

VOTO E SCRUTINIO ELETTRONICO

Alessandro Fabio Farè - 910928

Il progetto consiste nella realizzazione di un sistema di voto e scrutinio elettronico.

Concretamente si tratta di un applicativo in grado di automatizzare due principali funzioni:

1) Gestione della sessione di voto

E' possibile configurare una sessione di voto, scegliendo se si tratti di elezioni (quindi votazione a candidati e/o partiti) o di referendum.

In base alla scelta fatta è possibile in seguito specificare la modalità di votazione e di vincita, inserire le informazioni relative a candidati e/o partiti o le proposte di legge con eventuale quorum, e infine far partire la fase di votazione.

2) Votazione

Una volta configurata la sessione di voto l'applicazione dà la possibilità agli elettori di votare.

L'elettore deve inizialmente registrarsi.

- Se il voto viene espresso a distanza, l'elettore dovrà svolgere una fase di autenticazione, in cui verrà generato automaticamente il campo relativo al suo codice fiscale, identificato in base ai dati inseriti, attraverso un sistema implementato nell'applicativo, e potrà passare alla fase di votazione solo in caso in cui la verifica del diritto al voto darà esito positivo.
In seguito, potrà decidere per quale scheda elettorale votare e infine concludere la fase di valutazione votando secondo le sue preferenze.
- Se il voto viene espresso di persona in un seggio elettorale, la fase di identificazione e verifica del diritto al voto saranno eseguiti manualmente, e l'elettore passerà direttamente alla fase di votazione.

Infine sarà possibile visionare l'andamento delle votazioni e visualizzare l'estrazione del vincitore attraverso un sistema di scrutinio, che si basa sulle modalità di voto e vincita.

1 Descrizione del problema

1.1 Obiettivi

- Fornire uno strumento di voto e scrutinio elettronico che possa essere familiare a qualsiasi persona, utilizzando uno stile minimale in modo da garantire chiarezza e facilità d'uso
- Garantire sicurezza di autenticazione e registrazione e totale segretezza del voto espresso dall'elettore

- Permettere una gestione similare per progetti diversi
- Consentire implementazioni alternative o aggiunta di nuove funzionalità grazie all'uso di alta coesione e lasco accoppiamento a livello implementativo

1.2 Analisi e specifica dei requisiti

Requisiti Funzionali

1.2.1 Requisiti Utente

- L'utente deve essere identificato, perché può essere un Elettore, che vota a distanza o di persona, oppure il gestore del sistema (o Admin)
- L'elettore può effettuare la votazione sia a distanza che in presenza
Se a distanza deve svolgere la fase di autenticazione prima di poter votare
- L'elettore può votare se è almeno maggiorenne
- L'elettore può votare una volta sola
- Come Elettore devo poter effettuare la votazione secondo una delle seguenti modalità:
 - Voto Ordinale
 - Voto Categorico
 - Voto Categorico con Preferenza
 - Referendum
- Come Admin, oltre a dover configurare la sessione di voto con le modalità descritte precedentemente, devo poter scegliere le modalità di vincita che si distinguono in base alle modalità di voto scelte.
Se ho scelto la modalità di Voto ordinale, Categorico o Categorico con Preferenza, le modalità di vincita configurabili possono essere:
 - Maggioranza
 - Maggioranza Assoluta

Altrimenti, nel caso di scelta del Referendum, le modalità di vincita configurabili possono essere:

- Referendum con Quorum
 - Referendum senza Quorum
- Come Admin, se durante la creazione della sessione di voto ho scelto la modalità Referendum, dovrò poter inserire la proposta di legge, la modalità di vincita ed eventualmente il quorum
- Come Admin, se durante la creazione della sessione di voto ho scelto la modalità di Voto Ordinale, Categorico o Categorico con Preferenza, dovrò poter inserire le liste di partiti, di candidati e la modalità di voto e di vincita

1.2.2 Requisiti di Sistema

- Il sistema deve tener conto di due tipologie di utente: Elettore e Admin
- Il sistema deve permettere all'utente di potersi registrare e autenticare, qualora la votazione avvenisse a distanza, tramite dispositivo elettronico
La fase di identificazione non verrà attuata nel caso in cui l'Elettore si presenti all'urna, in quanto si assume che l'identità di quest'ultimo venga controllata dallo Scrutinatore
- Il sistema deve garantire a ogni tipo di utente la possibilità di svolgere le operazioni a lui associate, evitando che un qualsiasi elettore possa accedere alle funzionalità di gestione della sessione di voto
 - Se Elettore:
 - Il sistema deve garantire di poter effettuare la votazione secondo le modalità espresse
 - Il sistema deve garantire la registrazione delle informazioni relative all'Elettore, sia in fase di registrazione che di eventuale autenticazione
 - Il sistema deve garantire la verifica del diritto al voto
 - Il sistema deve garantire la generazione del codice fiscale in automatico in base alle informazioni ricevute dall'Elettore
 - Il sistema deve garantire la segretezza del voto espresso dall'Elettore
 - Il sistema deve garantire all'Elettore la possibilità di potersi astenere dalla votazione, qualora presentasse Scheda Bianca
 - Se Admin:
 - Il sistema deve garantire all'Admin, dopo la fase di registrazione, la creazione della sessione di voto in base alle modalità di voto scelte
- Il sistema deve calcolare il conteggio dei voti visualizzando l'esito finale della scheda elettorale

