειρά της, θα πρέπει να εκχωρήσει τους βαθμούς στην βάση δεδομένων της σχολής με σκοπό να ενημερωθούν κατάλληλα οι

καρτέλες των φοιτητών. Από εκεί και πέρα πρόσβαση και δυνατότητα αλλαγών στις βαθμολογίες έχει η γραμματεία της σχολής και ο διαχειριστής της Βάσης Δεδομένων του πανεπιστημίου.



Εικόνα 4 Απλή διαδικασία βαθμολόγησης

4.1.3 Πιθανά σημεία βελτίωσης-Κίνητρα

Στην παραπάνω διαδικασία παρατηρούμε ότι υπάρχουν διάφοροι χρήστες οι οποίοι έχουν εξουσία πάνω στην εγκυρότητα των βαθμολογιών. Αρχικά οι καθηγητές έχουν την εξουσία πάνω στις βαθμολογίες σε όλη τη διάρκεια του μαθήματος, ενώ την ίδια δυνατότητα έχει και ο διαχειριστής του Moodle. Στη συνέχεια θα πρέπει πάλι με ευθύνη των καθηγητών να γίνει μετατροπή των αρχείων ώστε να είναι στην κατάλληλη μορφή που είναι απαραίτητη για να παραδοθούν στην γραμματεία. Στη συνέχεια η γραμματεία θα έχει στη διάθεση της σε πρώτη φάση αρχεία βαθμών και υπ' ευθύνη της να εισαχθούν αυτά στο σύστημα της σχολής. Τέλος όταν οι βαθμοί ανέβουν στις καρτέλες των φοιτητών, ο διαχειριστής των υπολογιστικών συστημάτων του πανεπιστημίου όπως και η γραμματεία είναι οι υπεύθυνη για τη συντήρηση των βαθμών και για πιθανές αλλαγές. Παρατηρούμε λοιπόν ότι πρόκειται για μια διαδικασία κατά την οποία υπάρχουν αρκετοί συμμετέχοντες που έχουν πλήρη εξουσία πάνω από ευαίσθητες πληροφορίες όπως είναι τα αρχεία βαθμών. Επομένως, τίθεται υπό αμφισβήτηση η ιδιωτικότητα των δεδομένων, αφού πολλοί, σίγουρα περισσότεροι από τους ελάχιστους δυνατούς, χρήστες έχουν πρόσβαση σε αυτά ενώ πλήττεται η ακεραιότητα των βαθμών και απειλείται η διαφάνεια της διαδικασίας εφόσον καθηγητές, γραμματεία και διαχειριστές έχουν κάθε δυνατότητα επεξεργασίας στα αρχεία. Τέλος, υπάρχουν μέρη της διαδικασίας όπως η εξαγωγή από το Moodle, η

μετατροπή στην κατάλληλη μορφή και η αποστολή στη γραμματεία και η εκ νέου εισαγωγή στο φοιτητολόγιο που μπορούν να αυτοματοποιηθούν ή να αφαιρεθούν. Πρόκειται μια διαδικασία που απαιτεί εμπιστοσύνη σε πολλούς παράγοντες για να εξασφαλιστεί η προστασία των δεδομένων και το κύρος των μαθημάτων.

Όλες οι παραπάνω αδυναμίες λοιπόν μας οδήγησαν σε αυτή την εργασία να μελετήσουμε πως θα μπορούσαμε να υποστηρίξουμε την διαδικασία διαχείρισης με σκοπό να εξαλείψουμε κάποιους από τους παράγοντες εμπιστοσύνης. Η προσέγγισή μας, δεν αντικαθιστά σε καμία περίπτωση την υπάρχουσα διαδικασία αλλά έχει σκοπό να την υποστηρίξει και να ενισχύσει σε κάποιο βαθμό τα 4 χαρακτηριστικά που αναφέραμε παραπάνω.

4.2 Περιγραφή της προσέγγισης

Στόχος μας με αυτή την εργασία είναι, να αναπτύξουμε ένα επιπρόσθετο στοιχείο(plugin) στην πλατφόρμα του Moodle το οποίο θα επικοινωνεί με ένα ιδιωτικό δίκτυο Blockchain και θα αποθηκεύει εκεί τα δεδομένα βαθμολογίας. Αυτό το επιπρόσθετο στοιχείο θα εμφανίζεται σε επίπεδο μαθήματος και θα είναι ορατό μόνο στους καθηγητές του κάθε μαθήματος. Ο διαχειριστής του Moodle θα έχει πρόσβαση στο στοιχείο για όλα τα μαθήματα. Με αυτό τον τρόπο και από την στιγμή που ο καθηγητής θα επιλέξει να φορτώσει τους βαθμούς στο Blockchain θα έχει πλέον εξασφαλισθεί ένα αντίγραφο των βαθμολογιών που δεν μπορεί να μεταβληθεί. Αυτό το αντίγραφο θα αποθηκεύεται σε ένα ιδιωτικό Blockchain το οποίο θα ανήκει στο εκάστοτε πανεπιστήμιο. Συνεπώς έτσι με αυτό το υποστηρικτικό στοιχείο μπορούμε να «θωρακίσουμε» ένα μεγάλο μέρος όλης της διαδικασίας διαχείρισης βαθμών. Ακόμα κρίνεται σημαντικό να έχουμε τη δυνατότητα να ανακτούμε συγκεκριμένες βαθμολογίες από το Blockchain ούτως ώστε να μπορούμε να επαληθεύσουμε ότι οι βαθμοί είναι αμετάβλητοι ή να διαπιστώσουμε πιθανές αλλαγές κατά τη διαδικασία διαχείρισης. Η δυνατότητα επικύρωσης κρίθηκε σκόπιμο να υπάρχει και αναπτύχθηκε για τους καθηγητές μέσα στην πλατφόρμα του Moodle, ώστε να μπορούν να επικυρώνουν βαθμολογίες από την αρχική υποβολή μέχρι και αμέσως πριν την αποστολή στην γραμματεία. Συνοψίζοντας λοιπόν, η λύση μας αφορά τους καθηγητές που αλληλοεπιδρούν με το Moodle. Σε μια πιθανή επέκταση αυτής της εργασίας η ανάκτηση βαθμολογιών θα μπορούσε να υλοποιηθεί κατευθείαν στο φοιτητολόγιο(περισσότερα στο Κεφάλαιο 7).

4.2.1 Γιατί Moodle;

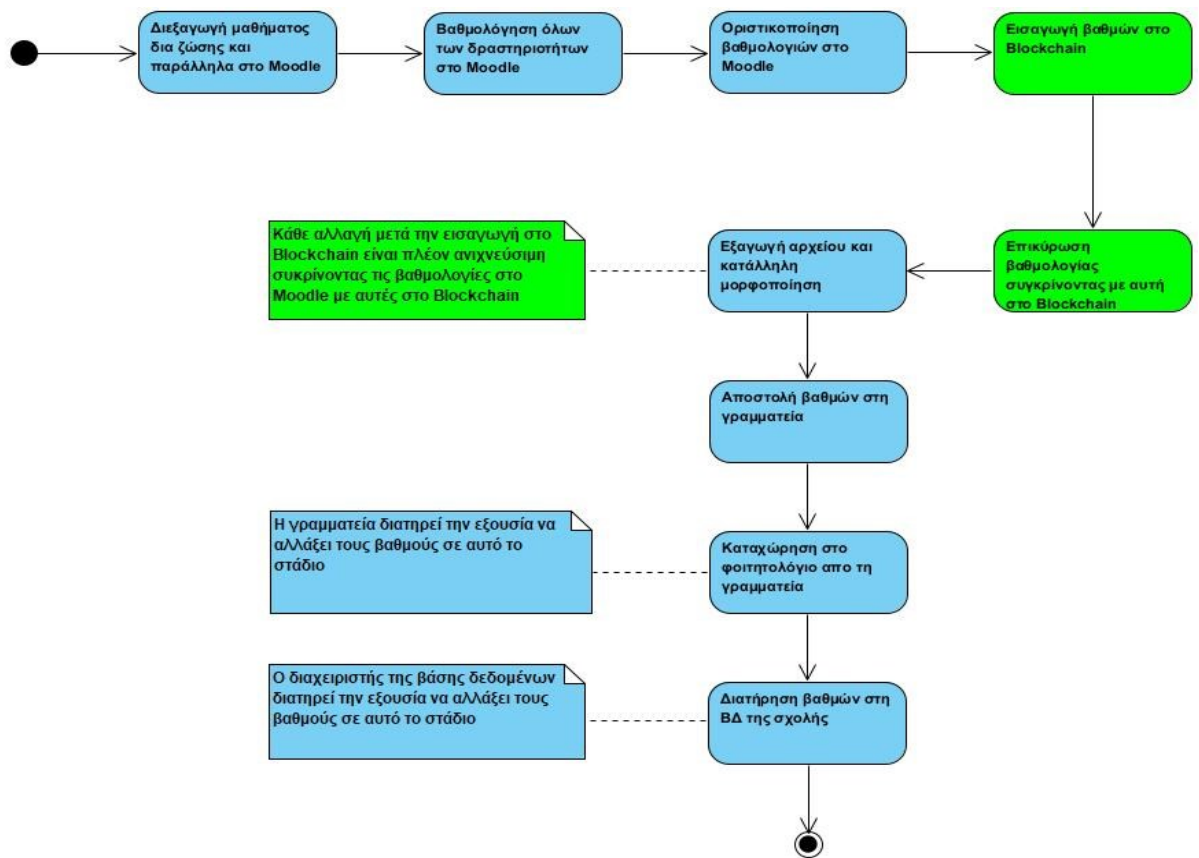
Όπως εξηγήθηκε στο κεφάλαιο 2 το Moodle είναι σε μεγάλο βαθμό επεκτάσιμο και υποστηρίζει την ανάπτυξη επιπρόσθετων στοιχείων, ενώ συνοδεύεται από μια μεγάλη κοινότητα

προγραμματιστών που δουλεύουν και αναπτύσσουν συνεχώς την πλατφόρμα. Ακόμα είναι ευρέως διαδομένο και χρησιμοποιείται από πολλά πανεπιστημιακά ιδρύματα. Όλα τα παραπάνω μας δίνουν την δυνατότητα να επεκτείνουμε την πλατφόρμα με την λειτουργικότητα που επιθυμούμε και να σχεδιάσουμε μια λύση που ενδεχομένως να απευθυνθεί σε μια μεγάλη κοινότητα όπως αυτή του Moodle. Σε ότι αφορά τη διαδικασία η πλατφόρμα ήδη υποστηρίζει-διευκολύνει την διαχείριση της μαθησιακής εμπειρίας και με την επέκταση μας αποσκοπούμε να διευκολύνουμε και να αυτοματοποιήσουμε περαιτέρω την διαχείριση των βαθμών.

4.2.2 Ιδιωτικό αδειοδοτούμενο δίκτυο Blockchain

Η λύση μας βασίζεται στην τεχνολογία του Blockchain και στα οφέλη που αυτή φέρνει. Σύμφωνα και με την ανάλυση που έγινε στο κεφάλαιο 3, τα δεδομένα που αποθηκεύονται στο Blockchain είναι πρακτικά αμετάβλητα γεγονός που ταιριάζει απόλυτα στην ανάγκη μας για διασφάλιση της ακεραιότητας των βαθμών. Ακόμα, με τη χρήση Blockchain για αποθήκευση των βαθμολογιών και ανάκτηση από εκεί των βαθμών, διατηρούμε μια λύση αποθήκευσης η οποία δεν απαιτεί εμπιστοσύνη σε κανένα από τα ενδιαφερόμενα μέρη. Είναι πλήρως καταμεμημένη και δεδομένου ότι τα δεδομένα αποθηκεύονται σε κάθε κόμβο, είναι αδύνατο να χάσουμε πληροφορία ακόμα και αν τεθεί εκτός λειτουργίας ή παραβιαστεί κάποιος κόμβος. Ο λόγος που επιλέξαμε να υλοποιήσουμε ένα ιδιωτικό αδειοδοτούμενο δίκτυο είναι για να εξασφαλίσουμε φθηνότερες και γρηγορότερες συναλλαγές, ρυθμίζοντας το δίκτυο για τις δικές μας περιπτώσεις χρήσεις, αλλά και αφού οι χρήστες που θέλουμε να αλληλοεπιδρούν με τα δεδομένα είναι καθορισμένοι (καθηγητές, διαχειριστές, γραμματείς).

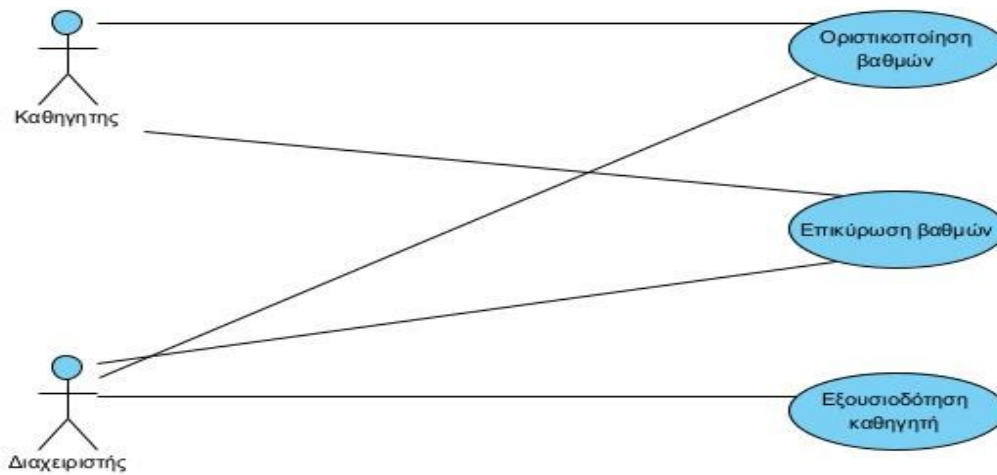
4.2.3 UML διάγραμμα δραστηριοτήτων της νέας διαδικασίας



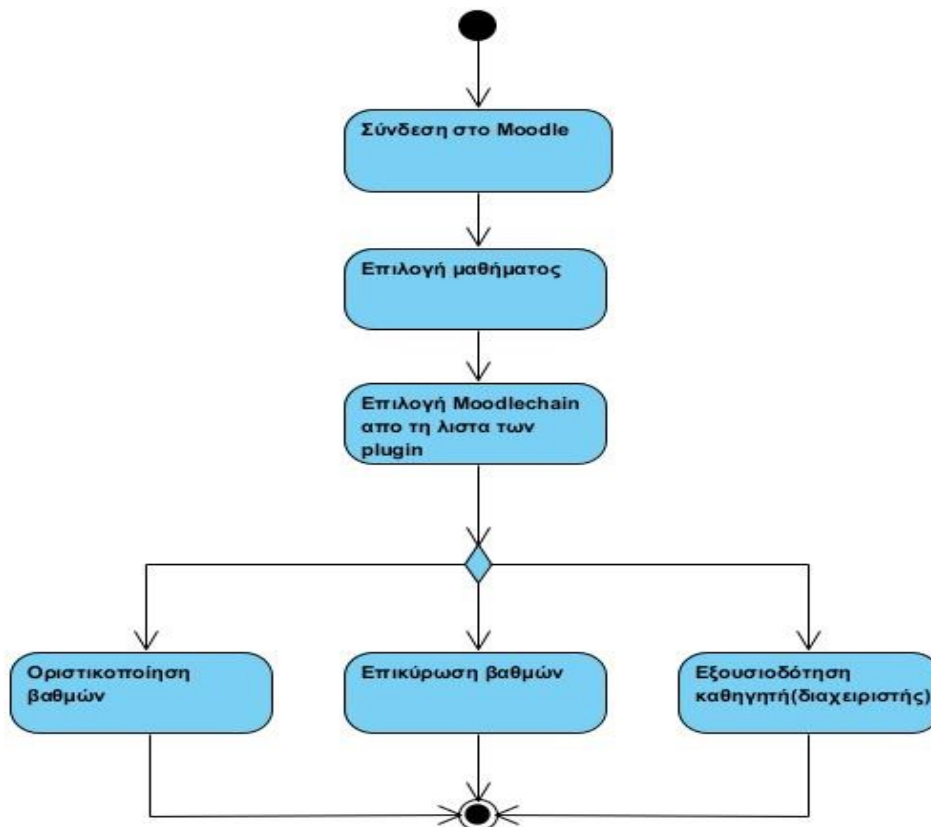
Εικόνα 5 Η διαδικασία βαθμολόγησης μετά την προσθήκη του στοιχείου

4.3 Μοντελοποίηση των περιπτώσεων χρήσης με UML

Στο παρακάτω διάγραμμα βλέπουμε τις περιπτώσεις χρήσης της λύσης που σχεδιάσαμε σε συνδυασμό με τους χρήστες που έχουν πρόσβαση σε αυτές, ενώ στο αμέσως επόμενο το διάγραμμα ροής του εργαλείου που υλοποιήσαμε.



Εικόνα 6 Διάγραμμα χρηστών-περιπτώσεων χρήσης

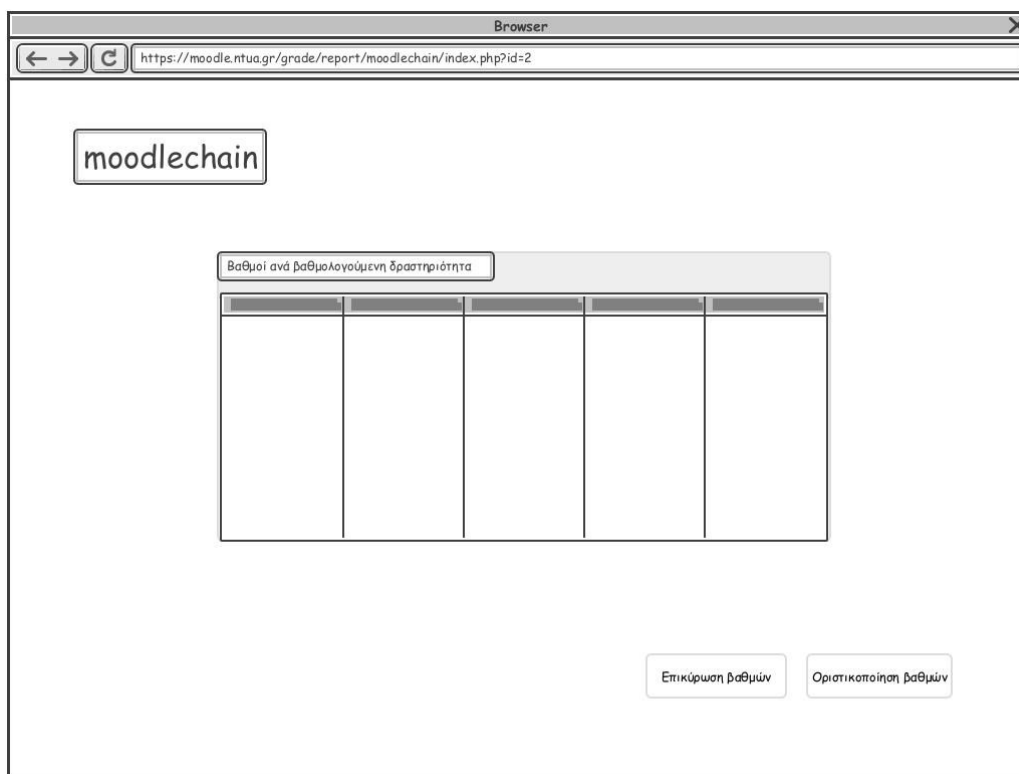


Εικόνα 7 Διάγραμμα ροής υψηλού επιπέδου

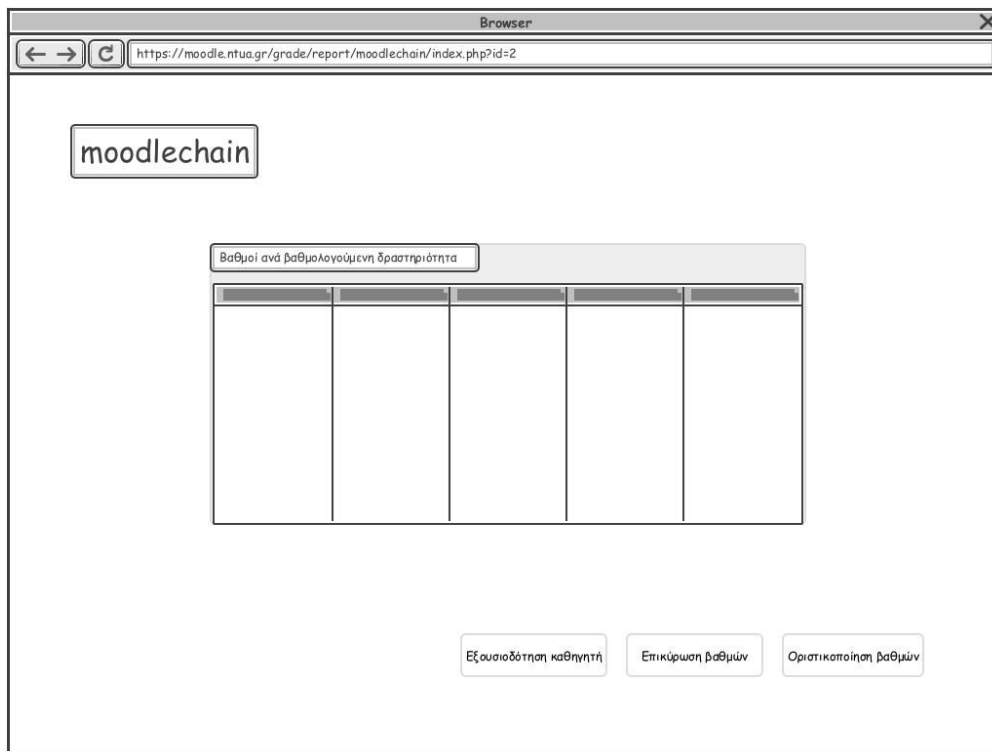
Παρατηρώντας τα 2 διαγράμματα βλέπουμε ότι υπάρχουν 3 βασικές περιπτώσεις χρήσης(θα αναλυθούν παρακάτω) για το εργαλείο μας ενώ οι χρήστες μας κατηγοριοποιούνται σε καθηγητές και στους διαχειριστές του εργαλείου που είναι υπεύθυνοι για την διαχείριση και συντήρηση του Moodle αλλά του δικτύου Blockchain.

Δικαιώματα/ Χρήστη	Πρόσβαση στο plugin	Οριστικοποίηση βαθμών	Επικύρωση βαθμών	Εξουσιοδότηση καθηγητή
Διαχειριστής	Ναι	Ναι	Ναι	Ναι
Καθηγητής	Ναι	Ναι (Αν έχει εξουσιοδοτηθεί)	Ναι	Όχι
Φοιτητής	Όχι	Όχι	Όχι	Όχι

Το plugin μας θα αναπτυχθεί σε επίπεδο μαθήματος, δηλαδή ο χρήστης θα μπαίνει σε αυτό μέσω ενός συγκεκριμένου μαθήματος και η ορατότητα που θα έχει εξαρτάται από τον ρόλο του στο συγκεκριμένο μάθημα. Ο διαχειριστής του Moodle θα έχει την ίδια ορατότητα για το plugin σε όλα τα μαθήματα του πανεπιστημίου. Στα επόμενα δυο διαγράμματα βλέπουμε μια ελάχιστη προσομοίωση για την αρχική σελίδα του plugin που συναντά ένας καθηγητής στο μάθημα του και ο διαχειριστής του Moodle. Σε αυτή βλέπουμε έναν πίνακα με σειρές που αναφέρονται στους φοιτητές και στήλες που αναφέρονται στις βαθμολογούμενες δραστηριότητες του μαθήματος. Ο πίνακας είναι γεμάτος με τις επιμέρους αλλά και την συνολική βαθμολογία των φοιτητών. Ακόμα οι χρήστες βλέπουν και 3 κουμπιά, το καθένα για κάθε μια από τις 3 περιπτώσεις χρήσης που θα αναλύσουμε παρακάτω.



Εικόνα 8 Wireflow διάγραμμα για τον καθηγητή του μαθήματος



Εικόνα 9 Wireflow διάγραμμα για τον διαχειριστή του Moodle

4.4 Τα συστατικά της λύσης

Αρχικά μέρος του συστήματος μας, αποτελεί η εγκατάσταση του Moodle που υποστηρίζει κάθε πανεπιστήμιο σε δικούς του server. Εκεί θα πρέπει εμείς να εγκαταστήσουμε το επιπρόσθετο στοιχείο που υλοποιήσαμε με όνομα Moodlechain.

Οι χρήστες του Moodle αλληλοεπιδρούν με αυτό από τον προσωπικό τους υπολογιστή μέσω κάποιου web-browser.

