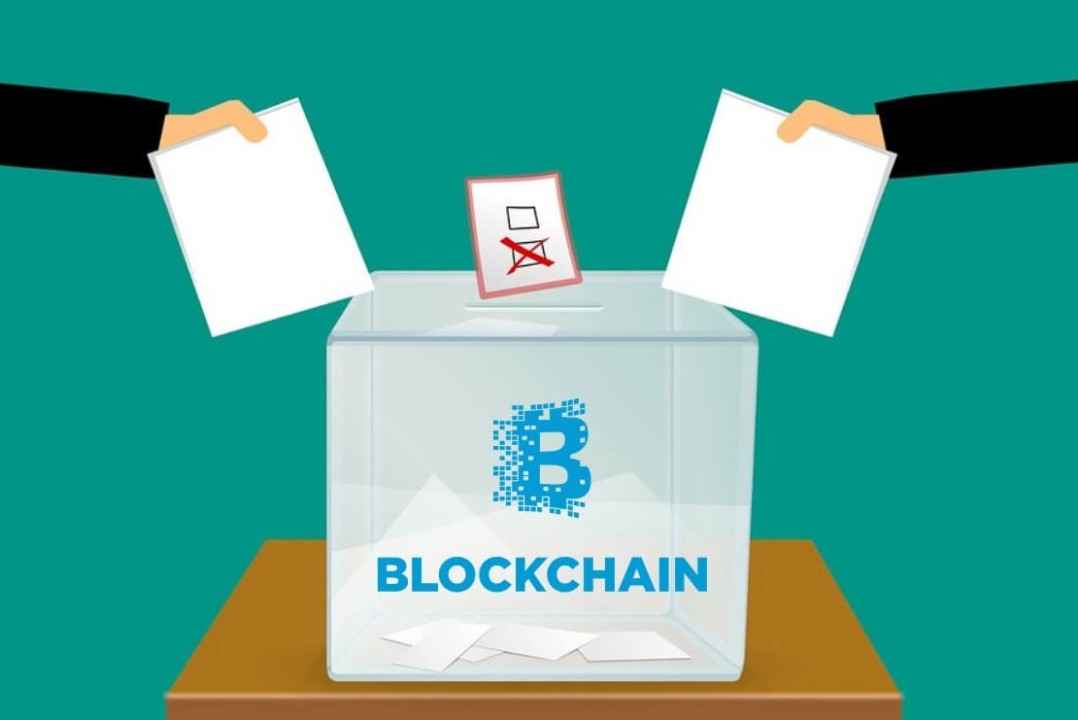
**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ**

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΠΜΣ «ΠΡΟΗΓΜΕΝΕΣ ΤΕΧΝΟΛΟΓΙΕΣ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ»**

**ΦΟΙΤΗΤΙΚΕΣ ΕΚΛΟΓΕΣ ΒΑΣΙΣΜΕΝΕΣ**

**ΣΤΗΝ ΤΕΧΝΟΛΟΓΙΑ ΤΟΥ BLOCKCHAIN**



**ΔΗΜΟΠΟΥΛΟΣ ΓΡΗΓΟΡΙΟΣ Α.Μ.: 23001**

**ΣΙΔΕΡΗΣ ΧΡΗΣΤΟΣ Α.Μ.: 23027**

**ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ ΣΤΟ ΜΑΘΗΜΑ**

**«Υπολογιστικά συστήματα υψηλών επιδόσεων για εφαρμογές blockchain»**

**Επιβλέπων καθηγητής**

**Κόγιας Δημήτριος**

**Αθήνα, 2024**

ΠΕΡΙΕΧΟΜΕΝΑ

[ΕΙΣΑΓΩΓΗ 3](#_Toc160482611)

[ΚΕΦΑΛΑΙΟ 1: Η ΤΕΧΝΟΛΟΓΙΑ BLOCKCHAIN ΣΤΗ ΔΙΑΔΙΚΑΣΙΑ ΤΩΝ ΦΟΙΤΗΤΙΚΩΝ ΕΚΛΟΓΩΝ 4](#_Toc160482612)

[1.1: Γιατί η τεχνολογία του Blockchain / DLT 4](#_Toc160482613)

[1.2: Ποια είναι τα οφέλη; 5](#_Toc160482614)

[1.3: Παρουσίαση σεναρίου για την εφαρμογή της λύσης 5](#_Toc160482615)

[1.4: Πηγές Έμπνευσης 7](#_Toc160482616)

[ΚΕΦΑΛΑΙΟ 2: Ο ΣΧΕΔΙΑΣΜΟΣ ΜΙΑΣ ΕΦΑΡΜΟΓΗΣ ΦΟΙΤΗΤΙΚΩΝ ΕΚΛΟΓΩΝ ΒΑΣΙΣΜΕΝΗ ΣΤΗΝ ΤΕΧΝΟΛΟΓΙΑ BLOCKCHAIN 9](#_Toc160482617)

[4.1 Τρόπος υλοποίησης 10](#_Toc160482618)

[4.2 Διαδικασία Ψηφοφορίας 12](#_Toc160482619)

[4.3 Δομή και Ανάλυση του Smart Contract της Ψηφοφορίας 14](#_Toc160482620)

[4.4 Προβλήματα και Απαιτήσεις Υλοποίησης Εφαρμογής 17](#_Toc160482621)

[4.5 O κώδικας του Smart Contract “Student\_Elections” και το Address του 18](#_Toc160482622)

[ΠΑΡΑΠΟΜΠΕΣ 25](#_Toc160482623)

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

[**Εικόνα 1.** Διάγραμμα ροής συστήματος φοιτητικών εκλογών με χρήση blockchain 6](#_Toc160482624)

[**Εικόνα 2.** Διεξαγωγή Ηλεκτρονικής Ψηφοφορίας μέσω Blockchain βασισμένο σε Πιστοποιητικά Εξακρίβωσης 9](#_Toc160482625)

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

[**Πίνακας 1.** Υιοθέτηση της τεχνολογίας Blockchain στις ηλεκτρονικές εκλογές και τα αναφερόμενα τρωτά σημεία. 8](#_Toc160482626)

[**Πίνακας 2.** Συνέπειες εκλογικών νοθειών σε διάφορες χώρες 8](#_Toc160482627)

# ΕΙΣΑΓΩΓΗ

Ο πιο συνηθισμένος τρόπος ψηφοφορίας είναι μέσω των ψηφοδελτίων. Αυτή η μέθοδος έχει επικριθεί ευρέως λόγω της δόλιας ψηφοφορίας και της σύλληψης μαρτύρων σε διάφορες χώρες παγκοσμίως. Έτσι, η χειροκίνητη ψηφοφορία αντικαταστάθηκε με ηλεκτρονικά μηχανήματα για την καταγραφή της ψήφου για μεμονωμένους πολίτες της χώρας. Οι μηχανές εξοικονόμησαν το κόστος χαρτιού και μείωσαν τον χρόνο και αντικατέστησαν τη χειροκίνητη άσκηση στη συμβατική καταμέτρηση και οδήγησε σε απόρριψη ψεύτικων ψήφων. Αυτού του είδους εκλογικές μηχανές κατάφερε περισσότερη διαφάνεια και δυνατότητα επαλήθευσης στους ψηφοφόρους της [1].

Ακόμη και μετά από όλες αυτές τις αλλαγές, εξακολουθούν να υπάρχουν αρκετές ανησυχίες για τους ψηφοφόρους. Η Τεχνολογία Κατανεμημένου Καθολικού (Distributed Ledger Technology - DLT) μπορεί να συνδυαστεί με τέτοιες μηχανές ψηφοφορίας για την παραγωγή του πιο εύρωστη και χωρίς λάθη. Το DLT είναι ασφαλές και αμετάβλητο μέσω της χρήσης σύνθετων αλγορίθμων κρυπτογράφησης. Με απλά λόγια, το Blockchain ορίζεται ως μια κατανεμημένη βάση δεδομένων της οποίας το αντίγραφο εκδίδεται σε όλους όσους εμπλέκονται στη διαδικασία συναλλαγής [2]. Ένας μπορεί να προσθέσει εγγραφές στη βάση δεδομένων αλλά δεν μπορεί να τις αλλάξει. Επομένως, τα δεδομένα αποθηκεύονται μέσα στο blockchain είναι ασφαλή, διαφανή και αδιάφθορα.

Ένα κατανεμημένο ψηφιακό καθολικό χρησιμοποιείται στην τεχνολογία Blockchain που χρησιμοποιείται για την καταγραφή συναλλαγών που συμβαίνουν μεταξύ δύο μερών. Κατά τη δημιουργία αλυσίδας μπλοκ, κάθε μπλοκ αποτελείται από δεδομένα και τα συσχετισμένη τιμή κατακερματισμού του μπλοκ που δημιουργήθηκε προηγουμένως σε μια τέτοια αλυσίδα [3]. Τα αποθηκευμένα δεδομένα μέσα σε τέτοια μπλοκ μπορεί να εξαρτάται από τον τύπο του Blockchain, ειδικά την έκδοσή του. Το "Hash" είναι το δεύτερο στοιχείο που είναι πάντα μοναδικό, πολύ παρόμοιο με ένα ανθρώπινο δακτυλικό αποτύπωμα που μπορεί να αναγνωριστεί ανάμεσα σε τρισεκατομμύρια χέρια. Το "Hash" υπολογίζεται αμέσως μετά τη δημιουργία ενός μπλοκ και το Hash προσδιορίζει το μπλοκ μαζί με το περιεχόμενό του. Τυχόν χειρισμοί στο μπλοκ θα προκαλούν αυτόματα αλλαγές στη σχετική τιμή του hash [4]. Έτσι, ο ρόλος του Hash είναι πολύ σημαντικός για την αναγνώριση οποιουδήποτε μπλοκ, εάν αυτό τροποποιηθεί. Αυτό δίνει μοναδικά χαρακτηριστικό για το Blockchain, και κάθε μπλοκ είναι συνδεδεμένο ή αλυσοδεμένο με σειρά το ένα με το άλλο. Το πρώτο μπλοκ είναι ένα αρχικό μπλοκ, και επομένως δεν έχει καμία τιμή hash και είναι επίσης γνωστό ως “genesis block”. Όταν κάποιος προσπαθεί να τροποποιήσει ή να αλλάξει τα δεδομένα σε οποιοδήποτε μπλοκ, η τιμή hash που σχετίζεται με τα μπλοκ τροποποιείται επίσης, κάτι που βοηθά στην αναγνώριση τέτοιο μπλοκ και το καταστεί ως "άκυρο". Αυτό το σενάριο κάνει μια αλυσίδα μπλοκ πιο ασφαλή και αμετάβλητη.