Requisiti non Funzionali

1.2.3 Requisiti di Sicurezza

- Nessuno deve poter risalire all'Elettore dalla scheda elettorale
- Il voto dell'Elettore deve restare segreto
- Il sistema deve fornire un sistema di auditing in modo da tener traccia di tutte le operazioni eseguite dagli utenti e degli esiti finali di votazione

1.2.4 Requisiti Implementativi

- L'implementazione del sistema utilizza il linguaggio Java
- L'interfaccia del sistema è stata implementata tramite la libreria grafica Swing di Java

1.2.5 Requisiti Normativi

- L'Elettore deve poter votare una volta sola e solo se maggiorenne

1.3 Glossario

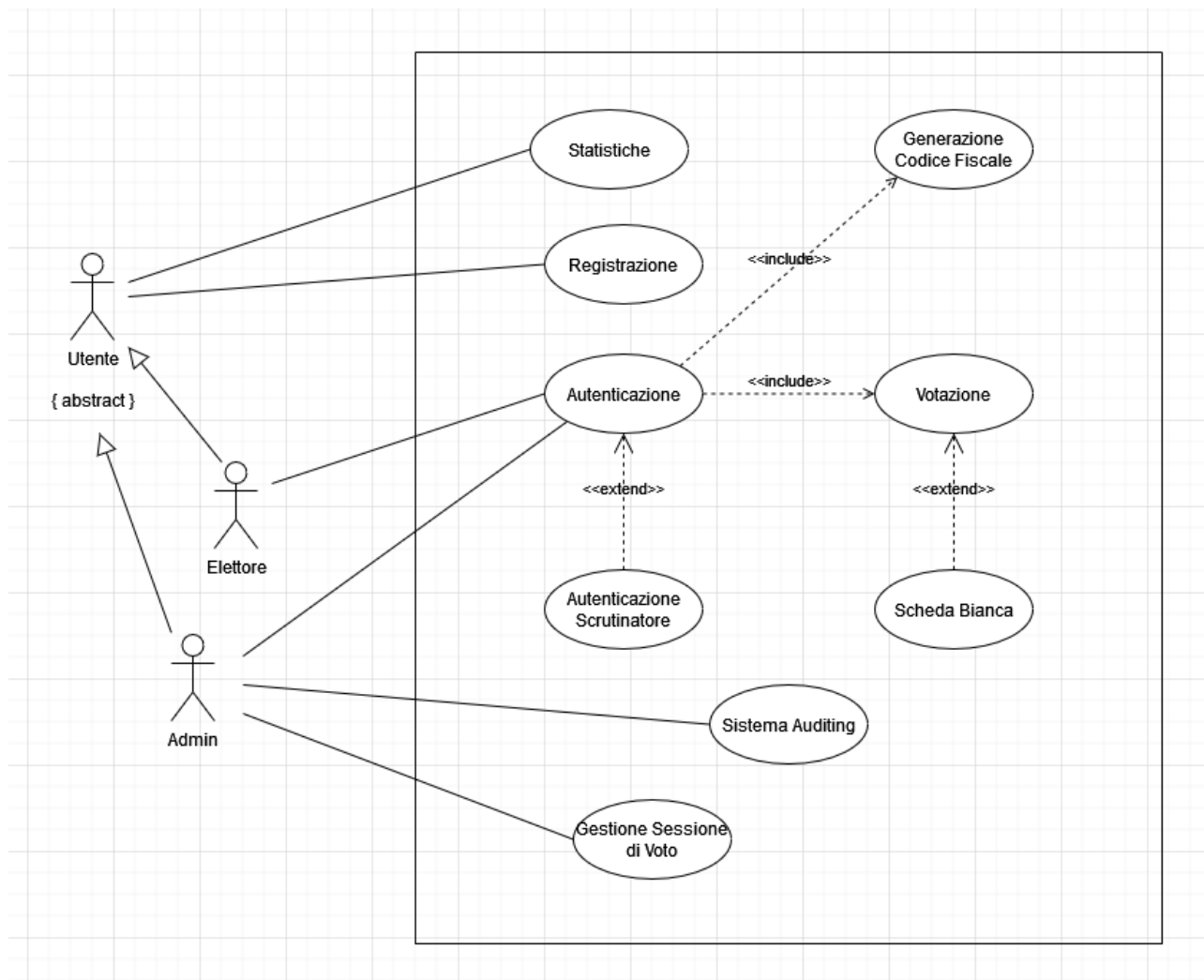
TERMINE	DESCRIZIONE
<i>Elettore</i>	Cittadino che ha o esercita il diritto di voto
<i>Admin</i>	Individuo che è preposto all'identificazione degli elettori, in caso di votazione in seggio elettorale, e alla gestione del sistema
<i>Seggio Elettorale</i>	Luogo fisico dove si può votare
<i>Partito</i>	Un partito politico è un'associazione tra persone accomunate da una stessa visione, identità o finalità politica di interesse pubblico, relativa alla gestione dello Stato e della società o su temi specifici e particolari
<i>Referendum</i>	Proposta di legge fatta agli elettori per decidere se si è favorevoli o contrari a un particolare quesito dato
<i>Maggioranza Assoluta</i>	Il vincitore ottiene almeno il 50% + 1 dei voti che sono stati dati
<i>Maggioranza Base</i>	Il vincitore è il candidato che ha ottenuto il maggior numero di voti
<i>Quorum</i>	Numero di elettori necessario affinché una votazione sia valida
<i>Voto Categorico</i>	L'elettore inserisce una preferenza per un candidato o partito
<i>Voto Categorico con Preferenza</i>	E' come il voto categorico ma è possibile inserire preferenze tra i candidati di un partito
<i>Voto Ordinale</i>	L'Elettore deve specificare un ordine di preferenza dei vari partiti o candidati alle elezioni

