

ΤΕΧΝΟΛΟΓΙΕΣ BLOCKCHAIN ΚΑΙ ΑΠΟΚΕΝΤΡΩΜΕΝΕΣ ΕΦΑΡΜΟΓΕΣ

2^η ΕΡΓΑΣΙΑ - Crowdfunding dApp

Ανδρέας Χατζηαντώνης - ics21138

YouTube : https://www.youtube.com/watch?v=BoZRT5_RpXg

Web Page: <https://ahadjiantonis-dev.github.io/CrowdFunding-dApp/>

**Contract Address on Sepolia
0x4DF29b10f3f2a18C470e2CB93FBCd6E4Bedf814b**

Επιτελική σύνοψη - Backend

Η παρούσα εργασία αποτελεί την υλοποίηση μιας εφαρμογής crowdfunding (πληθοχρηματοδότησης) τύπου <https://www.kickstarter.com/> η οποία έχει βασιστεί στην τεχνολογία Blockchain στο δίκτυο Ethereum και για την λειτουργία της υλοποιήθηκε ένα έξυπνο συμβόλαιο σε Solidity.

Για την υλοποίηση του συμβόλαιού χρησιμοποιήθηκαν οι εξής λειτουργίες.

Δημιουργία καμπάνιας. Εγγραφή από επιχειρηματία (entrepreneur) πληρώνοντας κάποιο αντίτυπο στο συμβόλαιο και δηλώνοντας τις πληροφορίες της (ένα όνομα, το κόστος μετοχής και το πλήθος των μετοχών που απαιτούνται για την ολοκλήρωσή της). Κάθε εκστρατεία αποκτά έναν μοναδικό αναγνωριστικό αριθμό. Μπορεί οποιοσδήποτε (εκτός του ιδιοκτήτη του συμβόλαιού) να εκτελέσει την ενέργεια αυτή ακόμη και επανειλημμένα με την προϋπόθεση πως δεν είναι στην λίστα αποκλεισμένων (banned list). Όποτε την εκτελεί, πρέπει να αποστέλλει στο συμβόλαιο την αξία του τέλους καμπάνιας όπως αυτή έχει καθοριστεί σφαιρικά, ίδια για όλες τις καμπάνιες (π.χ. 0,02 Ethers).

Χρηματοδότηση καμπάνιας. Αγορά μετοχών από επενδυτές (backers) και διατήρηση των χρημάτων στο συμβόλαιο έως ότου ολοκληρωθεί η εκστρατεία. Μπορεί οποιοσδήποτε να εκτελέσει την ενέργεια αυτή, ακόμη και επανειλημμένα υπό την προϋπόθεση πως η καμπάνια δεν έχει ακόμη ολοκληρωθεί ή ακυρωθεί. Όποτε την εκτελεί, πρέπει να πληρώνει στο συμβόλαιο την αξία των μετοχών, το πλήθος των οποίων δηλώνει πως θέλει να αγοράσει.

Ακύρωση καμπάνιας. Να επιτρέπει την επιστροφή των χρημάτων που φυλάσσονται στο συμβόλαιο για μια συγκεκριμένη εκστρατεία στους επενδυτές. Χαρακτηρισμός της εκστρατείας ως άκυρης. Την ενέργεια αυτή μπορεί να την εκτελέσει μόνο ο επιχειρηματίας που την εκκίνησε ή ο ιδιοκτήτης του συμβόλαιού υπό την προϋπόθεση πως η καμπάνια δεν έχει ακόμη ολοκληρωθεί ή ακυρωθεί.

Αποζημίωση επενδυτή. Επιστροφή των χρημάτων που έχουν επενδυθεί από συγκεκριμένο επενδυτή σε ακυρωμένες εκστρατείες. Μπορεί οποιοσδήποτε να καλέσει την ενέργεια αυτή. Αν υπάρχουν χρήματα τα οποία έχει επενδύσει σε μία ή περισσότερες καμπάνιες που έχουν ακυρωθεί, τα χρήματα αυτά του επιστρέφονται. Οι συγκεκριμένες καμπάνιες δεν του χρωστάνε πλέον επιστροφή.

Ολοκλήρωση καμπάνιας. Μεταφορά του 80% των χρημάτων που φυλάσσονται στο συμβόλαιο για την συγκεκριμένη εκστρατεία προς τον επιχειρηματία που την εκκίνησε. Χαρακτηρισμός της ως ολοκληρωμένης. Την ενέργεια αυτή μπορεί να την εκτελέσει μόνο ο επιχειρηματίας που την εκκίνησε ή ο ιδιοκτήτης του συμβόλαιού υπό την προϋπόθεση πως δεν είναι ακυρωμένη και η καμπάνια έχει συμπληρώσει το πλήθος των μετοχών που δηλώθηκε αρχικά (ότι την ολοκληρώνουν).

Κώδικας και Συμπεριφορά του Συμβολαίου

Βασικές Μεταβλητές

- **address public Owner**
 - Αποθηκεύει τη διεύθυνση του ιδιοκτήτη του συμβολαίου.
- **address public specialOwner**
 - Ειδική διεύθυνση που φέρει δικαιώματα ιδιοκτήτη αλλά και απλού χρήστη.
- **uint public campaignFee**
 - Το σταθερό κόστος σε Ether που πρέπει να πληρώσει κάθε νέος επιχειρηματίας για να δημιουργήσει καμπάνια (π.χ. 0,02 ETH).
- **uint public reservationFeePercentage**
 - Το ποσοστό που παρακρατείται από κάθε ολοκληρωμένη καμπάνια υπέρ του ιδιοκτήτη (π.χ. 20%).
- **uint public campaignCount**
 - Μετρητής του πλήθους των συνολικών καμπανιών που έχουν δημιουργηθεί (ανεξαρτήτως αν ακυρώθηκαν ή ολοκληρώθηκαν).
- **uint public totalFeesAccumulated**
 - Το σύνολο των τελών που έχουν εισρεύσει στο συμβόλαιο (π.χ. από δημιουργία καμπάνιας ή τέλη ολοκλήρωσης).
- **bool public destroyed**
 - Δήλωση κατάστασης του συμβολαίου (αν έχει καταστραφεί ή όχι).

Δομή (Struct) δεδομένων καμπάνιας

- **campaignId:** Μοναδικός αναγνωριστικός αριθμός.
- **entrepreneur:** Διεύθυνση επιχειρηματία που δημιούργησε την καμπάνια.
- **title:** Τίτλος/Όνομα της καμπάνιας.
- **pledgeCost:** Κόστος ανά μετοχή που ορίζει ο επιχειρηματίας.
- **pledgesNeeded:** Συνολικός αριθμός μετοχών που πρέπει να αγοραστούν για να ολοκληρωθεί η καμπάνια.
- **pledgesCount:** Τρέχων αριθμός αγορασμένων μετοχών.
- **fulfilled:** Κατάσταση ολοκλήρωσης (true/false).
- **withdraw:** Φανερώνει αν έχουν αποσυρθεί χρήματα από την καμπάνια (true/false).
- **cancelled:** Κατάσταση ακύρωσης (true/false).
- **backers:** Λίστα διευθύνσεων όσων έχουν αγοράσει μετοχές.
- **backersPledges:** Mapping από κάθε επενδυτή στο πλήθος των μετοχών που κατέχει.

Επιπλέον Δομές Δεδομένων

- **mapping(uint => Campaign) public campaigns**
Αποθηκεύει όλα τα αντικείμενα καμπάνιας ανά μοναδικό campaignId.
- **mapping(string => bool) public titles**
Αποθηκεύει όλους τους τίτλους από καμπάνιες που έχουν δημιουργηθεί.
- **mapping(address => bool) public bannedEntrepreneurs**
Καταχωρίζει τις αποκλεισμένες διευθύνσεις.
- **address[] public bannedEntrepreneursList**
Λίστα με όλες τις αποκλεισμένες διευθύνσεις.

Συμβάντα (Events)

Για κάθε κύρια λειτουργία, ορίζεται ένα αντίστοιχο συμβάν (event) που ενεργοποιείται όταν ολοκληρωθεί η εκάστοτε συναρτησιακή κλήση:

- **CampaignCreated**
Ενεργοποιείται όταν δημιουργείται μια καινούργια καμπάνια.
- **CampaignFunded**
Ενεργοποιείται με κάθε αγορά μετοχής από έναν επενδυτή.
- **CampaignCancelled**
Ενεργοποιείται όταν μια καμπάνια ακυρώνεται.
- **BackerCompensated**
Ενεργοποιείται κατά την επιστροφή κεφαλαίων σε επενδυτές ακυρωμένων καμπανιών.
- **CampaignFulfilled**
Ενεργοποιείται όταν μια καμπάνια ολοκληρώνεται επιτυχώς.
- **ContractDestroyed**
Ενεργοποιείται όταν ο ιδιοκτήτης καταστρέφει το συμβόλαιο.

Τα events αυτά διευκολύνουν την παρακολούθηση των αλλαγών στο UI (frontend) σε πραγματικό χρόνο και την ενημέρωση των χρηστών χωρίς να χρειάζεται η χειροκίνητη ενημέρωση.

Θα μπορούσαν να υπάρχουν περισσότερα συμβάντα όμως για τον σκοπό υλοποίησης της εργασίας σε μία πιο απλούστερη μορφή επιλέχθηκαν μόνο τα προαναφερθέντα.