Η σύλληψη της ιδέας της τεχνολογίας Blockchain δημιουργεί μια σειρά από εγγραφές, δηλαδή μπλοκ, που διατηρούν δεδομένα και τη σχετική δημιουργούμενη τιμή hash (πάντα μοναδική). Αυτό δημιουργεί ένα κατανεμημένο καθολικό (ledger) που διατηρεί μία εγγραφή όλων των δεδομένων για κάθε συναλλαγή [5]. Οι τρεις κύριοι πυλώνες του Blockchain συνεπάγονται ότι είναι: αμετάβλητο, αποκεντρωμένο και διαφανές.

# ΚΕΦΑΛΑΙΟ 1: Η ΤΕΧΝΟΛΟΓΙΑ BLOCKCHAIN ΣΤΗ ΔΙΑΔΙΚΑΣΙΑ ΤΩΝ ΦΟΙΤΗΤΙΚΩΝ ΕΚΛΟΓΩΝ

## 1.1: Γιατί η τεχνολογία του Blockchain / DLT

Η τεχνολογία του Blockchain είναι η ιδανικότερη επιλογή για την ανάπτυξη ενός προηγμένου και κατανεμημένου συστήματος για τη διαδικασία των φοιτητικών εκλογών κάθε πανεπιστημίου. Οι τρεις κύριοι λόγοι είναι:

* **Αποκέντρωση**

Η ανάγκη του αποκεντρωμένου συστήματος μπορεί να γίνει κατανοητή μόνο όταν έχουμε επίγνωση σχετικά με τα τρωτά σημεία ενός κεντρικού συστήματος που χρησιμοποιείται σε ένα παραδοσιακό σύστημα μεταφοράς κεφαλαίων. Οι τράπεζες και το μοντέλο πελάτη-διακομιστή είναι παραδείγματα του κεντρικού συστήματος στο οποίο η τράπεζα ως κεντρική αρχή ελέγχει ολόκληρη τη διαδικασία συναλλαγής [6]. Για την αντιμετώπιση τέτοιων περιορισμών, εισάγεται η ιδέα ενός αποκεντρωμένου συστήματος στο οποίο δεδομένα έχουν χρησιμοποιηθεί για την αποθήκευση, καταγραφή και συγχρονισμό συναλλαγών σε διαφορετικούς κόμβους. Στην αποκέντρωση, κάθε κόμβος μπορεί να κάνει συναλλαγή που σχετίζεται με τα δεδομένα. Η Blockchain τεχνολογία έχει καθιερωθεί με τη βοήθεια κατανεμημένων δικτύων, ψηφιακής υπογραφής και τεχνικές κρυπτογράφησης/αποκρυπτογράφησης από τον τομέα ασφαλείας. Ένα αποκεντρωμένο σύστημα χρησιμοποιεί δίκτυα peer-to-peer (P2P) στα οποία κάθε κόμβος μπορεί να κατέχει το αντίγραφο των πλήρων δεδομένων στο την αλυσίδα των μπλοκ [7].

* **Διαφάνεια**

Γενικά, οι συναλλαγές Blockchain δεν είναι κρυπτογραφημένες. Το τρέχον μπλοκ αποθηκεύει το hash του προηγούμενου μπλοκ. Η τεχνική κρυπτογράφησης χρησιμοποιείται στο Blockchain το οποίο τελικά προστατεύει τα δεδομένα. Έτσι, αυτό το χαρακτηριστικό επιτρέπει στην τεχνολογία Blockchain να διατηρεί τη διαφάνεια και το απόρρητο σε όλους τους κόμβους δικτύου ομότιμων συνδέσεων. Η ταυτότητα ενός μεμονωμένου κόμβου διατηρείται κρυφή μέσω της χρήσης σύνθετων κρυπτογραφικών μοναδικών αλφαριθμητικών χαρακτήρων και συνήθως συμβολίζεται μόνο από το δημόσιο αναγνωριστικό/διεύθυνσή του [8].

* **Αμετάβλητο**

Αυτός ο όρος χρησιμοποιείται για να απεικονίσει κάτι που έχει εισέλθει στην αλυσίδα των μπλοκ και δεν μπορεί ποτέ να τροποποιηθεί ή να τροποποιηθεί ούτως ή άλλως. Παρόλο που τα δεδομένα μπορούν να προστεθούν στην αλυσίδα, αλλά τα ήδη υπάρχοντα μπλοκ δεδομένων δεν μπορούν να τροποποιηθούν. Η διαδικασία κρυπτογράφησης κατακερματισμού (hash) είναι μια βασική ιδιότητα της τεχνολογίας Blockchain. Επιπλέον, ο κατακερματισμός είναι η μεθοδολογία ή τεχνική στην οποία το μήκος των δεδομένων εισόδου είναι μεταβλητή ποσότητα ενώ το μήκος εξόδου είναι σταθερό [9].

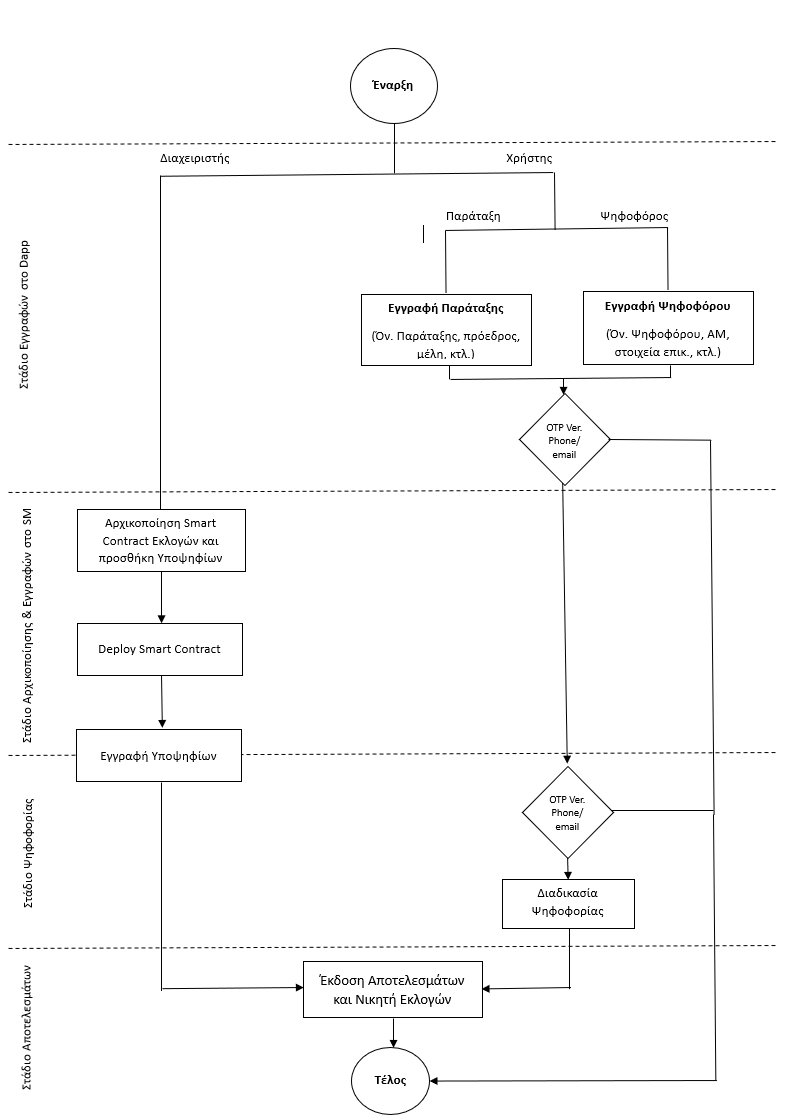
## 1.2: Ποια είναι τα οφέλη;

Το blockchain όχι μόνο μπορεί να χρησιμοποιηθεί στις φοιτητικές εκλογές, αλλά μπορεί επίσης να χρησιμοποιηθεί για τη συνολική βελτίωση διακυβέρνηση με την ενσωμάτωσή του σε συστήματα κτηματολογίου [10], στον τραπεζικό τομέα του δημόσιου τομέα [11], υγειονομική περίθαλψη [12] και οικοδόμηση έξυπνων πόλεων [13]. Όλοι οι τομείς που είναι επιρρεπείς σε εγκλήματα στον κυβερνοχώρο μπορούν να γίνουν ασφαλείς με τη βοήθεια του Blockchain. Έτσι, η καινοτομία της συγκεκριμένης εργασίας έγκειται στην ανάπτυξη ενός συστήματος βασισμένο σε Blockchain Τεχνολογία με χρήση ανιχνεύσιμων πιστοποιητικών. Τα σημαντικότερα οφέλη του Blockchain στις φοιτητικές εκλογές, αλλά και σε οποιαδήποτε άλλη εκλογική διαδικασία, είναι τα παρακάτω [14]:

* Το πρώτο όφελος που μπορεί να επιφέρει το Blockchain είναι η **διαφάνεια**. Το αποκεντρωμένο καθολικό (ledger) των αρχείων Blockchain έχουν ως αποτέλεσμα την ακρίβεια και την ασφάλεια, διασφαλίζοντας έτσι την εμπιστοσύνη σε κάθε στάδιο της διαδικασίας ψηφοφορίας
* Το **αμετάβλητο** δημόσιο ledger επιτρέπει την παρακολούθηση και την καταμέτρηση των ψήφων ενώ είναι ορατή στον καθένα. Αυτή η δυνατότητα του Blockchain παρέχει νομιμότητα της ψηφοφορίας
* Το Blockchain και το κατανεμημένο ledger του παρέχουν ένα σύστημα που δεν μπορεί να **παραβιαστεί**, καθώς δεν υπάρχει εμπλοκή εσφαλμένου ή διεφθαρμένου κεντρικού φορέα
* Το Blockchain επιτρέπει την **ανωνυμία** κατά την ψηφοφορία παρέχοντας **ιδιωτικά κλειδιά** στους ψηφοφόρους. Αυτές οι εφαρμογές τέτοιων ιδιωτικών κλειδιών διατηρούν ανώνυμες τις ψήφους που ρωτήθηκαν από τους ψηφοφόρους.
* Ο χρόνος επεξεργασίας μειώνεται στο Blockchain επειδή τα αποτελέσματα μπορούν να συγκεντρωθούν και να **υποβληθούν σε επεξεργασία αμέσως** μετά την ολοκλήρωση της φάσης της ψηφοφορίας.
* Το Blockchain είναι η απόλυτη λύση για τη **διασφάλιση** των εκλογών

## 1.3: Παρουσίαση σεναρίου για την εφαρμογή της λύσης

Οι φοιτητικές εκλογές μπορούν να ξεκινήσουν μόνο μετά την ολοκλήρωση του smart contract. Ο λογαριασμός που θα χρησιμοποιεί το contract θα δίνεται σε καθεστώς διαχειριστή και έτσι θα μπορεί να έχει πρόσβαση στη διαχειριστική πλευρά του ιστοτόπου. Αυτό θα επιτρέψει στον διαχειριστή να αρχικοποίηση την έναρξη και την λήξη των εκλογών. Ακολουθεί το διάγραμμα ροής του συστήματος φοιτητικών εκλογών με χρήση τεχνολογίας Blockchain.