Στη συνέχεια πρέπει να μελετήσουμε τις διάφορες υλοποιήσεις που υπάρχουν για ιδιωτικά δίκτυα Blockchain ώστε να σχεδιάσουμε τα υπόλοιπα συστατικά της λύσης. Υπάρχουν διάφορες εναλλακτικές όπως θα δούμε παρακάτω.[16]

1. R3 Corda είναι μια καινοτόμα πλατφόρμα Blockchain που αποσκοπεί στη μείωση του κόστους συναλλαγών και στην αύξηση της ταχύτητας τους. Αυτό το έργο είχε αρχικά σχεδιαστεί για τον χρηματοοικονομικό τομέα αλλά βρίσκει και άλλες εφαρμογές όπως ο τομέας υγείας και περίθαλψης, η εφοδιαστική αλυσίδα, κυβερνητικές και δημόσιες εφαρμογές καθώς και οι χρηματιστηριακές συναλλαγές. [17]

2. Η πλατφόρμα του Quorum δημιουργήθηκε αρχικά από την εταιρία JP Morgan και πρόκειται για μια εμπορική έκδοση του Ethereum. Σχεδιάστηκε για να δώσει τη δυνατότητα σε εταιρίες να εκμεταλλευτούν το Ethereum στις δικές τους υψηλής αξίας εφαρμογές. Οι επιχειρήσεις μπορούν να βασιστούν στο λογισμικό ανοιχτού κώδικα που παρέχεται από το Quorum και να ενσωματώσουν πάνω του εργαλεία ανεπτυγμένα είτε από τις ίδιες τις εταιρίες είτε από άλλες για να χτιστούν επεκτάσιμες και υψηλής απόδοσης εφαρμογές. Πλέον έχει πωληθεί και συντηρείται από την εταιρία Consensus. [18]
3. Η πλατφόρμα του Ripple έχει χτιστεί πάνω στο XRP ledger το οποίο κανονίζει τις συναλλαγές σε 3-5 δευτερόλεπτα. Αυτό το δίκτυο υποστηρίζεται από ένα διεθνές δίκτυο από επαληθευτές και είναι σχεδιασμένο να είναι γρήγορο, επεκτάσιμο και ενεργειακά αποδοτικό. Το Ripple Blockchain υποστηρίζει προϊόντα ανοιχτού κώδικα που αποσκοπούν στην ενίσχυση του XRP ledger. Τέλος, η πλατφόρμα χρησιμοποιείται από μεγάλες εταιρίες όπως η American Express, η Deloitte και η SBI Holdings που πειραματίζονται με τις δυνατότητες της πλατφόρμας κυρίως γύρω από τις διαδικασίες ηλεκτρονικών πληρωμών. [19]
4. Το Hyperledger Fabric είναι ένα έργο ανοιχτού κώδικα που δημιουργήθηκε από το Linux Foundation. Είναι ένα επεκτάσιμο framework Blockchain και πρότυπο για εταιρικές πλατφόρμες Blockchain. Προορίζεται ως βάση για την ανάπτυξη εταιρικών εφαρμογών και λύσεων στη μεγάλη βιομηχανία ενώ η αρθρωτή και επεκτάσιμη αρχιτεκτονική του το κάνει να φιλοξενεί ένα ευρύ φάσμα περιπτώσεων χρήσης. Διαθέτει προηγμένα στοιχεία ελέγχου απόρρητου έτσι ώστε μόνο τα δεδομένα που είναι θεμιτό να κοινοποιηθούν να κοινοποιούνται μεταξύ των συμμετεχόντων στο δίκτυο. Καταλαβαίνουμε λοιπόν ότι παρέχει μια λύση που ταιριάζει στις περιπτώσεις χρήσης μας.[20]
5. Το Ethereum Blockchain είναι μια ανοιχτή πλατφόρμα Blockchain που δίνει τη δυνατότητα στους προγραμματιστές να αναπτύξουν και να τρέξουν εκεί τις δικές τους αποκεντρωμένες εφαρμογές. Υποστηρίζει ανάπτυξη έξυπνων συμβολαίων και παρέχει ένα σύνολο εργαλείων για να υποστηρίξει την ανάπτυξη συμβολαίων συμπεριλαμβανομένης και μιας γλώσσας ειδικού σκοπού της Solidity που είναι σχεδιασμένη για ανάπτυξη συμβολαίων. Υπάρχει πολύ ενεργή κοινότητα από προγραμματιστές και για αυτό θεωρείται πολύ εύκολη επιλογή για να ξεκινήσει κανείς την ανάπτυξη αποκεντρωμένων εφαρμογών.

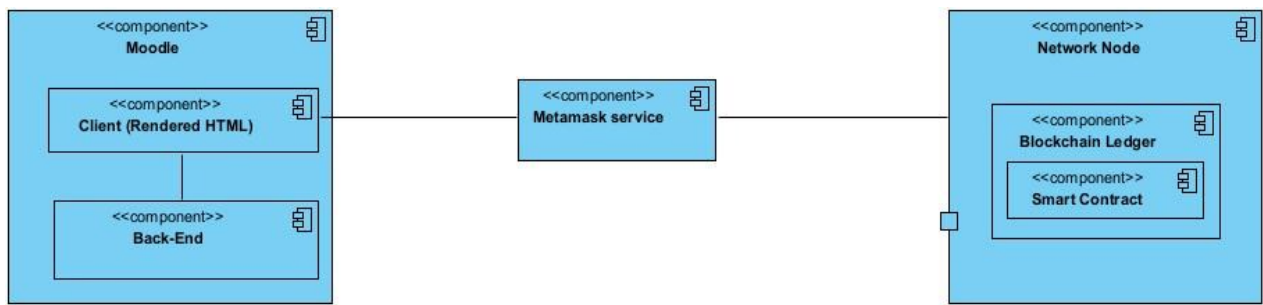
Από τις παραπάνω επιλογές οι δύο τελευταίες ήταν οι επικρατέστερες, καθώς οι τρεις πρώτες θεωρήθηκαν πολύ εξειδικευμένες για την λύση μας. Εκ των δύο εν τέλη επιλέχθηκε η λύση του Ethereum Blockchain. Ο κύριος λόγος είναι ότι αυτή η επιλογή συνοδεύεται από πληθώρα εργαλείων ανάπτυξης (Remix, ganache) αλλά και εργαλεία που διευκολύνουν την εγκατάσταση ενός τέτοιου

δικτύου ενώ έχει πολύ μεγάλη κοινότητα για υποστήριξη και πλούσια τεκμηρίωση. Από την άλλη το Hyperledger Fabric κρίθηκε λίγο πιο σύνθετο όπως αποδείχθηκε και εδώ.[21] Έτσι λοιπόν καταλήγουμε σε υλοποίηση δικτύου στο Ethereum Blockchain.

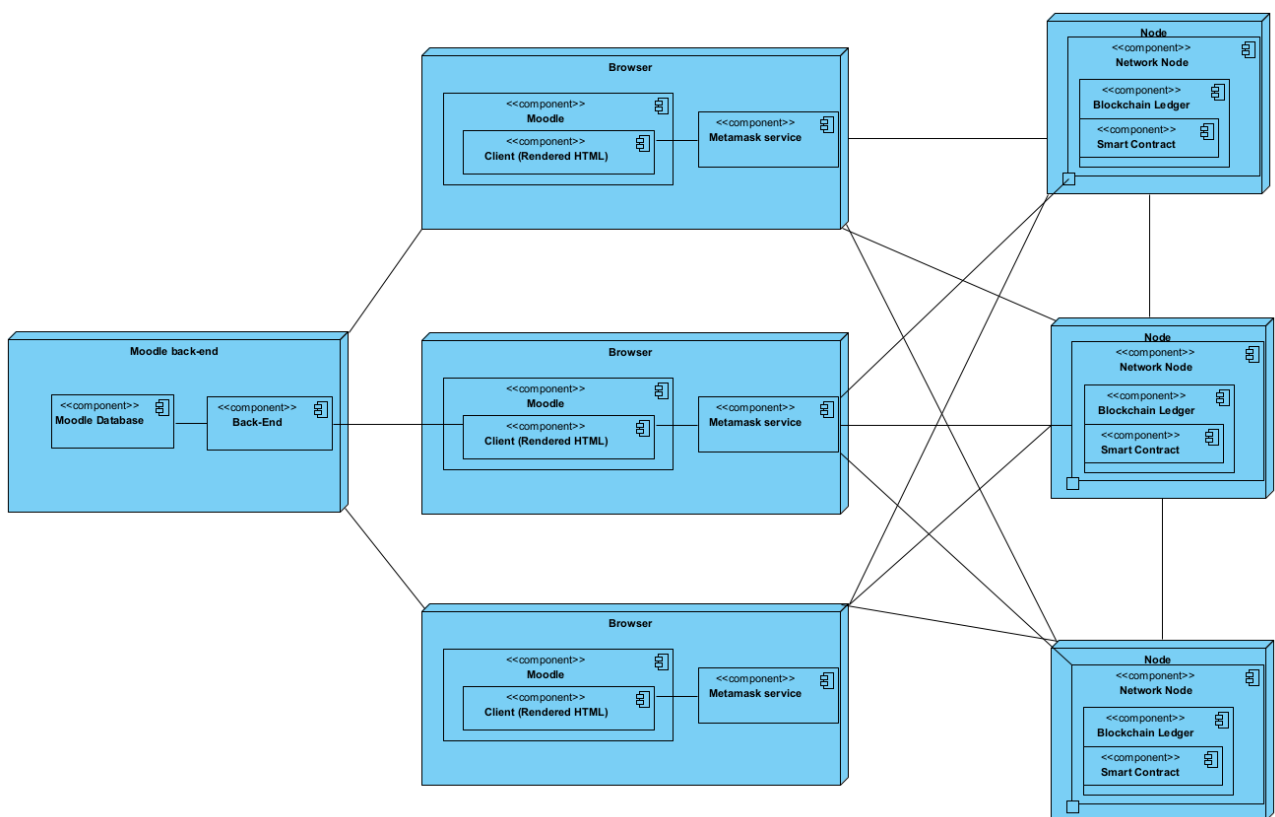
Για να υλοποιήσουμε την επιχειρησιακή λογική που επιθυμούμε, για να διαχειριζόμαστε τις βαθμολογίες και να μπορούμε να τις λαμβάνουμε πίσω προς επικύρωση είναι αναγκαίο να υλοποιήσουμε ένα έξυπνο συμβόλαιο που θα φορτώνουμε στο Blockchain δίκτυο. Η πιο διαδεδομένη γλώσσα για ανάπτυξη έξυπνων συμβολαίων στο Ethereum Blockchain είναι η Solidity η οποία αναπτύχθηκε από την ομάδα του Ethereum. Υπολογίζεται ότι [22] το 90% των έξυπνων συμβολαίων είναι ανεπτυγμένο σε Solidity. Πρόκειται για μια γλώσσα αντικειμενοστραφούς προγραμματισμού, υψηλού επιπέδου και έχει πάρει πολλά στοιχεία της από τις πολύ γνωστές γλώσσες προγραμματισμού C++, Python και Javascript [23]. Μερικές συνηθισμένες περιπτώσεις χρήσης για τη Solidity είναι η ανάπτυξη συμβολαίων για ψηφοφορίες, τυφλές δημοπρασίες, συλλογική χρηματοδότηση κ.α. Καταλαβαίνουμε λοιπόν ότι πρόκειται για μια ευρέως διαδεδομένη γλώσσα, με πολύ ενεργή κοινότητα. Όλα αυτά σε συνδυασμό με τα διάφορα πλαίσια ανάπτυξης που υπάρχουν, μας οδήγησαν στην επιλογή αυτής της γλώσσας για την ανάπτυξη του συμβολαίου που θα δούμε στη συνέχεια της εργασίας.

Τέλος, χρειαζόμαστε έναν τρόπο ώστε να αλληλοεπιδρούμε από το Moodle με το δίκτυο Blockchain. Αυτή την δυνατότητα θα μας την δώσει η υπηρεσία του Metamask. Το Metamask δίνει την δυνατότητα στους χρήστες να επικοινωνήσουν με ένα Ethereum Blockchain δίκτυο μέσω μιας επέκτασης από το web browser. Δημιουργήθηκε από την εταιρία Consensus και δίνει τη δυνατότητα στους χρήστες να αποθηκεύουν και να διαχειρίζονται τους λογαριασμούς τους, να συναλλάσσουν tokens κρυπτονομισμάτων που βασίζονται στο Ethereum, ενώ χρησιμοποιείται κυρίως από τους χρήστες για την επικοινωνία τους με αποκεντρωμένες εφαρμογές στο Ethereum. Η επέκταση του Metamask θα πρέπει να είναι εγκατεστημένη στο browser του κάθε χρήστη που επιθυμεί να χρησιμοποιήσει το επιπρόσθετο στοιχείο μας.[24][25]

Στα διαγράμματα που ακολουθούν σχεδιάσαμε όλα τα συστατικά του που απαρτίζουν την λύση που υλοποιήσαμε, όπως αυτά συζητήθηκαν σε αυτή την ενότητα.



Εικόνα 10 Διάγραμμα συστατικών του Plugin

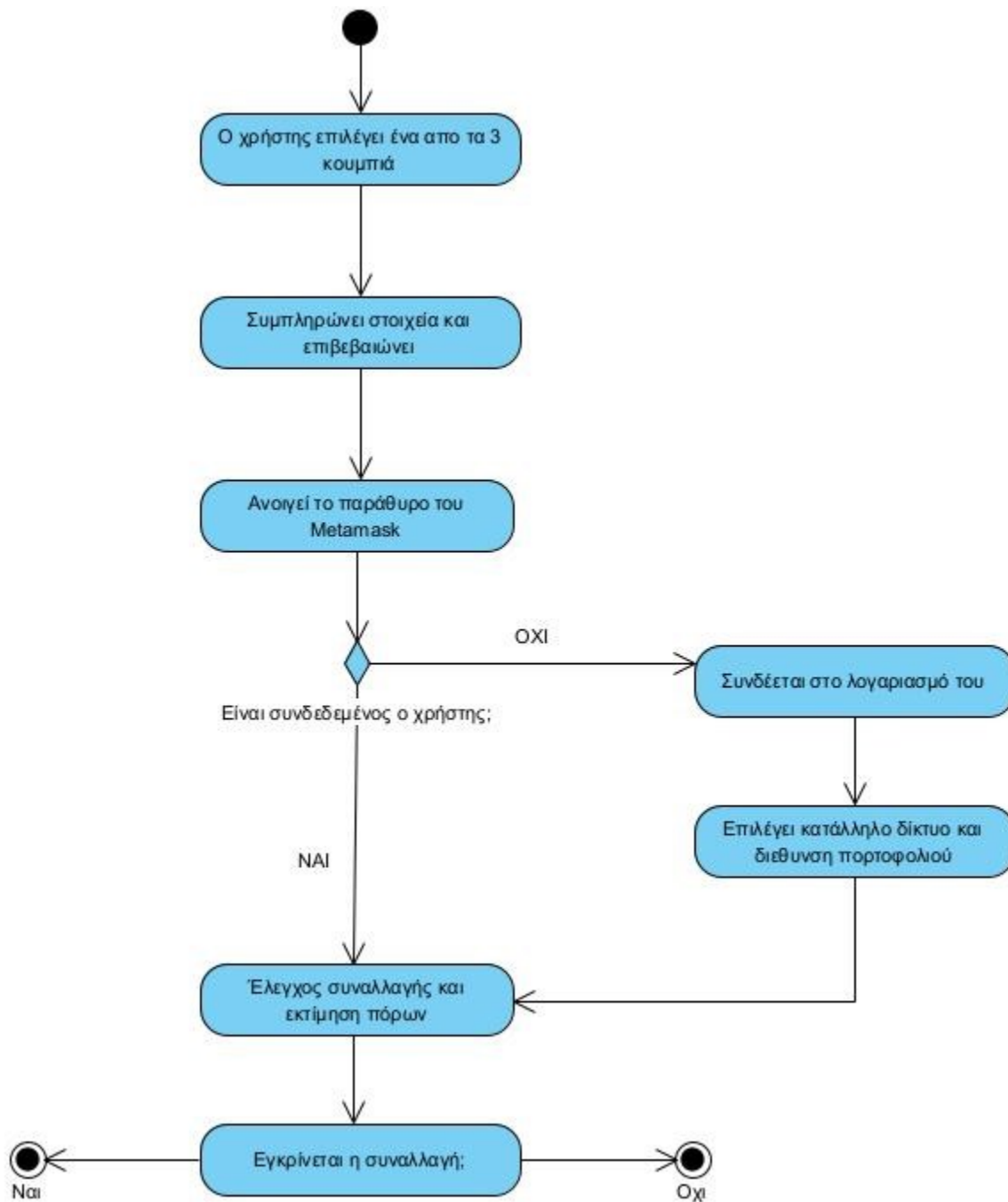


Εικόνα 11 Διάγραμμα deployment

4.5 Αλληλεπίδραση με την υπηρεσία Metamask

Όπως αναφέρθηκε για την επικοινωνία ανάμεσα στο Moodle και το δίκτυο Blockchain είναι αναγκαία η υπηρεσία του Metamask η οποία αναφέρθηκε παραπάνω. Ο χρήστης του plugin κατά την αρχική εγκατάσταση θα πρέπει να ρυθμίσει στο Metamask το δίκτυο με το οποίο θα επικοινωνεί σύμφωνα με τις οδηγίες του διαχειριστή του εκάστοτε πανεπιστημίου και να εισάγει τον λογαριασμό

του. Στη συνέχεια αφού επιλέξει οποιαδήποτε από τις δραστηριότητες που παρέχουμε με το plugin που απαιτούν επικοινωνία με το Blockchain θα χρειαστεί να αλληλοεπιδράσει με το αναδυόμενο παράθυρο του Metamask μέσω του browser του. Στο παρακάτω διάγραμμα βλέπουμε τις δραστηριότητες που πρέπει να εκπληρωθούν από το χρήστη κατά την αλληλεπίδραση του με το Metamask για να φτάσουμε σε μια επιτυχημένη συναλλαγή. Αυτό το διάγραμμα είναι πιο ακριβές για τις δύο πρώτες περιπτώσεις χρήσης. Σε αυτές ο χρήστης επιθυμεί να αλλάξει την κατάσταση του Blockchain, προσθέτοντας νέα δεδομένα και για αυτό γίνεται εκτίμηση κόστους συναλλαγής και ο χρήστης πρέπει να το αποδεχτεί. Στην τελευταία περίπτωση χρήσης ο χρήστης, δεν προσθέτει αλλά διαβάζει κάποια δεδομένα οπότε δεν χρειάζεται να αποδεχτεί την συναλλαγή για να ολοκληρωθεί, η αλληλεπίδραση με το παράθυρο του Metamask τελειώνει μόλις επιλέξει τον λογαριασμό του.



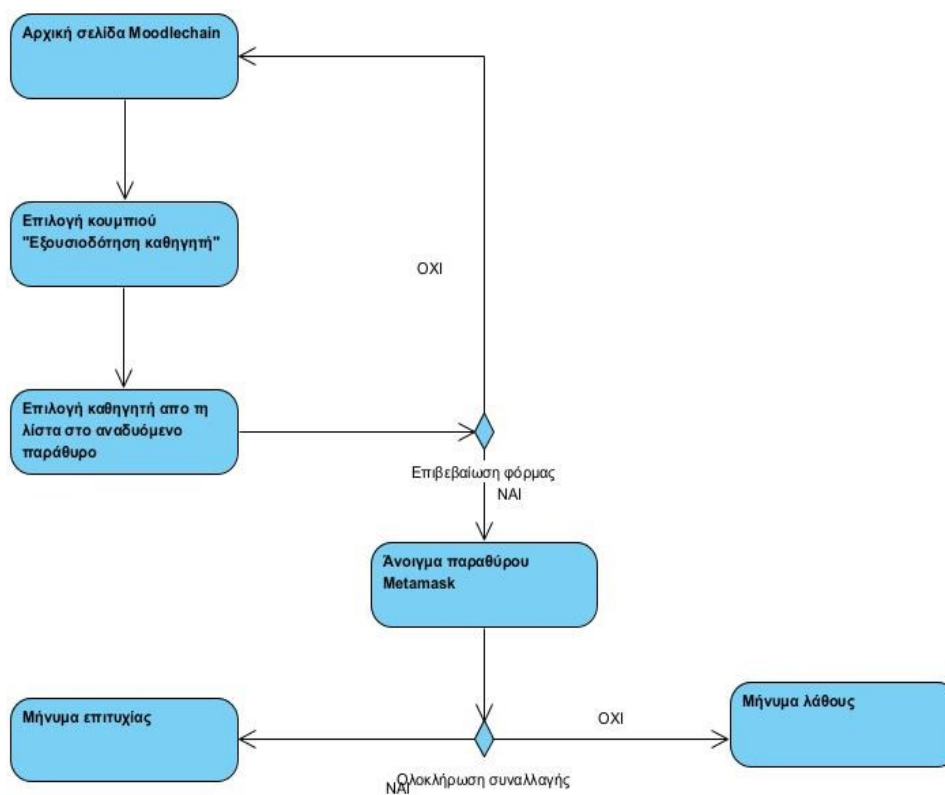
Εικόνα 12 Γενικό διάγραμμα διαδικασίας χρήσης της υπηρεσίας του Metamask

4.6 Περίπτωση χρήσης 1: Εξουσιοδότηση καθηγητή

Η συγκεκριμένη περίπτωση χρήσης αφορά μόνο τον διαχειριστή του Moodle. Ο διαχειριστής επιλέγοντας το αντίστοιχο κουμπί, θα επιλέγει έναν καθηγητή από την λίστα καθηγητών του μαθήματος, για να τον εξουσιοδοτήσει ώστε να εισάγει βαθμούς στο Blockchain για το συγκεκριμένο μάθημα. Με αυτό τον τρόπο θελήσαμε να διαχωρίσουμε τον ρόλο του καθηγητή στο

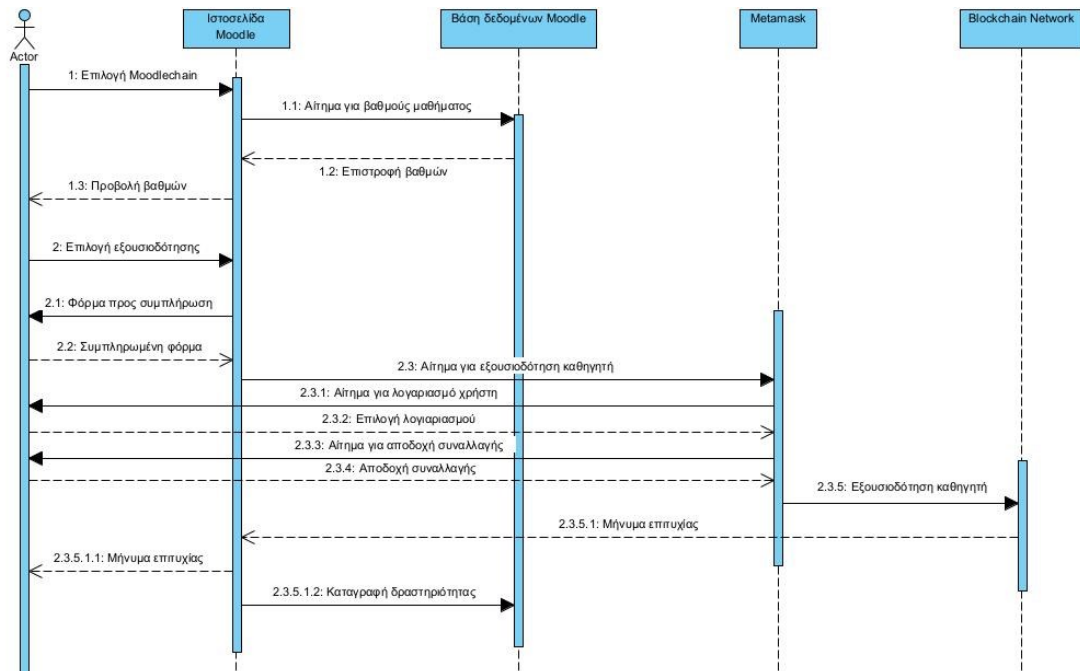
Moodle, με τη δυνατότητα να προσθέσει κανείς βαθμολογίες στο Blockchain. Περιορίζουμε λοιπόν έτσι τους καθηγητές του μαθήματος από τους οποίους το θα δέχεται βαθμούς το έξυπνο συμβόλαιο. Η συγκεκριμένη περίπτωση χρήσης, μας εξυπηρετεί γιατί σε πολλά πραγματικά μαθήματα στο Moodle, υπάρχουν αρκετοί χρήστες οι οποίοι έχουν τον ρόλο του καθηγητή όπως π.χ. βοηθοί εργαστηρίου, μεταπτυχιακοί και διδακτορικοί φοιτητές εφόσον είναι βολικό για να διαχειρίζονται το υλικό του μαθήματος, αλλά δεν θέλουμε αυθαίρετα όλοι να μπορούν να στείλουν τις βαθμολογίες στο Blockchain. Προσθέτουμε έτσι ένα επιπλέον στρώμα δικαιωμάτων τα οποία διαχειρίζεται ο διαχειριστής. Παρακάτω σχεδιάσαμε μερικά διαγράμματα για να μοντελοποιήσουμε την διαδικασία.