2 Progettazione del Sistema

Nota: Nel caso i diagrammi non fossero sufficientemente chiari all'interno del documento, si possono trovare le immagini ad alta risoluzione relative ai diagrammi e alle GUI all'interno delle relative cartelle, images_diagram e images_gui

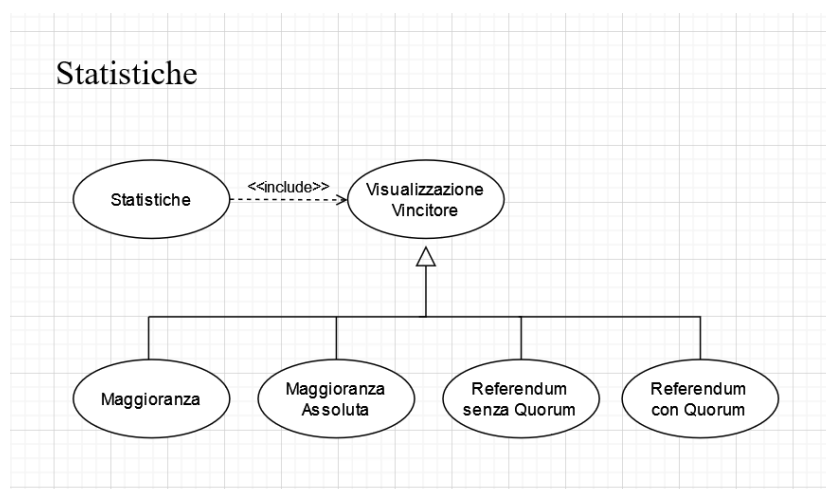
2.1 Diagramma dei casi d'uso

I sorgenti ad alta risoluzione si trovano nella cartella use case all'interno della cartella images_diagram.



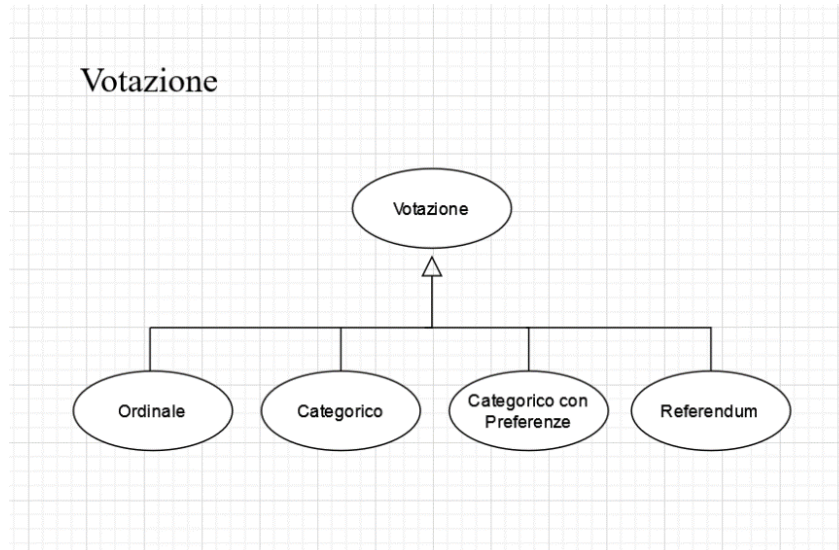
Di seguito una breve descrizione per ciascun scenario, con alcune specificazioni:

- **Statistiche:** l'utente, che sia Elettore o Admin, può visualizzare l'esito delle schede elettorali attraverso la pagina di Statistiche, raggiungibile dal menù principale