**Εικόνα 1.** Διάγραμμα ροής συστήματος φοιτητικών εκλογών με χρήση blockchain

Βήμα 1ο: Επαλήθευση Χρήστη μέσω OTP

Για να έχουν δικαίωμα εγγραφής και ψήφου, όλοι οι χρήστες θα πρέπει να επαληθεύσουν τον καταχωρημένο αριθμό κινητού τηλεφώνου ή email σε μια υπηρεσίας παρόχου επαλήθευσης OTP προτού ολοκληρώσουν την εγγραφή τους στην πλατφόρμα. Μετά την επαλήθευση μέσω OTP, θα έχουν πρόσβαση στην πλατφόρμα ψηφοφορίας.

Βήμα 2ο: Έναρξη εκλογών

Ο διαχειριστής θα πρέπει να ορίσει στο smart contract τα ονόματα των παρατάξεων που έβαλαν υποψηφιότητα και μετά να κάνει deploy το smart contract σε ένα EVM δίκτυο, ορίζοντας τη διάρκεια των εκλογών όπου θα μπορούν οι έγκυροι χρήστες να ψηφίσουν.

Βήμα 3ο: Προσθήκη ψηφοφόρων

Ο διαχειριστής θα μπορεί να προσθέσει τις μοναδικές διευθύνσεις όλων των ταυτοποιημένων ψηφοφόρων στο smart contract. Κάθε ψηφοφόρος ορίζεται από τη συνάρτηση να έχει δικαίωμα για μία ψήφο.

Βήμα 4ο: Διεξαγωγή Ψηφοφορίας

Μόλις επαληθευτεί το **κινητό ή email μέσω OTP** ενός ψηφοφόρου, θα του δοθεί δικαίωμα ψήφου. Στην ιστοσελίδα ψηφοφορίας θα βρουν μια λίστα με όλους τους υποψηφίους από τους οποίους θα μπορούν να ψηφίσουν μόνο έναν για να κερδίσει.

Βήμα 5ο: Έκδοση Αποτελεσμάτων

Αμέσως μετά την λήξη της ψηφοφορίας, όλοι οι χρήστες του EVM δικτύου θα μπορούν να έχουν πρόσβαση στην έκδοση των τελικών αποτελεσμάτων και του νικητή καλώντας τις αντίστοιχες συναρτήσεις του smart contract.

## 1.4: Πηγές Έμπνευσης

Αυτή η ιδέα του Blockchain για τη διεξαγωγή ασφαλών εκλογών έχει εφαρμοστεί στο παρελθόν από εταιρείες όπως η Agora και η Polys αλλά η πρώτη δεν μπόρεσε να δικαιολογήσει την παρουσία της και είχε μια διαμάχη με την κυβέρνηση της Σιέρα Λεόνε ενώ η τελευταία δεν πέτυχε ποτέ επεκτασιμότητα σε οποιεσδήποτε πολιτειακές εκλογές. Υπήρξαν αρκετές προκλήσεις που πρέπει να διατηρηθούν κατά τον σχεδιασμό του συστήματος για τη διεξαγωγή ηλεκτρονικών εκλογών [14][15].

* Δυσκολία ενοποίησης με παλαιού τύπου συστήματα.
* Πολυπλοκότητα και έλλειψη ταλαντούχου προσωπικού Blockchain.
* Έλλειψη επεκτασιμότητας.
* Έλλειψη διαλειτουργικότητας.
* Έλλειψη καλής διακυβέρνησης.
* Έλλειψη εμπειρίας χρήστη και εκπαίδευσης.

Οι πηγές έμπνευσης ήταν αρκετά αξιοσημείωτες προσπάθειες που έγιναν τα τελευταία χρόνια σε διάφορες χώρες στον κόσμο και αφορούσαν εκλογές μέσω τεχνολογίας Blockchain. Ωστόσο καμία από αυτές δεν έχει επιτύχει την επεκτασιμότητα που απαιτείται για να είναι επιτυχής η ψηφοφορία με βάση το Blockchain, όπως φαίνεται στο Πίνακας 1 [15][16].

**Πίνακας 1.** Υιοθέτηση της τεχνολογίας Blockchain στις ηλεκτρονικές εκλογές και τα αναφερόμενα τρωτά σημεία.

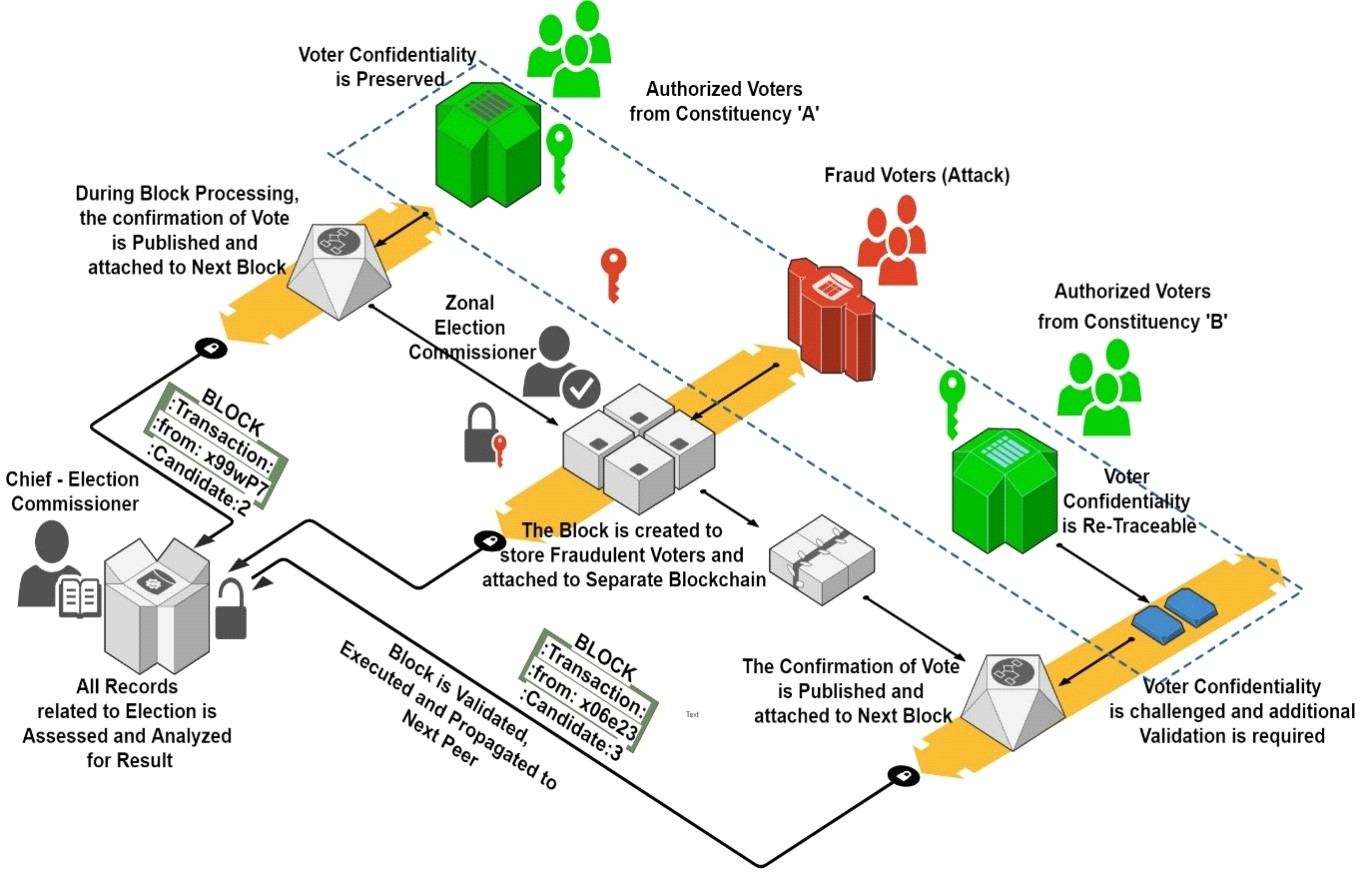
|  |  |  |
| --- | --- | --- |
| **Έτος** | **Χώρα** | **Αποτελέσματα** |
| Αύγουστος 2018 | Ιαπωνία | Δοκιμασμένο μόνο για κοινωνικούς σκοπούς αλλά όχι για εκλογές. (Κρατική χορηγία) |
| Νοέμβριος 2018 | ΗΠΑ | Δεν καλύπτονταν όλα τα τρωτά σημεία.  (εφαρμογή με όνομα Voatz με έδρα τη Βοστώνη) |
| Μάρτιος 2018 | Σιέρα Λεόνε | Επισήμως δεν έγινε αποδεκτό. (Επωνυμία εταιρείας με έδρα την Ελβετία Agora) |
| Ιούνιος 2019 | Ρωσία | Διεξήχθησαν εκλογές για την πόλη της Μόσχας |
| Ιούνιος 2019 | Αφρικανικά Έθνη | Ευελιξία και επαρκής ασφάλεια στην εκλογική διαδικασία. |

Λόγω της παρουσίας των προαναφερθέντων ιδιοκτησιών στον παραπάνω πίνακα και αφού εξέτασαν τις απάτες που συνέβη σε ψηφιακά εκλογικά συστήματα όπως αναλύεται στον Πίνακα 2, η τεχνολογία Blockchain συνιστάται να χρησιμοποιείται σε Ηλεκτρονικές Μηχανές Ψηφοφορίας ούτως ώστε να γίνουν πιο έξυπνες και ασφαλής [17][18].

**Πίνακας 2.** Συνέπειες εκλογικών νοθειών σε διάφορες χώρες

|  |  |
| --- | --- |
| **ΧΩΡΑ** | **ΖΗΤΗΜΑ** |
| Ινδία | Κατάληψη του θαλάμου και νοθεία (booth) |
| ΗΠΑ | Νοθεία μέσω hacking |
| Ρωσία | Περισσότεροι ψηφοφόροι από το ανώτερο όριο |
| Ηνωμένο Βασίλειο | Ψηφοφορία μέσω αντιπροσώπου |
| Νιγηρία | Απομίμηση ψηφοφόρων και σύλληψη περιπτέρου |
| Γερμανία | Τα EVM είναι επιρρεπή σε hacking |
| Ολλανδία | Έλλειψη διαφάνειας στα EVM |
| Ιρλανδία | Έλλειψη διαφάνειας και εμπιστοσύνης στα EVM |

Η αντίληψη που σχετίζεται με τη διεξαγωγή ψηφοφορίας από εξουσιοδοτημένους ψηφοφόρους μπορεί να απεικονιστεί στην **Εικόνα 2**, που δικαιολογεί τη χρήση της τεχνολογίας Blockchain σε ένα τέτοιο σύστημα[19][20]. Όταν ένας απατεώνας ψηφοφόρος διεισδύσει στο σύστημα μέσω πλαστών διαπιστευτηρίων μπορεί αμέσως να γίνει αντιληπτό στο γραφείο διοίκησης από τις αρχές. Τέτοιες κακόβουλες δραστηριότητες μπορούν εύκολα να προσδιοριστεί μέσω της τεχνολογίας Blockchain [21]. Το πρώτο μπλοκ που ονομάζεται genesis block δημιουργείται με νόμιμα δεδομένα που σχετίζονται με την ταυτότητα συναλλαγής, τη διεύθυνση πηγής/προορισμού, στοιχεία ψηφοφόρου/υποψηφίου κ.λπ.