Modifiers

Προτού εκτελεστούν διάφορες συναρτήσεις, χρησιμοποιούνται **modifiers** για έλεγχο προ-συνθηκών:

Σχετικά με τον Ιδιοκτήτη

- **onlyOwner**: Επιτρέπει την κλήση συνάρτησης **μόνο** από τον ιδιοκτήτη του συμβολαίου.
- **notOwner**: Αποκλείει τον ιδιοκτήτη από την κλήση (π.χ. για δημιουργία καμπάνιας).
- **validNewOwner**: Βεβαιώνει πως η νέα διεύθυνση ιδιοκτήτη δεν είναι μηδενική.

Σχετικά με τον Επιχειρηματία

- **notBanned**: Αποκλείει από την εκτέλεση συνάρτησης τις διευθύνσεις που είναι στη λίστα banned.
- **alreadyBanned**: Εμποδίζει το διπλό ban σε περίπτωση που η διεύθυνση είναι ήδη αποκλεισμένη.

Σχετικά με την Καμπάνια

- **validCampaignFee**: Ελέγχει αν το ποσό που αποστέλλεται στη `createCampaign` είναι ίσο με το `campaignFee`.
- **campaignExists**: Επιβεβαιώνει ότι η καμπάνια υπάρχει στο mapping.
- **uniqueTitle**: Ελέγχει ότι ο τίτλος της καμπάνιας δεν έχει ήδη καταχωρηθεί.
- **onlyIfCancelled**: Επιτρέπει την κλήση μόνο αν η καμπάνια είναι ακυρωμένη.
- **notCancelled**: Αποκλείει την εκτέλεση αν η καμπάνια έχει ακυρωθεί.
- **notFulfilled**: Αποκλείει την εκτέλεση αν η καμπάνια έχει ήδη ολοκληρωθεί.
- **fullyFunded**: Επιτρέπει την κλήση μόνο αν έχει συγκεντρωθεί ο απαιτούμενος αριθμός μετοχών.
- **notAlreadyCompleted**: Αποτρέπει ολοκλήρωση μιας μη ολοκληρωμένης καμπάνιας.
- **hasRefundablePledges**: Ελέγχει ότι ο επενδυτής έχει μετοχές σε ακυρωμένες καμπάνιες που μπορούν να αποζημιωθούν.

Σχετικά με τους Επενδυτές

- **validPayment**: Ελέγχει ότι το ποσό που πληρώνεται (`msg.value`) ταιριάζει ακριβώς με το `pledgeCost * numShares`.

Σχετικά με τον Επιχειρηματία ή Επενδυτή

- **onlyAuthorized**: Επιτρέπει την κλήση συνάρτησης μόνο στον ιδιοκτήτη του συμβολαίου ή στον επιχειρηματία της συγκεκριμένης καμπάνιας.

Σχετικά με το Συμβόλαιο

- **notDestroyed**: Αποτρέπει την κλήση συναρτήσεων αν το συμβόλαιο είναι ήδη κατεστραμμένο.

Κύριες Συναρτήσεις

1. Διαχείριση του Συμβολαίου

- **setDamaged()**
Θέτει το συμβόλαιο σε κατάσταση destroyed = true, απαγορεύοντας περαιτέρω αλληλεπιδράσεις.
- **getContractBalance()**
Επιστρέφει το υπόλοιπο (ETH) του συμβολαίου.
- **changeOwner(address newOwner)**
Αλλάζει τον ιδιοκτήτη σε μια νέα διεύθυνση, αν δεν είναι μηδενική (για σκοπούς αποφυγής τυχαίου κλικ) (validNewOwner).
- **banEntrepreneur(address entrepreneur)**
Προσθέτει έναν επιχειρηματία στη λίστα αποκλεισμένων, αν δεν είναι ήδη.
- **getBannedTraders()**
Επιστρέφει τη λίστα όλων των αποκλεισμένων διευθύνσεων.
- **getAccumulatedFees()**
Επιστρέφει τα συνολικά τέλη που έχουν συγκεντρωθεί από τη δημιουργία καμπανιών.

2. Λειτουργίες Crowdfunding

- **createCampaign(string memory title, uint pledgeCost, uint pledgesNeeded)**
Δημιουργεί μια καινούργια καμπάνια, έπειτα από πληρωμή του campaignFee.
- **fundCampaign(uint campaignId, uint numPledges)**
Επιτρέπει την αγορά μετοχών σε συγκεκριμένη καμπάνια, προσθέτοντας κεφάλαια στο συμβόλαιο.
- **cancelCampaign(uint campaignId)**
Επιτρέπει στον επιχειρηματία ή τον ιδιοκτήτη να ακυρώσει την καμπάνια, εφόσον δεν έχει ολοκληρωθεί.
- **compensateBacker()**
Επιστρέφει τα χρήματα που έχουν επενδυθεί σε ακυρωμένες καμπάνιες, μηδενίζοντας τα μερίδια του χρήστη.
- **completeCampaign(uint campaignId)**
Ολοκληρώνει μια καμπάνια, μεταφέροντας το 80% των κεφαλαίων στον επιχειρηματία και σημειώνει ότι μπορεί να αποσυρθεί το υπόλοιπο 20% από τον ιδιοκτήτη.

3. Ερωτήματα Πληροφόρησης

- **getActiveCampaigns()**
Επιστρέφει λίστες με τις ενεργές καμπάνιες (ID, τίτλο, κόστος μετοχής κ.λπ.).
- **getCompletedCampaigns()**
Επιστρέφει λίστες με τις ολοκληρωμένες καμπάνιες.
- **getCancelledCampaigns()**
Επιστρέφει λίστες με τις ακυρωμένες καμπάνιες.
- **getBackers(uint campaignId)**
Επιστρέφει τις διευθύνσεις και τα μερίδιά τους για μια συγκεκριμένη καμπάνια.
- **getBackerPledges(uint campaignId, address backer)**
Επιστρέφει το πλήθος μετοχών που κατέχει ο backer σε μια συγκεκριμένη καμπάνια.

4. Διαχείριση Τελών

- **getTotalFees()**
Υπολογίζει τα συνολικά τέλη από ολοκληρωμένες καμπάνιες και τα τέλη δημιουργίας.
- **withdrawFees()**
Επιτρέπει στον ιδιοκτήτη να αποσύρει τα συγκεντρωμένα τέλη (το 20% από τις ολοκληρωμένες καμπάνιες + τα campaignFees).

Ενδελεχής Ανάλυση Κύριων Συναρτήσεων

Δημιουργία καμπάνιας – CreateCampaign

Η συνάρτηση έχει στόχο τη δημιουργία μιας καμπάνιας από οποιονδήποτε επιχειρηματία, εξαιρουμένου του ιδιοκτήτη του συμβολαίου, καθώς και των αποκλεισμένων χρηστών. Η συνάρτηση δέχεται ως ορίσματα τον τίτλο της καμπάνιας, το κόστος ανά μετοχή, καθώς και τον συνολικό αριθμό μετοχών που απαιτούνται για τη χρηματοδότηση.

Η μεταβλητή `title` έχει δηλωθεί ως τύπου `memory`, καθώς τα δεδομένα της χρειάζεται να αποθηκευτούν προσωρινά μόνο κατά τη διάρκεια της εκτέλεσης της συνάρτησης. Στη συνέχεια, τα δεδομένα αυτά αποθηκεύονται μόνιμα σε μια άλλη θέση, επομένως δεν απαιτείται να διατηρηθούν καθολικά στη μνήμη του blockchain. Οι μεταβλητές τύπου `uint` δεν χρειάζονται ειδική δήλωση αποθήκευσης, καθώς χειρίζονται απευθείας ως τιμές.

Η συνάρτηση είναι δηλωμένη ως `external` και `payable`. Το χαρακτηριστικό `external` υποδηλώνει ότι η συνάρτηση καλείται αποκλειστικά από εξωτερικούς χρήστες και όχι από άλλες συναρτήσεις του συμβολαίου. Το χαρακτηριστικό `payable` χρησιμοποιείται επειδή η δημιουργία της καμπάνιας απαιτεί την καταβολή ενός προκαθορισμένου τέλους (fee), το οποίο προστίθεται στο υπόλοιπο του συμβολαίου.

Η μεταβλητή `totalFeesAccumulated` αποτελεί έναν αθροιστικό μετρητή που διατηρεί το σύνολο των τελών που έχουν συλλεχθεί μέσω των καλεσμάτων της συνάρτησης. Παράλληλα, η μεταβλητή `campaignCount` λειτουργεί ως μοναδικός αριθμητικός αναγνωριστικός δείκτης (ID) για κάθε καμπάνια, αυξάνοντας κατά ένα σε κάθε νέα δημιουργία καμπάνιας.

Η μεταβλητή `newCampaign`, τύπου `Campaign`, αποτελεί έναν δείκτη που συνδέεται με τη θέση αποθήκευσης της τρέχουσας καμπάνιας στο mapping `campaigns`, βασισμένη στον μοναδικό δείκτη `campaignId`. Αυτό επιτρέπει την άμεση αποθήκευση των δεδομένων της καμπάνιας χωρίς την ανάγκη περιττής αντιγραφής δεδομένων μέσω της επίδειξης του storage. Στη μεταβλητή `newCampaign` αποθηκεύονται τα στοιχεία της καμπάνιας, όπως ο τίτλος, το κόστος ανά μετοχή (στην κλίμακα του Ether) και ο απαιτούμενος αριθμός μετοχών. Παρά το γεγονός ότι οι τιμές ορίζονται στην κλίμακα του Ether, υποστηρίζονται μικρότερες τιμές μέσω δεκαδικών.