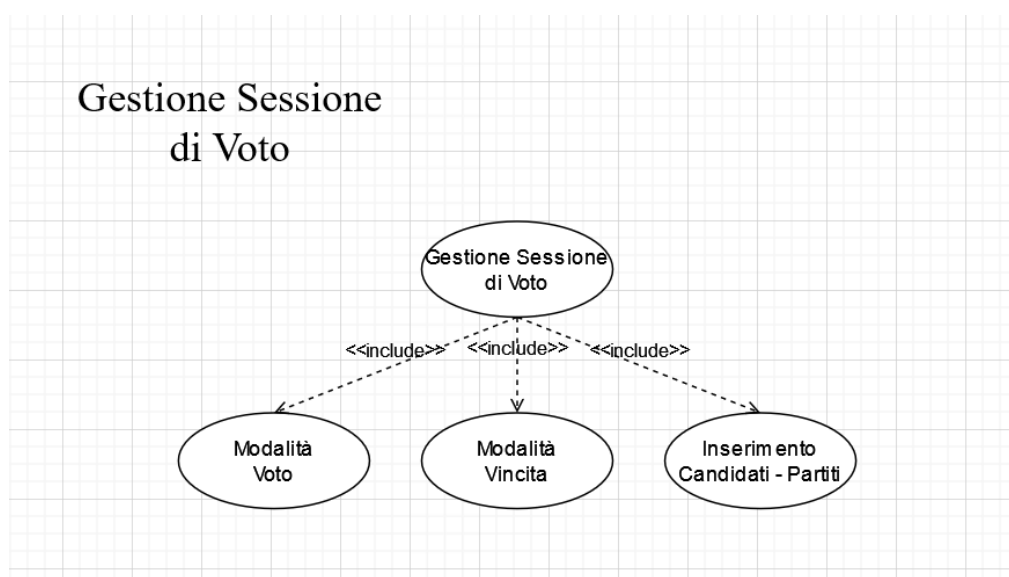


- **Registrazione:** l'utente deve registrarsi al sistema prima di poter svolgere la sessione di votazione, in caso di Elettore, o di gestione del sistema, in caso di Admin.
In questa fase, una volta inseriti username e password, il sistema riconoscerà se l'utente è l'Admin o un Elettore, e saprà inoltre se la votazione avverrà a distanza o in presenza.
Nel caso in cui si fosse registrato l'Admin, il sistema reindirizzerà l'utente alla fase di gestione del sistema.
Nel caso in cui si fosse registrato un Elettore che voterà in presenza, sarà reindirizzato subito alla fase di Votazione.
Nel caso in cui si fosse registrato un Elettore che voterà a distanza, sarà reindirizzato alla fase di Autenticazione.
- **Autenticazione:** l'elettore deve autenticarsi al sistema inserendo le proprie informazioni relative a nome, cognome, sesso, nascita, città e stato.
Una volta inserite queste informazioni, il sistema genererà il codice fiscale dell'Elettore e registrerà quest'ultimo.
Se l'Elettore è maggiorenne e non ha ancora votato, sarà reindirizzato alla fase di Votazione.
Opzionalmente, se la votazione sarà in presenza, si assume che questa fase di identificazione venga fatta dallo Scrutinatore, e quindi l'elettore sarà reindirizzato subito alla fase di Votazione, senza passare da quella di Autenticazione.

- **Votazione:** l'elettore potrà votare sul tipo di scheda elettorale scelta (Elezione o Referendum).
In base alle modalità di voto sarà possibile votare candidati o partiti, inserire un ordine di preferenza per candidati o partiti o votare la proposta di legge.
Opzionalmente, l'elettore potrà non mettere alcun voto e presentare Scheda Bianca.



- **Sistema Auditing:** l'Admin, attraverso un sistema di log, verifica il corretto funzionamento del programma, scrivendo tutte le operazioni eseguite dall'Admin stesso e dagli Elettori su un file di estensione txt.
- **Gestione Sessione di Voto:** l'Admin, dopo la fase di registrazione, deve configurare la sessione di voto, scegliendo la modalità di votazione della scheda elettorale, e in base a ciò inserire candidati, partiti, modalità di vincita o proposte di legge ed eventuale quorum.



2.2 Descrizione degli scenari

Nome	Statistiche
Scopo	Permettere all'utente di visualizzare gli esiti finali delle schede elettorali
Attore/i	Utente
Pre-condizioni	L'utente deve essere presente all'interno del sistema sulla pagina di Avvio
Trigger	L'utente vuole visualizzare l'andamento delle votazioni
Descrizione sequenza eventi	<ol style="list-style-type: none">1. L'Utente clicca il bottone relativo alle Statistiche2. L'Utente sceglie la scheda elettorale di cui vuole visualizzare gli esiti3. Vengono visualizzati all'utente gli esiti finali di votazione e il relativo vincitore candidato, con il numero di voti ottenuti e l'appartenenza a un dato partito, o l'esito della proposta di legge
Alternativa/e	3.a Non c'è ancora un vincitore
Post-condizioni	L'utente viene reindirizzato alla pagina di Avvio

Nome	Registrazione
Scopo	Permettere all'utente di entrare nel sistema ed effettuare le operazioni a lui consentite
Attore/i	Utente
Pre-condizioni	L'utente deve essere presente all'interno del sistema sulla pagina di Registrazione
Trigger	L'utente vuole accedere al sistema
Descrizione sequenza eventi	<ol style="list-style-type: none">1. L'Utente clicca il bottone relativo alla Registrazione2. L'Utente inserisce le credenziali (username e password)3. L'Utente se vuole gestire il sistema seleziona la checkbox di Admin e inserisce le credenziali esatte dell'Admin4. L'Utente se vuole votare a distanza seleziona la checkbox relativa alla votazione a distanza5. L'Utente clicca sul bottone di conferma
Alternativa/e	3.a L'Utente è già registrato
Post-condizioni	L'utente viene registrato e reindirizzato alla pagina di Autenticazione se è un Elettore che vuole votare a distanza, o alla pagina di Votazione se è un elettore che vuole votare in presenza, o alla pagina di Gestione del sistema se è Admin

Nome	Autenticazione
Scopo	Registrare l'elettore, generare il codice fiscale e farlo votare
Attore/i	Elettore, Admin
Pre-condizioni	L'elettore deve essere registrato
Trigger	L'elettore ha passato la fase di registrazione
Descrizione sequenza eventi	<ol style="list-style-type: none"> 1. L'Elettore inserisce i suoi dati riguardo nome, cognome, sesso, nascita, città e stato. 2. L'Admin registra l'elettore 3. Viene generato il codice fiscale dell'elettore in base alle informazioni ricevute 4. L'Elettore clicca sul bottone conferma e potrà votare
Alternativa/e	<ol style="list-style-type: none"> 1.a I dati inseriti non sono corretti 4.b L'elettore non è maggiorenne o ha già votato e sarà reindirizzato alla pagina di Avvio
Post-condizioni	L'utente viene reindirizzato alla pagina di Votazione