**Εικόνα 2.** Διεξαγωγή Ηλεκτρονικής Ψηφοφορίας μέσω Blockchain βασισμένο σε Πιστοποιητικά Εξακρίβωσης

# ΚΕΦΑΛΑΙΟ 2: Ο ΣΧΕΔΙΑΣΜΟΣ ΜΙΑΣ ΕΦΑΡΜΟΓΗΣ ΦΟΙΤΗΤΙΚΩΝ ΕΚΛΟΓΩΝ ΒΑΣΙΣΜΕΝΗ ΣΤΗΝ ΤΕΧΝΟΛΟΓΙΑ BLOCKCHAIN

Τα ιδρύματα τριτοβάθμιας εκπαίδευσης, όπως τα πανεπιστήμια, διενεργούν συχνά φοιτητικές εκλογές για διάφορες θέσεις φοιτητών, εκπροσώπους διοικητικών συμβουλίων σχολών και άλλους διοικητικούς ρόλους. Η αξιοπιστία, η διαφάνεια και η αποτελεσματικότητα αυτών των εκλογών είναι πρωταρχικής σημασίας για να διασφαλιστεί ότι οι εκλεγμένοι αντιπρόσωποι αντιπροσωπεύουν πραγματικά τη βούληση της πανεπιστημιακής κοινότητας.

Οι παραδοσιακές φοιτητικές εκλογές περιλαμβάνουν συχνά ψηφοδέλτια σε χαρτί, τα οποία καταναλώνουν πόρους, είναι επιρρεπή σε ανθρώπινα λάθη και υστερούν σε έλεγχο. Υπάρχει μια αυξανόμενη ανάγκη για λύσεις εξ αποστάσεως ψηφοφορίας, ειδικά για πανεπιστήμια με πολλαπλές πανεπιστημιουπόλεις ή σημαντικό αριθμό φοιτητών εξ αποστάσεως εκπαίδευση. Επίσης υπάρχουν ανησυχίες για παραποίηση ψηφοδελτίων, νοθεία των ψηφοφόρων και τη συνολική ακεραιότητα των εκλογικών αποτελεσμάτων.

## 4.1 Τρόπος υλοποίησης

Για να μπορέσουμε να υλοποιήσουμε μια τέτοια εφαρμογή η οποία να εξυπηρετεί όλα όσα προαναφέραμε στο προηγούμενο κεφάλαιο θα πρέπει να χρησιμοποιήσουμε τα συγκεκριμένα εργαλεία-πλατφόρμες για να μπορέσουμε να αλληλοεπιδράσουμε με το Smart Contract μας που θα βρίσκεται στο δίκτυο του Blockchain. Για να μπορέσει ένας χρήστης να εγγραφεί και να ψηφίσει στις εκλογές θα πρέπει απλά να χρησιμοποιήσει ένα Dapp το οποίο θα αντιπροσωπεύει όλη τη λειτουργία του συστήματος.

Τα κύρια εργαλεία για δημιουργηθεί ένα σύστημα ψηφοφορίας είναι:

1. **Μια εφαρμογή ιστοτόπου**: Η συγκεκριμένη εφαρμογή θα αλληλοεπιδράει με τους αντιπροσώπους παρατάξεων και τους ψηφοφόρους. Οι σελίδες της εφαρμογής θα είναι η εγγραφή της παράταξης, η εγγραφή του ψηφοφόρου και αυτή της ψηφοφορίας. Οι φοιτητές μέσω της *σελίδας εγγραφής* ψηφοφόρου θα έχουν τη δυνατότητα να ψηφίσουν τις παρατάξεις αφού πρώτα δηλώσουν τα βασικά τους στοιχεία (πχ. ονοματεπώνυμο, ΑΜ, κτλ.), τα στοιχεία επικοινωνίας (κινητό τηλέφωνο, φοιτητικό e-mail, κτλ.) και να επικυρώσουν την ταυτότητα τους με τη χρήση μια υπηρεσίας παρόχου επαλήθευσης OTP (πχ. SMSCountry, 2Factor, MojoAuth, Signicat κτλ.). Σημαντικότερο όλων είναι οι ψηφοφόροι να έχουν δημιουργήσει ένα ηλεκτρονικό πορτοφόλι στο MetaMask ή κάποιο άλλο (Ledger, Guarda, κτλ.) και να δηλώσει την διεύθυνση πορτοφολιού του στη σελίδα εγγραφή ψηφοφόρου ούτως ώστε να μπορεί να συναλλαχθεί με το EVM δίκτυο. Από την άλλη μεριά οι υποψήφιες παρατάξεις θα πρέπει να μπούνε στην *σελίδα εγγραφής παράταξης* για να δηλώσουν τα στοιχεία της παράταξης (πχ. Όνομα παράταξης, μέλη, email παράταξης κτλ.) και να επικυρώσουν κι αυτοί την ταυτότητα τους με τη χρήση μια υπηρεσίας παρόχου επαλήθευσης OTP. Έπειτα αφού έχει ολοκληρωθεί το στάδιο εγγραφών στον ιστότοπο, ο διαχειριστής πρέπει να αρχικοποιήσει στο smart contract τα ονόματα των παρατάξεων και σε δεύτερο στάδιο να εκτελέσει το deploy του συμβολαίου στο δίκτυο. Με το deploy του συμβολαίου ο διαχειριστής θα έχει ένα χρονικό διάστημα να εγγράψει τις διευθύνσεις του κάθε εγγεγραμμένου ψηφοφόρου στη σελίδα καλώντας τη συνάρτηση *registerVoter* δημιουργώντας συναλλαγές για κάθε ψηφοφόρο στα smart contract. Μετά το στάδιο της εγγραφής των ψηφοφόρων στο Blockchain, οι εγγεγραμμένοι χρήστες έχουν δικαίωμα να ψηφίσουν στην σελίδα ψηφοφορίας την παράταξη που επιθυμούν να κερδίσει. Για να μπορέσει η εφαρμογή να τρέξει θα πρέπει να είναι συνδεδεμένη με ένα εικονικό δίκτυο Blockchain του Ethereum. Για να γίνει η διασύνδεση εφαρμογής και EVM θα χρειαστεί να ενσωματωθεί μια εφαρμογή MetaMask ή να έχουμε ένα λογαριασμό στο MetaMask. Με την ολοκλήρωση της ψηφοφορίες οι πληροφορίες (τα ονόματα παρατάξεων, το σύνολο των ψήφων, οι διευθύνσεις των εγγεγραμμένων ψηφοφόρων καθώς και οι ψήφοι τους) μεταφέρονται από την ιστοσελίδα στο δίκτυο Blockchain.
2. **Ο Διαχειριστής των συμβάντων**: Ο Διαχειριστής είναι αυτός που θα δημιουργήσει το smart contract στο εικονικό δίκτυο Ethereum και θα περιλαμβάνει όλα τα δεδομένα που θα λαμβάνει από την ιστοσελίδα. Οπότε στη διάθεση του θα πρέπει κατέχει ένα Ethereum πορτοφόλι (MetaMask , Ledger, κτλ.) με μια διεύθυνση για να χτίσει το smart contract και πρόσβαση στα off-chain δεδομένα εφαρμογής που αφορούν τις διευθύνσεις των εγγεγραμμένων φοιτητών και είναι αποθηκευμένα σε ένα κατανεμημένο δίκτυο κόμβων αποθήκευσης μέσω του συστήματος. Επιπροσθέτως θα μπορούσε να χτιστεί μια εφαρμογή όπου θα συνδέεται με ένα πληροφοριακό σύστημα του πανεπιστημίου και θα κάνει την ταυτοποίηση των στοιχείων των χρηστών.
3. **Έξυπνα συμβόλαια**: Το smart contract είναι κώδικας, συνήθως σε Solitidy ή Vyper, ο οποίος έχει σκοπό να διασυνδέσει τους συμμετέχοντες με ένα ιδιωτικό δίκτυο Blockchain. Στην προκειμένη περίπτωση το Smart Contract “*Student\_Elections*” είναι γραμμένο σε κώδικα Solidity και περιλαμβάνεις τις εξής functions: *registerVoters*, *vote* και *showWinner* οι οποίες έχουν σκοπό να μεταφέρουν δεδομένα στο EVM με τα στοιχεία της ψηφοφορίας. Σε επόμενο κεφάλαιο θα αναλυθεί η δομή και η λειτουργία του smart contract που δημιουργήθηκε για το σκοπό της κατανόησης της εργασίας. Το συγκεκριμένο συμβόλαιο εν ολίγοις προσθέτει τους ψηφοφόρους και εκκινεί την ψηφοφορία ορίζοντας μία ψήφο για κάθε ψηφοφόρο. Ουσιαστικά επικυρώνει τους χρήστες και κατά τη διάρκεια της ψηφοφορίας αυξάνει τις ψήφους της κάθε παράταξης ανάλογα τις προτιμήσεις των ψηφοφόρων και στο τέλος εκδίδει τα συνολικά αποτελέσματα και τον νικητή των φοιτητικών εκλογών.
4. **Ιnfura - IPFS (InterPlanetary File System)**: Το IPFS είναι ένα πρωτόκολλο, υπερμέσο και peer-to-peer δίκτυο για την αποθήκευση και την κοινή χρήση δεδομένων σε ένα κατανεμημένο σύστημα αρχείων. Το IPFS χρησιμοποιεί τη διεύθυνση περιεχομένου για να προσδιορίζει μοναδικά κάθε αρχείο σε έναν παγκόσμιο χώρο ονομάτων που συνδέει κεντρικούς υπολογιστές IPFS. Στη συγκεκριμένη διαδικασία είναι απαραίτητο γιατί μέσω της αποκεντρωμένης διαδικτυακής εφαρμογής οι ψηφοφόροι θα έχουν τη δυνατότητα να ψηφίσουν από οποιαδήποτε σημείο του κόσμου στο διαδίκτυο. Με την χρήση του Infura μας δίνεται η δυνατότητα να αποθηκεύσουμε τα στοιχεία των παρατάξεων και των φοιτητών εκτός αλυσίδας (off-chain), όπου για να μην υπάρξει αλλοίωση δεδομένων στο μέλλον, το παραγόμενο hash δεδομένων από το IPFS μπαίνει στο blockchain και στο ledger. Ο πάροχος κόμβου(MetaMask ) του δικτύου είναι αυτός που συνδέεται με το κατανεμημένο δίκτυο κόμβων αποθήκευσης όπου βρίσκονται τα off-chain δεδομένα.
5. **MetaMask**: είναι λογισμικό (μια επέκταση στο browser ή mobile app) στο οποίο δημιουργούμε ένα πορτοφόλι crypto για το δίκτυο του Blockchain και μπορούμε να αλληλοεπιδράσουμε με αποκεντρωμένες εφαρμογές “Dapps” στο δικό μας browser. Το MetaMask (όπως και το Infura, Alchemy και Quicknode) είναι ένας πάροχος κόμβου όπου μέσω του πρωτοκόλλου Web 3.0 ουσιαστικά ο χρήστης επικοινωνεί με την Dapp εφαρμογή του στο EVM δίκτυο και στο IPFS κατανεμημένο δίκτυο αρχείων. Η επέκταση του MetaMask εισάγει το **Ethereum Web3 API** και το περιβάλλον js κάθε ιστοσελίδας, ούτως ώστε να μπορούν να επικοινωνήσουν με το blockchain. Επιπλέον ο χρήστης μπορεί να έχει τις δικές του ταυτότητες (διευθύνσεις) μέσω των ιδιωτικών κλειδιών και έτσι μόλις το Dapp θέλει να φτιάξει μια συναλλαγή να την μεταφέρει στο δίκτυο του Blockchain. Συνοπτικά για να συνδεθούμε στο κύριο δίκτυο Ethereum, θα χρειαστούμε ένα συμβατό πορτοφόλι ή πρόγραμμα-πελάτη λογισμικού που να υποστηρίζει την αλληλεπίδραση με αυτό το δίκτυο. Οι δημοφιλείς επιλογές πέρα από το MetaMask είναι το MyEtherWallet και το Geth. Αφού επιλέξουμε τη μέθοδο σύνδεσης που προτιμάμε, θα πρέπει να ακολουθήσουμε τις αντίστοιχες οδηγίες για τη ρύθμιση και την πρόσβαση στο κύριο δίκτυο Ethereum ή σε ένα EVM. Ουσιαστικά το πρόγραμμα-πελάτη μας παρέχει ένα κόμβο στο blockchain όπου μας δίνει τη δυνατότητα να χρησιμοποιούμε το Ethereum με έναν πραγματικά ιδιωτικό, αυτάρκη και αξιόπιστο τρόπο. Δεν χρειαζόμαστε να εμπιστευόμαστε τις πληροφορίες που λαμβάνουμε, επειδή μπορούμε να επαληθεύσουμε τα δεδομένα μόνοι μας μέσω του MetaMask . Τέλος ο χρήστης πριν την ολοκλήρωση της ψηφοφορίας θα λαμβάνει μια ειδοποίηση για την έγκριση ή όχι της ψήφου (δηλαδή τη οριστικοποίηση της εγγραφή της συναλλαγής στο Blockchain).