4.6.1 UML Activity Διάγραμμα



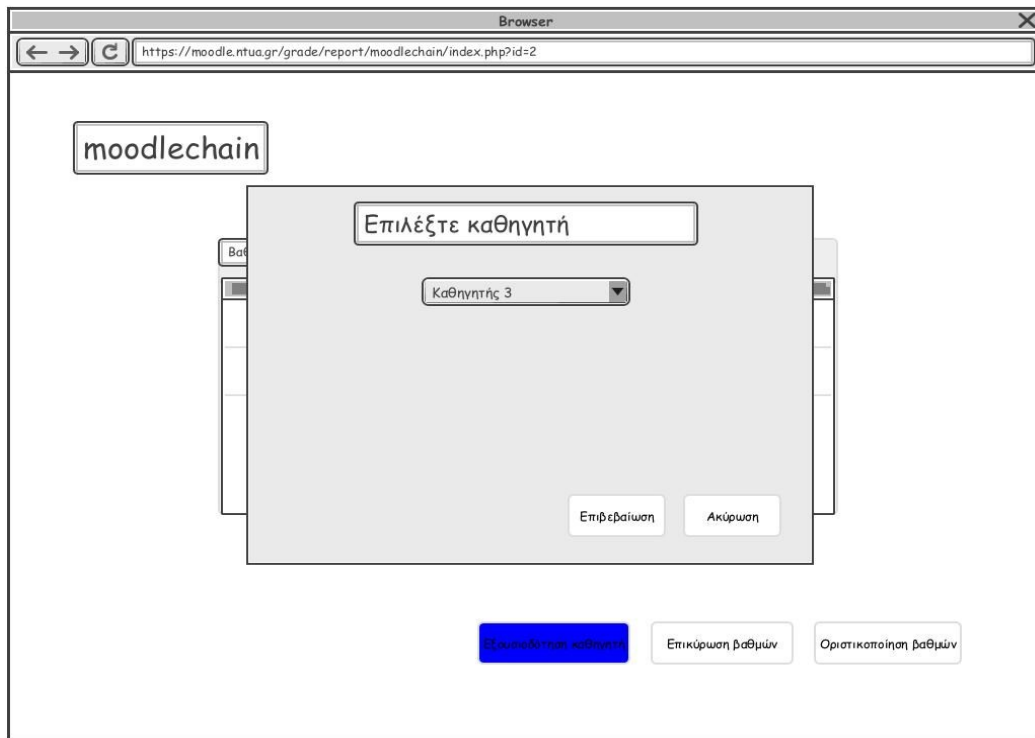
Εικόνα 13 Διάγραμμα της διαδικασίας εξουσιοδότησης

4.6.2 UML Sequence διάγραμμα



Εικόνα 14 Διάγραμμα ακολουθίας εξουσιοδότησης

4.6.3 Wireflow διαγράμματα εξουσιοδότησης



Εικόνα 15 Διάγραμμα wireflow εξουσιοδότησης καθηγητή : επιλογή καθηγητή

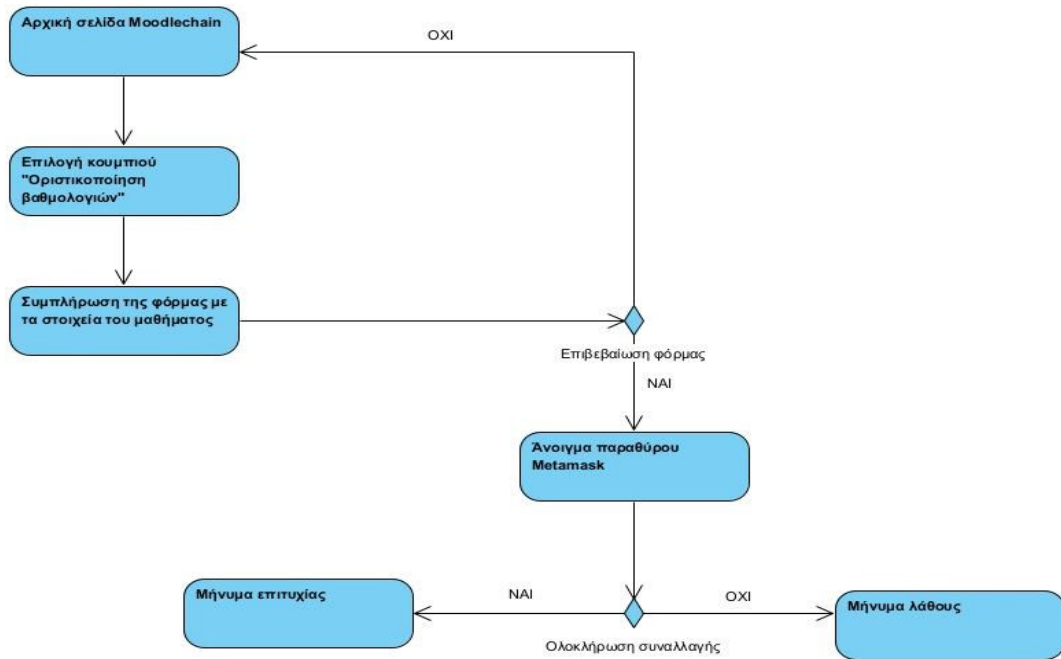


Εικόνα 16 Διάγραμμα wireflow εξουσιοδότησης καθηγητή : επιτυχής εξουσιοδότηση

4.7 Περίπτωση χρήσης 2: Οριστικοποίηση βαθμολογιών

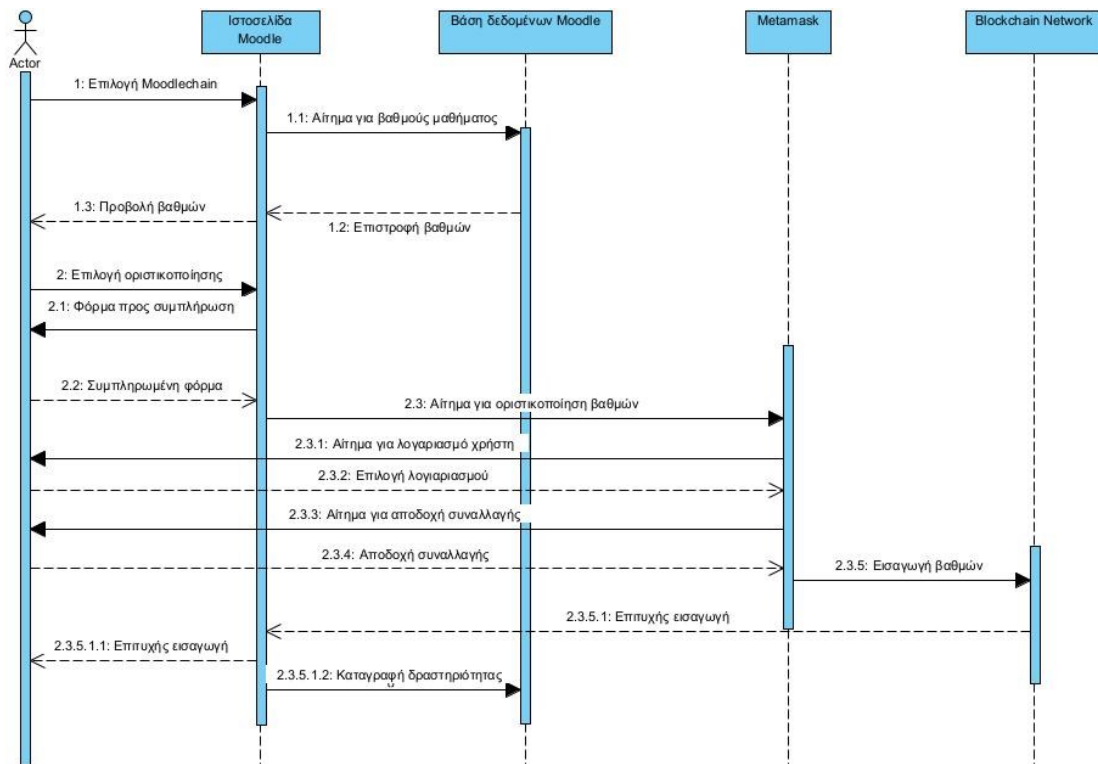
Η συγκεκριμένη περίπτωση χρήσης περιγράφει την εισαγωγή των βαθμολογιών στο Blockchain. Είναι διαθέσιμη στο Moodle για όλους τους καθηγητές του μαθήματος καθώς και για τον διαχειριστή. Για την ολοκλήρωση της διαδικασίας ο χρήστης θα πρέπει να συμπληρώσει μια φόρμα με μερικά επιπλέον δεδομένα που αφορούν το μάθημα όπως εξάμηνο, έτος και σχολή. Μόλις συμπληρωθεί η φόρμα ξεκινά η αλληλεπίδραση με το παράθυρο του Metamask και όταν ο χρήστης αποδεχτεί τη συναλλαγή, τα δεδομένα βαθμολογιών μαζί με τα δεδομένα της φόρμας που συμπλήρωσε ο χρήστης αποστέλλονται στο Blockchain. Πριν όμως τα δεδομένα εισαχθούν και διεκπεραιωθεί η συναλλαγή, θα ελέγχεται μέσω του έξυπνου συμβολαίου ότι ο χρήστης είτε είναι ο διαχειριστής, είτε έχει λάβει εξουσιοδότηση να προσθέσει βαθμούς στο συγκεκριμένο μάθημα. Αν κάτι τέτοιο δεν ισχύει, τότε η συναλλαγή θα απορριφθεί από το σύστημα και οι βαθμοί δεν θα εισαχθούν στο Blockchain.

4.7.1 UML Activity Διάγραμμα



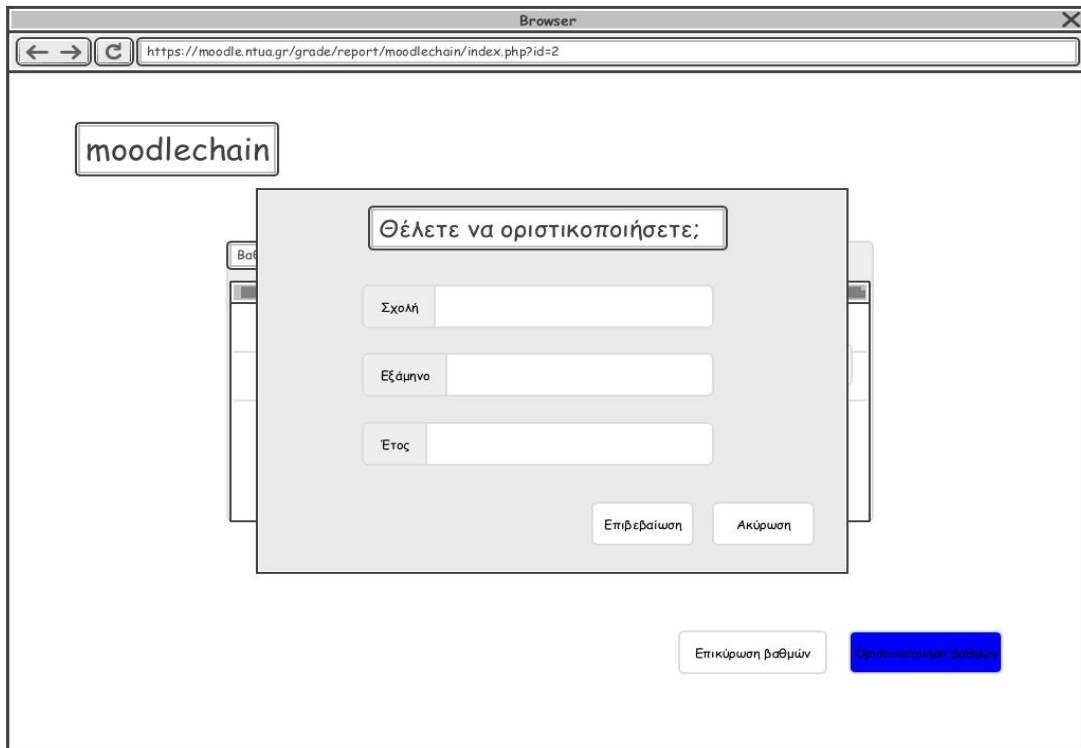
Εικόνα 17 Διάγραμμα διαδικασίας οριστικοποίησης βαθμολογιών

4.7.2 UML Sequence διάγραμμα

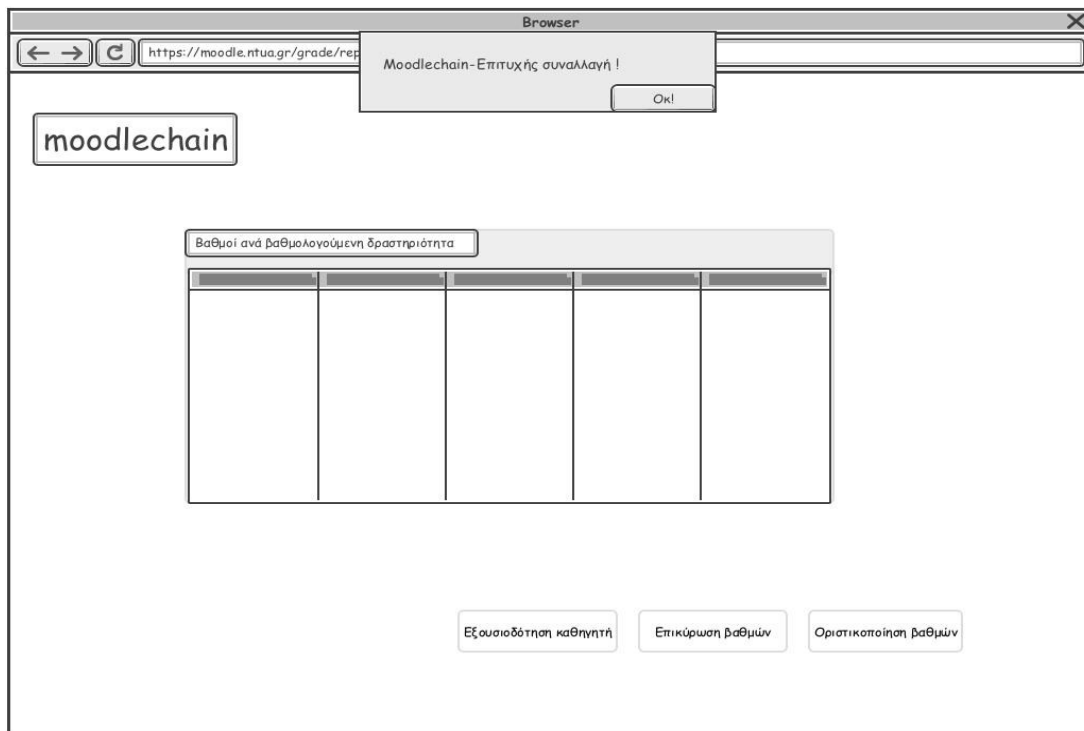


Εικόνα 18 Διάγραμμα ακολουθίας οριστικοποίησης βαθμολογιών

4.7.3 Wireflow διαγράμματα οριστικοποίησης



Εικόνα 19 Διάγραμμα διαδικασίας οριστικοποίησης βαθμολογιών: συμπλήρωση στοιχείων μαθήματος

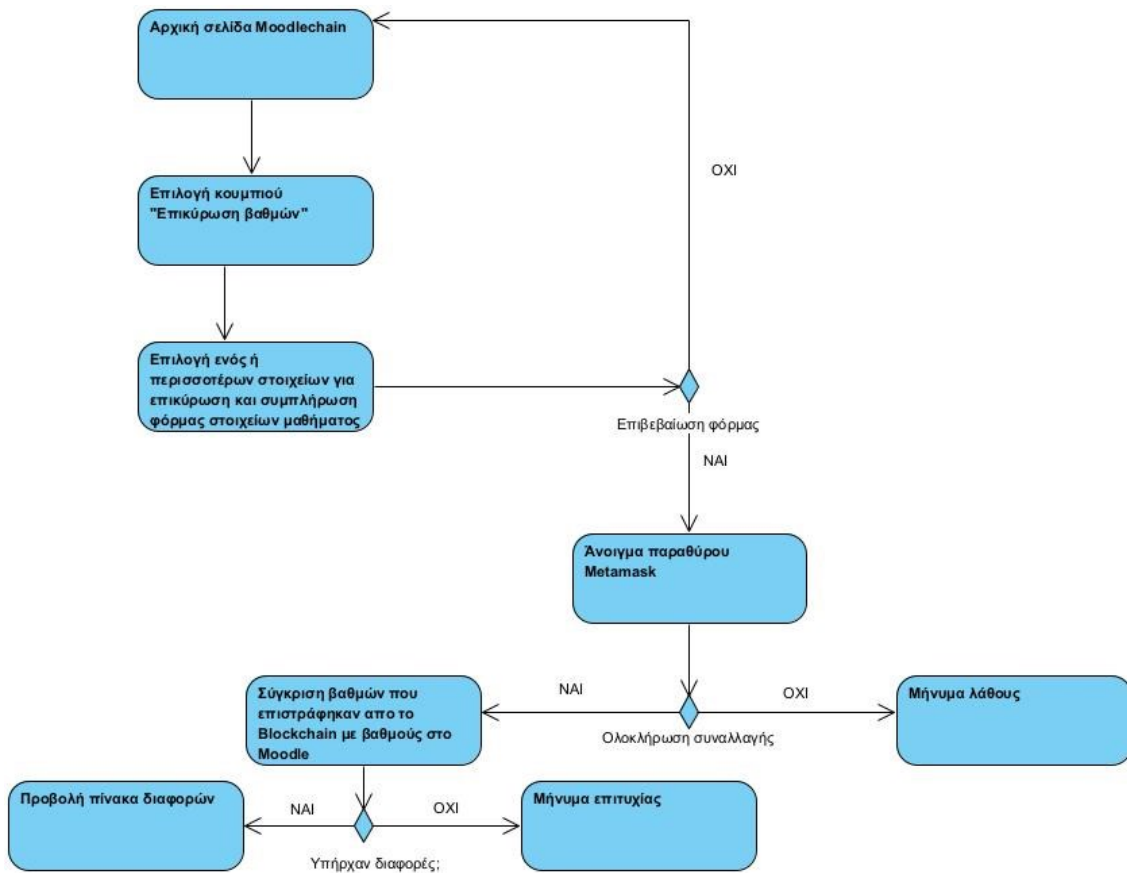


Εικόνα 20 Διάγραμμα διαδικασίας οριστικοποίησης βαθμολογιών: επιτυχής προσθήκη βαθμών

4.8 Περίπτωση χρήσης 3: Επικύρωση βαθμών

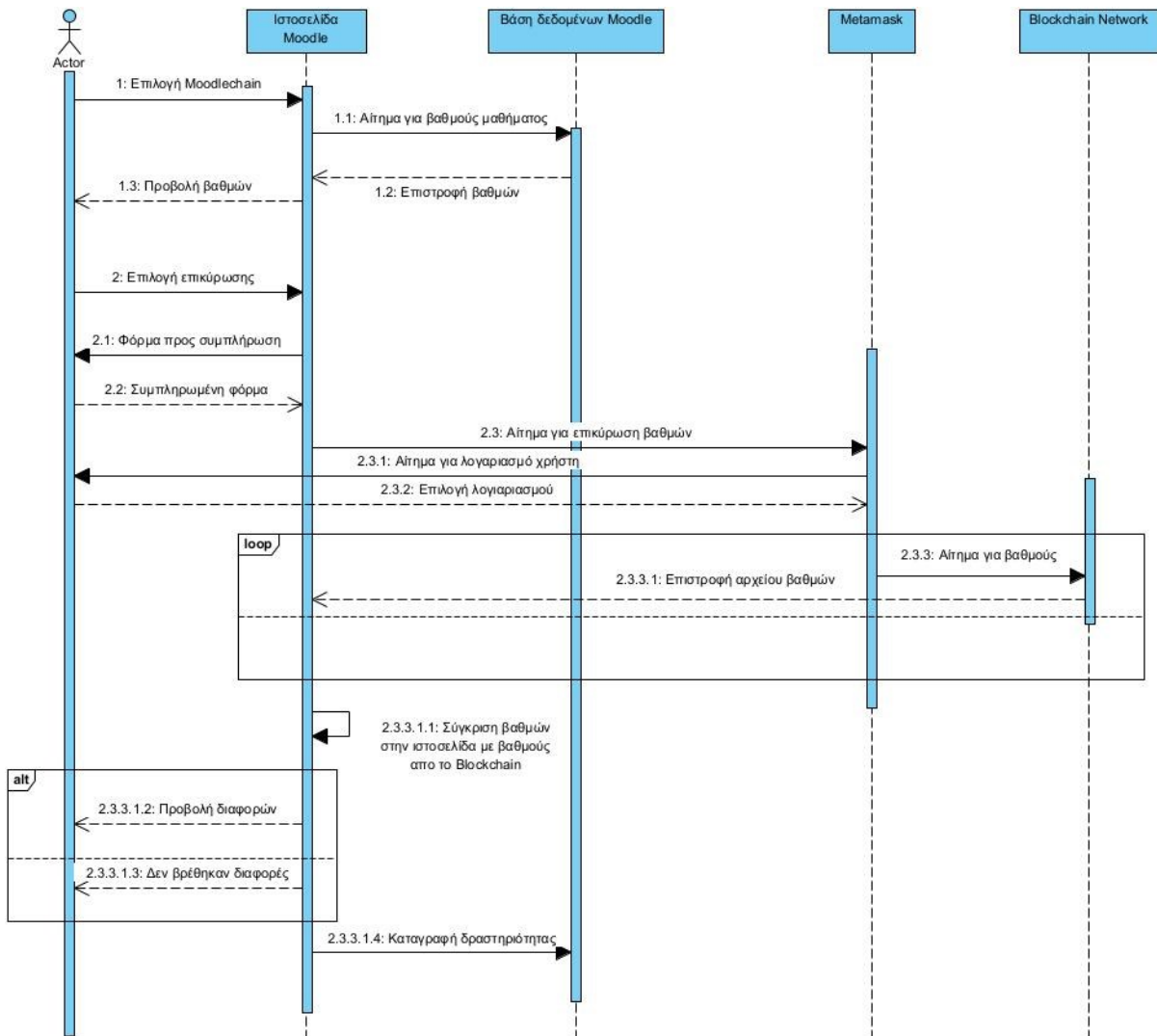
Κατά τη συγκεκριμένη περίπτωση χρήσης, ο χρήστης (καθηγητής ή διαχειριστής) επιδιώκει να επικυρώσει ότι οι βαθμολογίες που υπάρχουν στο Moodle, είναι ίδιες με αυτές που είναι φορτωμένες στο Blockchain. Αρχικά πρέπει να επιλέξει τουλάχιστον μια από τις βαθμολογούμενες δραστηριότητες του μαθήματος την οποία θέλει να επικυρώσει. Στη συνέχεια αφού συμπληρώσει μια φόρμα με στοιχεία που αφορούν το μάθημα στέλνει αίτημα στο Blockchain για να λάβει τις βαθμολογίες. Η συγκεκριμένη συναλλαγή δεν χρειάζεται όπως αναφέραμε στο 4.5 να γίνει αποδεκτή αλλά ολοκληρώνεται αμέσως μόλις ο χρήστης επιλέξει το λογαριασμό του στο Metamask. Στη συνέχεια μόλις επιστρέφονται τα στοιχεία, γίνεται η σύγκριση και είτε εμφανίζεται στο χρήστη ένας πίνακας διαφορών όπως βλέπουμε στην Εικόνα 25, ή ένα μήνυμα ότι δεν βρέθηκαν διαφορές που σημαίνει ότι οι βαθμοί είναι επικυρωμένοι.

4.8.1 UML Activity Διάγραμμα



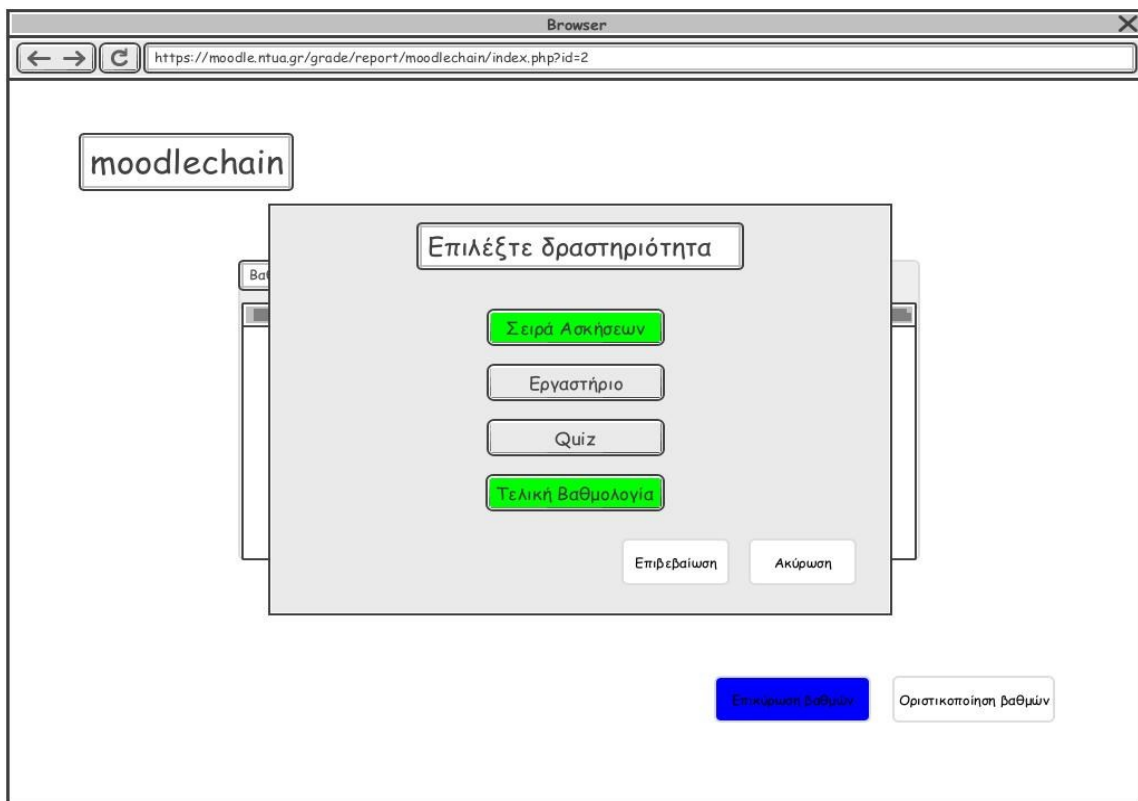
Εικόνα 21 Διάγραμμα διαδικασίας επικύρωσης βαθμών