Nome	Votazione
Scopo	Permettere all'elettore di poter votare la scheda elettorale scelta
Attore/i	Elettore, Admin
Pre-condizioni	L'elettore deve essere autenticato, o dal sistema o dallo scrutatore
Trigger	L'utente ha passato la fase di autenticazione e vuole votare
Descrizione sequenza eventi	<ol style="list-style-type: none"> 1. L'elettore sceglie se votare per il referendum o per candidati e/o partiti 2. L'elettore entra all'interno della sezione in cui avviene la votazione 3. L'elettore sceglie all'interno della scheda le sue preferenze 4. I dati delle preferenze saranno aggiornati dall'Admin che modificherà il numero di voti delle schede elettorali, senza porre alcun riferimento all'Elettore 5. L'Elettore conferma il voto 6. L'Elettore sarà reindirizzato alla pagina di Avvio
Alternativa/e	<ol style="list-style-type: none"> 3.a L'elettore non esprime alcun voto, scegliendo Scheda Bianca 4.a I dati inseriti non sono corretti
Post-condizioni	L'utente viene reindirizzato alla pagina di Avvio

Nome	Gestione Sessione di Voto
Scopo	Permettere all'Admin di creare e avviare una sessione di voto
Attore/i	Admin
Pre-condizioni	L'Admin deve registrarsi con le credenziali giuste
Trigger	Gestire una sessione di voto
Descrizione sequenza eventi	<ol style="list-style-type: none"> 1. L'Admin sceglie se inserire una scheda elettorale riguardo il referendum o su candidati e/o partiti 2. L'Admin inserisce candidati e/o partiti, modalità di vincita e voto, o proposte di legge ed eventuale quorum 3. L'Admin registra tutti i dati 4. L'Admin avvia la sessione di voto
Alternativa/e	3.a I dati sono già stati inseriti
Post-condizioni	Viene creata la sessione di voto e l'Admin viene reindirizzato alla pagina di Avvio

2.3 Diagramma delle classi

E' stato implementato un diagramma delle classi unico che comprende le associazioni tra i vari packages del sistema, ciascuno contenente le proprie classi, e un diagramma delle classi per il package delle views della GUI, in modo da avere più chiarezza delle classi implementate.

- Il sistema è stato implementato attraverso il design pattern MVC, di cui sono riportati i packages relativi.
- E' inoltre presente il package Document, all'interno del quale si trovano le classi CitiesCodes e StatesCodes. Queste classi sono servite per poter generare il codice fiscale dell'Elettore in fase di autenticazione in base alla città di nascita e allo stato di provenienza.
Sono stati infatti implementati degli HashMap mappando i codici e la relativa città o stato, in modo che ad esempio quando l'elettore sceglie una determinata città automaticamente il sistema ricava il codice relativo, inserendolo poi nell'ordine corretto all'interno della stringa del codice fiscale.
- C'è un ulteriore package, Connection, all'interno del quale si trova la classe ConnectionFactory utilizzata per potersi connettere al Database.
- Infine troviamo il package Utils, all'interno del quale si trovano classi statiche utilizzate in diversi punti del progetto.
Le classi DesignUtils e RoundJTextField sono state usate per customizzare gli elementi grafici di Java Swing, in particolare i JButton, JtextField, JLabel e JPanel.
La classe Utils è invece usata per gestire alcuni dati inseriti dall'Elettore.
La classe Logger è utilizzata per il sistema di auditing.

