

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RAD   |  |  | | --- | --- | | Riferimento |  | | Versione | 1.0 | | Data |  | | Destinatario | Prof. Alfredo De Santis, Prof. Christian Esposito | | Presentato da | Carmine D’Angelo, Matteo Della Rocca |   E-Voting  “Proud to vote” |

TEAM COMPOSITION

|  |  |  |
| --- | --- | --- |
| **Ruolo** | **Nome** | **Posizione** |
| Top Manager | Alfredo De Santis | Cliente |
| Top Manager | Christian Esposito | Cliente |
| Team Member | Carmine D’Angelo |  |
| Team Member | Matteo Della Rocca |  |

SOMMARIO

[TEAM COMPOSITION 2](#_Toc125726058)

[1. Introduzione 4](#_Toc125726059)

[Questo è un progetto di gruppo per l'esame di Sicurezza dei Dati 2022/2023 presso l'Università degli Studi di Salerno, Laurea Magistrale in Informatica. 4](#_Toc125726060)

[1.1 Dominio 4](#_Toc125726061)

[1.2 Obiettivi e Criteri di Successo 4](#_Toc125726062)

[1.3 Sistema corrente 4](#_Toc125726063)

[1.4 Attori 4](#_Toc125726064)

[2. Sistema proposto 5](#_Toc125726065)

[2.1 Requisiti Funzionali 5](#_Toc125726066)

[2.2 Requisiti Non Funzionali 6](#_Toc125726067)

[2.2.1 Usabilità 6](#_Toc125726068)

[2.2.2 Affidabilità 6](#_Toc125726069)

[2.2.3 Prestazioni 6](#_Toc125726070)

[2.2.4 Manutenibilità 6](#_Toc125726071)

[2.2.5 Implementazione 6](#_Toc125726072)

[2.2.6 Interfaccia 6](#_Toc125726073)

[2.2.7 Sicurezza 6](#_Toc125726074)

[3. Modello del Sistema 7](#_Toc125726075)

[3.1 Scenari 7](#_Toc125726076)

[3.2 Use Cases 10](#_Toc125726077)

1. Introduzione

Questo è un progetto di gruppo per l'esame di Sicurezza dei Dati 2022/2023 presso l'Università degli Studi di Salerno, Laurea Magistrale in Informatica.

* 1. Dominio

Questo progetto è stato preso in considerazione in quanto pensiamo che l'integrità elettorale sia essenziale per un paese democratico e per la fiducia degli elettori nel proprio stato. I metodi di voto politico possono essere cruciali in questo senso, infatti, dal punto di vista del governo, le tecnologie di voto elettronico possono aumentare la partecipazione e la fiducia degli elettori e riaccendere l'interesse per il sistema di voto.

* 1. Obiettivi e Criteri di Successo

I protocolli di voto elettronico hanno un unico controllore che supervisiona l'intero processo di voto. Questa tecnica porta a selezioni errate dovute alla disonestà dell'autorità centrale (commissione elettorale), che è difficile correggere con i metodi esistenti. La rete decentralizzata può essere utilizzata come una moderna tecnica di voto elettronico per aggirare l'autorità centrale.

I sistemi di voto elettronico devono essere legittimi, accurati, sicuri e convenienti quando vengono utilizzati per le elezioni. Hanno un grande potenziale per ridurre i costi organizzativi e aumentare l'affluenza alle urne. Eliminano la necessità di stampare le schede elettorali o di aprire i seggi elettorali: gli elettori possono votare ovunque ci sia una connessione Internet. L'adozione può essere limitata da potenziali problemi associati ai sistemi di voto elettronico. La tecnologia blockchain è nata per superare questi problemi e offre nodi decentralizzati per il voto elettronico e viene utilizzata per produrre sistemi di voto elettronico principalmente a causa dei loro vantaggi di verifica end-to-end.

1.3 Sistema corrente

Rispetto ai metodi di voto convenzionali, il voto elettronico può migliorare sia l'efficienza che l'integrità del processo. A causa della sua flessibilità, semplicità d'uso e costo economico rispetto alle elezioni generali, il voto elettronico può essere ampiamente utilizzato in varie decisioni.