Τέλος, με την ολοκλήρωση της διαδικασίας, ενεργοποιείται το συμβάν `CampaignCreated`, ενημερώνοντας έτσι το σύστημα και τους χρήστες για τη δημιουργία της νέας καμπάνιας.

Αυτοαξιολόγηση :

Η συνάρτηση `createCampaign` καλύπτει το μεγαλύτερο μέρος του ζητήματος, καθώς επιτρέπει τη δημιουργία καμπανιών από εξουσιοδοτημένους χρήστες με τους βασικούς κανόνες και περιορισμούς. Δεν υπάρχουν έλεγχοι για την εγκυρότητα των δεδομένων εισόδου, όπως το αν ο τίτλος (`title`) είναι κενός ή αν οι τιμές του κόστους ανά μετοχή (`pledgeCost`) και του αριθμού μετοχών (`pledgesNeeded`) είναι μη ρεαλιστικές. Η χρήση του `campaignCount` ως μοναδικό ID καλύπτει τις βασικές ανάγκες, αλλά δεν έχει δοκιμαστεί σε εξαιρετικά σενάρια, όπως η επαναχρησιμοποίηση ID μετά από ακύρωση καμπάνιας. Γενικά, η συνάρτηση καλύπτει το **90% του ζητήματος**, αλλά θα μπορούσε να βελτιωθεί με την προσθήκη ελέγχων στα δεδομένα εισόδου, τη διασφάλιση της σταθερότητας της μεταβλητής `totalFeesAccumulated` και την καλύτερη πληροφόρηση του χρήστη σχετικά με τις απαιτήσεις της λειτουργίας.

Χρηματοδότηση καμπάνιας – fundCampaign

Η συνάρτηση **fundCampaign** έχει ως στόχο να επιτρέπει σε εξωτερικούς χρήστες να χρηματοδοτήσουν μια καμπάνια αγοράζοντας μετοχές σε αυτήν. Η συνάρτηση λαμβάνει ως ορίσματα το αναγνωριστικό της καμπάνιας (`campaignId`) και τον αριθμό των μετοχών (`numShares`) που επιθυμεί να αγοράσει ο χρήστης. Είναι δηλωμένη ως `external` και `payable`, καθώς καλείται από εξωτερικούς χρήστες και απαιτεί την αποστολή πληρωμής σε Ether ίση με το κόστος των μετοχών. Η συνάρτηση περιλαμβάνει μια σειρά από ελέγχους μέσω modifiers για να διασφαλίσει ότι:

- Το συμβόλαιο δεν βρίσκεται σε κατάσταση βλάβης (`notDestroyed`).
- Η καμπάνια που επιλέχθηκε δεν έχει ακυρωθεί (`notCancelled`).
- Η καμπάνια δεν έχει ολοκληρωθεί ήδη (`notFulfilled`).
- Η πληρωμή που συνοδεύει τη συναλλαγή είναι ακριβής, δηλαδή αντιστοιχεί στο κόστος των μετοχών που επιθυμεί να αγοράσει ο χρήστης (`validPayment`).

Αφού περάσει τους ελέγχους, η συνάρτηση αποκτά πρόσβαση στη συγκεκριμένη καμπάνια μέσω του mapping `campaigns` και προσθέτει τον χρήστη στη λίστα των επενδυτών της καμπάνιας εάν δεν είναι ήδη καταχωρισμένος. Στη συνέχεια, ενημερώνει τον αριθμό μετοχών που κατέχει ο χρήστης στην καμπάνια (`backerPledges`) και αυξάνει τον συνολικό αριθμό μετοχών που έχουν πωληθεί για την καμπάνια (`pledgesCount`).

Τέλος, ενεργοποιείται το συμβάν **PledgeMade**, ενημερώνοντας το σύστημα και τους χρήστες για την επιτυχή αγορά μετοχών στην καμπάνια. Η συνάρτηση διασφαλίζει αποδοτική αποθήκευση και διαχείριση δεδομένων, ενώ καλύπτει τα βασικά σενάρια χρηματοδότησης με τους απαραίτητους ελέγχους.

Η συνάρτηση **fundCampaign** καλύπτει επαρκώς τη διαδικασία χρηματοδότησης καμπανιών, διασφαλίζοντας ότι οι επενδυτές μπορούν να αγοράζουν μετοχές με ορθούς ελέγχους και σωστή διαχείριση των δεδομένων. Παρέχει πλήρη προστασία μέσω modifiers, αποτρέποντας τη χρηματοδότηση ακυρωμένων ή ολοκληρωμένων καμπανιών, ενώ επαληθεύει ότι η πληρωμή που συνοδεύει τη συναλλαγή είναι ακριβής. Η χρήση του storage για τη διαχείριση της καμπάνιας εξασφαλίζει αποδοτική ενημέρωση των δεδομένων χωρίς περιττή αντιγραφή. Παράλληλα, η ενημέρωση της λίστας επενδυτών και των μετοχών τους (`backerPledges`) είναι άμεση και ορθολογική. Ωστόσο, η συνάρτηση παρουσιάζει ορισμένες ελλείψεις. Δεν υπάρχουν έλεγχοι για την αποφυγή υπερχείλισης στον υπολογισμό της συνολικής αξίας των μετοχών (`pledgeCost * numShares`), κάτι που θα μπορούσε να προκαλέσει προβλήματα σε σπάνιες περιπτώσεις όμως στο δικό μας σενάριο αυτό είναι επιτρεπτό. Επιπλέον, δεν γίνεται επαλήθευση για το αν η τιμή `numShares` είναι μηδενική ή μη έγκυρη, γεγονός που θα μπορούσε να οδηγήσει σε άσκοπα αιτήματα, όμως λόγω των ελέγχων στο ψηφιακό πορτοφόλι καθώς δεν υπάρχουν αρνητικές τιμές ETHs αυτό καλύπτεται. Τέλος, η συνάρτηση δεν λαμβάνει υπόψη περιπτώσεις όπου η καμπάνια έχει ήδη φτάσει το μέγιστο αριθμό μετοχών που απαιτούνται. Συνολικά, η συνάρτηση καλύπτει το **95% του ζητήματος**, αλλά θα μπορούσε να βελτιωθεί με πρόσθετους ελέγχους εισόδου και μηχανισμούς προστασίας για εξαιρετικές περιπτώσεις.

Ακύρωση Καμπάνιας – cancelCampaign

Η συνάρτηση **cancelCampaign** έχει ως στόχο να επιτρέπει την ακύρωση μιας καμπάνιας από εξουσιοδοτημένα μέλη, διασφαλίζοντας ότι η καμπάνια πληροί συγκεκριμένα κριτήρια πριν ακυρωθεί. Λαμβάνει ως όρισμα το μοναδικό αναγνωριστικό της καμπάνιας (campaignId) και ελέγχει μέσω modifiers ότι:

- Ο καλών είναι εξουσιοδοτημένος να ακυρώσει την καμπάνια, δηλαδή είτε ο ιδιοκτήτης του συμβολαίου είτε ο επιχειρηματίας που τη δημιούργησε (onlyAuthorized).
- Η καμπάνια δεν έχει ολοκληρωθεί ήδη (notFulfilled).
- Η καμπάνια δεν έχει ακυρωθεί προηγουμένως (notCancelled).

Αφού περάσει τους ελέγχους, η συνάρτηση αποκτά πρόσβαση στη συγκεκριμένη καμπάνια μέσω του mapping campaigns και ορίζει τη μεταβλητή cancelled σε true, σηματοδοτώντας ότι η καμπάνια έχει ακυρωθεί. Τέλος, ενεργοποιεί το συμβάν **CampaignCancelled**, ενημερώνοντας το σύστημα και τους χρήστες για την ακύρωση της καμπάνιας.

Η συνάρτηση διασφαλίζει ότι μόνο εξουσιοδοτημένα μέρη μπορούν να ακυρώσουν μια καμπάνια και ότι αυτή η ακύρωση πραγματοποιείται με ασφαλή και αποδοτικό τρόπο, αποτρέποντας αχρείαστες ή μη έγκυρες τροποποιήσεις.

Αυτοαξιολόγηση:

Η συνάρτηση **cancelCampaign** καλύπτει επαρκώς τη διαδικασία ακύρωσης μιας καμπάνιας, διασφαλίζοντας μέσω modifiers ότι μόνο εξουσιοδοτημένα μέλη (ο επιχειρηματίας ή ο ιδιοκτήτης του συμβολαίου) μπορούν να την εκτελέσουν. Παράλληλα, αποτρέπει την ακύρωση καμπανιών που έχουν ήδη ολοκληρωθεί ή ακυρωθεί, προστατεύοντας από αχρείαστες ή λανθασμένες τροποποιήσεις. Η χρήση του **storage** για την τροποποίηση της κατάστασης της καμπάνιας (**cancelled**) καθιστά τη διαδικασία αποδοτική και άμεση, ενώ το συμβάν **CampaignCancelled** ενημερώνει με διαφάνεια το σύστημα για την ακύρωση. Δεν ελέγχει εάν η ακύρωση μπορεί να προκαλέσει παρενέργειες, όπως η απώλεια κεφαλαίων από επενδυτές που έχουν ήδη δεσμεύσει μετοχές όμως αυτές οι μετοχές δεν χάνονται καθώς οι λειτουργίες αυτές επικαλύπτονται σε επόμενη συνάρτηση. Συνολικά, η συνάρτηση καλύπτει το **100% του ζητήματος**.