4.8.2 UML Sequence διάγραμμα

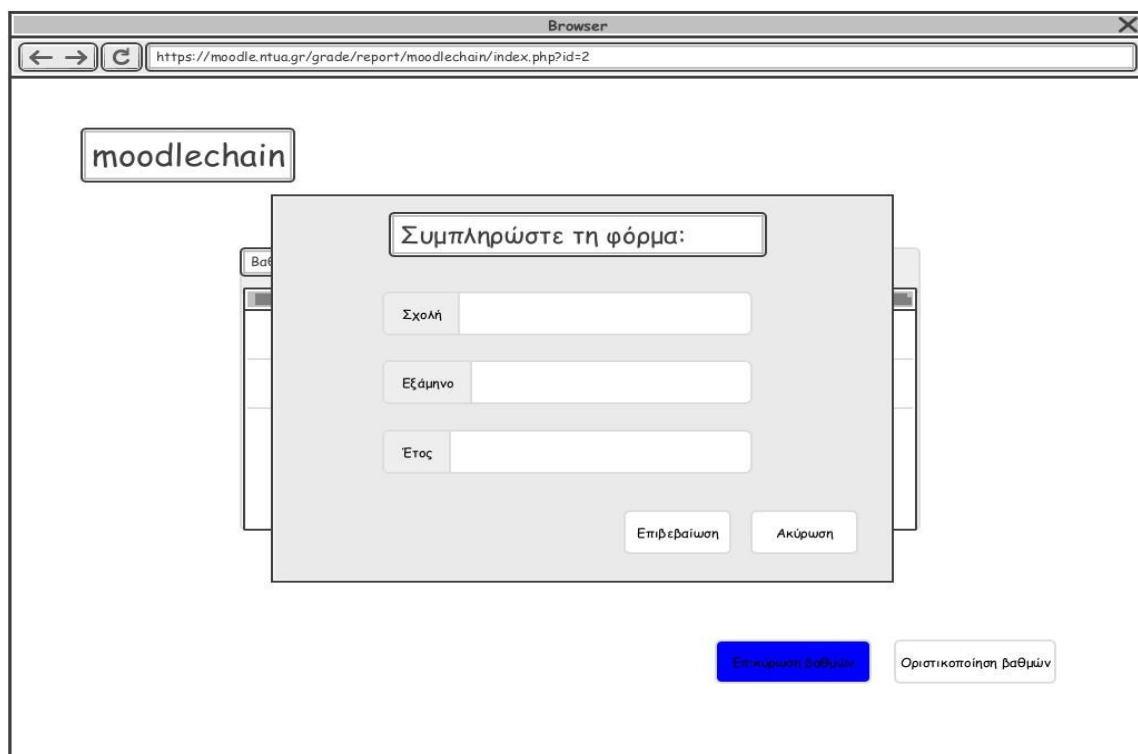


Εικόνα 22 Διάγραμμα ακολουθίας επικύρωσης βαθμών

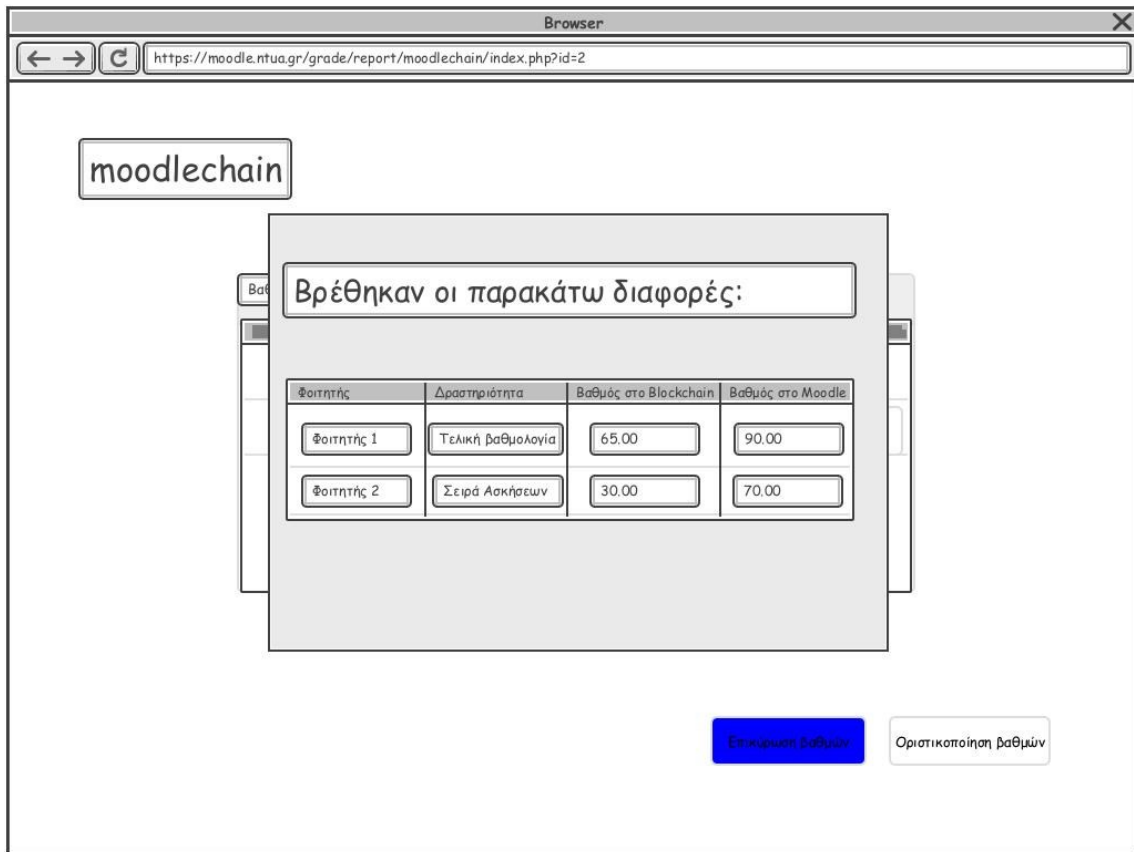
4.8.3 Wireflow διαγράμματα οριστικοποίησης



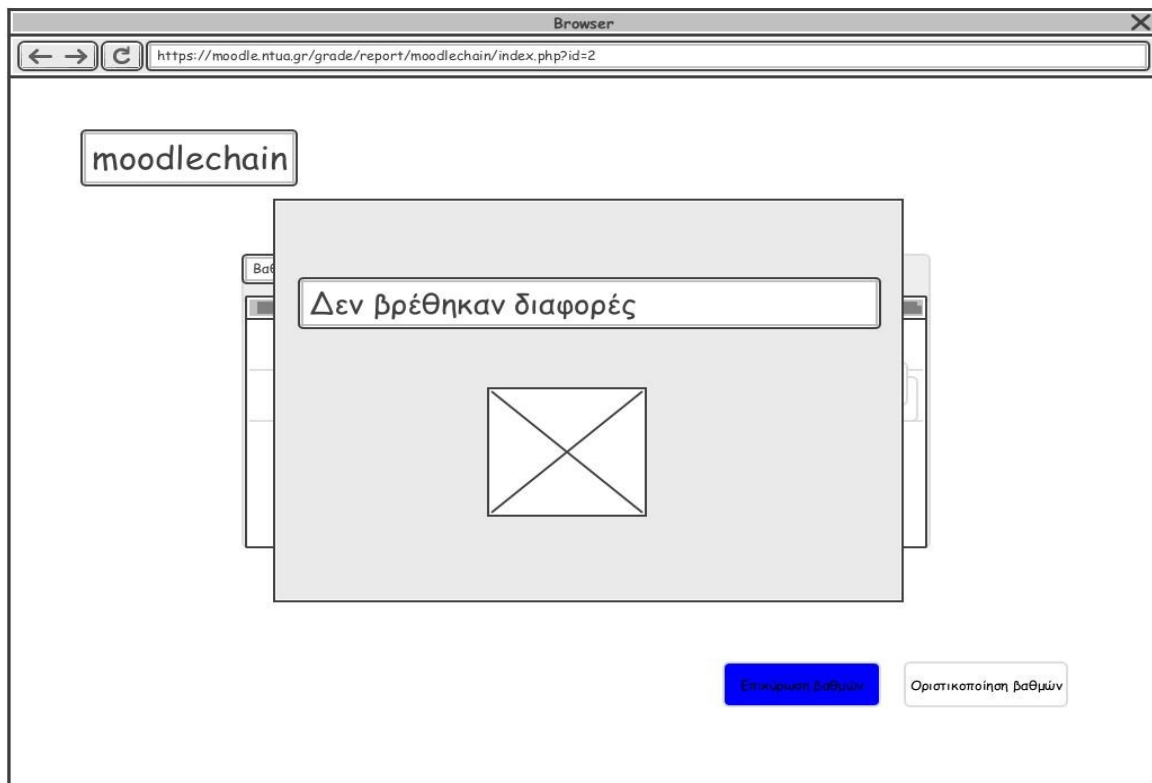
Εικόνα 23 Διάγραμμα wireflow επικύρωσης βαθμών: Επιλέξτε δραστηριότητα για επικύρωση



Εικόνα 24 Διάγραμμα wireflow επικύρωσης βαθμών: Συμπλήρωση στοιχείων μαθήματος

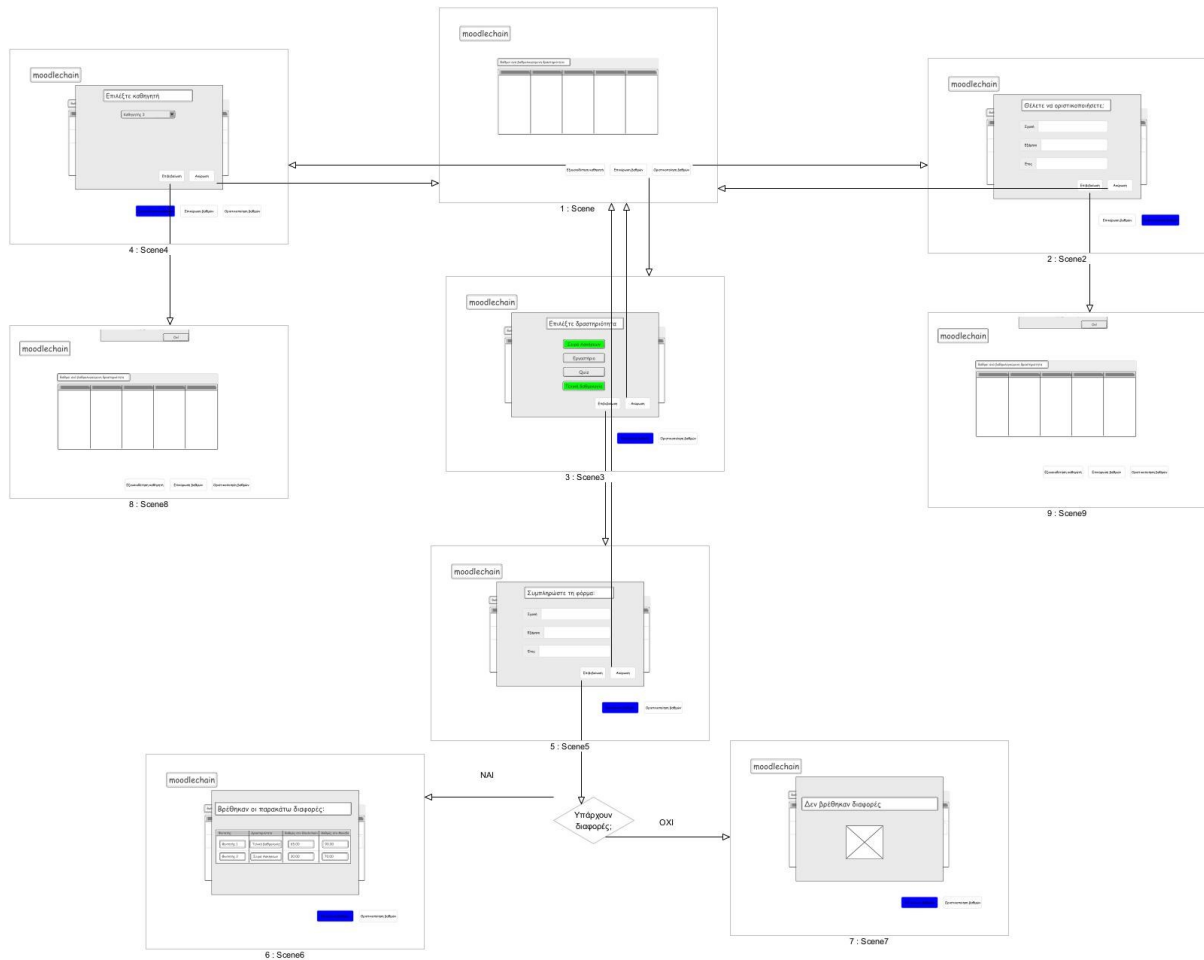


Εικόνα 25 Διάγραμμα wireflow επικύρωσης βαθμών: Πίνακας διαφορών



Εικόνα 26 Διάγραμμα wireflow επικύρωσης βαθμών: Δεν βρέθηκαν διαφορές

4.9 Συνολικό wireflow διάγραμμα



Εικόνα 27 Συνολικό wireflow διάγραμμα

Κεφάλαιο 5 Υλοποίηση

Σε αυτό το κεφάλαιο θα δούμε όλα τα βήματα που ακολουθήσαμε για να υλοποιήσουμε όλες τις απαιτήσεις που περιγράψαμε στο παραπάνω κεφάλαιο, από εγκαταστάσεις του απαιτούμενου λογισμικού και ρυθμίσεις έως την ανάπτυξη κώδικα.

5.1 Εγκατάσταση Moodle

Το πρώτο απαραίτητο βήμα για να ξεκινήσει η ανάπτυξη του plugin, είναι να εγκαταστήσουμε και να «σηκώσουμε» ένα Moodle site, πάνω στο οποίο θα αρχίσουμε να χτίζουμε το plugin. Για να εγκαταστήσουμε το Moodle σε λειτουργικό Linux, ακολουθήσαμε οδηγίες που βρέθηκαν στην τεκμηρίωση του Moodle.[26] Αρχικά έπρεπε να τρέξουμε κάποιες εντολές για να εγκαταστήσουμε το απαιτούμενο λογισμικό:

```
sudo apt install apache2 mysql-client mysql-server php7.4 libapache2-mod-php7.4
```

```
sudo apt install graphviz aspell ghostscript clamav php7.4-pspell php7.4-curl php7.4-gd php7.4-intl php7.4-mysql php7.4-xml php7.4-xmlrpc php7.4-ldap php7.4-zip php7.4-soap php7.4-mbstring
```

Όπως φαίνεται χρησιμοποιούμε τα συστατικά που αναφέρθηκαν στο κεφάλαιο 2.4.1 όπως ο Apache server, η MySQL βάση και η php ως γλώσσα προγραμματισμού. Στη συνέχεια αφού εγκαταστήσαμε το λογισμικό διαχείρισης MySQL βάσεων δεδομένων με την εντολή:

```
sudo mysql_secure_installation
```

Προχωράμε στο κατέβασμα του πηγαίου κώδικα του Moodle. Αρχικά χρειάστηκε να εγκαταστήσουμε το σύστημα ελέγχου εκδόσεων git:

```
sudo apt install git
```

Και στη συνέχεια κατεβάσαμε με αυτόν τον τρόπο τον πηγαίο κώδικα που moodle από το επίσημο αποθετήριο(repository):


```
sudo git clone git://git.moodle.org/moodle.git
```

Στην συνέχεια επιλέξαμε το “branch” του repository που αντιστοιχεί στην έκδοση που θα χρησιμοποιήσουμε. Εδώ επιλέξαμε την τελευταία 4.0 έκδοση του Moodle.

```
cd moodle  
sudo git branch --track MOODLE_400_STABLE origin/MOODLE_400_STABLE  
sudo git checkout MOODLE_400_STABLE
```

Αφού επιλέξαμε έκδοση που θέλουμε και κάναμε την αντίστοιχη αλλαγή (checkout) στο branch, θα χρειαστεί να αντιγράψουμε τον πηγαίο κώδικα του Moodle στην τοποθεσία /var/www/html/. Η συγκεκριμένη τοποθεσία γνωστή και ως “web root” για το Moodle, είναι η προεπιλεγμένη τοποθεσία από όπου τρέχει το περιεχόμενο του ένας Apache server. Στη συνέχεια θα πρέπει να δημιουργήσουμε τη βάση δεδομένων που θέλουμε για το Moodle. Συνδεόμαστε σαν root χρήστης στην κονσόλα του συστήματος διαχείρισης βάσεων:

```
sudo mysql -u root -p
```

Και για να δημιουργήσουμε τη βάση πληκτρολογούμε:

```
CREATE DATABASE moodle DEFAULT CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci;
```

Στη συνέχεια είναι αναγκαίο να δημιουργήσουμε έναν χρήστη στο σύστημα βάσεων μέσω του οποίου το Moodle site μας, θα αποθηκεύει αλλαγές στη βάση:

```
create user 'moodledude'@'localhost' IDENTIFIED BY '*****';
```

Τα στοιχεία αυτά θα τα χρειαστούμε παρακάτω. Και να του δώσουμε δικαιώματα στη βάση:

```
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,CREATE TEMPORARY  
TABLES,DROP,INDEX,ALTER ON moodle.* TO 'moodledude'@'localhost';
```

Για να ολοκληρωθεί η εγκατάσταση μπορούμε να ανοίξουμε έναν browser στη διεύθυνση <http://IP.ADDRESS.OF.SERVER/moodle> συμπληρώνοντας την αντίστοιχη διεύθυνση. Αν όλα έχουν γίνει σωστά θα πρέπει να οδηγηθούμε στην παρακάτω οθόνη:

Installation

Language

Choose a language

Please choose a language for the installation. This language will also be used as the default language for the site, though it may be changed later.

Language

Next >



Εικόνα 28 Εγκατάσταση Moodle, αρχική σελίδα

Αρκεί να προχωρήσουμε με τις προεπιλεγμένες ρυθμίσεις και να συμπληρώσουμε στην κατάλληλη οθόνη τα στοιχεία του χρήστη της βάσης που δημιουργήσαμε παραπάνω, για να ολοκληρώσουμε την εγκατάσταση.

Database

Database settings

Improved MySQL (native/mysqli)

The database is where most of the Moodle settings and data are stored and must be configured here.

The database name, username, and password are required fields; table prefix is optional.

The database name may contain only alphanumeric characters, dollar (\$) and underscore (_).

If the database currently does not exist, and the user you specify has permission, Moodle will attempt to create a new database with the correct permissions and settings.

Database host

Database name

Database user

Database password

Tables prefix

Database port

Unix socket

< Previous Next >

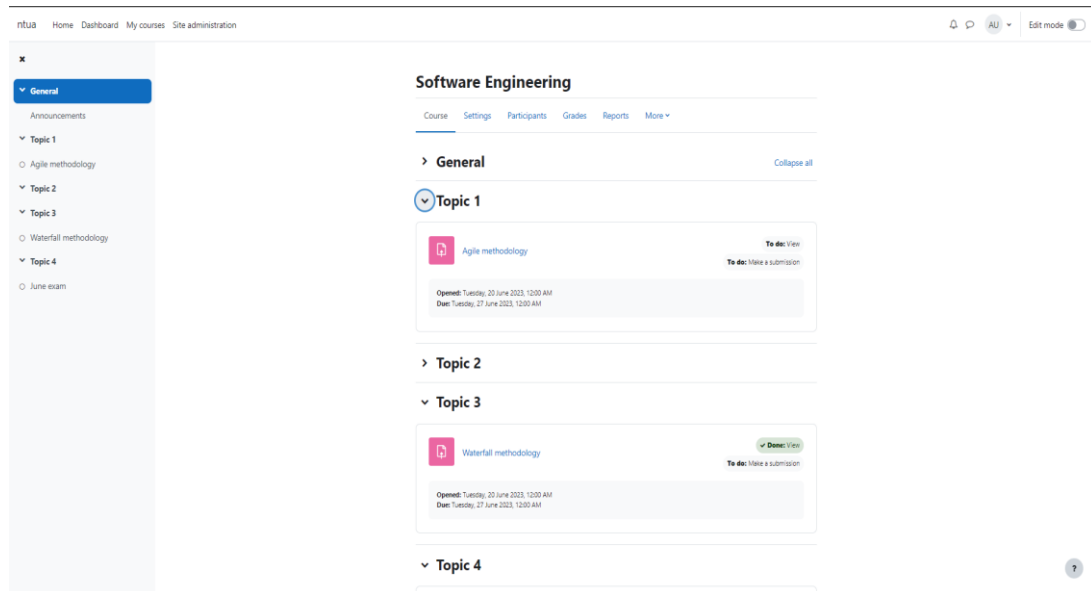
Εικόνα 29 Εγκατάσταση Moodle, ρύθμιση βάσης δεδομένων

5.1.1 Δημιουργία μαθήματος

Αφού ολοκληρώσουμε την εγκατάσταση και δημιουργήσαμε τον χρήστη-διαχειριστή, το επόμενο μας βήμα είναι να «γεμίσουμε» αυτήν την εγκατάσταση Moodle με περιεχόμενο, που θα μας

βοηθήσει κατά την ανάπτυξη του plugin, ενώ στη συνέχεια θα μας χρειαστεί ώστε να επιδείξουμε τις λειτουργίες αφού ολοκληρώσουμε την ανάπτυξη.

Αρχικά μέσω της ιστοσελίδας και ως διαχειριστής δημιουργήσαμε ένα μάθημα, ενώ προσθέσαμε μερικές βαθμολογούμενες δραστηριότητες.



Εικόνα 30 Δημιουργία μαθήματος στο Moodle

Επόμενο βήμα, είναι να προσθέσουμε χρήστες στο site και να τους εγγράψουμε στο μάθημα. Το Moodle μας δίνει την δυνατότητα να ανεβάσουμε ένα αρχείο για να εισάγουμε πολλούς χρήστες ταυτόχρονα στο site. Γι' αυτό λοιπόν δημιουργήθηκε το παρακάτω πρόγραμμα (script) σε JavaScript με το οποίο δημιουργούμε το αρχείο χρηστών.

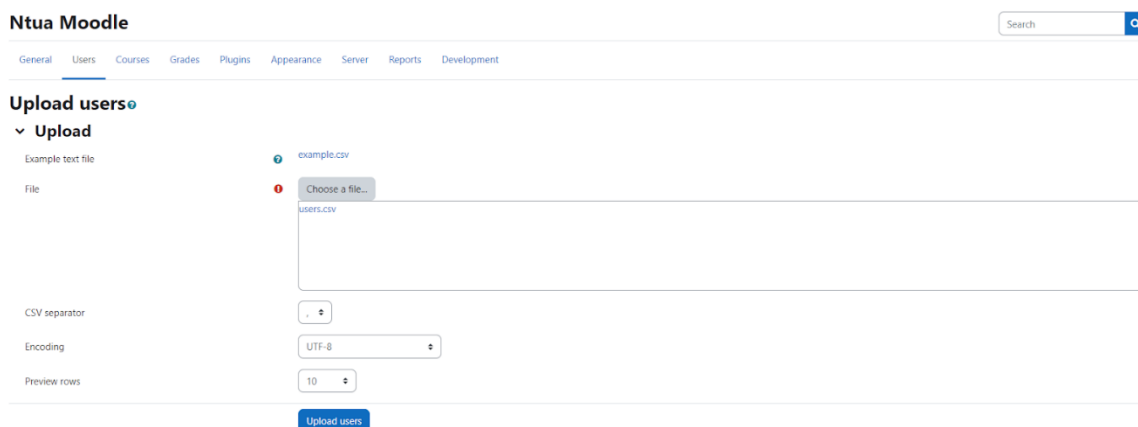
```
const fs = require('fs');
const faker = require('faker');
const NUM_ROWS = 200; // number of rows to generate
// create write stream
const stream = fs.createWriteStream('users.csv');
// write header row
stream.write('username,firstname,lastname,email\n');
// generate fake data
for (let i = 0; i < NUM_ROWS; i++) {
  const username = faker.internet.userName();
  const firstname = faker.name.firstName();
  const lastname = faker.name.lastName();
  const email = faker.internet.email();
  stream.write(`${username},${firstname},${lastname},${email}\n`);
} // close stream
stream.end();
```

Με τη σταθερά NUM_ROWS, επιλέγουμε το πλήθος των χρηστών που θα δημιουργήσουμε ενώ χρησιμοποιήσαμε τη βιβλιοθήκη “faker” της Javascript για την παραγωγή δεδομένων. Το τελικό αρχείο μοιάζει κάπως έτσι:

```
1 username,firstname,lastname,email
2 Rupert_Yost29,Nya,Mraz,Mireya_Terry@hotmail.com
3 Watson_Towne88,Anya,Dickens,Travon72@gmail.com
4 Reece.Romaguera,Heber,Schoen,Blanca_Bosco@gmail.com
5 Conner.Klein8,Maiya,Reichert,Dante_Schmidt@hotmail.com
6 Zakary.Wisoky,Jerel,Harvey,Mariah_Goyette@yahoo.com
7 Daisy_Wyman,Demetris,Upton,Sherwood7@yahoo.com
8 Payton.Schaden98,Lennie,Kassulke,Verla.Reinger8@yahoo.com
9 Ila_Hane,Genevieve,Monahan,Jerrell.Ryan90@hotmail.com
10 Claud Blanda,Makayla,Rosenbaum,Rachel Weissnat@yahoo.com
```

Εικόνα 31 Μορφή αρχείου χρηστών

Και στη συνέχεια ανεβάζουμε το αρχείο στο Moodle:



Εικόνα 32 Ανέβασμα αρχείου χρηστών

Επιπλέον, για να μη χρειαστεί να εγγράψουμε όλους τους χρήστες στο μάθημα «έναν-έναν» χρησιμοποιήσαμε ένα plugin που υπάρχει στο Moodle plugin directory, που δίνει τη δυνατότητα για εγγραφή σε μαθήματα πολλών χρηστών ταυτοχρόνως. Το plugin ονομάζεται “user bulk enrolment” [27] και ουσιαστικά απαιτεί μια λίστα με τα email των χρηστών τους οποίους πρέπει να εισάγει στο μάθημα. Για να φτιάξουμε αυτή τη λίστα, δημιουργήσαμε ένα άλλο πρόγραμμα που διαβάζει το αρχείο χρηστών και τη δημιουργεί.

```

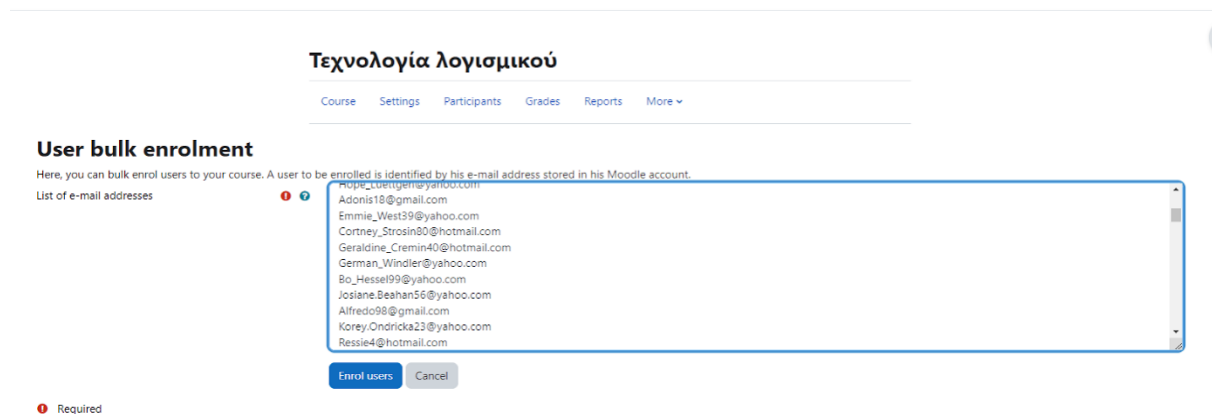
const fs = require('fs');
const parse = require('csv-parse');

const input = fs.createReadStream('users.csv');
const output = fs.createWriteStream('emails.txt');
const parser = parse({ columns: true });

input.pipe(parser).on('data', (row) => {
  output.write(row.email + '\n');
});

```

Και στη συνέχεια ανεβάσαμε τα email των χρηστών:



Εικόνα 33 Bulk Enrolment plugin, αρχική σελίδα

Τελευταίο μας βήμα ήταν να ορίσουμε καθηγητές και φοιτητές στο μάθημα και να βαθμολογήσουμε τους φοιτητές. Η τελική βαθμολογία του μαθήματος, φαίνεται παρακάτω:

All participants: 200/200

First name:

Last name:

1 2 3 4 5 6 7 8 9 10 ... 40 >

First name / Last name ^		Email address	Software Engineering			
			Agile Methodology	Waterfall methodology	Final exam	Course total
PA	Presley Ankunding	Markus.King@gmail.com	✗ 1.00	✓ 60.00	100.00	86.10
DA	Dario Armstrong	Ariane.Cassin11@gmail.com	✓ 100.00	✓ 55.00	90.00	87.50
EA	Eulalia Armstrong	Marley39@yahoo.com	✓ 50.00	✓ 55.00	50.00	50.50
SB	Sally Batz	Rickey.Beier@gmail.com	✓ 60.00	✓ 55.00	40.00	43.50
GB	Glenda Beahan	Nicklaus7@gmail.com	✓ 60.00	✓ 55.00	80.00	75.50
Overall average			55.64	69.57	64.94	64.72