Nonostante ciò, i metodi di voto elettronico esistenti corrono il pericolo di un'autorità eccessiva e di dettagli manipolati, limitando l'equità fondamentale, la privacy, la segretezza, l'anonimato e la trasparenza nel processo di voto. Poiché le procedure di voto elettronico sono centralizzate, autorizzate dall'autorità critica, controllate, misurate e monitorate in un sistema di voto elettronico, è un problema in sé per un processo di voto trasparente. È quindi essenziale garantire che la sicurezza nel voto non diminuisca.

1.4 Attori

Il sistema prevede la presenza di due attori: l’organizzatore dell’elezione e l’elettore.

2. Sistema proposto

Il sistema proposto prevede di facilitare le operazioni, di cui una gran parte gestite manualmente, servendo una piattaforma intuitiva che agevoli l’organizzazione e la fruizione dei servizi riguardanti elezioni sia da parte dei votanti che da parte degli organizzatori.

2.1 Requisiti Funzionali

|  |  |  |
| --- | --- | --- |
|  | **Priorità** | |
| **Requisiti** | **Elettore** | **Organizzatore Elezione** |
| Aggiunta candidati | - | ALTA |
| Eliminazione candidati | - | ALTA |
| Apertura elezione | - | ALTA |
| Chiusura elezione | - | ALTA |
| Reset elezione | - | ALTA |
| Votazione | ALTA | - |
| Visualizzazione risultati dell’elezione | ALTA | - |
| Visualizzazione dei candidati | ALTA | ALTA |

* L'organizzatore dell'elezione deve poter aggiungere candidati;
* L'organizzatore dell'elezione deve poter eliminare candidati;
* L'organizzatore dell'elezione deve poter resettare l'elezione per poterne creare un'altra;
* L'organizzatore dell'elezione deve poter aprire le elezioni;
* L'organizzatore dell'elezione deve poter chiudere le elezioni;
* L'elettore deve poter votare una sola volta;
* L'elettore deve poter visualizzare i risultati dell'elezione;
* L’elettore e l’organizzatore devono poter visualizzare i candidati.

2.2 Requisiti Non Funzionali

2.2.1 Usabilità

* Il sistema dovrà essere semplice da utilizzare in modo da permettere all’utente di imparare ad utilizzare le funzionalità base in un tempo minore di 10 minuti;

2.2.2 Affidabilità

* Il sistema controllerà il formato dei dati inseriti nei form. In caso di dati mancanti o con formato non valido, l’utente visualizzerà appositi messaggi di errore;
* Almeno il 75% delle componenti del sistema deve essere testato e funzionante;

2.2.3 Prestazioni

* Il sistema deve garantire tempi di risposta per ogni funzionalità, complessivamente inferiori ai 10 secondi;

2.2.4 Manutenibilità

* Il sistema dovrà essere facilmente modificabile in modo efficace ed efficiente senza introdurre difetti.

2.2.5 Implementazione

L’implementazione del sistema prevede l’utilizzo di:

* Smart Contract, costituiti da tutte le regole e protocolli necessari per il voto elettronico;
* Blockchain Network, per implementare il Contratto. Per simulare la Blockchain Network è stato usato Ganache;
* Sito Web, creato con HTML, CSS (Bootstrap) e Javascript;
* Gestione dell'access control con openzeppelin;
* Lo standard di codifica dei caratteri sarà UTF-8.

2.2.6 Interfaccia

Il sistema è un’architettura Client-Server web, accessibile da dispositivi fissi e mobili.

2.2.7 Sicurezza

L’accesso al sistema per utilizzare le funzionalità in base alla tipologia di utente è protetto da autenticazione. In particolare, l’utente dovrà autenticarsi tramite Metamask.   
L’anonimato e l’integrità del voto sono rispettati grazie all’utilizzo della blockchain.

3. Modello del Sistema

In questo capitolo sono riportati **alcuni** scenari di alcune funzionalità.