Αποζημίωση Επενδυτή – compensateBacker

Η συνάρτηση **compensateBacker** επιτρέπει σε έναν επενδυτή να λάβει επιστροφή χρημάτων για τις μετοχές που έχει επενδύσει σε ακυρωμένες καμπάνιες. Εκτελείται από εξωτερικούς χρήστες και λειτουργεί με βάση τις ακόλουθες ενέργειες:

Αναζήτηση Καμπανιών: Η συνάρτηση επαναλαμβάνει όλες τις καμπάνιες που έχουν δημιουργηθεί, χρησιμοποιώντας έναν βρόχο που διατρέχει από το 1 έως το campaignCount.

Έλεγχος Ακυρωμένων Καμπανιών: Για κάθε καμπάνια, ελέγχει αν η καμπάνια έχει ακυρωθεί (campaign.cancelled) και ο επενδυτής κατέχει μετοχές στην καμπάνια (campaign.backerPledges[msg.sender] > 0).

Υπολογισμός Αποζημίωσης: Αν πληρούνται τα παραπάνω κριτήρια, η συνάρτηση υπολογίζει την αποζημίωση για τον επενδυτή, πολλαπλασιάζοντας τον αριθμό των μετοχών που κατέχει ο επενδυτής με το κόστος ανά την υφιστάμενη τιμή μετοχής (refundAmount = backerPledges[msg.sender] * pledgeCost).

Ενημέρωση Δεδομένων:

Το ποσό αποζημίωσης προστίθεται στο σύνολο της επιστροφής (totalRefund) καθώς μπορεί να υπάρχουν περισσότερες από μία ακυρωμένης καμπάνιες.

Ο αριθμός των μετοχών του επενδυτή στη συγκεκριμένη καμπάνια μηδενίζεται (backerPledges[msg.sender] = 0).

Απαίτηση Επιστροφής: Μετά την επανάληψη όλων των καμπανιών, η συνάρτηση ελέγχει αν το συνολικό ποσό επιστροφής είναι μεγαλύτερο από μηδέν. Αν όχι, ακυρώνεται η συναλλαγή με το μήνυμα "No refundable shares available".

Μεταφορά Ποσού: Εάν υπάρχουν διαθέσιμες αποζημιώσεις, το συνολικό ποσό επιστρέφεται στον επενδυτή μέσω της εντολής payable(msg.sender).transfer(totalRefund).

Ενεργοποίηση Συμβάντος: Για κάθε καμπάνια που επιστρέφονται χρήματα, ενεργοποιείται το συμβάν **BackerCompensated**, το οποίο ενημερώνει για τον επενδυτή, την καμπάνια, και το ποσό της επιστροφής.

Η συνάρτηση διασφαλίζει ότι οι επενδυτές που έχουν επενδύσει σε ακυρωμένες καμπάνιες μπορούν να λάβουν πίσω τα χρήματά τους, ενώ παράλληλα ενημερώνει σωστά τα δεδομένα για τις μετοχές τους.

Αυτοαξιολόγηση:

Η συνάρτηση **compensateBacker** καλύπτει αποτελεσματικά τη διαδικασία επιστροφής χρημάτων στους επενδυτές για μετοχές που έχουν επενδύσει σε ακυρωμένες καμπάνιες. Διασφαλίζει την ορθή επεξεργασία όλων των καμπανιών, τον υπολογισμό της αποζημίωσης και την ενημέρωση των δεδομένων του συμβολαίου, ενώ αποτρέπει επαναλαμβανόμενες απαιτήσεις μέσω του μηδενισμού των μετοχών του επενδυτή. Το συμβάν **BackerCompensated** παρέχει διαφάνεια, ενημερώνοντας τους χρήστες για τις συναλλαγές. Η συνάρτηση δεν είναι αποδοτική σε περιπτώσεις μεγάλου αριθμού καμπανιών, καθώς διατρέχει όλες τις καμπάνιες ανεξαρτήτως κατάστασης, κάτι που μπορεί να αυξήσει το κόστος (gas). Συνολικά, η συνάρτηση καλύπτει το **95% του ζητήματος**, αλλά θα μπορούσε να βελτιωθεί με την προσθήκη φίλτρων για πιο αποδοτική αναζήτηση καμπανιών και την ενσωμάτωση μηχανισμών αποτυχίας μεταφοράς.

Ολοκλήρωση καμπάνιας – fulfillCampaign

Η συνάρτηση **fulfillCampaign** επιτρέπει την ολοκλήρωση μιας καμπάνιας, εφόσον πληρούνται συγκεκριμένες προϋποθέσεις. Η λειτουργία της εξασφαλίζει ότι η καμπάνια είναι έτοιμη να χαρακτηριστεί ως ολοκληρωμένη και να διανεμηθούν τα κεφάλαια στον επιχειρηματία. Η συνάρτηση λαμβάνει ως όρισμα το μοναδικό αναγνωριστικό της καμπάνιας (campaignId) και πραγματοποιεί τους ακόλουθους ελέγχους μέσω τροποποιητών:

- Ο καλών πρέπει να είναι εξουσιοδοτημένος να ολοκληρώσει την καμπάνια (onlyAuthorized).
- Η καμπάνια δεν πρέπει να έχει ακυρωθεί (notCancelled).
- Η καμπάνια πρέπει να έχει επιτύχει την απαιτούμενη χρηματοδότηση (fullyFunded).
- Η καμπάνια δεν πρέπει να έχει ολοκληρωθεί ήδη (notAlreadyCompleted).
- Η καμπάνια πρέπει να υπάρχει (campaignExists).

Μετά τους ελέγχους, η συνάρτηση αποκτά πρόσβαση στη συγκεκριμένη καμπάνια μέσω του mapping campaigns. Υπολογίζει το ποσό πληρωμής (ayout) που αντιστοιχεί στο 80% των συνολικών κεφαλαίων που συγκεντρώθηκαν, αφήνοντας το υπόλοιπο 20% διαθέσιμο για ανάληψη από τον ιδιοκτήτη του συμβολαίου. Το ποσό αυτό μεταφέρεται στον επιχειρηματία μέσω της εντολής transfer, ενώ η κατάσταση της καμπάνιας ενημερώνεται με την ένδειξη fulfilled = true, δηλώνοντας ότι η καμπάνια έχει ολοκληρωθεί. Επιπλέον, η μεταβλητή withdraw ορίζεται σε true, επιτρέποντας στον ιδιοκτήτη του συμβολαίου να κάνει ανάληψη του υπολοίπου.

Τέλος, ενεργοποιείται το συμβάν **CampaignFulfilled**, ενημερώνοντας για την επιτυχή ολοκλήρωση της καμπάνιας, τον καλούντα, το αναγνωριστικό της καμπάνιας και το ποσό πληρωμής. Η συνάρτηση διασφαλίζει την αποδοτική και διαφανή ολοκλήρωση των καμπανιών, προστατεύοντας τα δικαιώματα των επιχειρηματιών και του ιδιοκτήτη.

Αυτοαξιολόγηση:

Η συνάρτηση **fulfillCampaign** καλύπτει αποτελεσματικά τη διαδικασία ολοκλήρωσης μιας καμπάνιας, διασφαλίζοντας ότι πληρούνται όλες οι προϋποθέσεις πριν τη μεταφορά των κεφαλαίων στον επιχειρηματία. Μέσω modifiers, η συνάρτηση προστατεύει από μη εξουσιοδοτημένες ενέργειες, ακυρωμένες ή ήδη ολοκληρωμένες καμπάνιες, καθώς και καμπάνιες που δεν έχουν συγκεντρώσει την απαιτούμενη χρηματοδότηση. Η χρήση του **storage** εξασφαλίζει αποδοτική διαχείριση των δεδομένων, ενώ η εντολή **transfer** επιτρέπει την άμεση καταβολή του ποσού στον επιχειρηματία. Παρόλα αυτά, παρουσιάζονται ορισμένες ελλείψεις. Δεν υπάρχει μηχανισμός για την αντιμετώπιση αποτυχιών στη μεταφορά του ποσού, γεγονός που μπορεί να οδηγήσει σε μη ολοκληρωμένες συναλλαγές σε σπάνιες περιπτώσεις. Συνολικά, η συνάρτηση καλύπτει το **95% του ζητήματος**, αλλά μπορεί να βελτιωθεί με την προσθήκη ελέγχων αποτυχίας μεταφοράς.