DIAGRAMMA CLASSI UNICO

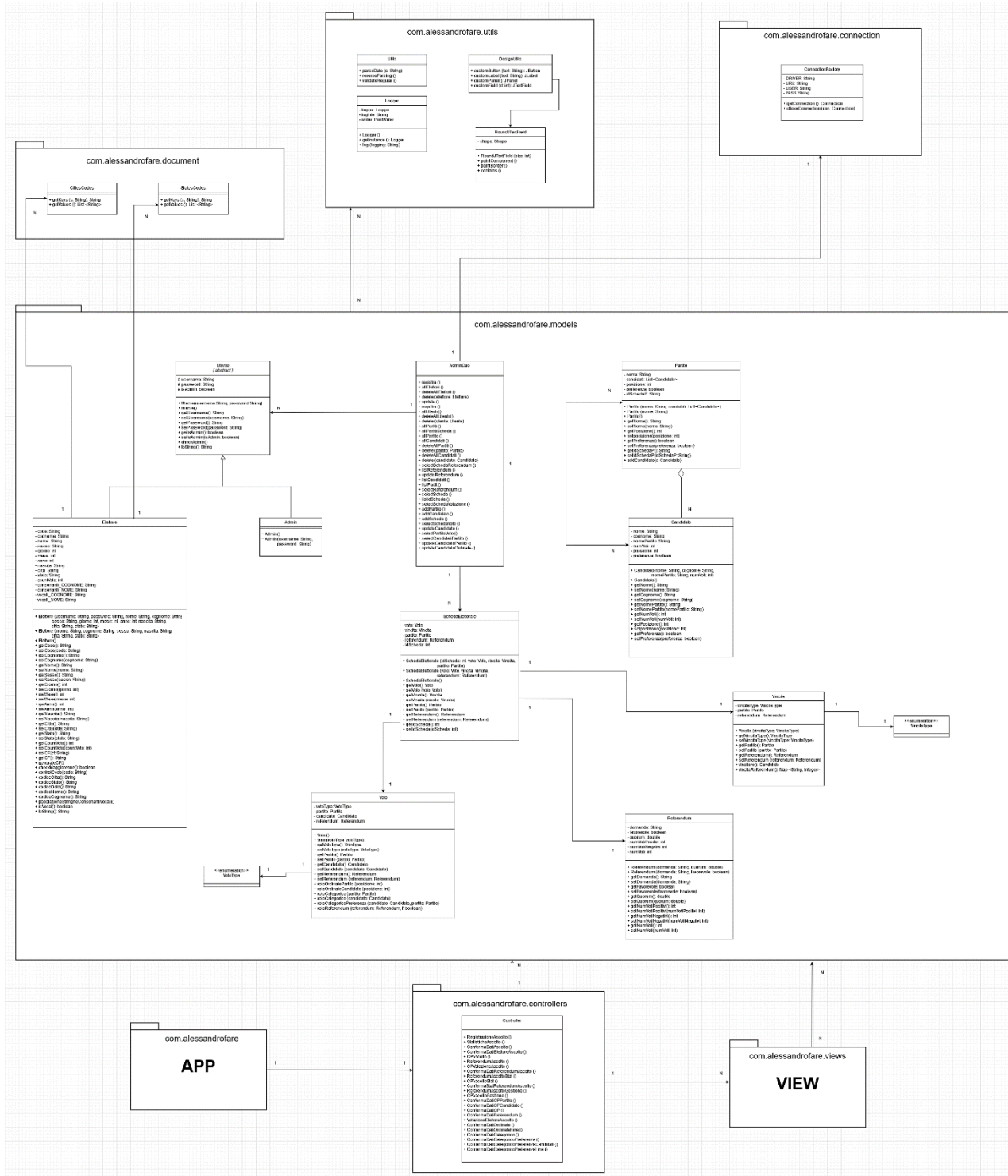
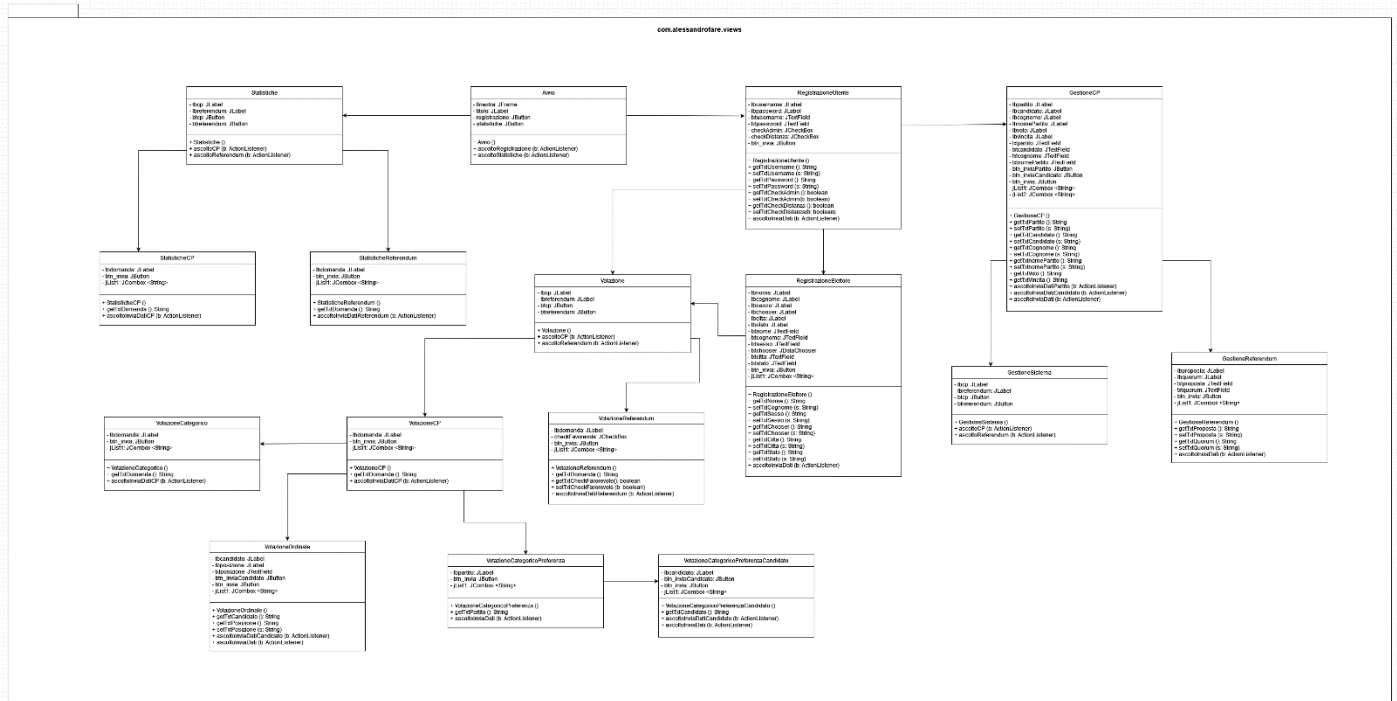