## 4.2 Διαδικασία Ψηφοφορίας

Η διαδικασία της ψηφοφορίας είναι δημοκρατική και θα πρέπει να είναι ασφαλή και διαφανής για την έκδοση των τελικών αποτελεσμάτων της. Η υλοποίηση ενός Dapp για την ψηφοφορίας είναι κομβική επειδή θα έχει σκοπό την επιλογή του επικρατέστερου υποψηφίου ανάμεσα σε μία λίστα υποψηφίων. Στην προκειμένη περίπτωση θα πρέπει να αναπτύξουμε ένα Dapp από το οποίο θα πρέπει να επιλεχτεί η επικρατέστερη φοιτητική παράταξη ενός πανεπιστημίου.

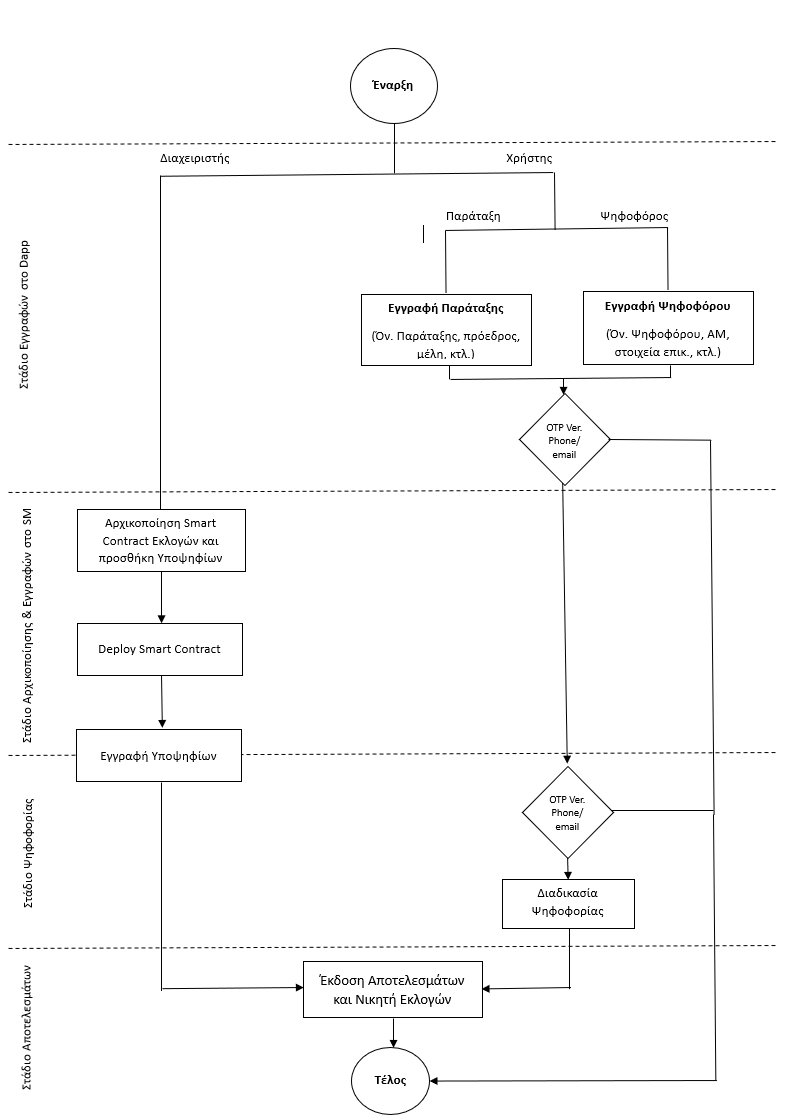
Οι κανόνες και περιορισμοί του συστήματος ψηφοφορίας Blockchain είναι οι εξής:

* + Οι εγγραφές των ψηφοφόρων και των παρατάξεων στην ιστοσελίδα της ψηφοφορίας είναι απαραίτητες για να δηλωθούν και να εξακριβωθούν τα στοιχεία των φοιτητών του πανεπιστημίου.
  + Κατά την ολοκλήρωση της εγγραφής στην ιστοσελίδα υπάρχει ένα βήμα 2-way-authentication όπου ο χρήστης επαληθεύει το ακαδημαϊκού του email ή το κινητό του τηλέφωνο.
  + Ο διοργανωτής-διαχειριστής της ψηφοφορίας ενεργοποιεί το smart contract και αρχικοποιεί τη λίστα παρατάξεων προς ψήφιση στο smart contract.
  + Όλοι οι ψηφοφόροι ταυτοποιούνται από τις μοναδικές διευθύνσεις των λογαριασμών των ηλεκτρονικών πορτοφολιών τους.
  + Την κεντρική διαχείριση των εγγραφών των ψηφοφόρων στο έξυπνο συμβόλαιο την αναλαμβάνει μόνο ένας, ο διοργανωτής (chairperson).
  + Δικαίωμα στην ψηφοφορία έχουν μόνο οι εγγεγραμμένοι χρήστες και μπορούν να ψηφίσουν μόνο μια φορά από τη διεύθυνση πορτοφολιού τους που έχουν δηλώσει κατά την εγγραφή τους στην ιστοσελίδα.
  + Οι ψηφοφόροι μπορούν να ψηφίζουν μόνο από τη λίστα εγγεγραμμένων υποψήφιων παρατάξεων που θα παρέχετε.
  + Η διαδικασίας της ψηφοφορίας θα ακολουθήσει 3 βήματα: την εγγραφή των ψηφοφόρων από των διαχειριστή, την διαδικασία ψηφοφορίας από τους εγγεγραμμένους φοιτητές μόνο, και την έκδοση του νικητή της ψηφοφορίας και των τελικών αποτελεσμάτων.
  + Ο ιδιοκτήτης-διαχειριστής του Smart Contract δεν έχει δικαίωμα να ψηφίσει.
  + Η ολοκλήρωση της διαδικασίας ψηφοφορίας γίνεται μετά την πάροδο του χρονικού διαστήματος που έχει οριστεί κατά το deploy του συμβολαίου από τον ιδιοκτήτη. Θα μπορούσαμε να ορίσουμε και μια συνθήκη στο συμβόλαιο όπου όταν όλοι οι ψηφοφόροι έχουν ψηφίσει να προχωράει στην έκδοση τελικών αποτελεσμάτων. Ωστόσο κατά τη διάρκεια της ψηφοφορίας δίνουμε τη δυνατότητα στον διαχειριστή του συμβολαίου να κάνει κάποια επιπρόσθετη εγγραφή ψηφοφόρου σε περίπτωση ανθρώπινου λάθους.

Τα στάδια της ψηφοφορίας του συστήματος είναι τα ακόλουθα:

* Εγγραφή: Η λίστα των ψηφοφόρων και των παρατάξεων δημοσιεύονται στο διαδίκτυο από τον διοργανωτική αρχή των εκλογών και αφού δεν υπάρχουν αντιρρήσεις εκδίδεται η τελική λίστα ψηφοφόρων και παρατάξεων
* Ψηφοφορία: Περιλαμβάνει δύο φάσεις, την εγγραφή ψηφοφόρων από των διαχειριστή και την ψηφοφορία. Στη φάση της εγγραφής ο διαχειριστής του smart contract ενημερώνει το συμβόλαιο με τις διευθύνσεις των ψηφοφόρων και στη φάση ψηφοφορίας οι φοιτητές στέλνουν την ψήφο τους στην παράταξη που θέλουν να κερδίσει.
* Καταμέτρηση: Η διαδικασία των εκλογών τερματίζεται μετά το πέρας ενός χρονικού διαστήματος. Ουσιαστικά το έξυπνο συμβόλαιο τερματίζει και κανείς δεν μπορεί να συναλλαχθεί με αυτό πια. Έπειτα τα τελικά αποτελέσματα εκδίδονται στο blockchain δίκτυο όπου όλοι μπορούν να έχουν πρόσβασης σε αυτά.

Ακολουθεί το διάγραμμα ροής της εφαρμογής Dapp των φοιτητικών εκλογών το οποίο ακολουθεί τα βήματα και στάδια αυτής της εργασίας (**Εικόνα 1**).