Επιπλέον συναρτήσεις

getActiveCampaigns

Η συνάρτηση **getActiveCampaigns** επιστρέφει πληροφορίες για όλες τις ενεργές καμπάνιες, δηλαδή εκείνες που δεν έχουν ακυρωθεί και δεν έχουν ολοκληρωθεί. Διατρέχει όλες τις καμπάνιες που είναι αποθηκευμένες στο mapping campaigns και δημιουργεί arrays με τα αναγνωριστικά (ids), τους επιχειρηματίες (entrepreneurs), τους τίτλους (titles), το κόστος ανά μετοχή (pledgeCosts), και τις απαιτούμενες μετοχές (pledgesNeeded). Αυτά τα δεδομένα επιστρέφονται στον καλούντα, επιτρέποντας την εύκολη προβολή όλων των ενεργών καμπανιών.

getFulfilledCampaigns

Η συνάρτηση **getFulfilledCampaigns** επιστρέφει πληροφορίες για όλες τις καμπάνιες που έχουν ολοκληρωθεί. Διατρέχει τις καμπάνιες και δημιουργεί arrays με το αναγνωριστικό, τον επιχειρηματία, τον τίτλο, το κόστος ανά μετοχή, τις απαιτούμενες μετοχές και τον αριθμό των μετοχών που συγκεντρώθηκαν (pledgesCount). Αυτή η συνάρτηση επιτρέπει την προβολή των επιτυχημένων καμπανιών και των αποτελεσμάτων τους.

getCancelledCampaigns

Η συνάρτηση **getCancelledCampaigns** επιστρέφει πληροφορίες για όλες τις ακυρωμένες καμπάνιες. Παρομοίως με τις προηγούμενες συναρτήσεις, συλλέγει τα δεδομένα για τις καμπάνιες που έχουν ακυρωθεί, όπως το αναγνωριστικό, τον επιχειρηματία, τον τίτλο, το κόστος ανά μετοχή και τις απαιτούμενες μετοχές, και τα επιστρέφει σε πίνακες.

getBackers

Η συνάρτηση **getBackers** επιστρέφει τη λίστα των επενδυτών και τον αριθμό των μετοχών που κατέχει ο καθένας για μια συγκεκριμένη καμπάνια, η οποία προσδιορίζεται μέσω του campaignId. Διατρέχει τη λίστα των επενδυτών που σχετίζονται με την καμπάνια και επιστρέφει πίνακες με τις διευθύνσεις και τον αριθμό των μετοχών τους.

getbackerPledges

Η συνάρτηση **getBackerPledges** επιστρέφει όλες τις καμπάνιες στις οποίες ένας επενδυτής έχει μετοχές, καθώς και τον αριθμό των μετοχών που κατέχει σε κάθε μία από αυτές. Διατρέχει όλες τις καμπάνιες και συλλέγει τα δεδομένα σε arrays που περιλαμβάνουν τα IDs των καμπανιών (campaignIds) και τις μετοχές του επενδυτή.

getTotalFees

Η συνάρτηση **getTotalFees** υπολογίζει τα συνολικά τέλη που έχουν συγκεντρωθεί από ολοκληρωμένες καμπάνιες. Διατρέχει τις καμπάνιες που έχουν ολοκληρωθεί και από τις οποίες δεν έχει γίνει ακόμα ανάληψη, υπολογίζει τα τέλη (20% από τα συνολικά κεφάλαια που συγκεντρώθηκαν και ένα σταθερό ποσό campaignFee), και επιστρέφει το άθροισμα αυτών των ποσών.

withdrawFees

Η συνάρτηση **withdrawFees** επιτρέπει στον ιδιοκτήτη του συμβολαίου να κάνει ανάληψη των συνολικών τελών που έχουν συγκεντρωθεί από ολοκληρωμένες καμπάνιες. Ελέγχει ότι τα τέλη **δεν έχουν ήδη αποσυρθεί** για κάθε καμπάνια, υπολογίζει το συνολικό ποσό και το μεταφέρει στον ιδιοκτήτη μέσω της εντολής transfer. Μετά την ανάληψη, η συνάρτηση ενημερώνει τη μεταβλητή withdraw σε κάθε καμπάνια ώστε να διασφαλιστεί ότι τα τέλη μπορούν να αποσυρθούν μόνο μία φορά.

Αυτοαξιολόγηση και Προοπτικές Βελτίωσης

- **Πληρότητα Λειτουργιών:** Το συμβόλαιο καλύπτει όλες τις βασικές λειτουργίες μιας πλατφόρμας crowdfunding. Οι επενδυτές μπορούν να το χρησιμοποιούν με ασφάλεια, ενώ οι επιχειρηματίες διαχειρίζονται τις καμπάνιες τους με διαφάνεια.
- **Ασφάλεια:** Αξιοποιούνται **modifiers** για τον έλεγχο δικαιωμάτων και καταστάσεων (π.χ. notCancelled, fullyFunded), περιορίζοντας τις κακόβουλες ή λανθασμένες κλήσεις.
- **Διαχείριση Σφαλμάτων:** Θα μπορούσαν να ενσωματωθούν επιπλέον έλεγχοι για άκρα σενάρια, π.χ. υπερχείλιση, αποτυχία μεταφοράς Ether σε περίπτωση fallback λειτουργιών κ.λπ.
- **Κλιμάκωση:** Για πολύ μεγάλες λίστες καμπανιών ή επενδυτών, ενδέχεται να χρειαστούν βελτιστοποιημένες δομές δεδομένων ή διαχωρισμός των αποθηκευτικών δομών.
- **Επεκτασιμότητα:** Μπορεί να προστεθούν λειτουργίες όπως:
 - Εύκολες κλήσεις για μαζικές επιστροφές.
 - Έλεγχος χρονικής διάρκειας μιας καμπάνιας (deadline).
 - Πρόσθετη λογική ανταμοιβών (rewards) για επενδυτές.

Συνολικά, το έξυπνο συμβόλαιο καταφέρνει να προσφέρει μια πλήρη βάση για την ανάπτυξη εφαρμογής crowdfunding στο δίκτυο Ethereum, επιτρέποντας ασφαλείς συναλλαγές και ξεκάθαρους ρόλους (ιδιοκτήτης, επιχειρηματίες, επενδυτές). Συνδυαζόμενο με ένα κατάλληλο **frontend**, δίνει μια ολοκληρωμένη εμπειρία πληθοχρηματοδότησης (crowdfunding) σε αποκεντρωμένο περιβάλλον.

Επιτελική Σύνοψη - Frontend

Η παρούσα εργασία αποτελεί την υλοποίηση μίας εφαρμογής crowdfunding (πληθοχρηματοδότησης) τύπου <https://www.kickstarter.com/> η οποία έχει βασιστεί στην τεχνολογία Blockchain χρησιμοποιώντας το Ethereum αναπτύσσοντας μια εφαρμογή web με απλή React. Για την λειτουργία της υλοποιήθηκε ένα έξυπνο συμβόλαιο στην γλώσσα Solidity .

Για την υλοποίηση του συμβολαίου χρησιμοποιήθηκαν οι εξής λειτουργίες.

Δημιουργία καμπάνιας. Εγγραφή από επιχειρηματία (entrepreneur) πληρώνοντας κάποιο αντίτιμο στο συμβόλαιο και δηλώνοντας τις πληροφορίες της (ένα όνομα, το κόστος μετοχής και το πλήθος των μετοχών που απαιτούνται για την ολοκλήρωσή της). Κάθε εκστρατεία αποκτά έναν μοναδικό αναγνωριστικό αριθμό. Μπορεί οποιοσδήποτε (εκτός του ιδιοκτήτη του συμβολαίου) να εκτελέσει την ενέργεια αυτή ακόμη και επανειλημμένα με την προϋπόθεση πως δεν είναι στην λίστα αποκλεισμένων (banned list). Όποτε την εκτελεί, πρέπει να αποστέλλει στο συμβόλαιο την αξία του τέλους καμπάνιας όπως αυτή έχει καθοριστεί σφαιρικά, ίδια για όλες τις καμπάνιες (π.χ. 0,02 Ethers).

Χρηματοδότηση καμπάνιας. Αγορά μετοχών από επενδυτές (backers) και διατήρηση των χρημάτων στο συμβόλαιο έως ότου ολοκληρωθεί η εκστρατεία. Μπορεί οποιοσδήποτε να εκτελέσει την ενέργεια αυτή, ακόμη και επανειλημμένα υπό την προϋπόθεση πως η καμπάνια δεν έχει ακόμη ολοκληρωθεί ή ακυρωθεί. Όποτε την εκτελεί, πρέπει να πληρώνει στο συμβόλαιο την αξία των μετοχών, το πλήθος των οποίων δηλώνει πως θέλει να αγοράσει.

Ακύρωση καμπάνιας. Να επιτρέπει την επιστροφή των χρημάτων που φυλάσσονται στο συμβόλαιο για μια συγκεκριμένη εκστρατεία στους επενδυτές. Χαρακτηρισμός της εκστρατείας ως άκυρης. Την ενέργεια αυτή μπορεί να την εκτελέσει μόνο ο επιχειρηματίας που την εκκίνησε ή ο ιδιοκτήτης του συμβολαίου υπό την προϋπόθεση πως η καμπάνια δεν έχει ακόμη ολοκληρωθεί ή ακυρωθεί.

Αποζημίωση επενδυτή. Επιστροφή των χρημάτων που έχουν επενδυθεί από συγκεκριμένο επενδυτή σε ακυρωμένες εκστρατείες. Μπορεί οποιοσδήποτε να καλέσει την ενέργεια αυτή. Αν υπάρχουν χρήματα τα οποία έχει επενδύσει σε μία ή περισσότερες καμπάνιες που έχουν ακυρωθεί, τα χρήματα αυτά του επιστρέφονται. Οι συγκεκριμένες καμπάνιες δεν του χρωστάνε πλέον επιστροφή.