Εικόνα 34 Βαθμολογία μαθήματος Software Engineering

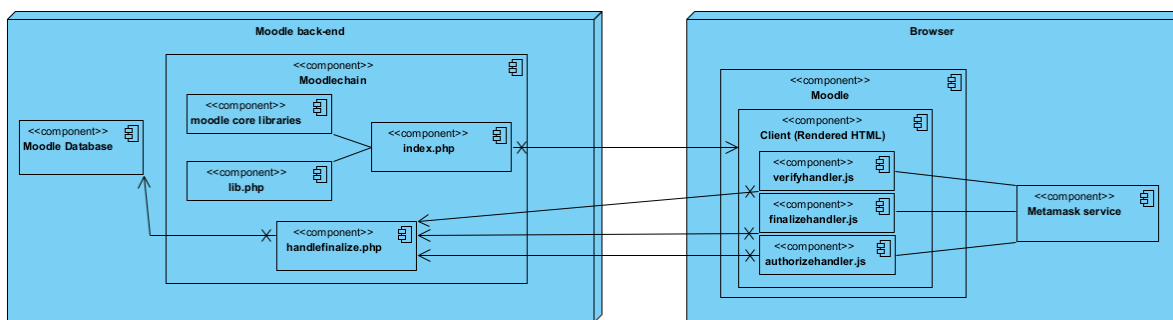
5.2 Ανάπτυξη επιπρόσθετου στοιχείου – Moodlechain

5.2.1 Τύπος plugin και δομή

Όπως προαναφέρθηκε στο Κεφάλαιο 2, στην πλατφόρμα του Moodle υπάρχουν κάποιοι προκαθορισμένοι τύποι plugin. Από τους διάφορους τύπους επιλέχθηκε ο “grade-report”. Το κριτήριο επιλογής είναι η λειτουργικότητα που παρέχει το στοιχείο καθώς και τα κοινά στοιχεία που έχει με τα άλλα plugins αυτού του τύπου. Έτσι λοιπόν η ανάπτυξη του στοιχείου γίνεται μέσα στο φάκελο “moodle\gradelreport” του πηγαίου κώδικα του Moodle όπου εκεί υπάρχει ένας ξεχωριστός φάκελος για κάθε plugin αυτού του τύπου. Παρακάτω βλέπουμε τη δομή του στοιχείου:

```
[moodlechain]
├── [css]
│   ├── finalize_button.css
│   ├── styles.css
│   └── table.css
├── [db]
│   ├── access.php
│   └── install.xml
├── handleFinalize.php
├── index.php
├── [js]
│   ├── [handlers]
│   │   ├── authorizeHandler.js
│   │   ├── finalizeHandler.js
│   │   └── verifyHandler.js
│   └── [helpers]
│       ├── accessContract.js
│       ├── contractConfig.js
│       └── popUpWindows.js
├── [lang]
│   ├── [el]
│   │   └── gradereport_moodlechain.php
│   └── [en]
│       └── gradereport_moodlechain.php
├── lib.php
├── [templates]
│   └── allButtons.mustache
└── version.php
```

Εικόνα 35 Moodlechain-Δομή plugin



Εικόνα 36 Moodlechain διάγραμμα συστατικών

Στη συνέχεια θα γίνει μια μικρή ανάλυση στα αρχεία του πηγαίου κώδικα, τηρώντας τη λογική σειρά. Ο πηγαίος κώδικας θα παρέχεται όπου αυτό κρίνεται σημαντικό:

version.php: Το πρώτο αρχείο που πρέπει να φτιάξει κανείς όταν αναπτύσσει στοιχεία στο Moodle. Περιέχει ένα σύνολο ιδιοτήτων οι οποίες χρησιμοποιούνται κατά την αρχική εγκατάσταση του Plugin καθώς και σε πιθανές αναβαθμίσεις. Σε αυτό το αρχείο δηλώνουμε ποια ελάχιστη έκδοση του Moodle χρειάζεται το Plugin για να λειτουργήσει, σε ποια έκδοση βρίσκεται τώρα καθώς και το όνομα του Plugin. Επίσης η μεταβλητή MATURITY χρησιμεύει στους Moodle admins ώστε να ξέρουν το στάδιο ανάπτυξης του plugin.

```
defined('MOODLE_INTERNAL') || die();

$plugin->version = 2023013000; // The current plugin version
(Date: YYYYMMDDXX)
$plugin->requires = 2014051200; // Requires this Moodle version
$plugin->component = 'gradereport_moodlechain'; // Full name of the plugin
(used for diagnostics)
$plugin->maturity = MATURITY_STABLE;
```

DB/access.php: Σε αυτό το αρχείο καθορίζονται τα «capabilities» που ορίζονται με αυτό το plugin. Τα capabilities στο Moodle, είναι αυτά που σχετίζουν ορισμένους ρόλους με τα δικαιώματα που έχουν να κάνουν κάποιες ενέργειες στο plugin. Πιο συγκεκριμένα για το plugin ορίστηκε 1 capability που καθορίζει τη χρήση του και δόθηκε μόνο στους καθηγητές ενός μαθήματος. Έτσι λοιπόν το plugin αυτό είναι εμφανές μόνο στους καθηγητές και στον διαχειριστή του συστήματος.

```

defined('MOODLE_INTERNAL') || die();
$capabilities = array(
    'gradereport/moodlechain:view' => array(
        'riskbitmask' => RISK_PERSONAL,
        'captype' => 'read',
        'contextlevel' => CONTEXT_COURSE,
        'archetypes' => array(
            'teacher' => CAP_ALLOW,
            'editingteacher' => CAP_ALLOW,
            'manager' => CAP_ALLOW
        )
    ),
);

```

DB/install.xml: Με αυτό το αρχείο μπορεί κανείς να ορίσει αν χρειάζεται πίνακες για τη βάση δεδομένων του Moodle που να σχετίζονται με το plugin. Ορίσαμε λοιπόν έναν πίνακα για να καταγράψουμε την δραστηριότητα που υπάρχει. Σε κάθε γραμμή λοιπόν αποθηκεύουμε ποιος Moodle user, έκανε ποια δραστηριότητα (προσθήκη βαθμών στο blockchain, επικύρωση βαθμών ή εξουσιοδότηση καθηγητή) καθώς και σε ποιο μάθημα.

```

<TABLES>
  <TABLE NAME="gradereport_moodlechain_history" COMMENT="each record is a message">
    <FIELDS>
      <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
      <FIELD NAME="userid" TYPE="text" LENGTH="10" NOTNULL="true" />
      <FIELD NAME="courseid" TYPE="text" LENGTH="10" NOTNULL="true" />
      <FIELD NAME="teacherid" TYPE="text" LENGTH="10" NOTNULL="false" />
      <FIELD NAME="schoolid" TYPE="text" LENGTH="10" NOTNULL="false" />
      <FIELD NAME="semesteryearcourse" TYPE="text" NOTNULL="false"/>
      <FIELD NAME="actiontype" TYPE="int" LENGTH="10" NOTNULL="true" COMMENT="0 for addGrades, 1 for verify, 2 for authorizeteacher"/>
      <FIELD NAME="time" TYPE="int" LENGTH="10" NOTNULL="true"/>
    </FIELDS>
    <KEYS>
      <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
    </KEYS>
  </TABLE>
</TABLES>

```


handleFinalize.php: Αυτό το script, καλείται αφού ολοκληρώνεται οποιαδήποτε δραστηριότητα στο plugin και χειρίζεται την αποθήκευση των δεδομένων που αναλύσαμε παραπάνω στον πίνακα καταγραφής που ορίσαμε για το plugin.

```
require_once(' ../../../../config.php');
require_once($CFG->libdir . '/csvlib.class.php');
global $USER, $DB;
$userid = $USER->id;
$actionArray = array(
    'addGrades'=> 0,
    'verifyGrades'=> 1,
    'authorizeTeacher'=> 2,
);
$action = required_param('action', PARAM_TEXT);
$isAuthorize = $action == 'authorizeTeacher';

$courseid = required_param('courseId', PARAM_TEXT);
$schoolId = $isAuthorize ? "" : required_param('schoolId', PARAM_TEXT);
$semesterYearCourse = $isAuthorize ? "" :
required_param('semesterYearCourse', PARAM_TEXT);
$record = new stdClass();
$record->userid = $userid;
$record->courseid = $courseid;
$record->teacherid = $isAuthorize ? required_param('teacherId', PARAM_TEXT)
: "";
$record->schoolid = $isAuthorize ? "" : $schoolId;
$record->semesteryearcourse = $isAuthorize ? "" : $semesterYearCourse;
$record->actiontype = $actionArray[$action];
$record->time = time();
var_dump($record);
echo $DB->insert_record('gradereport_moodlechain_history', $record);
```

lib.php: Εδώ ορίζουμε το object που χρειαζόμαστε για την εισαγωγή των βαθμών στο Blockchain, αλλά και κατά την επικύρωση. Πιο συγκεκριμένα, στο αρχείο επεκτείνουμε μια υπάρχουσα κλάση που χρησιμοποιείται στα grade report τύπου plugins η οποία παρέχει συναρτήσεις και αντικείμενα για ανάκτηση βαθμολογιών του μαθήματος, ανά χρήστη και ανά βαθμολογούμενη δραστηριότητα. Συνεπώς χρησιμοποιούμε ήδη ορισμένες συναρτήσεις κάνοντας μικρές προσαρμογές για να εξάγουμε το τελικό αντικείμενο που μετέπειτα θα στείλουμε στο smart contract. Επίσης χρησιμοποιούμε αυτό το αντικείμενο, για να συγκρίνουμε τις βαθμολογίες που υπάρχουν στο σύστημα, με τις βαθμολογίες που παίρνουμε πίσω από το Blockchain. Το grade object που δημιουργούμε από αυτό το αρχείο έχει την παρακάτω μορφή:

```
[
  {
    "activityName": "exam",
    "grades": [
      {
        "studentId": "191",
        "studentName": "Dario Armstrong",
        "email": "Ariane.Cassin11@gmail.com",
        "activityName": "exam",
        "submittedOn": "2023-05-30 14:13:11",
        "gradedOn": "2023-05-31 17:14:31",
        "gradedBy": "Admin",
        "rawGrade": "50.00000"
      }
    ],
  },
  {
    "activityName": "Final Grade",
    "grades": [
      {
        "studentId": "191",
        "studentName": "Dario Armstrong",
        "email": "Ariane.Cassin11@gmail.com",
        "activityName": "Final Grade",
        "submittedOn": "2023-05-31 17:14:31",
        "gradedOn": "2023-05-31 17:14:31",
        "gradedBy": "System",
        "rawGrade": "50.00000"
      }
    ],
  }
]
```

Όπως βλέπουμε, πρόκειται για έναν πίνακα όπου το μήκος του ισοδυναμεί με τον αριθμό των βαθμολογούμενων δραστηριοτήτων που υπάρχουν στο μάθημα, συνυπολογίζοντας και την τελική βαθμολογία ως ένα από αυτά. Κάθε στοιχείο του πίνακα περιέχει κάποια πεδία το activityName το

όνομα δηλαδή της δραστηριότητας στην οποία αντιστοιχεί καθώς και το πεδίο με «κλειδί» grades που αντιστοιχεί σε έναν πίνακα με μήκος ίσο με τον αριθμό των φοιτητών που έχουν βαθμολογηθεί και κάθε στοιχείο του περιέχει μεταξύ άλλων δεδομένα όπως το id και το όνομα του φοιτητή και την βαθμολογία του στη συγκεκριμένη δραστηριότητα.

index.php: Αυτό το αρχείο λειτουργεί ως το «σημείο εισόδου» για το plugin. Εδώ ορίζουμε το user interface για το plugin, που περιέχει έναν πίνακα με τις βαθμολογίες των μαθητών. Ο πίνακας δημιουργείται με χρήση της grade_report κλάσης που είναι ήδη ορισμένη σε άλλα plugins. Επίσης σε αυτό το αρχείο, πέρα από τον πίνακα των βαθμών εμφανίζουμε στο χρήστη τα 2 (ή 3 για τον διαχειριστή) κουμπιά μέσω των οποίων μπορεί να επιλέξει μια από τις λειτουργίες που παρέχονται από το plugin. Αυτό το αρχείο περιέχει php κώδικα για τον ορισμό των απαιτούμενων αντικειμένων (βαθμολογίες, χρήστες κ.α.) αλλά χρησιμοποιείται και html για την εμφάνιση των αντικειμένων στο χρήστη.

templates/allButtons.mustache: Σε αυτό το αρχείο ορίζουμε τα 3 κουμπιά. Χρησιμοποιούμε την σύνταξη κατά mustache για την δημιουργία της html για το κάθε κουμπί αλλά φορτώνοντας κάθε φορά, δυναμικά, διαφορετικά δεδομένα. Κάθε κουμπί «κουβαλά» τα ανάλογα δεδομένα και έχει τον αντίστοιχο event-handler που καλείται κάθε φορά που κάποιος το πατά. Οι handlers είναι γραμμένοι σε Javascript, φορτώνονται και τρέχουν στον browser του client. Παρακάτω βλέπουμε τον κώδικα του template, για την εμφάνιση των κουμπιών. Χρησιμοποιούμε την showAll μεταβλητή που έχει καθοριστεί από το index.php για να δείχνουμε ή όχι το authorize κουμπί μόνο στον admin. Επίσης βλέπουμε τις μεταβλητές που φορτώνουμε στον dom μέσω του template, ώστε να τις χρησιμοποιήσει ο κάθε handler στη συνέχεια.

```
<div class = "button-container">
  {{#showAll}}
    <button id="authorize_button" data-course-id="{{courseId}}" data-
      grades="{{grades}}" data-teachers="{{teachers}}"

      class="finalize_button">{{authorizeButtonText}}
    </button>
    <script type="module"
      src="../../moodlechain/js/handlers/authorizeHandler.js">
    </script>
  {{/showAll}}
  <button id="verify_button" data-course-id="{{courseId}}" data-
    grades="{{grades}}"
  class="finalize_button">{{verifyButtonText}}
  </button>
  <script type="module"
    src="../../moodlechain/js/handlers/verifyHandler.js">
  </script>
  <button id="finalize_button" data-course-id="{{courseId}}" data-
    grades="{{grades}}" data-user-Id="{{userId}}"
    class="finalize_button">{{finalizeButtonText}}
  </button>
  <script type="module"
    src="../../moodlechain/js/handlers/finalizeHandler.js">
  </script>
</div>
```

js/helpers/contractConfig.js: Σε αυτό το αρχείο ορίζουμε στοιχεία σχετικά με το contract. Πρόκειται για παραμετροποιήσεις οι οποίες είναι απαραίτητες για την αλληλεπίδραση με το έξυπνο συμβόλαιο και περιλαμβάνει μεταβλητές όπως contract address, contract abi, network. Οι τιμές αυτών των μεταβλητών εξαρτώνται από την εγκατάσταση και το που έχουμε στήσει το Blockchain δίκτυο, οπότε ο διαχειριστής θα πρέπει κάθε φορά να τις ρυθμίζει εκ νέου.

js/helpers/accessContract.js: Σε αυτό το αρχείο και χρησιμοποιούμε την βιβλιοθήκη ethers.js καθώς και τις παραπάνω παραμετροποιήσεις για να αρχικοποιήσουμε τα αντικείμενα που χρειαζόμαστε για την αλληλεπίδραση με το έξυπνο συμβόλαιο. Πιο συγκεκριμένα αρχικά ελέγχουμε αν είναι εγκατεστημένο το Metamask extension στο browser και στη συνέχεια (αν πετύχει ο έλεγχος) περιμένουμε μέχρι να επιλεγθεί ο λογαριασμός Metamask που θα χρησιμοποιηθεί από τον χρήστη ώστε να γίνει η αρχικοποίηση του contract.

```
if (window.ethereum) {
  try {
    await ethereum.request({ method: 'eth_requestAccounts' })
      .then((accounts) => accounts[0])
      .catch((err) => console.Log(err));
    const provider = new ethers.BrowserProvider(window.ethereum)
    const providerNetwork = await provider.getNetwork();
    if (providerNetwork.chainId !== network.chainId) {
      console.Log(M.str.gradereport_moodlechain.wrongNetwork,
        providerNetwork.chainId);
      return;
    }
  }
}
```

Βλέπουμε πως όταν ολοκληρωθεί η αρχική αλληλεπίδραση του χρήστη με το Metamask παράθυρο, έχει αρχικοποιηθεί το αντικείμενο provider. Το συγκεκριμένο αντικείμενο είναι απαραίτητο για την αλληλεπίδραση με έναν κόμβο κάποιου Ethereum δικτύου. Ουσιαστικά μόλις ο χρήστης ρυθμίσει και συνδεθεί σε ένα δίκτυο Blockchain μέσω του παραθύρου του Metamask δημιουργείται το provider object, με το οποίο επιτρέπεται στο χρήστη να στείλει συναλλαγές και να διαβάσει δεδομένα από το Blockchain δίκτυο. Στη συνέχεια ελέγχουμε αν το δίκτυο του provider είναι ίδιο με αυτό που υπάρχει στις ρυθμίσεις contractConfig. Αυτός ο έλεγχος γίνεται σαν δικλείδα ασφαλείας, σε περίπτωση που ο χρήστης δεν έχει ρυθμίσει σωστά το δίκτυο που θέλει να συνδεθεί από το Metamask.

```
const signer = await provider.getSigner();
const contract = new ethers.Contract(address, abi, signer);
console.Log(M.str.gradereport_moodlechain.connectedToContract)
;
return contract;
}
catch (err) {
  alert(err.message);
}
else {
  alert(M.str.gradereport_moodlechain.metamaskNotInstalled);
}
}
```

Αν πάνε όλα καλά με την αρχικοποίηση του provider και αφού ο χρήστης επιλέξει ένα λογαριασμό αρχικοποιείται το αντικείμενο signer. Ο signer αντιπροσωπεύει ένα λογαριασμό Ethereum

και έχει τη δυνατότητα να υπογράψει συναλλαγές και μηνύματα χρησιμοποιώντας το κρυφό κλειδί που σχετίζεται με τον λογαριασμό. Στη συνέχεια θα χρειαστούμε τις παραμέτρους που ορίσαμε στο contractConfig αρχείο για να ορίσουμε το contract object. Το contract abi είναι μια αναπαράσταση σε μορφή JSON της διεπαφής του συμβολαίου που ορίζει τις συναρτήσεις(functions) και τα γεγονότα(events) του συμβολαίου. Χρησιμοποιείται από τη βιβλιοθήκη για να κωδικοποιεί τα δεδομένα που στέλνουμε και να αποκωδικοποιήσει τα δεδομένα που λαμβάνουμε όταν αλληλοεπιδρούμε με το συμβόλαιο. Επίσης αναγκαία μεταβλητή είναι η δεκαεξαδική διεύθυνση του συμβολαίου με το οποίο θέλουμε να αλληλοεπιδράσουμε. Χρησιμοποιώντας όλα τα παραπάνω αρχικοποιείται το contract object, το οποίο θα χρησιμοποιούμε για οποιαδήποτε ανταλλαγή δεδομένων με το Blockchain. (Θα δούμε πως ορίζονται τα contract abi και address στο κεφάλαιο 5.4)

js/helpers/popUpWindows.js: Σε αυτό το αρχείο ορίζουμε διάφορες συναρτήσεις κάθε μια από τις οποίες μας παρέχει ένα αναδυόμενο παράθυρο για να υποστηρίξουμε τις 3 λειτουργίες του plugin. Για την δημιουργία των αναδυόμενων παραθύρων χρησιμοποιήσαμε την βιβλιοθήκη SweetAlert2 που παρείχε έναν εύκολο και εύχρηστο τρόπο να προβάλλουμε λειτουργικά αναδυόμενα παράθυρα.

js/handlers/finalizeHandler.js: Αυτός ο handler καλείται όταν ο χρήστης πατήσει το κουμπί *Finalize Grades*, δηλαδή όταν επιχειρήσει να προσθέσει βαθμούς στο Blockchain. Αρχικά (και αφού έχουμε λάβει από δεδομένα όπως το object με τους βαθμούς που ορίσαμε στο lib.php, το id του μαθήματος κ.α.) καλεί τις ανάλογες συναρτήσεις από το "popUpWindows" για να εμφανίσει στο χρήστη αναδυόμενα παράθυρα, όπου θα πληκτρολογήσει κάποια ακόμα δεδομένα σχετικά με τη σχολή και το μάθημα. Αφού ολοκληρωθεί η υποβολή της φόρμας με τα στοιχεία, αρχικοποιείται το contract όπως εξηγήθηκε στο "accessContract" και αποστέλλουμε την συναλλαγή στο Blockchain. Ανάλογα με το αποτέλεσμα του αιτήματος, είτε εμφανίζεται μήνυμα επιτυχίας και η δραστηριότητα καταγράφεται στη βάση καλώντας το "finalizeHandler.php", είτε το ανάλογο μήνυμα λάθους.

```
const results = await getExtraDataPopUp(courseId, jsonData);
if (results === false) {
  return alert(M.str.gradereport_moodlechain.genericFailure);
}
const contract = await accessContract();
console.log("Finalizing grades for course with id "+ parseInt(courseId));
try {
  showLoading();
  await contract.addGrades(userId.toString(), parseInt(courseId),
  results, { gasLimit: 300000000 });
  console.log("Transaction completed, Logging event on database...")
  hideLoading();
  $.ajax({
    type: "POST",
    url: "handleFinalize.php",
    data: {
      action: "addGrades",
      courseId: courseId,
      schoolId: results[0].schoolId,
      semesterYearCourse: results[0].semesterYearCourse,
    },
    success: function (response) {
      console.log("Event logged successfully");
      alert(M.str.gradereport_moodlechain.finalizeSuccess)
    },
    error: ((jqXHR, textStatus, errorThrown) => {
      console.log(jqXHR, textStatus, errorThrown);
      alert(M.str.gradereport_moodlechain.genericFailure);
    })
  });
}
```

[js/handlers/verifyHandler.js](#): Αυτός ο handler καλείται όταν ο χρήστης πατήσει το κουμπί “Verify Grade”, όταν δηλαδή επιχειρήσει να κάνει επικύρωση βαθμών. Αφού έχουμε λάβει τα ανάλογα δεδομένα του μαθήματος όπως βαθμολογίες, course id, βαθμολογούμενες δραστηριότητες, αρχικά εμφανίζουμε ένα παράθυρο στο οποίο ο χρήστης επιλέγει ποια ή ποιες βαθμολογούμενες δραστηριότητες θέλει να επικυρώσει και στη συνέχεια συμπληρώνει κάποια στοιχεία σχετικά με τη σχολή και το μάθημα. Αφού ολοκληρωθεί αυτή η διαδικασία, αρχικοποιείται το contract όπως εξηγήθηκε στο “accessContract” και καλούμε την ανάλογη συνάρτηση του contract για να λάβουμε τους βαθμούς. Αξίζει να σημειωθεί ότι έχουμε δώσει την λειτουργικότητα στο χρήστη, να επιλέξει ποια ή ποιες δραστηριότητες θέλει να επικυρώσει. Συνεπώς όπως βλέπουμε παρακάτω, καλούμε τη συνάρτηση για να λάβουμε τους βαθμούς τόσες φορές, όσα και τα activities που επέλεξε ο χρήστης να επικυρώσει. Ο λόγος που υλοποιήθηκε κατ’ αυτόν τον τρόπο είναι για να μειώσουμε τον όγκο κάθε συναλλαγής σε περίπτωση που δεν θέλουμε να ελέγξουμε κάποια δραστηριότητα. Στη συνέχεια, χρησιμοποιώντας μερικές συναρτήσεις που έχουμε ορίσει, συγκρίνουμε τις βαθμολογίες που μας επιστράφηκαν από το Blockchain, με τις τρέχουσες βαθμολογίες. Ανάλογα με την ύπαρξη ή όχι διαφορών, εμφανίζουμε ένα νέο αναδυόμενο παράθυρο που επιδεικνύει στο χρήστη είτε ότι οι βαθμοί είναι έγκυροι, είτε ότι υπάρχουν διαφορές εμφανίζοντας έναν πίνακα που να δείχνει που ακριβώς βρίσκονται αυτές οι διαφορές.