3.1 Scenari

|  |  |
| --- | --- |
| **Nome scenario** | **Votazione** |
| **Istanze di Attori**  **Partecipanti** | Antonio: Elettore |
| **Flusso di eventi** | 1. Antonio accede al sistema per poter effettuare la votazione;  2. Antonio sceglie il candidato “Luca” e lo vota;  3. Antonio visualizza il nome del candidato votato quindi visualizza “Luca”. |

|  |  |
| --- | --- |
| **Nome scenario** | **Visualizzazione risultati dell’elezione** |
| **Istanze di Attori**  **Partecipanti** | Antonio: Elettore  Marello: Organizzatore dell’elezione |
| **Flusso di eventi** | 1. Marello ha appena chiuso l’elezione; 2. Antonio, venuto a conoscenza della chiusura dell’elezione, vuole visualizzare i risultati; 3. Antonio accede al sistema; 4. Antonio visualizza il risultato dell’elezione ed una lista dei candidati con annessi voti ciascuno; 5. Antonio può festeggiare poiché il suo candidato votato ha “Luca” ha vinto le elezioni; |

|  |  |
| --- | --- |
| **Nome scenario** | **Aggiunta Candidati** |
| **Istanze di Attori**  **Partecipanti** | Mattarella: Organizzatore dell’elezione |
| **Flusso di eventi** | 1. Mattarella è stato incaricato di gestire una votazione sulla piattaforma; 2. Mattarella accede al sistema; 3. Mattarella aggiunge il candidato “Matteo Renzi” appartenente al partito “IV” con l’annessa immagine di partito; 4. Mattarella aggiunge un altro candidato “Silvio Berlusconi” appartenente al partito “FDP” con l’annessa immagine di partito; |

|  |  |
| --- | --- |
| **Nome scenario** | **Reset Elezione** |
| **Istanze di Attori**  **Partecipanti** | Mattarella: Organizzatore dell’elezione |
| **Flusso di eventi** | 1. Mattarella è stato incaricato di gestire una votazione sulla piattaforma; 2. Mattarella accede al sistema; 3. Mattarella si accorge che un’elezione precedente era ancora presente nel sistema; 4. Mattarella con la funzione “Reset Elezione” rende possibile effettuare una nuova elezione; |

|  |  |
| --- | --- |
| **Nome scenario** | **Chiusura Elezione** |
| **Istanze di Attori**  **Partecipanti** | Mattarella: Organizzatore dell’elezione |
| **Flusso di eventi** | 1. Mattarella è stato incaricato di chiudere l’elezione precedentemente creata; 2. Mattarella accede al sistema; 3. Mattarella chiude l’elezione attuale con la funzione “Chiudi elezione”; 4. I votanti potranno visualizzare i risultati. |

|  |  |
| --- | --- |
| **Nome scenario** | **Apertura Elezione** |
| **Istanze di Attori**  **Partecipanti** | Mattarella: Organizzatore dell’elezione |
| **Flusso di eventi** | 1. Mattarella è stato incaricato di aprire l’elezione; 2. Mattarella accede al sistema; 3. Mattarella chiude l’elezione attuale con la funzione “Apri elezione”; 4. I votanti potranno effettuare la propria votazione. |