Ολοκλήρωση καμπάνιας. Μεταφορά του 80% των χρημάτων που φυλάσσονται στο συμβόλαιο για την συγκεκριμένη εκστρατεία προς τον επιχειρηματία που την εκκίνησε. Χαρακτηρισμός της ως ολοκληρωμένης. Την ενέργεια αυτή μπορεί να την εκτελέσει μόνο ο επιχειρηματίας που την εκκίνησε ή ο ιδιοκτήτης του συμβολαίου υπό την προϋπόθεση πως δεν είναι ακυρωμένη και η καμπάνια έχει συμπληρώσει το πλήθος των μετοχών που δηλώθηκε αρχικά (ότι την ολοκληρώνουν).

Διαχείριση εφαρμογής. Μεταφορά των κρατήσεων προς τον ιδιοκτήτη της εφαρμογής, εισαγωγή επιχειρηματία στη λίστα ανεπιθύμητων, αλλαγή ιδιοκτήτη εφαρμογής, καταστροφή εφαρμογής - συμβολαίου.

Κώδικας και συμπεριφορά της εφαρμογής (frontend)

Σκοπός του παρόντος Web App είναι να προσφέρει ένα ολοκληρωμένο περιβάλλον χρήσης για την αλληλεπίδραση με το έξυπνο συμβόλαιο που αναπτύχθηκε παραπάνω. Η εφαρμογή είναι υλοποιημένη σε απλή React και αξιοποιεί τη βιβλιοθήκη [web3.js](#) για τη σύνδεση, φόρτωση δεδομένων και εκτέλεση συναρτήσεων του έξυπνου συμβολαίου. Η εφαρμογή επιτρέπει σε επιχειρηματίες να δημιουργήσουν καμπάνιες, σε επενδυτές να τις χρηματοδοτήσουν, καθώς και στον ιδιοκτήτη του συμβολαίου να διαχειριστεί τις λειτουργίες διαχείρισης (π.χ. απόσυρση κερδών, αποκλεισμό επιχειρηματιών, καταστροφή του συμβολαίου κ.λπ.).

1. Ρόλος και Λειτουργία του Frontend

Η βασική ιδέα του frontend είναι:

- Σύνδεση με το έξυπνο συμβόλαιο:** Γίνεται με χρήση [web3.js](#) και του ABI του συμβολαίου (CrowdfundingABI).
- Προβολή ενεργών, ακυρωμένων και ολοκληρωμένων καμπανιών:** Οι χρήστες μπορούν να δουν τα δεδομένα καμπανιών (τίτλος, κόστος μετοχής, αριθμό μετοχών που έχουν ήδη αγοραστεί κ.λπ.).
- Δυνατότητα δημιουργίας καμπάνιας:** Ένας επιχειρηματίας (εκτός των αποκλεισμένων ή του ιδιοκτήτη του συμβολαίου) μπορεί να δημιουργήσει καμπάνια.
- Δυνατότητα χρηματοδότησης:** Ένας επενδυτής μπορεί να αγοράζει μετοχές μίας καμπάνιας μέσω συναλλαγής.
- Διαχείριση καμπανιών:** Ακύρωση, ολοκλήρωση, αποζημίωση επενδυτών κ.ά.
- Διαχειριστικές λειτουργίες (μόνο για τον ιδιοκτήτη του συμβολαίου):**
 - Απόσυρση τελών (withdrawFees).
 - Αλλαγή ιδιοκτήτη (changeOwner).
 - Αποκλεισμός (ban) επιχειρηματιών.
 - Καταστροφή του συμβολαίου (destroyContract).

Όλη η λογική παρουσιάζεται συγκεντρωμένη σε ένα React component (App.js), το οποίο διαχειρίζεται την κατάσταση (state) και τις συναλλαγές με το blockchain.

2. Τεχνολογίες και Βιβλιοθήκες

- React:** Χρησιμοποιείται για την υλοποίηση της διεπαφής χρήστη, την ενημέρωση του DOM και τη διαχείριση κατάστασης.
- Web3.js:** Χρησιμοποιείται για τη σύνδεση με το Metamask και την επικοινωνία με το έξυπνο συμβόλαιο στο δίκτυο.
- Bootstrap:** Χρησιμοποιείται για τη σχεδίαση των φορμών και των πινάκων.

3. Κύριες Μεταβλητές και Κατάσταση (State)

Το component `App` διατηρεί στο state μια σειρά από μεταβλητές, μεταξύ των οποίων:

- `account`: Η διεύθυνση του τρέχοντος συνδεδεμένου χρήστη στο Metamask.
- `connected`: Boolean μεταβλητή που δηλώνει αν υπάρχει επιτυχής σύνδεση με πορτοφόλι.
- `contractOwner`: Η διεύθυνση του ιδιοκτήτη του συμβολαίου.
- `contractBalance`: Το υπόλοιπο ETH που είναι αποθηκευμένο στο συμβόλαιο.
- `destroyed`: Κατάσταση που δηλώνει αν το συμβόλαιο έχει καταστραφεί (ή όχι).
- `title, pledgeCost, pledgesNeeded`: Πεδία φόρμας για τη δημιουργία νέας καμπάνιας.
- `campaigns, cancelledCampaigns, fulfilledCampaigns`: Λίστες που αποθηκεύουν δεδομένα καμπανιών.
- `isBanned`: Ελέγχει αν η τρέχουσα διεύθυνση χρήστη είναι αποκλεισμένη (banned) στο συμβόλαιο.
- `newOwnerAddress, entrepreneurAddress`: Χρησιμοποιούνται στο Control Panel για αλλαγή ιδιοκτήτη και αποκλεισμό.
- `collectedFees`: Τα συνολικά τέλη (fees) που έχουν συσσωρευτεί.

Η διαχείριση του state γίνεται κυρίως μέσω των μεθόδων και των event handlers (Class Component), προκειμένου να ανανεώνεται κατάλληλα και αυτόματα η διεπαφή χρήστη χωρίς την χρήση της SetInterval.

4. Διαδικασίες Αρχικοποίησης

Με την εκκίνηση της εφαρμογής, εκτελούνται μεταξύ άλλων:

- **`initializeWeb3()`:**
 - Συνδέεται στο Metamask, ζητάει άδεια λογαριασμών (accounts) και ορίζει το `web3` αντικείμενο.
 - Ορίζει το `contract` αντικείμενο με βάση το ABI και τη διεύθυνση του συμβολαίου.
- **`loadBlockchainData()`:**
 - Φορτώνει πληροφορίες για τον συνδεδεμένο λογαριασμό (`account`).
 - Διαβάζει τον ιδιοκτήτη του συμβολαίου (`contractOwner`) και το υπόλοιπο του συμβολαίου (`contractBalance`).
 - Ελέγχει τυχόν συσσωρευμένα τέλη (`collectedFees`).
- **Φόρτωση λιστών καμπανιών:**
 - `loadActiveCampaigns()`, `loadFulfilledCampaigns()`, `loadCancelledCampaigns()`.
- Ενεργοποίηση ελέγχου αν ο τρέχων χρήστης είναι αποκλεισμένος: `checkIfBanned()`.
- **`setupAccountListener()` και `setupEventListeners()`:**
 - Παρακολουθούν τυχόν αλλαγές λογαριασμών (Metamask) και γεγονότα από το συμβόλαιο.

5. Διασύνδεση με το Blockchain

5.1 Σύνδεση Μεταβλητών

- `web3`: δημιουργείται μέσω `new Web3(window.ethereum)`.
- `contract`: δημιουργείται μέσω `new web3.eth.Contract(CrowdfundingABI, contractAddress)`.

5.2 Συναρτήσεις Φόρτωσης Δεδομένων

- `loadActiveCampaigns()`: Καλεί τη συνάρτηση `getActiveCampaigns()` από το συμβόλαιο. Επιστρέφει πίνακες με τα IDs, τους τίτλους, το `pledgeCost` κ.λπ. Δημιουργεί έναν πίνακα `campaigns` όπου συνδυάζει τα δεδομένα και υπολογίζει παράλληλα πόσες μετοχές έχει ήδη κάποιος επενδυτής (`yourPledges`).
- `loadFulfilledCampaigns()` και `loadCancelledCampaigns()`: Παρόμοια λογική, αλλά για ολοκληρωμένες και ακυρωμένες καμπάνιες αντίστοιχα.

5.3 Χειρισμός Λογαριασμών

- `setupAccountListener()`: Παρακολουθεί αλλαγές στο Metamask. Μόλις αλλάξει ο λογαριασμός, ενημερώνεται το `state` και επαναφορτώνονται τα δεδομένα.

5.4 Παρακολούθηση Συμβάντων (Events)

- `setupEventListeners()`: Συνδέεται στα συμβάντα του smart contract (π.χ. `CampaignCreated`, `CampaignFulfilled`, `CampaignCancelled` κ.λπ.).
- Όταν π.χ. πυροδοτείται το `CampaignCreated`, η εφαρμογή επαναφορτώνει τα δεδομένα (π.χ. `loadActiveCampaigns()`).

6. Ενέργειες του Χρήστη (Event Handlers)

6.1 Δημιουργία Καμπάνιας – `createCampaign()`

1. Παίρνει τιμές από το `state` (`title`, `pledgeCost`, `pledgesNeeded`).
2. Υπολογίζει το πάγιο τέλος καμπάνιας (`campaignFee = 0.02 ETH`).
3. Καλεί τη συνάρτηση `createCampaign(title, pledgeCost, pledgesNeeded)` στο συμβόλαιο με `send(...)` και `value: campaignFee`.
4. Σε περίπτωση επιτυχίας, ενημερώνει τον χρήστη (`alert`) και επαναφορτώνει τις ενεργές καμπάνιες.