Αν υπάρξει κάποιο error κατά τη διαδικασία θα εμφανιστεί το ανάλογο μήνυμα λάθους, αλλιώς θα καταγραφεί η δραστηριότητα στη βάση του Moodle καλώντας το “finalizeHandler.php”.


```

const gradingActivities = await selectActivitiesPopUp(activities);
  if (gradingActivities === false) {
    return alert(M.str.gradereport_moodlechain.genericFailure);}
  console.log("Activities selected, retrieving grades from
blockchain...");
  const results = await getExtraDataPopUp(courseId, jsonData);
  if (results === false) {
    return alert(M.str.gradereport_moodlechain.genericFailure);
  }const [semester, year, course] =
results[0].semesterYearCourse.split("_");
  let objectToCompare = transformGrades(results);
  const contract = await accessContract();
  let incostisencias = [];
  showLoading();
  for (let i = 0; i < gradingActivities.length; i++) {
    try {
      const response = await
contract.getGrades(results[0].schoolId.toString(),
results[0].semesterYearCourse, gradingActivities[i].toString());

      if (response.Length == 0) {
        alert("No grades found");
        return;
      }else {
        incostisencias.push(compareGrades(response,
objectToCompare[gradingActivities[i]]));
      }
    }
    catch (err) {
      handleErrors(err);
      return; }
  }
  console.log("Grades retrieved from blockchain, comparing...")
  hideLoading();
  showIncotisencias(incostisencias, semester, year, course);
  console.log("Grades verified, logging event on database...")
$.ajax({
  type: "POST",
  url: "handleFinalize.php",
  data: {
    action: "verifyGrades",
    courseId: courseId,
    schoolId: results[0].schoolId,
    semesterYearCourse: results[0].semesterYearCourse,
  },
  success: function (response) {
    console.log("Event logged successfully");
  },
  error: ((jqXHR, textStatus, errorThrown) => {
    console.log(jqXHR, textStatus, errorThrown);
  })
});

```

js/handlers/authorizeHandler.js: Ο συγκεκριμένος handler καλείται όταν ο διαχειριστής του Moodle θέλει να εξουσιοδοτήσει έναν καθηγητή, για να μπορεί να προσθέσει βαθμολογίες στο Blockchain, όταν δηλαδή πατηθεί το κουμπί, “Authorize teacher”. Αρχικά και αφού έχουμε λάβει τα κατάλληλα δεδομένα, θα εμφανιστεί στο χρήστη ένα αναδυόμενο παράθυρο(“popUpWindows.js”), που περιέχει μια λίστα με τους καθηγητές για να επιλέξει ποιόν θέλει να εξουσιοδοτήσει. Αφού επιλεγεί ο καθηγητής, αρχικοποιούμε το contract object καλώντας το “accessContract” και καλούμε την αντίστοιχη συνάρτηση για την προσθήκη της άδειας του εκάστοτε καθηγητή. Ανάλογα με την έκβαση της συναλλαγής, είτε θα εμφανιστεί μήνυμα επιτυχίας και η δραστηριότητα θα καταγραφεί στην βάση του Moodle, είτε θα εμφανιστεί μήνυμα λάθους.

```
const teacherId = await showTeacherPopUp(teachers);
if (teacherId === false) {
    return alert(M.str.gradereport_moodlechain.genericFailure);
}
console.log(teachers[teacherId] + " selected for authorization,
adding permissions...");
const contract = await accessContract();
try {
    await contract.addPermissions(teacherId.toString(),
parseInt(courseId));
    console.log("Permissions added, logging event on database...")
$.ajax({
    type: "POST",
    url: "handleFinalize.php",
    data: {
        action: "authorizeTeacher",
        teacherId: teacherId,
        courseId: courseId,
    },
    success: function (response) {
        console.log("Event logged successfully");
        alert(M.str.gradereport_moodlechain.authorizedSuccess)
;
    },
    error: ((jqXHR, textStatus, errorThrown) => {
        console.log(jqXHR, textStatus, errorThrown);
        alert(M.str.gradereport_moodlechain.genericFailure);
    })
});
}
```

css: Όλα τα αρχεία σε αυτό το φάκελο συντελούν στο “styling” των HTML στοιχείων που εμφανίζουμε στον χρήστη.

lang/φεν & εl}: Ο φάκελος “lang” σε κάθε Moodle plugin, περιέχει μια λίστα από φακέλους, κάθε ένας εκ των οποίων αντιστοιχεί σε μια γλώσσα η οποία υποστηρίζεται από το plugin. Κάθε ένας εξ αυτών των φακέλων περιέχει αρχεία που έχουν κάποια συγκεκριμένα strings σε αυτή τη γλώσσα, τα οποία φορτώνονται στο interface ανάλογα με την επιλεγμένη γλώσσα. Στο συγκεκριμένο plugin υποστηρίζουμε ελληνικά και αγγλικά, έχοντας φτιάξει τους αντιστοίχους με τα ανάλογα strings.

5.3 Ανάπτυξη έξυπνου συμβολαίου

5.3.1 Remix

Η ανάπτυξη του έξυπνου συμβολαίου, έγινε με την γλώσσα ανάπτυξης Solidity που χρησιμοποιείται κατά κόρον στο Ethereum network και αναλύθηκε παραπάνω. Η διαδικασία ανάπτυξης αρχικά ξεκίνησε στην πλατφόρμα του Remix. Η συγκεκριμένη πλατφόρμα παρέχει ένα δωρεάν περιβάλλον ανάπτυξης για κώδικα Solidity και κρίθηκε πολύ χρήσιμη για την ανάπτυξη στο αρχικό στάδιο. Παρέχει διάφορες εκδόσεις από compilers για να κάνει κανείς πολύ εύκολα compile και debug τα contract καθώς και ένα περιβάλλον προσομοίωσης στο οποίο γίνεται deploy το συμβόλαιο σε ένα test-blockchain ώστε να δοκιμαστεί η αλληλεπίδραση με το συμβόλαιο και η λειτουργικότητα των αναπτυσσόμενων συναρτήσεων.

5.3.2 Δομές, μεταβλητές και συναρτήσεις

Η ανάπτυξη του συμβολαίου έγινε ώστε να υποστηρίζουμε την λειτουργικότητα που υλοποιήσαμε στο Moodle αλλά από την μεριά του δικτύου Blockchain. Οι συναρτήσεις που αναπτύχθηκαν λοιπόν, αφορούν την προσθήκη βαθμολογιών στο Blockchain, την δυνατότητα ανάκτησης τους καθώς και της συντήρησης μιας λίστας από καθηγητών που έχουν δικαιώματα να προσθέσουν βαθμολογίες για κάθε συγκεκριμένο μάθημα. Παρακάτω θα αναλύσουμε τις δομές μεταβλητές και συναρτήσεις που υλοποιήσαμε στο έξυπνο συμβόλαιο.

Ξεκινάμε από το παρακάτω πεδίο, που αντιπροσωπεύει την τιμή της διεύθυνσης master User. Κατά σύμβαση εδώ αυτή τη διεύθυνση θα πρέπει να χρησιμοποιεί ο διαχειριστής του Moodle, καθώς έτσι, όπως θα δούμε στη συνέχεια, εξασφαλίζει δικαίωμα να τρέξει όλες τις συναρτήσεις του συμβολαίου.

```
address masterUser;
```

Κατά το deployment του συμβολαίου, ο παρακάτω constructor καλείται και τρέχει μια φορά:

```
constructor() {
    masterUser = msg.sender;
}
```

Ουσιαστικά χρησιμοποιώντας την ενσωματωμένη συνάρτηση “msg.sender” δίνουμε τιμή στο πεδίο master User, ως την τιμή της διεύθυνσης του χρήστη που έκανε deploy το συμβόλαιο.

Στη συνέχεια ορίζουμε το παρακάτω mapping:

```
//mapping from teacher id to list of courses he is authorized for
mapping(string => uint[]) teacherPermissions;
```

Σκοπός του παραπάνω mapping είναι να αποθηκεύσουμε μια λίστα από μαθήματα στα οποία έχει δικαιώματα να βάλει βαθμούς ο κάθε καθηγητής. Έτσι λοιπόν δημιουργούμε μια αντιστοιχία ανάμεσα σε teacher id's και μια λίστα από ακέραια course id's και τα αποθηκεύουμε στο chain.

Αυτή η αντιστοιχία παίρνει τιμές με την παρακάτω συνάρτηση, που όπως βλέπουμε απαιτεί ο χρήστης να είναι ο masterUser για να αποθηκεύσει τα δεδομένα:

```
function addPermissions(string memory teacherId, uint courseId) public
{
    require(
        msg.sender == masterUser,
        "You dont have permission to do that"
    );
    teacherPermissions[teacherId].push(courseId);
}
```

Ακόμα ορίστηκε η παρακάτω βοηθητική συνάρτηση ώστε να μπορούμε να ελέγξουμε αν κάποιος καθηγητής έχει δικαίωμα να βαθμολογήσει κάποιο μάθημα. Αυτή η συνάρτηση θα καλείται όταν κάποιος χρήστης προσπαθήσει να προσθέσει βαθμούς.

```
function hasPermissions(string memory teacherId, uint courseId)
internal view returns (bool) {
    bool exists = false;
    for (uint i = 0; i < teacherPermissions[teacherId].length; i++) {
        if (teacherPermissions[teacherId][i] == courseId) {
            exists = true;
        }
    }
    return exists;
}
```

Πεδία εισόδου

teacherId: Το id του καθηγητή που επιλέχθηκε για έλεγχο δικαιωμάτων.

courseId: Το id του μαθήματος για το οποίο θα ελεγχθεί αν έχει άδεια ο καθηγητής.

Για την είσοδο των βαθμολογιών στο Blockchain ορίσαμε τις δύο παρακάτω δομές. Η πρώτη “Grade” περιέχει όλη την πληροφορία για μια συγκεκριμένη βαθμολογία, ενός φοιτητή σε μια από τις βαθμολογούμενες δραστηριότητες ενός μαθήματος.

Η δεύτερη δομή CourseData ουσιαστικά αναπαριστά τις βαθμολογίες όλων των φοιτητών, για μια συγκεκριμένη βαθμολογούμενη δραστηριότητα ενός μαθήματος. Περιέχει τα πεδία school id, semesterYearCourse και activityName με τα οποία ορίζεται αποκλειστικά μια δραστηριότητα ενός μαθήματος για ένα συγκεκριμένο εξάμηνο, μιας σχολής αλλά και μια λίστα από “Grades” δηλαδή τις βαθμολογίες όλων των φοιτητών.

```
struct Grade {
    string studentId;
    string studentName;
    string email;
    string activityName;
    string submittedOn;
    string gradedBy;
    string gradedOn;
    string rawGrade;
}
```

```
struct CourseData {
    string schoolId;
    string semesterYearCourse;
    string activityName;
    Grade[] grades;
}
```

Για την είσοδο των βαθμολογιών στο Blockchain ορίσαμε τις δύο παραπάνω δομές. Η πρώτη “Grade” περιέχει όλη την πληροφορία για μια συγκεκριμένη βαθμολογία, ενός φοιτητή σε μια από τις βαθμολογούμενες δραστηριότητες ενός μαθήματος.

Η δεύτερη δομή CourseData ουσιαστικά αναπαριστά τις βαθμολογίες όλων των φοιτητών, για μια συγκεκριμένη βαθμολογούμενη δραστηριότητα ενός μαθήματος. Περιέχει τα πεδία school id, semesterYearCourse και activityName, με τα οποία ορίζεται αποκλειστικά μια δραστηριότητα ενός μαθήματος για ένα συγκεκριμένο εξάμηνο, μιας σχολής αλλά και μια λίστα από “Grades” δηλαδή τις βαθμολογίες όλων των φοιτητών.

Οι δομές αυτές χρησιμοποιούνται για να μοντελοποιήσουμε την πληροφορία που λαμβάνουμε από το Moodle ώστε να την αποθηκεύσουμε εν τέλη χρησιμοποιώντας το παρακάτω mapping.

```
//schoolId->semester_year_course->activityName->grades Object  
mapping(string => mapping(string => mapping(string => Grade[][])))  
activityGrades;
```

Με αυτό το mapping χρησιμοποιώντας τα πεδία που συζητήσαμε παραπάνω, δημιουργείται μια αποκλειστική αντιστοιχία ανάμεσα σε μια συγκεκριμένη δραστηριότητα(κατά σχολή, εξάμηνο, έτος, μάθημα) και μια λίστα από λίστες βαθμολογιών. Ορίσαμε αυτά τα διαφορετικά επίπεδα αντιστοιχίας σχολή, εξάμηνο, έτος, μάθημα ώστε να είναι καλύτερα οργανωμένη η πληροφορία και να διατηρούμε αποδοτική τη διαδικασία του διαβάσματος των βαθμολογιών, χωρίς «ακριβές» αναζητήσεις σε μεγάλους όγκους δεδομένων. Ο λόγος που εν τέλη αποθηκεύουμε μια λίστα από λίστες είναι γιατί μπορεί να έχουμε ενημερώσεις στη βαθμολογία οπότε εδώ κρατάμε όλο το ιστορικό. Η «εξωτερική λίστα» έχει μήκος όσο αυτό των εκδόσεων, ενώ οι επιμέρους έχουν μήκος όσο των βαθμολογημένων φοιτητών κάθε φορά.

Αυτό το mapping, «γεμίζει» τιμές με την παρακάτω συνάρτηση.

```
function addGrades(string memory teacherId, uint courseId,  
CourseData[] memory data) public {  
    require( hasPermissions(teacherId, courseId) || msg.sender ==  
masterUser, "You have no permissions on this course, contact admin");  
    for (uint i = 0; i < data.length; i++) {  
        uint latest = activityGrades[data[i].schoolId][  
            data[i].semesterYearCourse  
][data[i].activityName].length;  
        activityGrades[data[i].schoolId][data[i].semesterYearCourse][  
            data[i].activityName  
].push();  
        for (uint j = 0; j < data[i].grades.length; j++) {  
            activityGrades[data[i].schoolId][data[i].semesterYearCourse  
e][  
                data[i].activityName  
][latest].push(data[i].grades[j]);  
        }  
    }  
}
```

Πεδία εισόδου

teacherId: Το id του καθηγητή που στέλνει το αίτημα για προσθήκη βαθμών.

courseId: Το id του μαθήματος το οποίο θέλουμε να βαθμολογηθεί.

CourseData[]: Μια λίστα από βαθμολογίες για το συγκεκριμένο μάθημα. Κάθε στοιχείο της λίστας περιέχει μια δεδομένα μορφής CourseData και αντιστοιχεί ουσιαστικά στους βαθμούς μιας συγκεκριμένης βαθμολογούμενης δραστηριότητας του μαθήματος.

Όπως βλέπουμε αρχικά αφού ελέγξουμε ότι ο καθηγητής έχει τα δικαιώματα να βαθμολογήσει το μάθημα, προσθέτουμε τις τιμές στο Blockchain χρησιμοποιώντας τις δομές που περιγράψαμε.

Παρακάτω βλέπουμε την συνάρτηση του συμβολαίου, μέσω της οποίας ανακτούμε τις βαθμολογίες από το Blockchain, για μια συγκεκριμένη βαθμολογούμενη δραστηριότητα.

```
function getGrades( string memory schoolId, string memory
semesterYearCourse, string memory activityName) public view returns
(Grade[] memory) {
    require(activityGrades[schoolId][semesterYearCourse][activityName]
.length >0, "No grades were found");
    uint latest =
activityGrades[schoolId][semesterYearCourse][activityName].length - 1;
    return
activityGrades[schoolId][semesterYearCourse][activityName][latest];
}
```

Πεδία εισόδου

schoolId: Το id της σχολής στην οποία ανήκει το μάθημα.

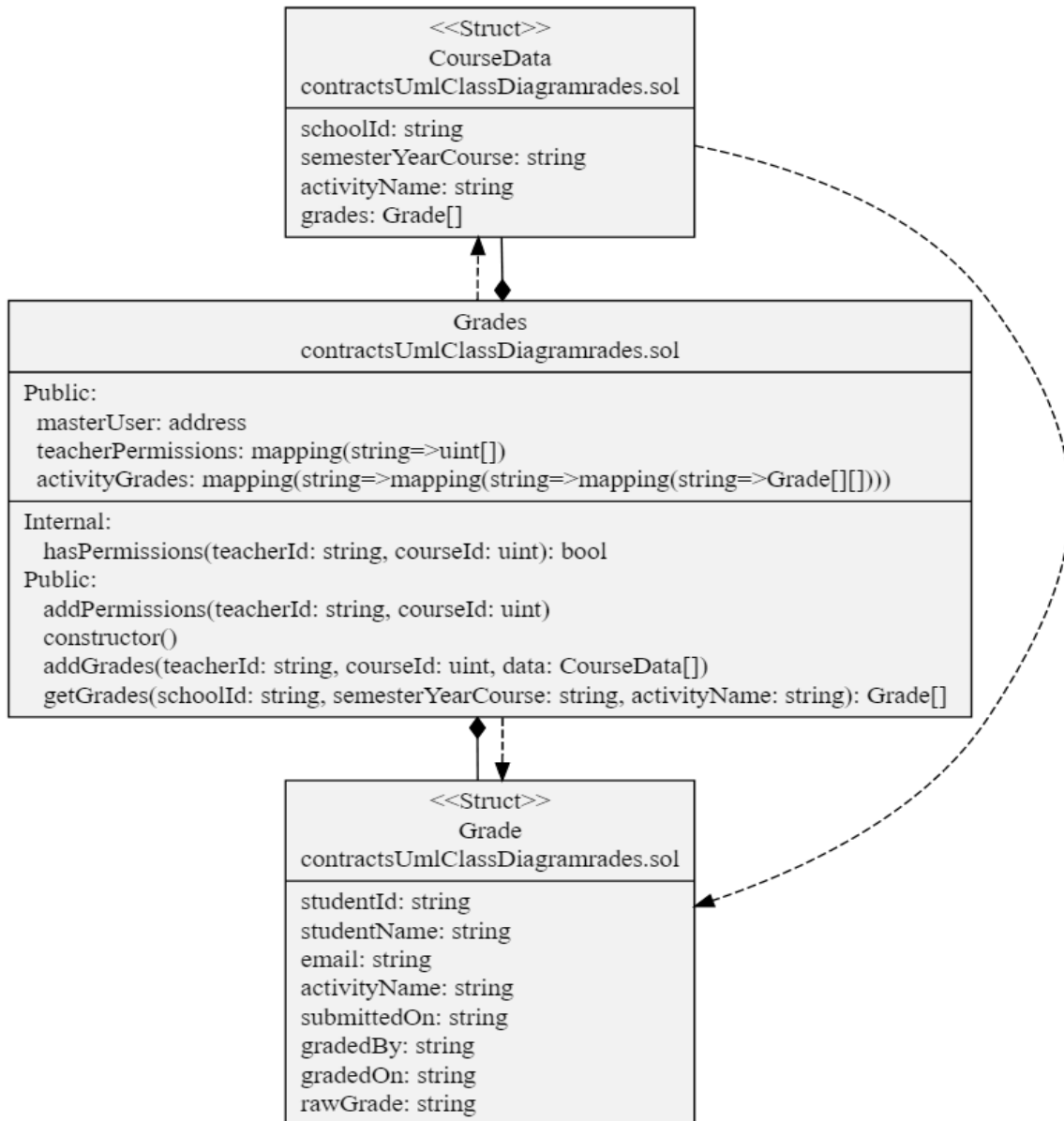
semesterYearCourse: Μια αναγνωριστική συμβολοσειρά που εξάγεται από το τρέχων εξάμηνο και έτος, αλλά και το id του μαθήματος.

activityName: Το όνομα της βαθμολογούμενης δραστηριότητας για την οποία θέλουμε να λάβουμε βαθμούς

Πεδία εξόδου

Λίστα από δεδομένα τύπου Grade[]: Επιστρέφεται μια λίστα από τους τελευταίους βαθμούς που προστέθηκαν στο Blockchain, για την συγκεκριμένη δραστηριότητα. Η λίστα που επιστρέφεται έχει μήκος όσο και το μήκος των φοιτητών που βαθμολογήθηκαν.

5.3.3 UML Class Diagram



Εικόνα 37 Uml class diagram του έξυπνου συμβολαίου

5.4 Εγκατάσταση Blockchain Network

Σε αυτή την εργασία δόθηκε περισσότερο βάρος στην ανάπτυξη του Moodle plugin και συνεπώς σε ότι αφορά το Blockchain δίκτυο επιλέχθηκε μια λύση που είναι εύκολη στη χρήση, προσομοιώνει βέβαια πλήρως ένα πραγματικό δίκτυο Blockchain αλλά θα φέρει ενδεχομένους κάποιους περιορισμούς σε μια μεγάλη παραγωγική εγκατάσταση.

Για την εγκατάσταση του δικτύου Blockchain θα χρησιμοποιήσουμε το εργαλείο “Ganache”[28] που μας παρέχεται από τη σουίτα του truffle[29]. Πρόκειται για ένα προσωπικό Blockchain που μας επιτρέπει να δημιουργήσουμε ένα ιδιωτικό δίκτυο Blockchain, βασισμένο στο Ethereum. Προσφέρει πολλές λειτουργικότητες για να διευκολύνει την δημιουργία του δικτύου και μας απαλλάσσει από το κόστος του να συνδεθούμε με το κύριο δίκτυο του Ethereum. Το ganache, παρέχεται και ως εφαρμογή με γραφικό περιβάλλον για το χρήστη, αλλά και ως πακέτο που χρησιμοποιείται εξ’ ολοκλήρου από τη γραμμή εντολών.

Αυτό το εργαλείο όπως αναφέραμε είναι μέρος της σουίτας “truffle” η οποία μας παρέχει και το αντίστοιχο πακέτο λογισμικού για την μεταγλώττιση και το deployment του έξυπνου συμβολαίου στο Blockchain δίκτυο του Ganache. Είναι σχεδιασμένα για να λειτουργούν μαζί. Το πακέτο “Truffle” λοιπόν παρέχει ένα περιβάλλον ανάπτυξης και επαλήθευσης(testing) για Blockchain, χρησιμοποιώντας το Ethereum Virtual Machine(EVM).

Παρακάτω ακολουθούν τα βήματα για την εγκατάσταση του δικτύου χρησιμοποιώντας μόνο τη γραμμή εντολών. Αρχικά τρέχουμε τις παρακάτω εντολές για κατεβάσουμε τα απαιτούμενα πακέτα που θα χρησιμοποιήσουμε.

```
npm install truffle -g  
npm install ganache -g
```

Στη συνέχεια αφού δημιουργήσουμε τον αντίστοιχο φάκελο και εισέλθουμε σε αυτόν πρέπει να αρχικοποιήσουμε το περιβάλλον εργασίας με την εντολή:

```
truffle init
```

Όταν ολοκληρωθεί η εντολή θα δούμε τους παρακάτω φακέλους να έχουν δημιουργηθεί:

- contracts/: Φάκελος για τα Solidity έξυπνα συμβόλαια.
- migrations/: Φάκελος για τα αρχεία deployment
- test/: Φάκελος για αρχεία test.
- truffle-config.js: Αρχείο παραμετροποιήσεων για το truffle.

Στη συνέχεια πρέπει να βάλουμε το συμβόλαιο μας μέσα στον φάκελο contracts και να ορίσουμε τις κατάλληλες ρυθμίσεις στο αρχείο truffle-config. Οι ρυθμίσεις αφορούν τον μεταγλωττιστή της solidity που θα χρησιμοποιήσουμε καθώς και τις ρυθμίσεις για το δίκτυο Blockchain. Μπορούμε να ορίσουμε περισσότερα από ένα δίκτυα Blockchain και να επιλέγουμε κάθε φορά σε ποιο δίκτυο θέλουμε να

ανεβάσουμε το έξυπνο συμβόλαιο μας. Οι ρυθμίσεις μας στο αρχείο `truffle-config` φαίνονται παρακάτω:

```
networks: {
  /* Useful for testing. The `development` name is special - truffle uses
  it by default if it's defined here and no other network is specified at
  the command line. You should run a client (like ganache, geth, or parity)
  in a separate terminal tab if you use this network and you must also set
  the `host`, `port` and `network_id` options below to some value.*/
  development: {
    host: ".*.*.*.*",      // Localhost (default: none)
    port: ****,           // Standard Ethereum port (default: none)
    network_id: "****",   // Any network (default: none)
    // gas: 4712388 // Gas limit used for deploys
  },
  ....
  compilers: {
    solc: {
      version: "0.8.19",
    }
  },
}
```

Για να σηκώσουμε το έξυπνο συμβόλαιο στο δίκτυο που ορίσαμε πρέπει αρχικά να το μεταγλωττίσουμε με την εντολή:

```
truffle compile
```

Κατά τη μεταγλώττιση του συμβολαίου θα ορισθεί το `contract abi`, στον φάκελο `/build/contracts` που θα δημιουργηθεί. Αφού ολοκληρωθεί η μεταγλώττιση πρέπει να σηκώσουμε το Blockchain δίκτυο και να ανεβάσουμε εκεί το έξυπνο συμβόλαιο μας, συνεπώς σε μια άλλη καρτέλα τερματικού πρέπει να τρέξουμε:

```
ganache -hostname *.*.*.* --port ****
```

Στην κονσόλα θα δούμε ότι το `ganache` έχει δημιουργήσει για εμάς 10 διευθύνσεις-λογαριασμούς τις οποίες μπορούμε να χρησιμοποιήσουμε για να αλληλοεπιδράσουμε με το δίκτυο.

Και στη συνέχεια στο άλλο τερματικό τρέχουμε:

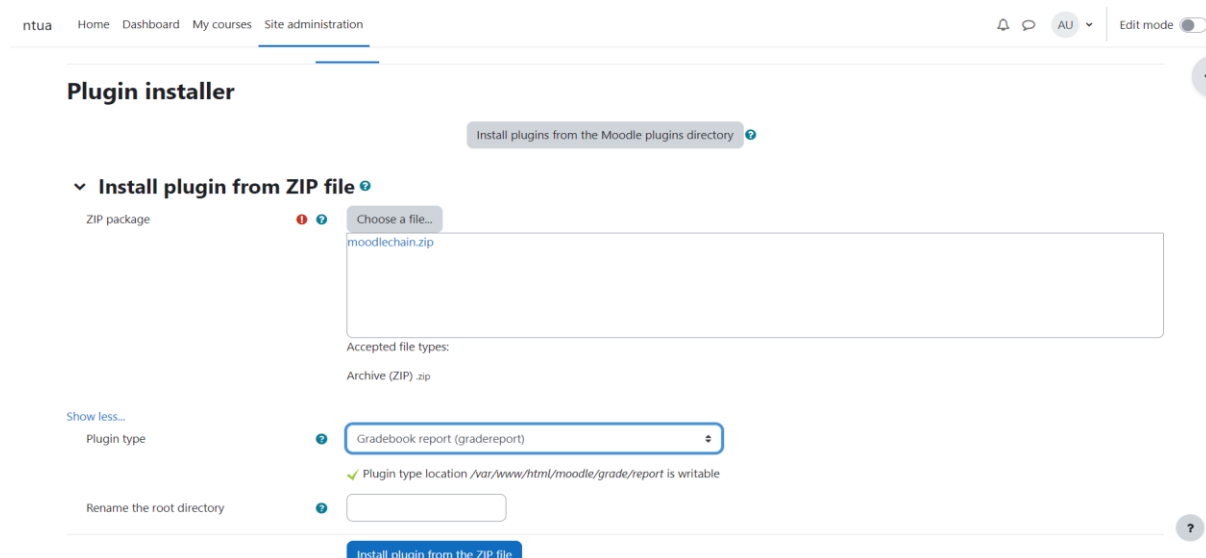
```
truffle migrate
```

Αν ολοκληρωθεί σωστά η διαδικασία, θα έχουμε πλέον σε λειτουργία ένα Blockchain δίκτυο και ανεβασμένο εκεί το έξυπνο συμβόλαιο μας. Στην κονσόλα θα δούμε πέρα από άλλες σχετικές πληροφορίες το contract address.

Κεφάλαιο 6 Demo

6.1 Εγκατάσταση plugin και βασική περιήγηση στο Moodle

Έχουμε πλέον τελειώσει με την ανάπτυξη κώδικα για το plugin, έχουμε μια εγκατάσταση του Moodle με το απαραίτητο περιεχόμενο (μάθημα, φοιτητές, βαθμολογούμενες δραστηριότητες). Σε αυτό το κεφάλαιο θα παρουσιάσουμε πως μπορεί κανείς να λάβει τον κώδικα του plugin, να το εγκαταστήσει στο Moodle site του και στη συνέχεια πως μπορεί κανείς να το χρησιμοποιήσει. Αρχικά ο κώδικας για το plugin είναι ανεβασμένος σε repository στο GitHub. [30] Από εκεί, επιλέγουμε να κατεβάσουμε τον πηγαίο κώδικα σε μορφή zip. Μέσα στο zip υπάρχει ο φάκελος “Moodlechain”, τον οποίο πρέπει να ανεβάσουμε ως zip στην κατάλληλη σελίδα του Moodle (Site administration > Plugins > Install plugins).