## 4.3 Δομή και Ανάλυση του Smart Contract της Ψηφοφορίας

Σε αυτή την εργασία προχωρήσαμε στη δημιουργία ενός smart contract και το κάναμε deploy στο δοκιμαστικό δίκτυο (EVM) Sepolia του Ethereum ούτως ώστε να τεστάρουμε τη λειτουργικότητα των φοιτητικών εκλογών και την αλληλοεπίδραση με ένα Blockchain δίκτυο. Για να μπορέσουμε να κατανοήσουμε την όλη διαδικασία της Dapp εφαρμογής, η οποία ουσιαστικά περνάει συναλλαγές στο δίκτυο επικοινωνώντας με το smart contract, είναι αναγκαίο να αναλύσουμε την δομή και την λειτουργία του συμβολαίου.

Το smart contract με όνομα “*Student\_Elections*” γράφτηκε σε γλώσσα Solidity και ο σκελετός του αποτελείται από structs, functions, δημόσιες μεταβλητές και ένα constructor για την αρχικοποίηση κάποιων μεταβλητών.

Οι δομές (structs) του smart contract είναι οι εξής:

* **Voter**: αποθηκεύει πληροφορίες σχετικές με τον υποψήφιο όπως εάν έχει δικαίωμα ψήφου ή πολλών ψήφων (weight), εάν έχει ψηφίσει (voted) και τον δείκτη της παράταξης που ψήφισε. Τέλος ο δείκτης στον πίνακα δομών που δημιουργείται στο συμβόλαιο έχει την μοναδική διεύθυνση του ψηφοφόρου όπου εν ολίγοις τον ταυτοποιεί.
* **Proposal**: αποθηκεύει πληροφορίες της παράταξης. Αυτές είναι το όνομα και ο αριθμός ψήφων της παράταξης.

H δημόσιες μεταβλητές του έξυπνου συμβολαίου είναι:

* **Chairperson**: αποθηκεύεται η διεύθυνση του διαχειριστή του συμβολαίου
* **Voters**: είναι ο πίνακας υποψηφίων όπου ο δείκτης πίνακα αντί για ακέραιο έχει αντικατασταθεί από την μοναδική διεύθυνση του ψηφοφόρου χρησιμοποιώντας mapping.
* **Proposals**: είναι ο πίνακας όπου καταχωρούνται όλες οι υποψήφιες παρατάξεις
* **startupTime**: κρατάει την ώρα όπου δημιουργήθηκε το smart contract στο δίκτυο blockchain
* **stageLimit**: είναι το χρονικό διάστημα του κάθε σταδίου της ψηφοφορίας και ορίζεται στον constructor
* **proposalNames**: είναι ο πίνακας ονομάτων των παρατάξεων τύπου bytes32

Ο constructor καλείται όταν γίνεται deploy το συμβόλαιο και παίρνει ένα όρισμα το οποίο είναι το χρονικό όριο του κάθε stage της ψηφοφορίας. Κατά το κάλεσμα του αρχικοποιεί κύριες μεταβλητές του συμβολαίου όπως τη διεύθυνση του διαχειριστή, στην ώρα δημιουργίας συμβολαίου, το χρονικό διάστημα του σταδίου και τη λίστα των παρατάξεων.

Τέλος οι συναρτήσεις του έξυπνου συμβολαίου οι οποίες ουσιαστικά κινούν όλα τα νήματα της διαδικασίας της ψηφοφορίας είναι:

* **getBytes32ArrayForConstructor (write):** Αυτή η συνάρτηση επιστρέφει ένα πίνακα bytes32, ο οποίος περιλαμβάνει τα ονόματα των υποψηφίων που ορίζει χειροκίνητα στον κώδικα του συμβολαίου ο διαχειριστής πριν το deploy.
* **registerVoters (write):** Είναι η συνάρτηση όπου εκτελείται μόνο από τον διαχειριστή συμβολαίου και χρησιμοποιείται ουσιαστικά για να δώσει δικαιώματα στα εγγεγραμμένα addresses να ψηφίσουν. Για να την καλέσουμε δίνουμε όρισμα την διεύθυνση του υποψηφίου και εσωτερικά στη συνάρτηση δημιουργεί μια θέση στη δομή Voter με τη διεύθυνση του και το δικαίωμα να ψηφίσει ορίζοντας default τιμές. Τέλος η συνάρτηση μπορεί να καλείται από τον ιδιοκτήτη κατά τη διάρκεια των εγγραφών αλλά και της ψηφοφορίας.
* **Vote (write):** Είναι η συνάρτηση όπου εκτελείται μόνο από τους ταυτοποιημένους και εγγεγραμμένους ψηφοφόρους στο συμβόλαιο οι οποίοι έχουν αποθηκευτεί στον πίνακα δομής Voters. Ο φοιτητής θα πρέπει κατά το κάλεσμα της συνάρτησης να ορίσει ένα index το οποίο αφορά την παράταξη που επιθυμεί να ψηφίσει. Φυσικά δεν μπορεί να ψηφίσει κάποιος που έχει ήδη ψηφίσει ή κάποιος που δεν έχει δικαίωμα στην ψηφοφορία. Εάν o δείκτης δεν αντιπροσωπεύει κάποια παράταξη, η συναλλαγή δεν ολοκληρώνεται και πετάει σφάλμα. Επίσης η συγκεκριμένη συνάρτηση είναι ενεργή μόνο κατά τη διάρκεια του σταδίου ψηφοφορίας!
* **showWinner (read):** Την συνάρτηση αυτή μπορεί να την καλέσει οποιοσδήποτε χρήστης στο blockchain δίκτυο χωρίς κάποια χρέωση για να εμφανίσει τον τελικό νικητή των εκλογών. Η ενεργοποίηση της συνάρτησης αυτής γίνεται στο τελικό στάδιο αποτελεσμάτων μετά την ολοκλήρωση της ψηφοφορίας. Σε σπάνια περίπτωση ισοβαθμίας εμφανίζει όλες τις ισοψηφείς παρατάξεις.
* **showFinalResults(read):** Την συνάρτηση αυτή μπορεί να την καλέσει οποιοσδήποτε χρήστης στο blockchain δίκτυο χωρίς κάποια χρέωση για να εμφανίσει τα τελικά αποτελέσματα των φοιτητικών εκλογών. Επίσης η ενεργοποίηση της συνάρτησης αυτής γίνεται στο τελικό στάδιο αποτελεσμάτων μετά την ολοκλήρωση της ψηφοφορίας.
* **sortProposals (write):** Η συνάρτηση αυτή καλείται κατά το στάδιο αποτελεσμάτων μόνο από τον ιδιοκτήτη συμβολαίου για να ταξινομήσει σε φθίνουσα σειρά τον πίνακα παρατάξεων με βάση τις συνολικές ψήφους των εκλογών.
* **bytes32ToString**: Βοηθητική συνάρτηση μετατροπής μια τιμής bytes32 τύπου μεταβλητής σε τύπου String.
* **uintToString**: Βοηθητική συνάρτηση μετατροπής μια τιμής uint τύπου μεταβλητής σε τύπου String.

Σημειώνουμε ότι όσες συναρτήσεις αναγράφονται ως “write” θα πρέπει ο χρήστης για να την καλέσει να καταβάλει ένα ειδικό φόρο για να μπορέσει να γίνει η συναλλαγή μέσω του MetaMask. Αντιθέτως, οι συναρτήσεις τύπου “read” μπορούν να γίνουν κλήση δωρεάν για την εμφάνιση των αποτελεσμάτων των εκλογών. Φυσικά προϋπόθεση για να συνδεθούμε στο smart contract είναι να διαθέτουμε ένα ηλεκτρονικό πορτοφόλι που να θα μας συνθέσει στο blockchain δίκτυο μέσω του web 3.0. Οι δοκιμές του έξυπνου συμβολαίου μας έχουν γίνει στο Sepolia Testnet όπου όλες οι συναλλαγές των συμβολαίων γίνονται με το τεστ κρυπτονόμισμα SepoliaETH. Στην περίπτωση που το συμβόλαιο μας βρισκόταν στο Ethereum Mainnet θα έπρεπε να πληρώσουμε το αντίστοιχο ποσό σε Ethereum, δηλαδή σε Gwei όπου είναι μια μικρότερη μονάδα του Ethereum και ισούται με 0.000000001 ETH.

Η διάρκεια ζωής του συμβολαίου ορίζεται κατά το κάλεσμα του constructor στο deploy του smart contract. Το πόσο διάρκεια θα έχει ένα συμβόλαιο θα οριστεί από την εκλογική αρχή του κάθε πανεπιστημίου. Το όρισμα στο constructor δηλώνει τη χρονικό διάστημα του κάθε σταδίου, όπου x μπορεί να είναι 3600sec (δηλαδή 1 ώρα) για το 1ο στάδιο και άλλη μια ώρα για το 2ο στάδιο. Το 3ο στάδιο είναι η έκδοση των αποτελεσμάτων που δεν έχει χρονικό περιορισμό μιας και εμφανίζονται εφάπαξ . Η εκλογική διαδικασία του έξυπνου συμβολαίου όπως προαναφέραμε έχει 3 στάδια:

* **Εγγραφών**: O διαχειριστής συμβολαίου έχει ένα χρονικό περιθώριο, ορισμένο κατά το deploy, όπου πρέπει να εγγράψει όλους τους ταυτοποιημένους φοιτητές κατοχυρώνοντας τις διευθύνσεις τους στο smart contract.
* **Ψηφοφορίας**: Οι φοιτητές έχουν ένα χρονικό διάστημα ίσο με το στάδιο εγγραφών όπου μπορούν να ψηφίσουν την παράταξη που υποστηρίζουν. Στο στάδιο αυτό ο διαχειριστής δεν μπορεί να ψηφίσει, ωστόσο μπορεί να συνεχίσει με κάποια επιπρόσθετη εγγραφή ψηφοφόρου σε περίπτωση παρουσίασης κάποιου ανθρώπινου ή τεχνικού λάθους ή κάποιας ένστασης.
* **Αποτελεσμάτων**: Είναι το τελικό στάδιο όπου τα αποτελέσματα και ο νικητής των φοιτητικών εκλογών μπορούν να εκδοθούν από οποιοδήποτε χρήστη στο blockchain καλώντας τις αντίστοιχες δύο συναρτήσεις τύπου read χωρίς καμία επιβάρυνση τελών. Επιπλέον ο διαχειριστής συμβολαίου μπορεί να ταξινομήσει τις παρατάξεις ούτως ώστε τα τελικά αποτελέσματα να εμφανίζονται σε φθίνουσα σειρά.