6.2 Χρηματοδότηση Καμπάνιας – `handlePledge(campaignId)`

1. Εντοπίζει την καμπάνια από τοπικό `state`.
2. Μετατρέπει το `pledgeCost` σε `Wei`.
3. Καλεί `fundCampaign(campaignId, 1)` στο smart contract με το αντίστοιχο `value`.
4. Ενημερώνει τον χρήστη και επαναφορτώνει τα δεδομένα (`campaigns`, `contractBalance` κ.λπ.).

6.3 Ακύρωση Καμπάνιας – `handleCancel(campaignId)`

1. Καλεί `cancelCampaign(campaignId)` από το συμβόλαιο (επιτρεπτό μόνο για τον ιδιοκτήτη ή τον επιχειρηματία της καμπάνιας).
2. Ενημερώνει την κατάσταση και ανανεώνει τη λίστα ακυρωμένων καμπανιών.

6.4 Ολοκλήρωση Καμπάνιας – `handleFulfill(campaignId)`

1. Καλεί `fulfillCampaign(campaignId)` στο συμβόλαιο.
2. Ενημερώνει την κατάσταση (π.χ. ενεργές/ολοκληρωμένες καμπάνιες) και το υπόλοιπο του συμβολαίου.

6.5 Αποζημίωση – `handleClaimAll()`

1. Καλεί `compensateBacker()` στο συμβόλαιο, όπου επιστρέφονται αυτόματα όλα τα ποσά για ακυρωμένες καμπάνιες που έχει χρηματοδοτήσει ο χρήστης.
2. Ενημερώνει τη λίστα ακυρωμένων καμπανιών και το υπόλοιπο.

6.6 Διαχειριστικές λειτουργίες (Control Panel)

- `withdrawFees()`: Απόδοση των συγκεντρωμένων τελών στον ιδιοκτήτη.
- `changeOwner()`: Αλλαγή ιδιοκτήτη του συμβολαίου.
- `banEntrepreneur()`: Εισαγωγή ενός επιχειρηματία στη λίστα αποκλεισμένων.
- `destroyContract()`: Καταστροφή του συμβολαίου εφόσον είμαστε ο ιδιοκτής.

7. Δομή και Εμφάνιση (Rendering)

Το αρχείο `App.js` χρησιμοποιεί διάφορες μεθόδους `render` για να διαχωρίσει λογικά την εμφάνιση:

1. **renderHeader()**
 - Εμφανίζει βασικές πληροφορίες όπως διεύθυνση τρέχοντος χρήστη, διεύθυνση ιδιοκτήτη, υπόλοιπο συμβολαίου, κ.λπ.
2. **renderCreateCampaign()**
 - Παρέχει φόρμα δημιουργίας καμπάνιας με τα πεδία `title`, `pledgeCost`, `pledgesNeeded`.
 - Κάνει `disable` τη φόρμα αν ο χρήστης είναι ο ιδιοκτήτης ή `banned` ή αν το συμβόλαιο έχει καταστραφεί.
3. **renderLiveCampaigns()**
 - Προβάλει πίνακα με τις ενεργές καμπάνιες.
 - Περιλαμβάνει κουμπιά για `Pledge`, `Cancel`, `Fulfill` αναλόγως της ιδιότητας του χρήστη.
4. **renderFulfilledCampaigns()**
 - Προβάλλει πίνακα με τις καμπάνιες που έχουν ολοκληρωθεί.
5. **renderCancelledCampaigns()**
 - Προβάλλει τις ακυρωμένες καμπάνιες.
 - Παρέχει δυνατότητα “Claim All” αν ο τρέχων χρήστης έχει μετοχές που μπορούν να αποζημιωθούν.
6. **renderControlPanel()**
 - Εμφανίζει ένα panel αποκλειστικά για τον ιδιοκτήτη (ή ειδικές εξουσιοδοτημένες διευθύνσεις).
 - Επιτρέπει λειτουργίες `ban`, `changeOwner`, `withdrawFees`, `destroyContract`.

Η διάταξη συνθέτει όλα τα παραπάνω τμήματα σε μία κεντρική σελίδα, δίνοντας τη δυνατότητα στους χρήστες να αλληλεπιδράσουν με όλες τις πτυχές της εφαρμογής.

8. Αυτοαξιολόγηση

1. Κάλυψη Απαιτήσεων:

- Το frontend καλύπτει όλες τις βασικές λειτουργίες της πλατφόρμας crowdfunding: δημιουργία, χρηματοδότηση, ακύρωση, ολοκλήρωση, αποζημίωση και διαχειριστικές ενέργειες.
- Παρακολουθούνται σε πραγματικό χρόνο τα συμβάντα (events) του έξυπνου συμβολαίου, ώστε η διεπαφή χρήστη να ενημερώνεται δυναμικά.

2. Ευκολία Χρήσης:

- Η χρήση React και Bootstrap προσφέρει ένα σχετικά φιλικό UI/UX.
- Κάθε ενέργεια διαθέτει κουμπιά, φόρμες και ειδοποίησεις (alerts) που καθοδηγούν τον χρήστη.

3. Βελτιώσεις:

- **Διάσπαση σε επιμέρους components:** Το αρχείο App.js συγκεντρώνει όλη την αλληλεπίδραση. Σε μεγαλύτερα έργα θα βοηθούσε ο επιμερισμός της λογικής σε ξεχωριστά components (π.χ. "CampaignList", "CreateCampaignForm", "ControlPanel").
- **Διαχείριση σφαλμάτων:** Θα μπορούσαν να χρησιμοποιηθούν πιο εξειδικευμένοι μηχανισμοί για τη διαχείριση των σφαλμάτων στις συναλλαγές (π.χ. προβολή user-friendly μηνυμάτων).
- **Βελτιστοποίηση:** Για εξοικονόμηση gas σε μεγαλύτερα συστήματα, ίσως απαιτούνται διαφορετικές δομές δεδομένων ή κλήσεις backend.
- **Περισσότερος έλεγχος έγκυρων διευθύνσεων:** Στις φόρμες εισόδου (π.χ. changeOwner, banEntrepreneur) θα μπορούσαν να προστεθούν έλεγχοι valid Ethereum address.

Σε γενικές γραμμές, το frontend καλύπτει ουσιαστικά όλο το απαιτούμενο φάσμα λειτουργιών του έξυπνου συμβολαίου και προσφέρει ένα επαρκές περιβάλλον χρήσης. Με λίγες βελτιώσεις στον κώδικα και την αρχιτεκτονική, η εφαρμογή μπορεί να επεκταθεί και να βελτιώσει ακόμη περισσότερο την εμπειρία χρήστη και την ανθεκτικότητά της σε διάφορες καταστάσεις.

Εγχειρίδιο Χρήσης - User Manual

1. Προαπαιτούμενα

1. **Metamask:** Θα πρέπει να έχετε εγκατεστημένο το Metamask στον browser σας.
2. **Διασύνδεση με το Δίκτυο:** Να έχετε συνδεδεμένο το Metamask στο δίκτυο Sepolia ETH.

2. Εκκίνηση Εφαρμογής & Σύνδεση με Metamask

1. **Φορτώστε την εφαρμογή:** Ανοίξτε τη σελίδα του DApp <https://ahadjiantonis-dev.github.io/CrowdFunding-dApp/>
2. **Σύνδεση Metamask:** Αν είστε εκτός σύνδεσης, το DApp θα σας ζητήσει να συνδεθείτε μέσω Metamask. Κάντε κλικ στο «Connect» (αν εμφανιστεί σχετικό μήνυμα) και επιλέξτε τον λογαριασμό που θέλετε να χρησιμοποιήσετε.
3. **Επιβεβαίωση:** Μόλις ολοκληρωθεί η σύνδεση, στο πάνω μέρος (Header) θα δείτε τη διεύθυνσή σας (Current Address) και τη διεύθυνση του ιδιοκτήτη του συμβολαίου (Contract Owner).

3. Βασική Επισκόπηση Διεπαφής Χρήστη

Η κύρια σελίδα χωρίζεται σε τμήματα:

1. **Header:**
 - Προβάλει την τρέχουσα διεύθυνση που έχετε συνδεδεμένη (Current Address).
 - Εμφανίζει τον ιδιοκτήτη του συμβολαίου (Contract Owner).
 - Φανερώνει το υπόλοιπο του συμβολαίου (Contract Balance) και τα συνολικά συλλεγμένα τέλη (Collected Fees).
2. **Create Campaign (Φόρμα Δημιουργίας Καμπάνιας):**
 - Επιτρέπει τη συμπλήρωση τίτλου, κόστους ανά μετοχή (ETH) και αριθμού μετοχών που απαιτούνται.
 - Ενεργή μόνο για χρήστες που **δεν** είναι αποκλεισμένοι (banned), και εφόσον το συμβόλαιο δεν είναι κατεστραμμένο.
 - Ανενεργή για τον ιδιοκτήτη του συμβολαίου.
3. **Live Campaigns (Ενεργές Καμπάνιες):**
 - Παρουσιάζει έναν πίνακα με όλες τις ενεργές καμπάνιες: Επιχειρηματία, Τίτλο, Κόστος, Αγορασμένες και Εναπομείνασες μετοχές.
 - Υπάρχουν κουμπιά για **Pledge** (αγορά μιας μετοχής), **Cancel** (ακύρωση καμπάνιας) και **Fulfill** (ολοκλήρωση), ανάλογα με τα δικαιώματα του χρήστη.
4. **Fulfilled Campaigns (Ολοκληρωμένες Καμπάνιες):**
 - Εμφανίζει πίνακα με όσες καμπάνιες έχουν φτάσει το στόχο τους και έχει γίνει η μεταφορά χρημάτων στον επιχειρηματία και στον ιδιοκτήτη.