3.2 Use Cases

In questo capitolo sono riportati **alcuni** use cases di funzionalità più importanti.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nome del caso d’uso** | | | | | **Autore** | |
| Votazione | | | | | Matteo Della Rocca | |
| **Descrizione** | | | | Il seguente use case descrive il processo di votazione da parte di un elettore. | |
| **Attore principale** | | | | **Elettore**  Intende effettuare la votazione | |
| **Attori secondari** | | | | N/A | |
| **Entry condition** | | | | L’elettore appartiene alla rete (ha il permesso di votare) | |
| **Exit condition**  On success | | | | L’elettore vota | |
| **Exit condition**  On failure | | | | L’utente non vota | |
| **Rilevanza/User Priority** | | | | Alta | |
| **Frequenza stimata** | | | | 1/anno | |
| **Flusso di Eventi Principale** | | | | | |
| **1** | **Elettore:** | | Si trova nella home del sistema ed esegue il login; | | |
| **2** | **Sistema:** | | Mostra una sezione con la lista di candidati da votare; | | |
| **3** | **Elettore:** | | Sceglie il candidato e tramite apposito pulsante lo vota; | | |
| **4** | **Sistema:** | | Verifica che:   * L’elezione sia aperta; * Il candidato sia un candidato valido; * L’elettore ha il permesso di votare; * L’elettore non abbia già votato. | | |
| **5** | **Sistema:** | | Aggiorna il numero di voti del candidato selezionato dall’utente e mostra all’utente il candidato votato; | | |
| **I Flusso di eventi alternativo:** | | | | Non c’è un’elezione disponibile | |
| **2a.1** | | **Sistema:** | | Mostra una sezione apposita per elezione non disponibile; | |
| **II Flusso di eventi alternativo:** | | | | L’elettore non è abilitato al voto | |
| **4b.1** | | **Sistema:** | | Mostra un messaggio di errore all’utente. | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nome del caso d’uso** | | | | **Autore** | |
| Visualizzazione risultati dell’elezione | | | | Carmine D’Angelo | |
| **Descrizione** | | | Il seguente use case descrive il processo di visualizzazione dei risultati da parte di un elettore. | |
| **Attore principale** | | | **Elettore**  Intende visualizzare i risultati post-chiusura | |
| **Attori secondari** | | | N.A | |
| **Entry condition** | | | L’elezione è chiusa | |
| **Exit condition**  On success | | | L’elettore visualizza i risultati dell’elezione | |
| **Exit condition**  On failure | | | L’elettore non visualizza i risultati dell’elezione | |
| **Rilevanza/User Priority** | | | Alta | |
| **Frequenza stimata** | | | 1/anno | |
| **Flusso di Eventi Principale** | | | | |
| **1** | **Elettore:** | Si trova nella home del sistema ed esegue il login; | | |
| **2** | **Sistema:** | Mostra una sezione con la lista dei candidati ed i voti totali, inoltre mostra il risultato dell’elezione; | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nome del caso d’uso** | | | | **Autore** | |
| Aggiunta candidato | | | | Carmine D’Angelo | |
| **Descrizione** | | | Il seguente use case descrive il processo di inserimento di un candidato da parte di un organizzatore. | |
| **Attore principale** | | | **Organizzatore dell’elezione**  Intende aggiungere un candidato | |
| **Attori secondari** | | | N.A | |
| **Entry condition** | | | L’elezione è chiusa | |
| **Exit condition**  On success | | | Il candidato viene aggiunto alla lista dei candidati | |
| **Exit condition**  On failure | | | Il candidato non viene aggiunto alla lista dei candidati | |
| **Rilevanza/User Priority** | | | Alta | |
| **Frequenza stimata** | | | 4/mese | |
| **Flusso di Eventi Principale** | | | | |
| **1** | **Organizzatore dell’elezione:** | | Accede all’area di gestione | |
| **2** | **Organizzatore dell’elezione:** | | Utilizza la funzione “Aggiungi candidato” | |
| **3** | **Organizzatore dell’elezione:** | | Inserisce:   * Nome/Cognome candidato; * Partito; * Immagine partito;   e li invia; | |
| **4** | **Sistema:** | | Verifica che:   * Tutti i campi obbligatori sono stati compilati; * Tutti i campi siano validi. | |
| **5** | **Sistema:** | | Aggiunge il candidato alla lista e mostra un messaggio di successo | | |
| **I Flusso di eventi alternativo:** | | | Qualche campo obbligatorio non è stato compilato | | |
| **4a.1** | | **Sistema:** | Mostra un messaggio di errore all’organizzatore | | |
| **II Flusso di eventi alternativo:** | | | Alcuni campi non sono validi. | | |
| **4b.1** | | **Sistema:** | Mostra un messaggio di errore all’organizzatore indicando il campo errato. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome del caso d’uso** | | | **Autore** | |
| Chiusura elezione | | | Matteo Della Rocca | |
| **Descrizione** | | Il seguente use case descrive il processo di chiusura di un’elezione da parte di un organizzatore. | |
| **Attore principale** | | **Organizzatore dell’elezione**  Intende chiudere un’elezione per calcolarne il risultato | |
| **Attori secondari** | | N.A | |
| **Entry condition** | | L’elezione è aperta | |
| **Exit condition**  On success | | L’elezione viene chiusa | |
| **Exit condition**  On failure | | L’elezione non viene chiusa | |
| **Rilevanza/User Priority** | | Alta | |
| **Frequenza stimata** | | 4/mese | |
| **Flusso di Eventi Principale** | | | |
| **1** | **Organizzatore dell’elezione :** | Accede all’area di gestione | |
| **2** | **Organizzatore dell’elezione :** | Utilizza la funzione “Chiusura elezione” | |
| **3** | **Sistema:** | Mostra un messaggio di successo | | |
| **4** | **Sistema:** | Calcola il risultato dell’elezione e rende i risultati disponibili | | |