## 4.4 Προβλήματα και Απαιτήσεις Υλοποίησης Εφαρμογής

Η ιδέα για τη δημιουργία μιας τέτοιας εφαρμογής αναλύθηκε παραπάνω, ωστόσο υπάρχουν μερικοί παράγοντες που μπορούν να σταθούν εμπόδιο με αποτέλεσμα να την επηρεάσουν ή ακόμα και να αποτρέψουν την υλοποίηση της. Αυτοί οι παράγοντες μπορεί να είναι ανθρώπινοι, τεχνικοί αλλά και νομοθετικοί. Κάποιοι από τα βασικότερα θέματα και απαιτήσεις είναι:

* **Ο νόμος GDPR (General Data Protection Regulation)** - Ο Γενικός Κανονισμός για την Προστασία των Δεδομένων τέθηκε σε ισχύ το 2018 και επηρεάζει το δικαιώματα σε τρίτους (όπως μπορεί να είναι ένας διαχειριστής μιας εφαρμογής για φοιτητικές εκλογές) να ελέγχει και να διαχειρίζεται διαχείριση προσωπικά δεδομένα φοιτητών. Τα προσωπικά δεδομένα μπορεί να είναι εκείνα που εισάγονται κατά την εγγραφή των χρηστών στο ιστότοπο, όπως το ονοματεπώνυμο, ηλικία, τηλέφωνο, e-mail, κτλ. Για να ρυθμιστεί αυτό το λεπτό ζήτημα θα πρέπει να τεθούν ισχυρές δικλείδες ασφαλείας για την προστασία των προσωπικών δεδομένων στην αποθήκευση off-chain δεδομένων σε ένα κατανεμημένο χώρο αποθήκευσης αρχείων IPFS και επίσης στο ledger του Blockchain δικτύου. Στην περίπτωση που η πανεπιστημιακή κοινότητα δεν το εγκρίνει οι εφαρμογή δεν θα μπορέσει να λειτουργήσει.
* **Οι κανόνες του Πανεπιστημίου** – Στα πανεπιστήμια ισχύουν κανονισμοί, νομοθεσίες και πρωτόκολλα τα οποία μπορούν να αποτρέψουν στην υλοποίηση μια τέτοιας καινοτόμας ιδέας. Το συμβούλιο των καθηγητών και ο πρύτανης είναι καθοριστικοί συντελεστές ούτως ώστε να αλλάξουν την παραδοσιακή ψηφοφορία σε μια τεχνολογικά εξελιγμένη διαδικασία.
* **Οι παρατάξεις και οι φοιτητές** – Εάν δεν συμφωνεί η φοιτητική κοινότητα και κυρίως οι αντιπρόσωποι και τα μέλη των παρατάξεων του πανεπιστημίου η υλοποίηση της εφαρμογής δεν θα γίνει ποτέ.
* **Το διαδίκτυο** – Μπορεί να ακούγεται παράξενο αλλά δεν είναι. Σίγουρα η συντριπτική πλειοψηφία των φοιτητών έχει πρόσβαση στο internet. Παρ’ όλ’ αυτά μια μικρή μειοψηφία η οποία δεν διαθέτει πρόσβαση στο διαδίκτυο μπορεί να αντισταθεί και να δημιουργήσει προβλήματα στο όλο εγχείρημα.
* **Πολυπλοκότητα διαδικασίας** – Η κλασική ψηφοφορία με την κάλπη είναι θεωρητικά η πιο εύκολη και γρήγορη διαδικασία αν αναλογιστούμε ότι δεν πρέπει να κάνουμε κάποια ιδιαίτερη προεργασία πιο πριν. Οπότε ένα μείζον ζήτημα είναι τα μέλη της πανεπιστημιακής κοινότητας να αισθάνονται άνετα χρησιμοποιώντας το ψηφιακό σύστημα ψηφοφορίας. Η διαδικασία μέσω blockchain απαιτεί από όλους τους υποψηφίους και ψηφοφόρους να ακολουθήσουν την διαδικασία εγγραφής στον ιστότοπο αφού πρώτα έχουν δημιουργήσει το δικό τους πορτοφόλι και προμηθευτεί μερικά κρυπτονομίσματα για τη συναλλαγή της ψηφοφορίας. Χωρίς την μοναδική διεύθυνση του πορτοφολιού κάθε χρήστης δεν υπάρχει η δυνατότητα ψηφίσματος. Η δυσκολία δημιουργίας ενός πορτοφολιού από άτομα τα οποία δεν είναι εξοικειωμένα με την τεχνολογία μπορεί να προκαλέσει μια παθογένεια στην εφαρμογή ψηφοφορίας. Τέλος ηδιαδικασία της χρέωσης κατά τη συναλλαγή για την προσθήκης δεδομένων στο ledger μπορεί να δημιουργήσει θέματα στους χρήστες κατά την ψηφοφορία εάν δεν διαθέτουν το επιθυμητό ποσό.
* **Τεχνική διαρροή (technical leakage) –** Υπάρχει πιθανότητα διαρροής ενός κόμβου αποθετηρίου και η εκλογές να τιναχθούν στον αέρα. Ουσιαστικά μπορεί διαρρεύσει ο πηγαίος κώδικας, τα κλειδιά API και τα διαπιστευτήρια με συνέπεια να έχουν πρόσβαση στο smart contract τρίτοι οι οποίοι έχουν σκοπό να σαμποτάρουν την διαδικασία της ψηφοφορίας. Στην προκειμένη περίπτωση θα πρέπει να υπάρχει πλήρη αξιοπιστία των προγραμματιστών της εφαρμογής και του ιδιοκτήτη του έξυπνου συμβολαίου.
* **Τεχνικά σφάλματα εφαρμογής** – Για να μπορεί μια τέτοια εφαρμογή να μπει σε λειτουργία θα πρέπει να δαπανηθεί αρκετός χρόνος για δοκιμές και ελέγχους πράγμα που μπορεί να επηρεάσει αρνητικά την λειτουργία της εφαρμογής στο τέλος. Οπότε είναι απαραίτητο ως πρώτο βήμα μια δοκιμαστική πλατφόρμα, πριν το λανσάρισμα της τελικής έκδοσης, για να επιβεβαιώσει την αξιοπιστία του προγράμματος.
* **Ενοποίηση με Πανεπιστημιακά Συστήματα** - Το σύστημα ψηφοφορίας blockchain πρέπει να ενσωματώνεται απρόσκοπτα με τις υπάρχουσες βάσεις δεδομένων και συστήματα πανεπιστημίων.
* **Οικονομικοί λόγοι** – Ένας τελευταίος και κομβικός παράγοντας είναι να μην υπάρξει καμία χρηματοδότη από το πανεπιστήμιο και έτσι να αποτρέψει την ανάπτυξη ενός διαδικτυακού προγράμματος ψηφοφορίας.

## 4.5 O κώδικας του Smart Contract “Student\_Elections” και το Address του

Το Smart Contract με όνομα “**Student\_Elections**” έχει ανέβει στο *Sepolia* *Testnet* δίκτυο του Ethereum.

Το *Address* του είναι: **0x314731d42122182aCf296dd9300bde36eF97bFa1**

και το *AΒΙ* του είναι:

[{"inputs":[{"internalType":"uint256","name":"stg\_lmt","type":"uint256"}],"stateMutability":"nonpayable","type":"constructor"},{"inputs":[{"internalType":"bytes32","name":"\_bytes32","type":"bytes32"}],"name":"bytes32ToString","outputs":[{"internalType":"string","name":"","type":"string"}],"stateMutability":"pure","type":"function"},{"inputs":[],"name":"chairperson","outputs":[{"internalType":"address","name":"","type":"address"}],"stateMutability":"view","type":"function"},{"inputs":[],"name":"getBytes32ArrayForConstructor","outputs":[{"internalType":"bytes32[4]","name":"b32Arr","type":"bytes32[4]"}],"stateMutability":"pure","type":"function"},{"inputs":[{"internalType":"uint256","name":"","type":"uint256"}],"name":"proposalNames","outputs":[{"internalType":"bytes32","name":"","type":"bytes32"}],"stateMutability":"view","type":"function"},{"inputs":[{"internalType":"uint256","name":"","type":"uint256"}],"name":"proposals","outputs":[{"internalType":"bytes32","name":"name","type":"bytes32"},{"internalType":"uint256","name":"voteCount","type":"uint256"}],"stateMutability":"view","type":"function"},{"inputs":[{"internalType":"address","name":"voter","type":"address"}],"name":"registerVoters","outputs":[],"stateMutability":"nonpayable","type":"function"},{"inputs":[],"name":"showFinalResults","outputs":[{"internalType":"string","name":"results","type":"string"}],"stateMutability":"view","type":"function"},{"inputs":[],"name":"showWinner","outputs":[{"internalType":"string","name":"winner","type":"string"}],"stateMutability":"view","type":"function"},{"inputs":[],"name":"sortProposals","outputs":[],"stateMutability":"nonpayable","type":"function"},{"inputs":[{"internalType":"uint256","name":"\_i","type":"uint256"}],"name":"uintToString","outputs":[{"internalType":"string","name":"\_uintAsString","type":"string"}],"stateMutability":"pure","type":"function"},{"inputs":[{"internalType":"uint8","name":"toProposal","type":"uint8"}],"name":"vote","outputs":[],"stateMutability":"nonpayable","type":"function"},{"inputs":[{"internalType":"address","name":"","type":"address"}],"name":"voters","outputs":[{"internalType":"uint256","name":"weight","type":"uint256"},{"internalType":"bool","name":"voted","type":"bool"},{"internalType":"uint8","name":"vote","type":"uint8"}],"stateMutability":"view","type":"function"}]

Ακόμη το Hashes για την εγκατάσταση του smart contract είναι:

|  |  |  |
| --- | --- | --- |
| **A/A** | **Txhash** | **Method** |
| 1 | 0xfc38b2fcf1b413262b319947f0e65164d3648163b11d05ddaa51d6e0ea325108 | 0x60806040 |
| 2 | 0x5b3ab3fc82460c0a62750dc1182db174afadbe91038bf5562cbdb0a4da5cf269 | Register Voters |
| 3 | 0xc714c411a92f7dd4a1df95aefb774326cdfeb0c886fba715976c383f293a4b85 | Vote |
| 4 | 0x4be06daf7da5e7e0955c493199f77775ac67b18cebdce4a907ad1e5fb6fcd968 | Register Voters |
| 5 | 0x8d3c458455f4108e58b81964ca222063e0ac5ed4725b8948207910bb90e3a3dc | Vote |
| 6 | 0x312ea7f1c62875034afb64518b3650541ba582c9f7fc57a707b636cd7542a127 | Vote |
| 7 | 0x07aa96559a21194fbea5a01f1f560c3308c261d50e4fa1b5f8b0f299030edcd4 | Vote |
| 8 | 0x44b0a249ff50f1728cfea85997dce2f73e4c197ead9b43150d49e897259ebcb0 | Vote |
| 9 | 0x0748a9c94fef46168cbf30364eb8c996022d78c54c2b4d9d503ce0717701514f | Vote |

Ακολουθεί αναλυτικά ο πηγαίος κώδικας του smart contract “**Student\_Elections**” σε γλώσσα προγραμματισμού Solidity:

// SPDX-License-Identifier: GPL-3.0

pragma solidity 0.8.24;

contract Student\_Elections {

    struct Voter {

        uint weight; // weight is accumulated by delegation

        bool voted;  // if true, that person already voted

        uint8 vote;   // index of the voted proposal

    }

    struct Proposal {

        bytes32 name;   // short name (up to 32 bytes)

        uint voteCount; // number of accumulated votes

    }

    // use this modifier in functions to give permission only to chairperson

    modifier onlyOwner() {

        require(msg.sender == chairperson, "Only chairperson has access to this function!");

        \_;

    }

    address public chairperson;

    mapping(address => Voter) public voters;

    Proposal[] public proposals;

    uint startupTime;

    uint256 stageLimit;

    bytes32[] public  proposalNames = getBytes32ArrayForConstructor();

    // chairperson assign the limit of each stage in seconds and initiate the candidate’s array

    constructor(uint256 stg\_lmt) {

        chairperson = msg.sender;

        startupTime = block.timestamp;

        stageLimit = stg\_lmt;

        for (uint i = 0; i < proposalNames.length; i++) {

            proposals.push(Proposal( {

                name: proposalNames[i],

                voteCount: 0

            }));

        }

    }

    // assign all candidates names in this function

    function getBytes32ArrayForConstructor() pure public returns (bytes32[4] memory b32Arr) {

        b32Arr = [

            bytes32("candidate1"),

            bytes32("candidate2"),

            bytes32("candidate3"),

            bytes32("candidate4")

        ];

    }

    // only chairperson is able to register voters to elections

    function registerVoters(address voter) public onlyOwner{

        if (stageLimit != 0)

            require(block.timestamp <= (startupTime + 2\*stageLimit), "The stage of Registration by chairperson has been finished.");

        require(chairperson != voter, "Chairperson cannot register to the elections.");

        require(voters[voter].weight == 0, "The voter has already registered.");

        voters[voter].weight = 1;

        voters[voter].voted = false;

    }

    // only registered Voters are able to cast a single vote to proposal $(toProposal)

    function vote(uint8 toProposal) public {

        if (stageLimit != 0) {

            require(block.timestamp > (startupTime + stageLimit) && block.timestamp <= (startupTime + 2\*stageLimit), "The stage of voting is not active.");

        }

        Voter storage sender = voters[msg.sender];

        require(msg.sender != chairperson, "Chairperson do not have the right to vote.");

        require(sender.weight != 0, "The voter do not have the right to vote.");

        require(sender.voted == false, "The voter has already voted.");

        require(toProposal < proposals.length, "There is no proposal with that index.");

        sender.voted = true;

        sender.vote = toProposal;

        proposals[toProposal].voteCount += sender.weight;

    }

    // everyone can display the Winner(s) of Elections

    function showWinner() public view returns (string memory winner) {

        require(block.timestamp > (startupTime + 2\*stageLimit), "The elections have not finished yet.");

        uint256 winnnerVotes = 0;

        for (uint8 p = 0; p < proposals.length; p++) {

            if (proposals[p].voteCount > winnnerVotes) {

                winnnerVotes = proposals[p].voteCount;

            }

        }

        for (uint8 p = 0; p < proposals.length; p++) {

            if (winnnerVotes == proposals[p].voteCount) {

                if (keccak256(bytes(winner)) == keccak256(bytes("")))

                    winner = string.concat("=> name: ", bytes32ToString(proposals[p].name), "  votes: ", uintToString(proposals[p].voteCount));

                else

                    winner = string.concat(winner, " , name: ", bytes32ToString(proposals[p].name), "  votes: ", uintToString(proposals[p].voteCount));

            }

        }

    }

    // everyone can display the Final Results of Elections

    function showFinalResults() public view returns (string memory results) {

        require(block.timestamp > (startupTime + 2\*stageLimit), "The elections have not finished yet.");

        for (uint8 i = 0; i < proposals.length; i++) {

            if (i == 0)

                results = string.concat("Election Results => ", uintToString(i+1), ".", bytes32ToString(proposals[i].name), "  votes: ", uintToString(proposals[i].voteCount));

            else

                results = string.concat(results, " , ", uintToString(i+1), ".", bytes32ToString(proposals[i].name), "  votes: ", uintToString(proposals[i].voteCount));

        }

        return results;

    }

    // FOLLOWING AUXILIARY FUNCTIONS

    // only chairperson can sort Proposals by votes when elections have been finished

    function sortProposals() public onlyOwner{

        require(block.timestamp > (startupTime + 2\*stageLimit), "The elections have not finished yet.");

        Proposal memory temp;

        bool swapped;

        for(uint8 i = 0 ; i < proposals.length-1; i++){

            swapped = false;

            for(uint8 j = 0 ; j < (proposals.length-i-1); j++){

                if(proposals[j].voteCount < proposals[j+1].voteCount){

                        //swap elements

                        temp = proposals[j];

                        proposals[j] = proposals[j+1];

                        proposals[j+1] = temp;

                        swapped = true;

                }

            }

            if (!swapped) break;

        }

    }

    // convert a bytes32 to string

    function bytes32ToString(bytes32 \_bytes32) public pure returns (string memory) {

        uint8 i = 0;

        while(i < 32 && \_bytes32[i] != 0) {

            i++;

        }

        bytes memory bytesArray = new bytes(i);

        for (i = 0; i < 32 && \_bytes32[i] != 0; i++) {

            bytesArray[i] = \_bytes32[i];

        }

        return string(bytesArray);

    }

    // convert a uint to string

    function uintToString(uint \_i) public pure returns (string memory \_uintAsString) {

        if (\_i == 0) {

            return "0";

        }

        uint j = \_i;

        uint len;

        while (j != 0) {

            len++;

            j /= 10;

        }

        bytes memory bstr = new bytes(len);

        uint k = len;

        while (\_i != 0) {

            k = k-1;

            uint8 temp = (48 + uint8(\_i - \_i / 10 \* 10));

            bytes1 b1 = bytes1(temp);

            bstr[k] = b1;

            \_i /= 10;

        }

        return string(bstr);

    }

}

# ΠΑΡΑΠΟΜΠΕΣ

1. Zaghloul, E.; Li, T.; Ren, J. *d*-BAME: Distributed Blockchain-Based Anonymous Mobile Electronic Voting. *IEEE Internet Things J.* **2021**, *8*, 16585–16597. [CrossRef]
2. Nakamoto, S. Bitcoin: A Peer-To-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 5 June 2022).
3. Park, S.; Specter, M.; Narula, N.; Rivest, R.L. Going from bad to worse: From Internet voting to blockchain voting. *J. Cybersecur.* **2021**, *7*, tyaa025. [CrossRef]
4. Hu, W.; Li, H. A blockchain-based secure transaction model for distributed energy in Industrial Internet of Things. *Alex. Eng. J.* **2020**, *60*, 491–500.
5. Dinh, T.N.; Thai, M.T. AI and Blockchain: A Disruptive Integration. *IEEE Comput.* **2018**, *51*, 48–53. [CrossRef]
6. Eyal, I. Blockchain Technology: Transforming Libertarian Cryptocurrency Dreams to Finance and Banking Realities. *IEEE Comput.* **2017**, *50*, 38–49. [CrossRef]
7. Febriyanto, E.; Triyono; Rahayu, N.; Pangaribuan, K.; Sunarya, P.A. Using Blockchain Data Security Management for E-Voting Systems. In Proceedings of the 2020 8th International Conference on Cyber and IT Service Management (CITSM), Pangkal, Indonesia, 23–24 October 2020; pp. 1–4.
8. Shabnam, S.; Sayyad, F. Voting Using Blockchain Technology. In *Intelligent Computing and Networking*; Springer: Singapore, 2021; pp. 285–291.
9. Mustafa, M.K.; Waheed, S. An E-Voting Framework with Enterprise Blockchain. In *Advances in Distributed Computing and Machine Learning*; Springer: Singapore, 2021; pp. 135–145.
10. Xie, J.; Yu, F.R.; Huang, T.; Xie, R.; Liu, J.; Liu, Y. A Survey on the Scalability of Blockchain Systems. *IEEE Netw.* **2019**, *33*, 166–173. [CrossRef]
11. Khalfan, M.; Azizi, N.; Haass, O.; Maqsood, T.; Ahmed, I. Blockchain Technology: Potential Applications for Public Sector E-Procurement and Project Management. *Sustainability* **2022**, *14*, 5791. [CrossRef]
12. Singh, L.; Kumar, A.; Singh, Y. Internet of Healthcare Things (IoHT) and Blockchain: An Efficient Integration for Smart Health Care Systems. In *Healthcare and Knowledge Management for Society 5.0*; CRC Press: Boca Raton, FL, USA, 2021; pp. 135–149. [CrossRef]
13. Rejeb, A.; Rejeb, K.; Simske, S.J.; Keogh, J.G. Blockchain technology in the smart city: A bibliometric review. *Qual. Quant.* **2022**, *56*, 2875–2906. [CrossRef]
14. Maesa, D.D.F.; Mori, P. Blockchain 3.0 applications survey. *J. Parallel Distrib. Comput.* **2020**, *138*, 99–114. [CrossRef]
15. Abayomi-Zannu, T.P.; Odun-Ayo, I.; Tatama, B.F.; Misra, S. Implementing a Mobile Voting System Utilizing Blockchain Technology and Two-Factor Authentication in Nigeria. In *Proceedings of the First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*; Springer: Singapore, 2020; pp. 857–872. [CrossRef]
16. Alvi, S.T.; Uddin, M.N.; Islam, L.; Ahamed, S. From Conventional Voting to Blockchain Voting: Categorization of Different Voting Mechanisms. In Proceedings of the 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka,
17. Bangladesh, 19–20 December 2020; pp. 1–6. [CrossRef]

Rubtcova, M.; Pavenkov, O. Implementation of Blockchain technology in electronic election in Sierra Leone. In Proceedings of the 2018 Conference: “Re-Thinking Regions in Global International Relations”, Davao City, Philippines, 23–24 March 2018; Volume 23.

1. Pawlak, M.; Poniszewska-Mara´nda, A. Trends in blockchain-based electronic voting systems. *Inf. Process. Manag.* **2021**, *58*, 102595. [CrossRef]
2. Khan, K.M.; Arshad, J.; Khan, M.M. Empirical analysis of transaction malleability within blockchain-based e-Voting. *Comput. Secur.* **2020**, *100*, 102081. [CrossRef]
3. Berdik, D.; Otoum, S.; Schmidt, N.; Porter, D.; Jararweh, Y. A Survey on Blockchain for Information Systems Management and Security. *Inf. Process. Manag.* **2020**, *58*, 102397. [CrossRef]
4. Hassan, N.U.; Yuen, C.; Niyato, D. Blockchain Technologies for Smart Energy Systems: Fundamentals, Challenges, and Solutions. *IEEE Ind. Electron. Mag.* **2019**, *13*, 106–118. [CrossRef]