Εικόνα 38 Εγκατάσταση Moodlechain

Στη συνέχεια το Moodle θα μας ενημερώσει ότι γίνονται αναβαθμίσεις και θα δούμε την παρακάτω οθόνη:

Plugins check

This page displays plugins that may require your attention during the upgrade, such as new plugins to be installed, plugins to be upgraded, missing plugins etc. Additional plugins are displayed if there is an available update for them. It is recommended that you check whether there are more recent versions of plugins available and update their source code before continuing with this Moodle upgrade.

[Check for available updates](#)

Plugins requiring attention

[Plugins requiring attention](#) 1 [All plugins](#) 428

Plugin name / Directory	Current version	New version	Requires	Source / Status
Gradebook reports				
Moodlechain <i>/grade/report/moodlechain</i>		2023013000	• Moodle 2014051200	Additional To be installed

[Reload](#)

[Upgrade Moodle database now](#)

Εικόνα 39 Εγκατάσταση Moodlechain, δημιουργία πίνακα στη βάση

Όταν ολοκληρωθεί η εγκατάσταση θα έχουμε πλέον πρόσβαση στο Moodle plugin μας, επιλέγοντας το μάθημα:

The screenshot shows the Moodle dashboard for 'ntua'. The navigation menu includes Home, Dashboard, My courses, and Site administration. The main content area is titled 'Ntua Moodle' and has sub-menus for Home, Settings, Participants, Reports, Question bank, and More. Under 'Available courses', there is a card for 'Software Engineering' with teachers: Beulah Adams, Ernestine Ankundung, and Admin User.

Εικόνα 40 Περιήγηση στο Moodle, αρχική σελίδα

Στη συνέχεια επιλέγοντας Grades και Moodlechain από την λίστα των plugins, μπαίνουμε στην αρχική σελίδα του Moodlechain(ως admin):

The screenshot shows the 'Moodlechain - Admin User' interface. It displays 'All participants: 200/200' and search filters for first and last names. Below is a table with columns for 'First name / Last name', 'Email address', and 'Software Engineering' with sub-columns for 'Agile Methodology', 'Waterfall methodology', 'Final exam', and 'Course total'. The table lists participants like Prestley Ankundung, Dario Armstrong, Eulalia Armstrong, Sally Bate, and Glenn Boshan with their respective scores. At the bottom, there are buttons for 'Authenticate teacher', 'Verify grades', and 'Finalize grades'.

First name / Last name	Email address	Agile Methodology	Waterfall methodology	Final exam	Course total
PA Prestley Ankundung	Markus.King@gmail.com	1.00	60.00	100.00	86.10
DA Dario Armstrong	Anara.Casin1@gmail.com	100.00	55.00	90.00	87.50
EA Eulalia Armstrong	Marley39@yahoo.com	50.00	55.00	50.00	50.50
SB Sally Bate	Rickey.Beier@gmail.com	60.00	55.00	40.00	43.50
GB Glenn Boshan	Nicklau7@gmail.com	60.00	55.00	80.00	75.50
Overall average		55.64	69.57	64.94	64.47

Εικόνα 41 Αρχική σελίδα Moodlechain

Τελευταίο βήμα για να χρησιμοποιήσουμε το plugin είναι η εγκατάσταση της επέκτασης του Metamask στο browser μας από την επίσημη ιστοσελίδα [31].

Αφού ολοκληρωθεί η εγκατάσταση πρέπει να δημιουργήσουμε το λογαριασμό μας και να προσθέσουμε εκεί τα στοιχεία του δικτύου Blockchain με το οποίο θέλουμε να επικοινωνήσουμε.

Δίκτυα > Προσθήκη ενός Δικτύου > Προσθήκη δικτύου με μη αυτόματο τρόπο

i Ένας κακόβουλος πάροχος δικτύου μπορεί να πει ψέμματα για την κατάσταση του blockchain και να καταγράψει τη δραστηριότητα του δικτύου σας. Να προσθέτετε μόνο προσαρμοσμένα δίκτυα που εμπιστεύεστε.

Ονομασία Δικτύου

Νέο RPC URL

Αναγνωριστικό Αλυσίδας ⓘ

Σύμβολο Νομίσματος

Αποκλεισμός διεύθυνσης URL Explorer (Προαιρετικό)

Ακύρωση **Αποθήκευση**

Εικόνα 42 Εισαγωγή δικτύου στο Metamask

- Η ονομασία του δικτύου είναι προσωπική επιλογή του χρήστη.
- Γενικά το RPC URL είναι το URL του κόμβου που έχει την RPC διεπαφή [32] στην περίπτωση μας αφού χρησιμοποιούμε Ganache για το δίκτυο, ταυτίζεται με το <http://networklp:port/> και τα networklp και Port που ορίσαμε στο αρχείο truffle-config.
- Το αναγνωριστικό αλυσίδας είναι αυτό που ορίσαμε στο truffle-config.
- Το σύμβολο νομίσματος είναι το 'eth' μιας και πρόκειται για δίκτυο Ethereum.

Στη συνέχεια ο κάθε χρήστης θα πρέπει να προσθέσει το private key που του αντιστοιχεί, για να εισάγει τον λογαριασμό του. Αυτό θα δοθεί στους χρήστες του δικτύου από τον διαχειριστή κατά την αρχική εγκατάσταση.

Εισαγωγή Λογαριασμού

Οι λογαριασμοί που εισάγονται δεν θα συσχετιστούν με τη Μυστική Φράση Ανάκτησης του λογαριασμού σας
MetaTask που δημιουργήθηκε αρχικά. Μάθετε
περισσότερα για τους εισηγμένους λογαριασμούς [εδώ](#)

Επιλέξτε Τύπο Ιδιωτικό Κλειδί

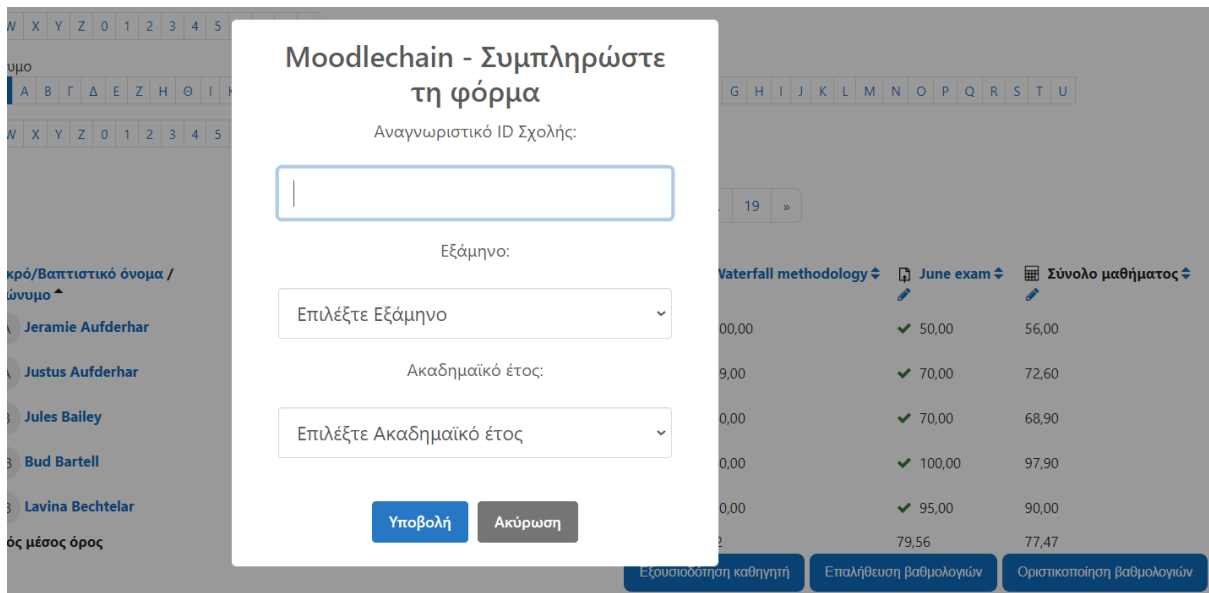
Επικολλήστε τη συμβολοσειρά ιδιωτικού κλειδιού εδώ:

Ακύρωση Εισαγωγή

Εικόνα 43 Εισαγωγή Λογαριασμού στο Metamask με ιδιωτικό κλειδί

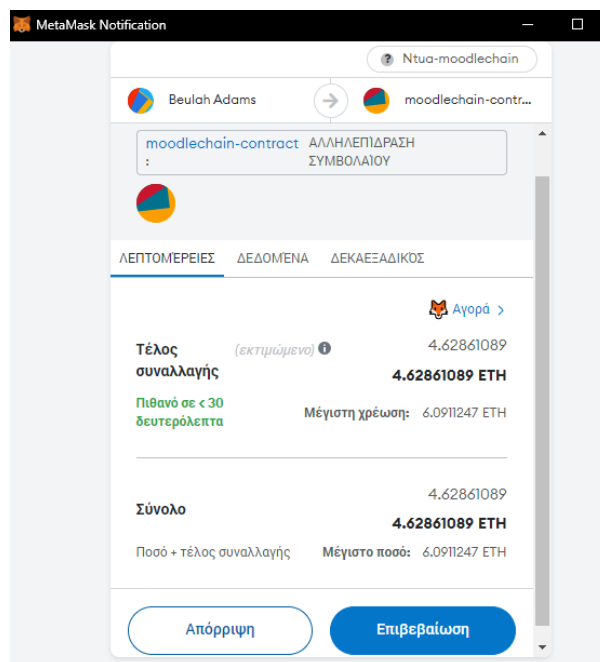
6.2 Καθηγητής που δεν έχει δικαιώματα-Προσπάθεια οριστικοποίησης

Σε αυτή και τις παρακάτω υποενότητες, θα επιδείξουμε μερικές περιπτώσεις χρήσης του plugin, χρησιμοποιώντας μερικά στιγμιότυπα οθόνης από τη σελίδα του Moodle. Ξεκινάμε με το σενάριο που ένας καθηγητής προσπαθεί να οριστικοποιήσει τη βαθμολογία του μαθήματος, χωρίς να έχει πάρει εξουσιοδότηση από τον διαχειριστή. Βλέπουμε ότι χρησιμοποιεί κανονικά το plugin, και αρχικά συμπληρώνει τα στοιχεία του μαθήματος:



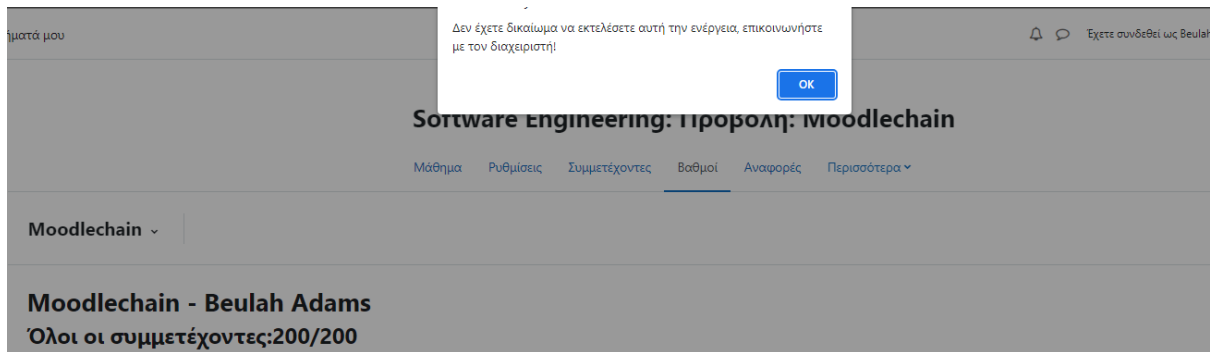
Εικόνα 44 Οριστικοποίηση βαθμών, συμπλήρωση στοιχείων μαθήματος

Αφού συμπληρώσει τη φόρμα, το παρακάτω αναδυόμενο παράθυρο εμφανίζεται στο χρήστη:



Εικόνα 45 Παράθυρο Metamask, εκτίμηση πόρων και επιβεβαίωση συναλλαγής

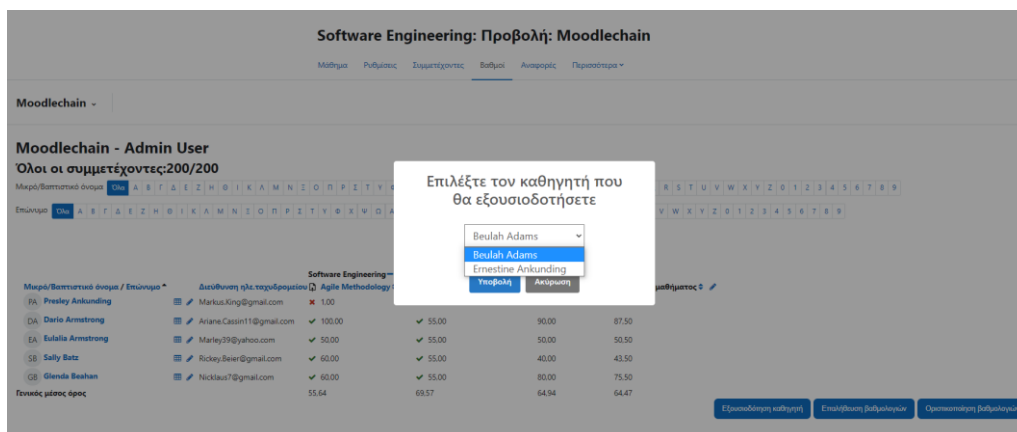
Η συναλλαγή απορρίπτεται από το Blockchain, αφού ο καθηγητής δεν έχει εξουσιοδοτηθεί από τον διαχειριστή.



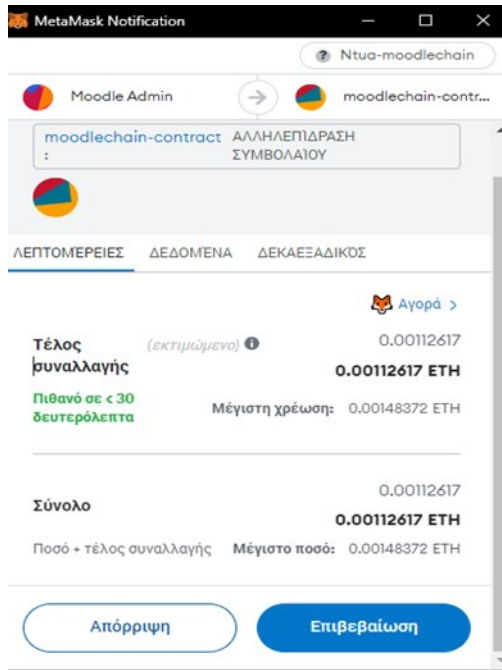
Εικόνα 46 Παράθυρο αποτυχίας συναλλαγής, ο καθηγητής δεν είχε εξουσιοδοτηθεί

6.3 Εξουσιοδότηση καθηγητή από διαχειριστή

Συνδεδεμένοι ως διαχειριστής, επιλέγουμε τώρα να εξουσιοδοτήσουμε έναν καθηγητή για να μπορεί να στείλει βαθμολογίες στο Blockchain. Στην παρακάτω φόρμα, αφού έχουμε επιλέξει το κουμπί «Εξουσιοδότηση καθηγητή», εμφανίζεται μια λίστα με τους καθηγητές του μαθήματος από όπου μπορούμε να επιλέξουμε ποιον θα εξουσιοδοτήσουμε.

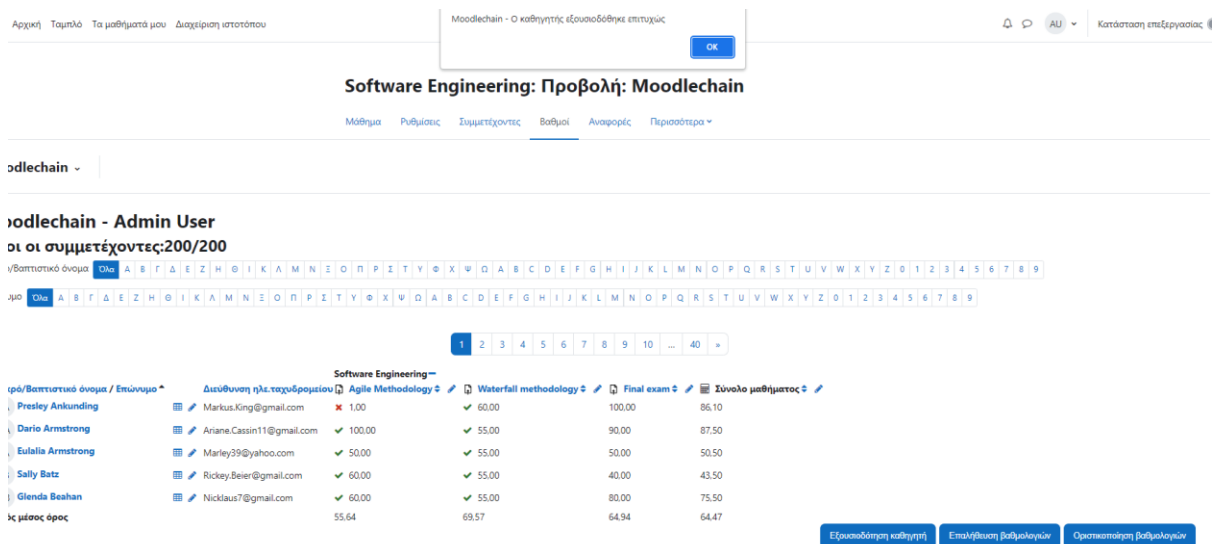


Εικόνα 47 Επιλογή καθηγητή για εξουσιοδότηση



Εικόνα 48 Παράθυρο Metamask-αποδοχή συναλλαγής

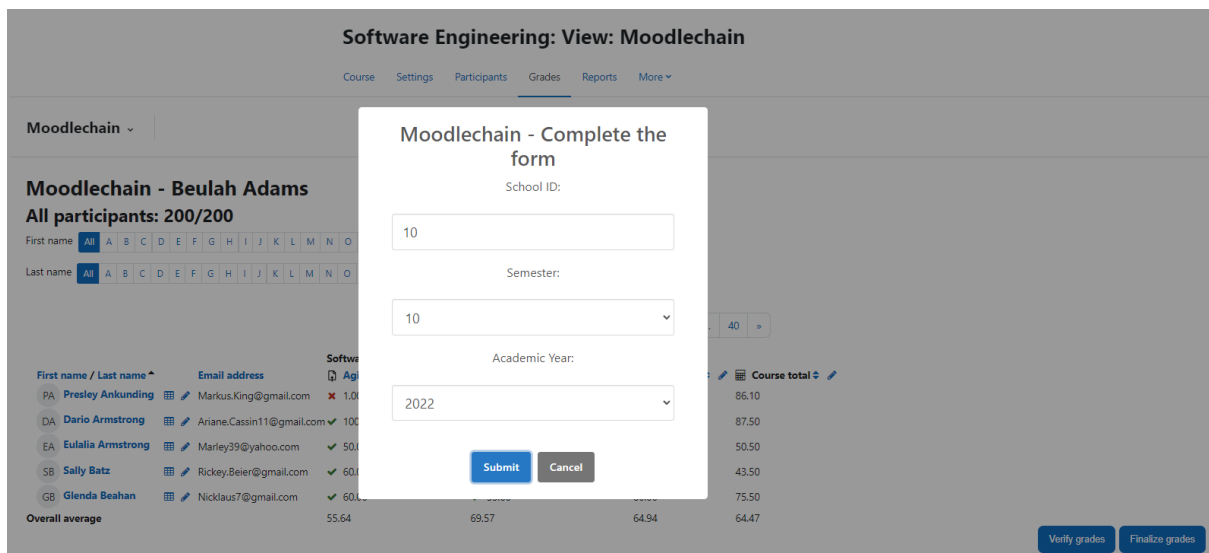
Αφού αποδεχτούμε τη συναλλαγή στο Metamask, αποστέλλεται στο Blockchain η συναλλαγή και βλέπουμε το μήνυμα επιτυχίας.



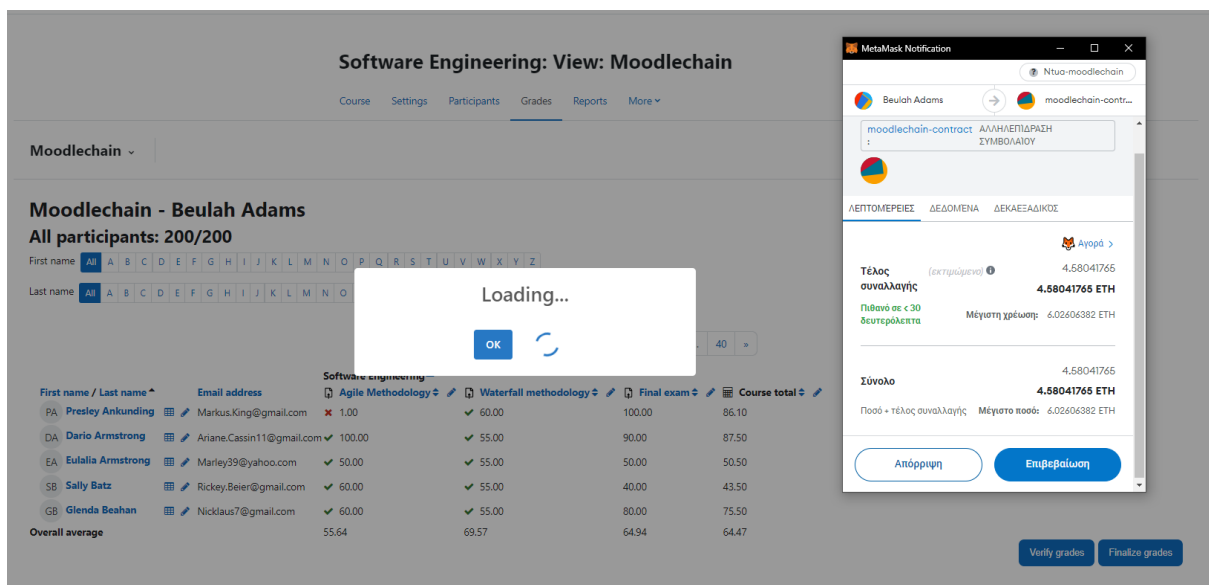
Εικόνα 49 Εξουσιοδότηση καθηγητή-Μήνυμα επιτυχίας

6.4 Καθηγητής που έχει δικαιώματα-Οριστικοποίηση

Συνδεόμαστε και πάλι ως ο καθηγητής “Beulah Adams” που εξουσιοδοτήσαμε νωρίτερα, για να στείλουμε εκ νέου βαθμούς στο Blockchain και βλέπουμε ότι αφού αποδεχτούμε τη συναλλαγή οι βαθμοί εισάγονται επιτυχώς. Επίσης για λόγους επίδειξης αλλάξαμε τη γλώσσα του Moodle και βλέπουμε το κείμενο στο Moodlechain στα αγγλικά.



Εικόνα 50 Οριστικοποίηση βαθμών-συμπλήρωση στοιχείων μαθήματος



Εικόνα 51 Οθόνη Moodle και παράθυρο Metamask

Software Engineering: View: Moodlechain

Course Settings Participants **Grades** Reports More

Moodlechain

Moodlechain - Beulah Adams

All participants: 200/200

First name All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Last name All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1 2 3 4 5 6 7 8 9 10 ... 40 >

First name / Last name	Email address	Software Engineering	Agile Methodology	Waterfall methodology	Final exam	Course total
PA Presley Ankunding	Markus.King@gmail.com	1.00	60.00	100.00	86.10	
DA Dario Armstrong	Ariane.Cassin11@gmail.com	100.00	55.00	90.00	87.50	
EA Eulalia Armstrong	Marley39@yahoo.com	50.00	55.00	50.00	50.50	
SB Sally Batz	Rickey.Beier@gmail.com	60.00	55.00	40.00	43.50	
GB Glenda Beahan	Nicklaus7@gmail.com	60.00	55.00	80.00	75.50	
Overall average		55.64	69.57	64.94	64.47	

Verify grades Finalize grades

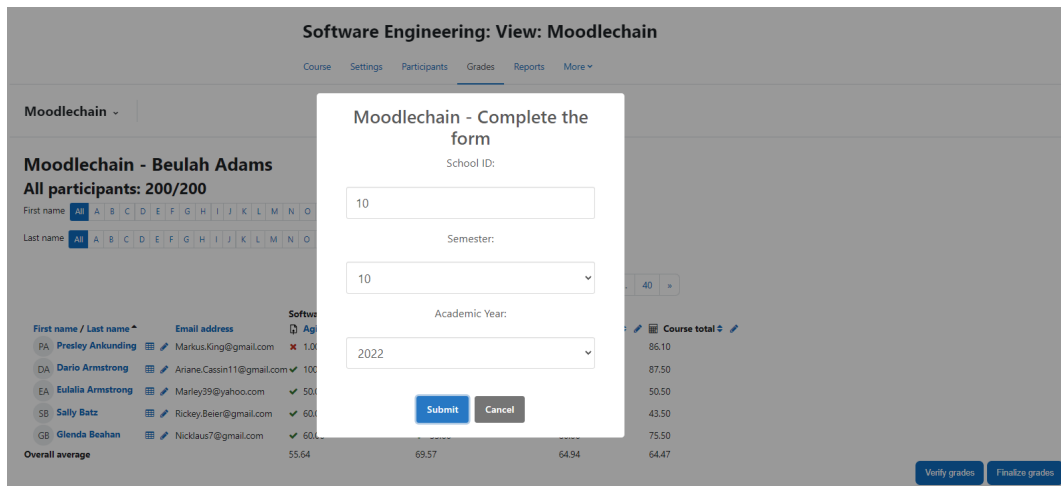
Εικόνα 52 Οριστικοποίηση βαθμών-Μήνυμα επιτυχίας

6.5 Επικύρωση βαθμών- δεν βρέθηκαν αλλαγές

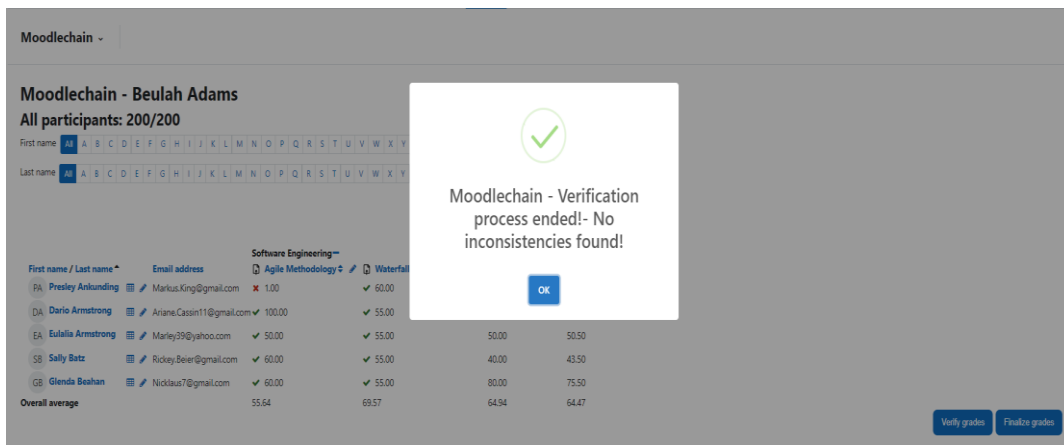
Σε συνέχεια του προηγούμενου σεναρίου, τώρα αιτούμαστε επικύρωση βαθμών για το ίδιο μάθημα που μόλις αποθηκεύσαμε. Αφού συμπληρώσουμε τα κατάλληλα στοιχεία βλέπουμε το μήνυμα στο αναδυόμενο παράθυρο που επιβεβαιώνει ότι δεν έχουν γίνει αλλαγές.