5. **Cancelled Campaigns (Ακυρωμένες Καμπάνιες):**
 - Εμφανίζει πίνακα με ακυρωμένες καμπάνιες.
 - Παρέχει δυνατότητα «Claim All» για τους επενδυτές, ώστε να επιστρέψουν μαζικά τα ποσά που είχαν επενδύσει σε αυτές.
6. **Control Panel** (μόνο για τον ιδιοκτήτη του συμβολαίου):
 - Κουμπί **Withdraw** για την απόσυρση των συγκεντρωμένων τελών.
 - Δυνατότητα αλλαγής ιδιοκτήτη (Change Owner).
 - Αποκλεισμός (Ban) επιχειρηματία.
 - Καταστροφή (Destroy) του συμβολαίου.

4. Χρήση και Λειτουργικότητες

4.1 Δημιουργία Νέας Καμπάνιας

1. **Μετακινηθείτε στη “New Campaign” Ενότητα:** Θα δείτε τρία πεδία — Title, Pledge Cost (ETH), Pledges Needed.
2. **Συμπληρώστε:**
 - **Title:** Ο τίτλος της καμπάνιας (π.χ. “sunCharge”).
 - **Pledge Cost (ETH):** Το κόστος ανά μετοχή σε Ether (π.χ. 1).
 - **Pledges Needed:** Ο συνολικός αριθμός μετοχών που πρέπει να αγοραστούν (π.χ. 10).
3. **Πατήστε “Create Campaign”:** Θα εμφανιστεί το Metamask για να επιβεβαιώσετε τη συναλλαγή.
4. **Απαιτούμενο τέλος (campaignFee):** Θα χρειαστεί να καταβάλετε 0.02 ETH ως τέλος. Μετά την επιτυχημένη συναλλαγή, η νέα καμπάνια θα εμφανιστεί στη λίστα “Live Campaigns”.

Σημείωση: Αν είστε ο ιδιοκτήτης του συμβολαίου ή αποκλεισμένος, η φόρμα δημιουργίας θα είναι απενεργοποιημένη.

4.2 Χρηματοδότηση Καμπάνιας (Pledge)

1. **Εντοπίστε την καμπάνια:** Στη λίστα “Live Campaigns”, βρείτε αυτήν που θέλετε να χρηματοδοτήσετε.
2. **Κουμπί “Pledge”:** Πατήστε το. Αυτόματα αποστέλλεται συναλλαγή προς το smart contract με **value = pledgeCost**.
3. **Επιβεβαίωση στο Metamask:** Θα σας ζητηθεί να εγκρίνετε την πληρωμή σε Ether.
4. **Ολοκλήρωση:** Μετά την επιβεβαίωση, η καμπάνια θα ενημερωθεί με τις νέες μετοχές που αποκτήσατε και το υπόλοιπο του συμβολαίου θα αυξηθεί.

4.3 Ακύρωση Καμπάνιας (Cancel)

1. **Ποιος μπορεί να ακυρώσει:** Μόνο ο επιχειρηματίας που την ξεκίνησε ή ο ιδιοκτήτης του συμβολαίου.
2. **Κουμπί “Cancel”:** Εμφανίζεται δίπλα στην καμπάνια, μόνο αν δικαιούστε να την ακυρώσετε.
3. **Επιβεβαίωση:** Αφού εγκριθεί η συναλλαγή, η καμπάνια μετακινείται στην ενότητα “Cancelled Campaigns”.

4.4 Ολοκλήρωση Καμπάνιας (Fulfill)

- Προϋπόθεση:** Ο συνολικός αριθμός μετοχών (`p1edgesNeeded`) πρέπει να έχει πωληθεί (ή ξεπεραστεί).
- Ποιος μπορεί να εκπληρώσει:** Ο επιχειρηματίας ή ο ιδιοκτήτης του συμβολαίου.
- Κουμπί “Fulfill”:** Εάν έχετε τα δικαιώματα και η καμπάνια είναι κατάλληλη για ολοκλήρωση, πατήστε το.
- Μεταφορά Κεφαλαίων:** Το 80% των κεφαλαίων πηγαίνει αυτόματα στον επιχειρηματία, ενώ το υπόλοιπο 20% παραμένει στο συμβόλαιο.
- Επιτυχία:** Η καμπάνια μετακινείται στην ενότητα “Fulfilled Campaigns”.

4.5 Αποζημίωση Επενδυτή (Claim All)

- Ενότητα “Cancelled Campaigns”:** Εμφανίζει τις καμπάνιες που ακυρώθηκαν.
- Καμπάνιες με Επιστρεφόμενες Μετοχές:** Αν έχετε επενδύσει σε ακυρωμένες καμπάνιες, θα δείτε πόσες μετοχές κατέχετε (Your Pledges).
- Κουμπί “Claim All”:** Πατήστε το για να ζητήσετε πίσω όλα τα ποσά που επενδύσατε στις ακυρωμένες καμπάνιες.
- Επιβεβαίωση συναλλαγής:** Μετά την ολοκλήρωσή της, τα ETH επιστρέφουν στο πορτοφόλι σας.

4.6 Control Panel (Μόνο για τον Ιδιοκτήτη)

Αν η διεύθυνση σας είναι η ίδια με αυτή του `ContractOwner`, θα βλέπετε τις παρακάτω λειτουργίες στην τελευταία ενότητα:

- Withdraw**
 - Αποσύρει τα συγκεντρωμένα τέλη που ανήκουν στον ιδιοκτήτη (20% από ολοκληρωμένες καμπάνιες + fee δημιουργίας).
- Change Owner**
 - Μεταβιβάζει την ιδιοκτησία του συμβολαίου σε μια νέα διεύθυνση.
 - Συμπληρώστε το πεδίο “Enter new owner’s wallet address” και πατήστε το κουμπί.
- Ban Entrepreneur**
 - Προσθέτει μια διεύθυνση στη “μαύρη λίστα” (banned). Οι banned διευθύνσεις δεν μπορούν να δημιουργήσουν καμπάνιες.
 - Συμπληρώστε το πεδίο “Enter entrepreneur’s address” και πατήστε το κουμπί.
- Destroy Contract**
 - Σημαντική, οριστική ενέργεια που ακυρώνει όλες τις ενεργές καμπάνιες και βάζει το συμβόλαιο σε κατάσταση `destroyed`.
 - Μετά την καταστροφή, **δεν** επιτρέπονται νέες συναλλαγές όπως η δημιουργία καμπανιών παρά μόνο αποζημιώσεις.

5. Επιπλέον Υποθέσεις & Σχεδιαστικές Αποφάσεις

1. **Έλεγχος Δικαιωμάτων:** Ορισμένα κουμπιά (Cancel, Fulfill, Control Panel) εμφανίζονται μόνο σε χρήστες με τα ανάλογα δικαιώματα (ιδιοκτήτης ή επιχειρηματίας).
2. **Refresh Δεδομένων:** Μετά από κάθε ενέργεια, η εφαρμογή καλεί εκ νέου τις συναρτήσεις του συμβολαίου (π.χ. `getActiveCampaigns()`) για να ανανεώσει τους πίνακες, κάτι που δεν χρειάζεται να γίνεται χειροκίνητα από τον χρήστη, με αυτό τον τρόπο αναπάσα στιγμή τα δεδομένα είναι ενημερωμένα με τις τελευταίες κινήσεις που έχουν γίνει στο συμβόλαιο.

6. Αυτοαξιολόγηση & Περιορισμοί

1. **Διαχείριση Λαθών:**
 - ο Επί του παρόντος, τα περισσότερα σφάλματα εμφανίζονται απλώς στην κονσόλα ή ως `alert`. Θα ήταν προτιμότερο ένας πιο φιλικός τρόπος προβολής (π.χ. ειδικό μήνυμα στη σελίδα).
 - ο Σε περίπτωση που μία συναλλαγή δεν εγκριθεί από το Blockchain δεν υπάρχει ενημέρωση στον χρήστη. Αν για παράδειγμα δοκιμάσετε να δημιουργήσετε μία καμπάνια και εντός λίγων δευτερολέπτων δεν εμφανιστεί μάλλον δεν έχει εγκριθεί και θα ήταν φρόνιμο να ελέγχετε την απόρριψη συναλλαγής στο πορτοφόλι σας.
2. **Ασφάλεια:**
 - ο Το UI στηρίζεται κυρίως σε ελέγχους του smart contract. Έτσι, ακόμα κι αν κάποιος προσπαθήσει να κάνει κάτι “παράνομο” από το UI, το έξυπνο συμβόλαιο διαθέτει τους αντίστοιχους ελέγχους (modifiers).
3. **Μελλοντικές Βελτιώσεις:**
 - ο Καλύτερο UX (π.χ. pagination για τις λίστες καμπανιών, φίλτρα αναζήτησης κλπ).

Παρά τις βελτιώσεις που θα μπορούσαν να γίνουν, το DApp αυτό αποτελεί πλήρη εφαρμογή πληθοχρηματοδότησης, επιτρέποντας σε επενδυτές και επιχειρηματίες να αλληλεπιδρούν με ασφάλεια και διαφάνεια.