DIAGRAMMA CLASSI VIEWS



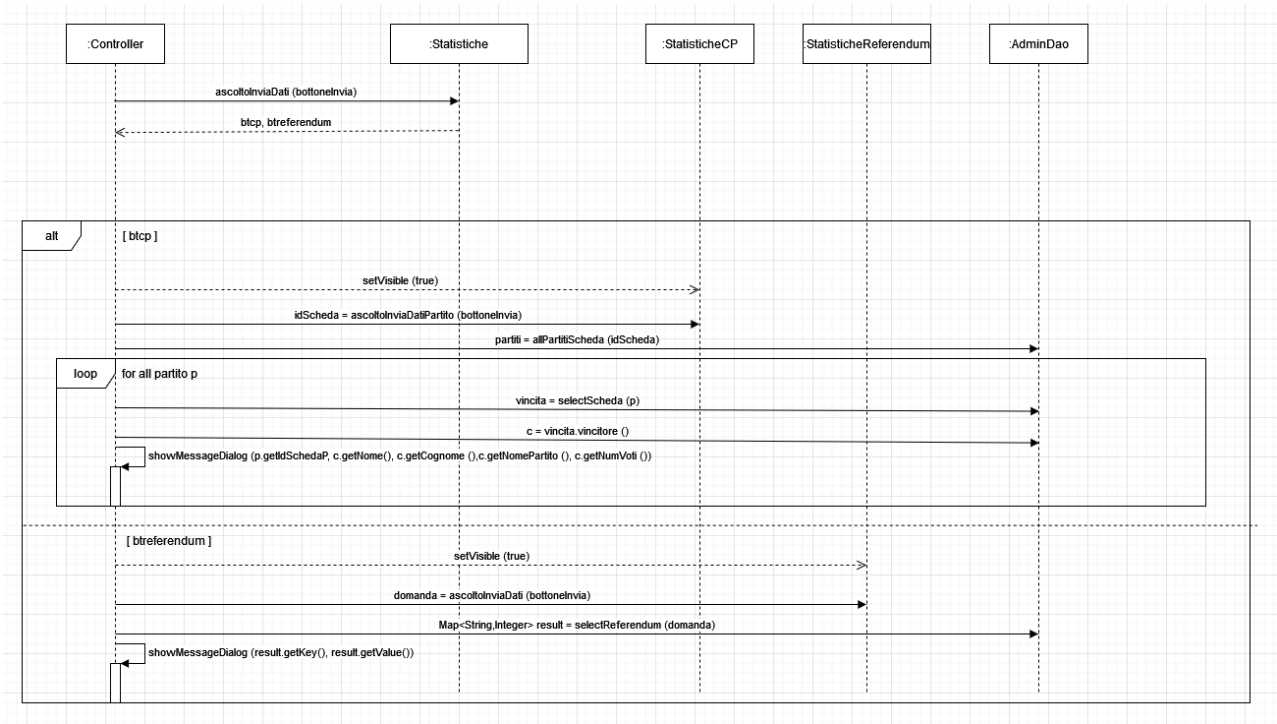
Il sistema parte dalla schermata di Avvio e può seguire diverse strade in base alle selezioni dell'Utente.

Sarà mostrata successivamente una navigation map in modo da spiegare meglio la navigazione.

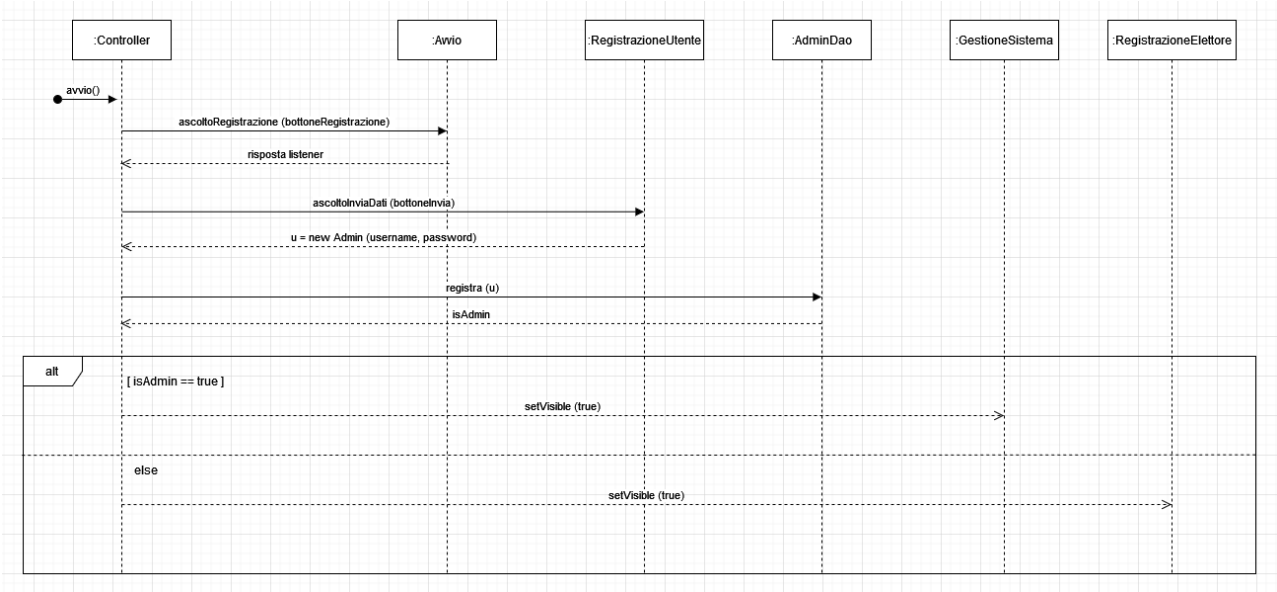
Per una maggiore chiarezza i sorgenti relativi ai diagrammi delle classi si trovano nella cartella class diagram contenuta nella cartella images diagram.