The screenshot shows the Moodlechain interface with a modal dialog box titled "Choose one or more activities to verify". The dialog contains four buttons: "Agile Methodology", "Waterfall methodology", "Final exam", and "Final Grade". Below these buttons are "Submit" and "Cancel" buttons. In the background, the grade list from the previous image is visible, showing the same data for participants and their scores in various activities.

Εικόνα 53 Επικύρωση βαθμών-Επιλογή δραστηριοτήτων για επικύρωση



Εικόνα 54 Επικύρωση βαθμών- Συμπλήρωση στοιχείων μαθήματος



Εικόνα 55 Επικύρωση βαθμών-Δεν βρέθηκαν διαφορές

6.6 Επικύρωση βαθμών-βρέθηκαν αλλαγές

Σε αυτό το σενάριο, κάνουμε κάποιες αλλαγές στις βαθμολογίες στο Moodle και συγκεκριμένα στο φοιτητή Presley Ankunding (ο πρώτος που εμφανίζεται). Αφού συμπληρώσουμε τα κατάλληλα στοιχεία βλέπουμε εν τέλη το ανάλογο παράθυρο που επιδεικνύει τις αλλαγές.

Moodlechain - Beulah Adams

All participants: 200/200

First name: [All] [A] [B] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W] [X] [Y] [Z]

Last name: [All] [A] [B] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W] [X] [Y] [Z]

1 2 3 4 5 6 7 8 9 10 ... 40 >

First name / Last name		Email address	Software Engineering			
			Agile Methodology	Waterfall methodology	Final exam	Course total
PA	Presley Ankunding	Markus.King@gmail.com	100.00	100.00	100.00	100.00
DA	Dario Armstrong	Ariane.Cassin11@gmail.com	100.00	55.00	90.00	87.50
EA	Eulalia Armstrong	Marley39@yahoo.com	50.00	55.00	50.00	50.50
SB	Sally Batz	Rickey.Beier@gmail.com	60.00	55.00	40.00	43.50
GB	Glenda Beahan	Nicklaus7@gmail.com	60.00	55.00	80.00	75.50
Overall average			56.13	69.77	64.94	64.54

Verify grades Finalize grades

Εικόνα 56 Αλλαγή βαθμολογίας φοιτητή Presley Ankunding

Software Engineering: View: Moodlechain


Course Settings Participants Grades Reports More

Moodlechain - Beulah Adams

All participants: 200/200

First name: [All] [A] [B] [C] [D] [E] [F]

Last name: [All] [A] [B] [C] [D] [E] [F]



Moodlechain - Verification process ended! Inconsistencies found:

Student name	Activity name	Grade on blockchain	Current grade
Presley Ankunding	Agile Methodology	1.00000	100.00000
Presley Ankunding	Waterfall methodology	60.00000	100.00000
Presley Ankunding	Final Grade	86.10000	100.00000

OK

First name / Last name		Email address	Software Engineering			
			Agile Methodology	Waterfall methodology	Final exam	Course total
PA	Presley Ankunding	Markus.King@gmail.com	100.00	100.00	100.00	100.00
DA	Dario Armstrong	Ariane.Cassin11@gmail.com	100.00	55.00	90.00	87.50
EA	Eulalia Armstrong	Marley39@yahoo.com	50.00	55.00	50.00	50.50
SB	Sally Batz	Rickey.Beier@gmail.com	60.00	55.00	40.00	43.50
GB	Glenda Beahan	Nicklaus7@gmail.com	60.00	55.00	80.00	75.50
Overall average			56.13	69.77	64.94	64.54

Verify grades Finalize grades

Εικόνα 57 Επικύρωση βαθμών, βρέθηκαν διαφορές

6.7 Επικύρωση βαθμών-δεν βρέθηκαν βαθμοί

Στο παρακάτω σενάριο, επιχειρούμε να επικυρώσουμε βαθμούς επιλέγοντας ένα τυχαίο schoolid για το οποίο δεν έχουμε ποτέ υποβάλλει βαθμούς και βλέπουμε το αντίστοιχο μήνυμα λάθους.

Moodlechain - Beulah Adams
All participants: 200/200

First name: [AM] [A] [B] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W] [X] [Y]
Last name: [AM] [A] [B] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W] [X] [Y]

Choose one or more activities to verify

- Agile Methodology
- Waterfall methodology
- Final exam
- Final Grade

Submit Cancel

First name / Last name	Email address	Software Engineering	Agile Methodology	Waterfall		
PA Presley Ankunding	Markus.King@gmail.com	1.00	60.00	60.00		
DA Dario Armstrong	Ariane.Cassin1@gmail.com	100.00	55.00	55.00	50.00	50.50
EA Eulalia Armstrong	Marley39@yahoo.com	50.00	55.00	55.00	50.00	50.50
SB Sally Batz	Rickey.Beier@gmail.com	60.00	55.00	55.00	40.00	43.50
GB Glenda Beahan	Nicklaus7@gmail.com	60.00	55.00	55.00	80.00	75.50
Overall average		55.64	69.57	64.94	64.94	64.47

Verify grades Finalize grades

Εικόνα 58 Επικύρωση βαθμών-Επιλογή δραστηριοτήτων για επικύρωση

Moodlechain - Complete the form

School ID:

12345678

Semester:

1

Academic Year:

1980

Submit Cancel

Εικόνα 59 Επικύρωση βαθμών-Συμπλήρωση στοιχείων μαθήματος

VM Exception while processing transaction: revert No grades were found

OK

Εικόνα 60 Επικύρωση βαθμών-Δεν υπάρχουν βαθμοί

Κεφάλαιο 7 Επίλογος

7.1 Συμπεράσματα

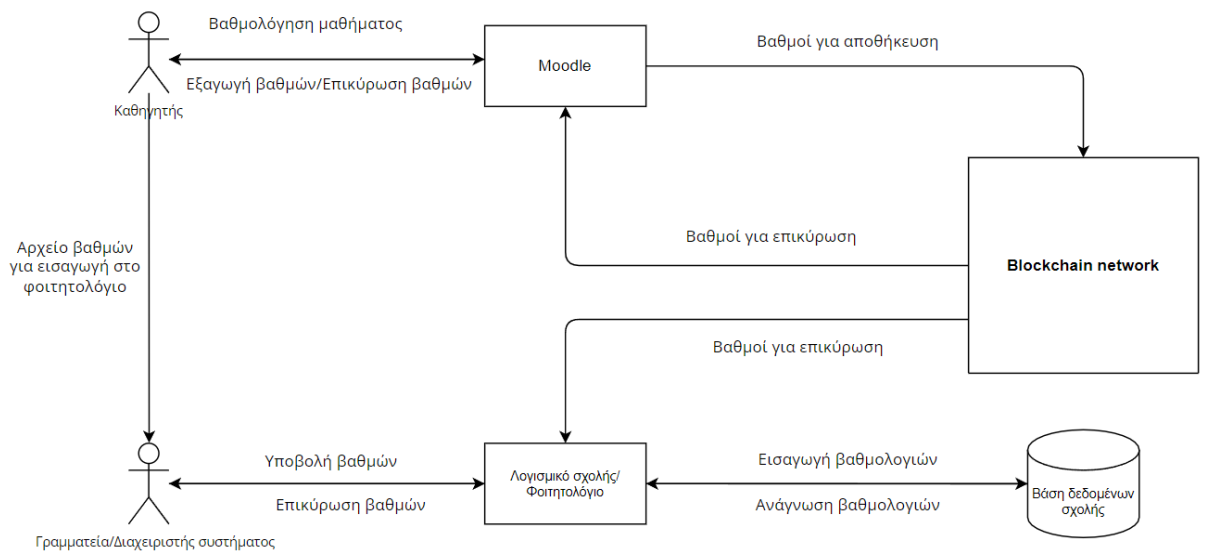
Σκοπός αυτής της εργασίας ήταν να μελετήσουμε πως η τεχνολογία του Blockchain θα μπορούσε να ωφελήσει την διαδικασία διαχείρισης βαθμών ενοποιημένο με μια πλατφόρμα υποστήριξης του e-Learning, καθώς και να υλοποιήσουμε ένα υποστηρικτικό εργαλείο στην πλατφόρμα του Moodle. Δημιουργήθηκε λοιπόν ένα λειτουργικό Moodle plugin το οποίο μπορούν να χρησιμοποιούν οι εμπλεκόμενοι (καθηγητές, διοικητικοί) για να αποθηκεύουν αλλά και να επικυρώνουν τις βαθμολογίες συγκρίνοντας αυτές που είναι αποθηκευμένες στο Blockchain, με τις τρέχουσες στην βάση δεδομένων του Moodle. Μέσω του εργαλείου η διαδικασία διαχείρισης των βαθμών πλέον αξιοποιεί όλα τα θετικά χαρακτηριστικά της τεχνολογίας του Blockchain. Πιο συγκεκριμένα, η κατακεκομμένη δομή του Blockchain προσφέρει ένα επιπλέον επίπεδο ασφάλειας για τα δεδομένα, αφού για να χαθεί η πληροφορία θα πρέπει να παραβιαστούν όλοι οι κόμβοι του συστήματος γεγονός πρακτικά αδύνατο. Επιπλέον, διασφαλίζεται η ακεραιότητα των δεδομένων για ένα μεγάλο μέρος της διαδικασίας, εφόσον από τη στιγμή που ο καθηγητής οριστικοποιήσει τις βαθμολογίες υπάρχει ένα αμετάβλητο, «καθαρό» αντίγραφο των βαθμών αποθηκευμένο στους κόμβους του δικτύου. Αυτό το αντίγραφο στη συνέχεια αξιοποιείται μέσω της λειτουργικότητας που υλοποιήσαμε για την επικύρωση βαθμών. Με αυτό τον τρόπο η διαδικασία διαχείρισης βαθμών, πιστώνεται σε διαφάνεια. Τέλος, το plugin υλοποιημένο στο Moodle και με βάση την φιλοσοφία της πλατφόρμας φέρνει όλα τα παραπάνω χωρίς να προσθέτει περαιτέρω πολυπλοκότητα στην διαδικασία αφού είναι πολύ εύκολο στη χρήση.

Βέβαια, το εργαλείο λειτουργεί υποστηρικτικά και δεν αντικαθιστά την υπάρχουσα διαδικασία διαχείρισης βαθμών. Επίσης περιορίζεται σε επικύρωση των βαθμών για το κομμάτι της διαδικασίας που οι βαθμολογίες είναι στην κατοχή των καθηγητών. Στο επόμενο κεφάλαιο θα μελετήσουμε πιθανά επόμενα βήματα που θα μπορούσαν μελλοντικά να ενισχύσουν περαιτέρω την υπάρχουσα διαδικασία ή και ενδεχομένως να την αντικαταστήσουν πλήρως.

7.2 Επόμενα βήματα

7.2.1 Διαδικασία διαχείρισης βαθμών

Το επόμενο βήμα για την περαιτέρω ενίσχυση της διαδικασίας είναι να παρέχουμε την δυνατότητα επικύρωσης και στα υπόλοιπα στάδια της διαδικασίας διαχείρισης των βαθμών. Μια πιθανή λύση για αυτό φαίνεται στο παρακάτω διάγραμμα.

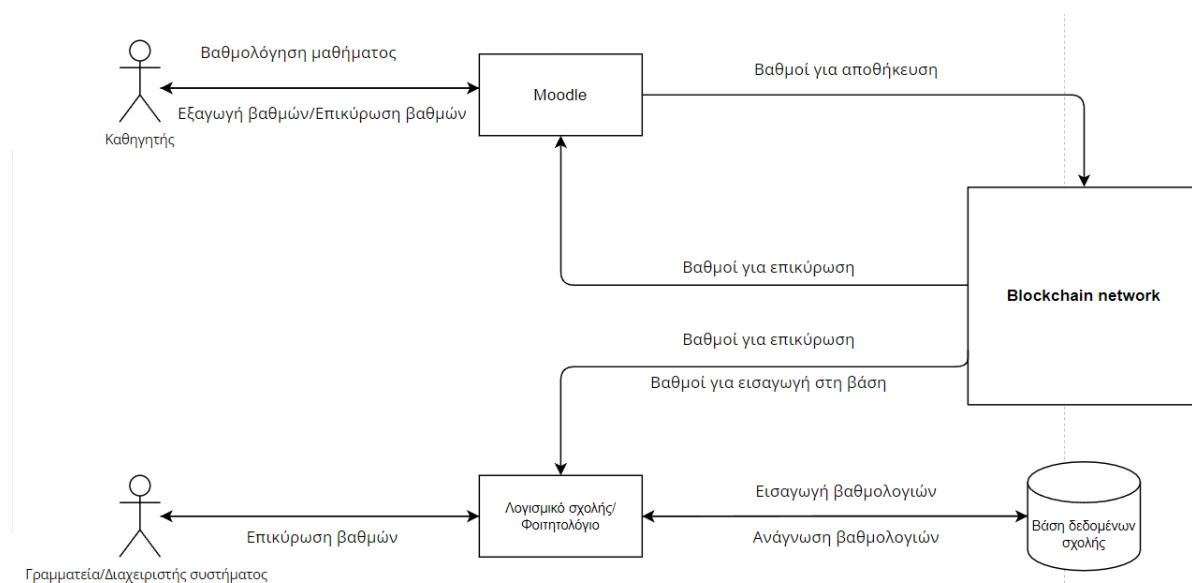


Εικόνα 61 Διάγραμμα ροής πρόταση 1

Για να υλοποιηθεί το παραπάνω σύστημα, θα έπρεπε να επεκτείνουμε επιπλέον το λογισμικό της σχολής ώστε να έχει τη δυνατότητα να αλληλοεπιδρά με το Blockchain δίκτυο. Στην προκειμένη, αρκεί να μπορεί να λαμβάνει τις βαθμολογίες από το δίκτυο και να τις συγκρίνει με αυτές που υπάρχουν στη βάση δεδομένων της σχολής. Η δυνατότητα επικύρωσης θα είναι εφικτή στο μεγαλύτερο μέρος της διαδικασίας και έτσι θα ενισχυθεί ακόμα περισσότερο η ακεραιότητα της διαδικασίας. Επιπλέον, θα μπορούσε να κοινοποιείται και με τους φοιτητές η κατάσταση των βαθμών(επικυρωμένοι ή όχι) ή τουλάχιστον η ημερομηνία της τελευταίας επικύρωσης.

Στην παραπάνω λύση, συζητήσαμε για επέκταση του υπάρχοντος λογισμικού κάθε σχολής για να μπορεί να επικοινωνεί με το δίκτυο Blockchain και να γίνεται η σύγκριση και η επικύρωση των βαθμών. Μιλάμε πάλι όμως για μια καθαρά υποστηρικτική διαδικασία, η οποία παραμένει εν μέρη λίγο πολύπλοκη για τα ενδιαφερόμενα μέρη. Παρακάτω θα αναλύσουμε μια πιο ανατρεπτική λύση η οποία δεν ενισχύεται αλλά στηρίζεται από το Blockchain.

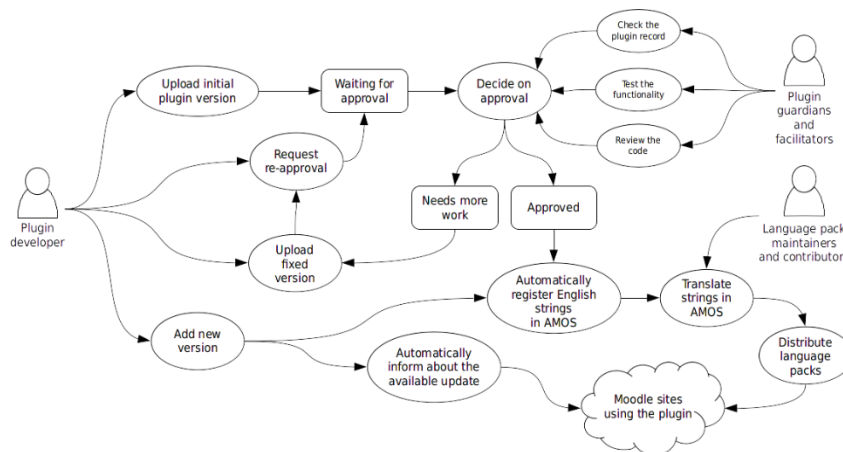
Αν παρατηρήσουμε την διαδικασία, από τη στιγμή που υλοποιήσαμε το plugin και το δίκτυο Blockchain «γεμίζει» με βαθμολογίες κατευθείαν από το Moodle, η μετατροπή και η αποστολή των βαθμών από τους καθηγητές στη γραμματεία μπορεί να θεωρηθεί περιττή. Αυτό ισχύει ακόμα περισσότερο στην παραπάνω πρόταση, αν έχουμε υλοποιήσει την αλληλεπίδραση του λογισμικού της σχολής με το δίκτυο Blockchain, αφού η βάση δεδομένων της σχολής θα μπορεί να παίρνει τα δεδομένα βαθμολογιών κατευθείαν από το δίκτυο. Αφαιρώντας λοιπόν το κομμάτι της διαδικασίας όπου οι καθηγητές μετατρέπουν και στέλνουν τις βαθμολογίες στη γραμματεία καθώς και την υποβολή των βαθμών από την γραμματεία στο σύστημα, σχεδιάζουμε μια διαδικασία πολύ πιο εύκολη, γρήγορη και παραγωγική για τα ενδιαφερόμενα μέρη, προστατεύουμε την ιδιωτικότητα αφού οι βαθμοί «περνούν από λιγότερα χέρια» και δημιουργείται μια πιο ασφαλής διαδικασία σε ότι αφορά την ακεραιότητα των δεδομένων. Παρακάτω βλέπουμε και το αντίστοιχο διάγραμμα για την διαδικασία που περιγράψαμε.



Εικόνα 62 Διάγραμμα ροής πρόταση 2

7.2.2 Moodle plugin

Ένα πιθανό επόμενο βήμα επίσης θα ήταν η περαιτέρω ανάπτυξη του Moodle plugin που έχουμε υλοποιήσει, με σκοπό να δημοσιευτεί. Το Moodle βασίζεται στην μεγάλη κοινότητα προγραμματιστών που εξελίσσουν και επεκτείνουν συνεχώς την πλατφόρμα με τα plugin που αναπτύσσουν. Συνεπώς υπάρχει μια εδραιωμένη ροή εργασίας, με συγκεκριμένα στάδια μέχρι να εισαχθεί το plugin στο “Moodle plugin directory”. Εκεί συντηρούνται όλα τα plugin που έχουν περάσει τα παραπάνω στάδια, και από εκεί μπορούν οι διαχειριστές Moodle να βρουν και να εγκαταστήσουν plugin της αρεσκείας τους.



Εικόνα 63 Ροή εργασιών για δημοσίευση plugin[33]

Τα βασικά προαπαιτούμενα για να εγκριθεί και να δημοσιευθεί ένα plugin απαριθμούνται παρακάτω:

1. Τύπος plugin: Είναι σημαντικό να συντηρείται η λογική συνέπεια σε μια πλατφόρμα που χρησιμοποιούν εκατομμύρια χρήστες. Για αυτό το λόγο υπάρχουν καθορισμένοι τύποι plugin που τα ομαδοποιούν ανάλογα με την λειτουργικότητα τους. Είναι κρίσιμο λοιπόν οι προγραμματιστές να έχουν επιλέξει τον κατάλληλο τύπο για το plugin τους. Το δικό μας plugin είναι του τύπου gradererort, εφόσον διαχειρίζεται βαθμολογίες.
2. Πρέπει να ακολουθεί τη Frankenstyle[34] σύμβαση ονομάτων που ορίζει κανόνες για την ονομασία του plugin, των φακέλων, των συναρτήσεων, των μεταβλητών του καθώς και πολλών άλλων στοιχείων του plugin.
3. Πρέπει να υπάρχει κάποιο αποθετήριο, συμβατό με το Git για έλεγχο εκδόσεων, στο διαδίκτυο για την αποθήκευση αλλά και την κοινοποίηση του κώδικα. Πέρα από τους λόγους που αυτό είναι απαραίτητο κατά την ανάπτυξη λογισμικού γενικά με αυτό τον τρόπο δίνεται η δυνατότητα και σε άλλους προγραμματιστές να συμμετάσχουν στην διαδικασία ανάπτυξης.
4. Είναι απαραίτητο να υπάρχει ένα σύστημα με το οποίο οι χρήστες του plugin θα μπορούν να αναφέρουν σφάλματα αλλά και να προτείνουν-ζητούν νέες λειτουργικότητες. Υπάρχει αλλά δεν είναι απαραίτητη η χρήση του αντίστοιχου συστήματος που υπάρχει από το Moodle.
5. Ο κώδικας πρέπει να συνοδεύεται από πολύ καλή τεκμηρίωση.
6. Τέλος, είναι αναγκαίο να υπάρχουν στιγμιότυπα οθόνης από τη χρήση του εργαλείου που να παρουσιάζουν την λειτουργία του, ώστε να γίνεται πιο εύκολο στη χρήση σε νέους χρήστες που το χρησιμοποιούν.

Δημοσιεύοντας το plugin στο Moodle plugin directory, το εργαλείο μας θα λάβει έκθεση σε μια μεγάλη κοινότητα προγραμματιστών αλλά και χρηστών του Moodle. Έτσι θα μπορέσουμε μέσω της

ανατροφοδότησης να βελτιώσουμε τις υπάρχουσες λειτουργίες αλλά και να προσθέσουμε νέες, ενώ ενδεχομένως να βοηθήσουμε στην βελτίωση της διαδικασίας διαχείρισης βαθμών και άλλα πανεπιστημιακά ιδρύματα.

Βιβλιογραφικές αναφορές

- [1] Ανάκτηση από <https://moodle.com/solutions/lms/>
- [2] Ανάκτηση από <https://stats.moodle.org/>
- [3] Ανάκτηση από <https://www.noc.ntua.gr/el/service/directory>
- [4] Ανάκτηση από <https://docs.moodle.org/402/en/Blocks>
- [5] Ανάκτηση από https://docs.moodle.org/402/en/IMS_content_package
- [6] Ανάκτηση από <https://docs.moodle.org/402/en/Resources>
- [7] Ανάκτηση από https://docs.moodle.org/400/en/Roles_and_permissions
- [8] Ανάκτηση από <https://docs.moodle.org/400/en/Context>
- [9] Ανάκτηση από https://docs.moodle.org/402/en/Installing_Moodle
- [10] Ανάκτηση από <https://www.ibm.com/topics/blockchain>
- [11] Ανάκτηση από <https://www.investopedia.com/terms/n/nonce.asp>
- [12] Ανάκτηση από <https://www.techtarget.com/searchcio/feature/What-are-the-4-different-types-of-blockchain-technology>
- [13] Ανάκτηση από <https://www.techtarget.com/whatis/definition/consensus-algorithm>
- [14] Ανάκτηση από <https://www.bitdegree.org/crypto/tutorials/what-is-a-smart-contract#what-is-a-smart-contract-what-yoursquare-going-to-find-in-this-guide>
- [15] Vitalik Buterin. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform. 2014.
- [16] Christos Chadjichristofi. Implementation of Blockchain Application for managing University grades
- [17] Ανάκτηση από <https://www.r3.com/trust-technology/>.
- [18] Ανάκτηση από [What is The Quorum Blockchain: JP Morgan to ConsenSys - Phemex Academy, ConsenSys Quorum | ConsenSys](#)
- [19] Ανάκτηση από [XRP - Digital Asset for Global Economic Utility | Ripple](#)
- [20] Ανάκτηση από Hyperledger Foundation. <https://www.hyperledger.org/>.
- [21] Starantzis Dimitrios. Software architectures for implementing decentralized autonomous organizations (DAO) with blockchain technologies. Diploma Thesis, SoftLab
- [22] <https://www.pluralsight.com/blog/software-development/what-is-solidity-smart-contracts>
- [23] Ανάκτηση από <https://docs.soliditylang.org/en/v0.8.20/>
- [24] Ανάκτηση από <https://metamask.io/>
- [25] Ανάκτηση από https://en.wikipedia.org/wiki/MetaMask#cite_note-CNET_2018-6

- [26] Ανάκτηση από https://docs.moodle.org/400/en/Step-by-step_Installation_Guide_for_Ubuntu
- [27] Ανάκτηση από https://moodle.org/plugins/local_bulkenrol
- [28] Ανάκτηση από <https://trufflesuite.com/docs/ganache/>
- [29] Ανάκτηση από <https://trufflesuite.com/docs/truffle/>
- [30] Ανάκτηση από <https://github.com/aggelako/moodlechain.git>
- [31] Ανάκτηση από <https://metamask.io/download/>
- [32] Ανάκτηση από [JSON-RPC API | MetaMask developer documentation](#)
- [33] Ανάκτηση από https://docs.moodle.org/dev/Plugin_contribution
- [34] Ανάκτηση από <https://docs.moodle.org/dev/Frankenstyle>