2.4 Diagrammi di sequenza

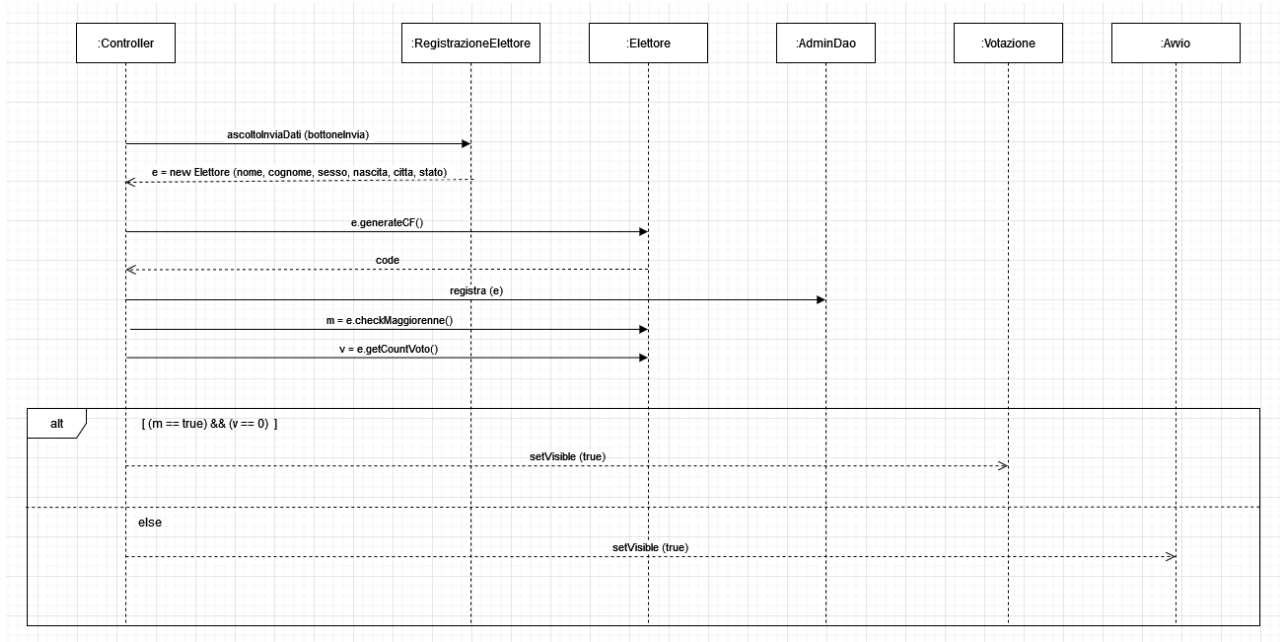
2.4.1 Statistiche



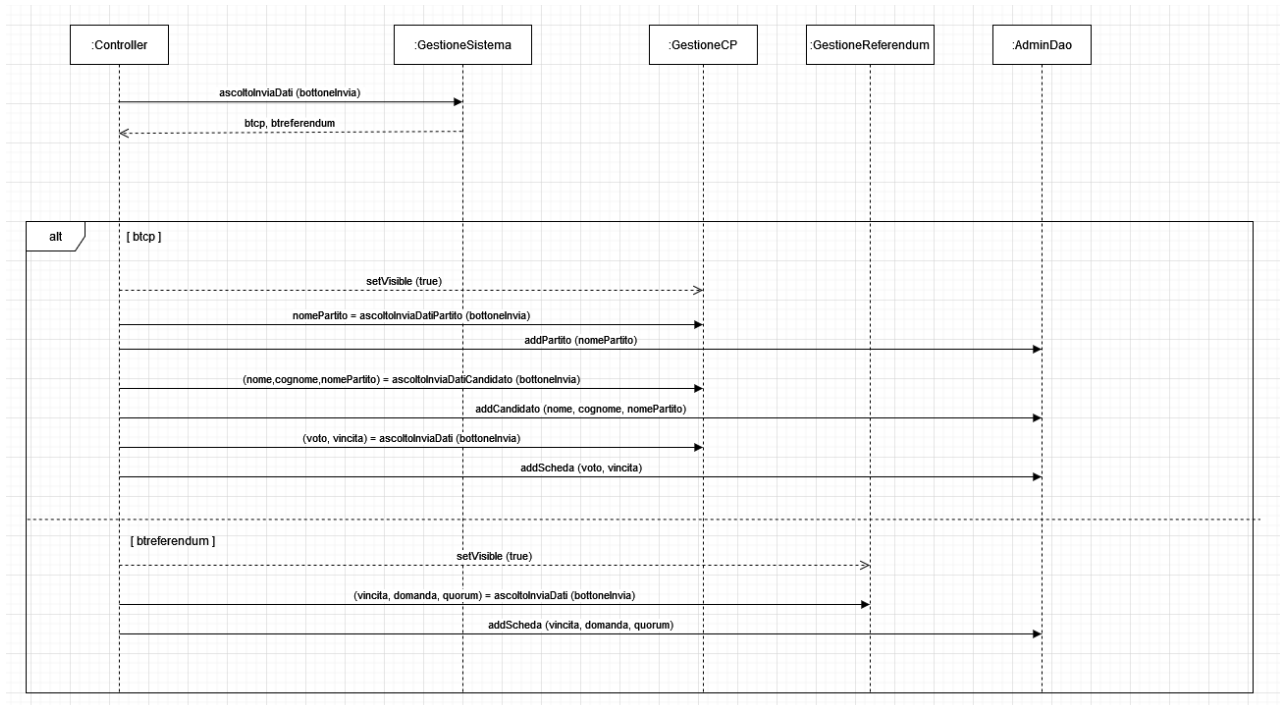
2.4.2 Registrazione



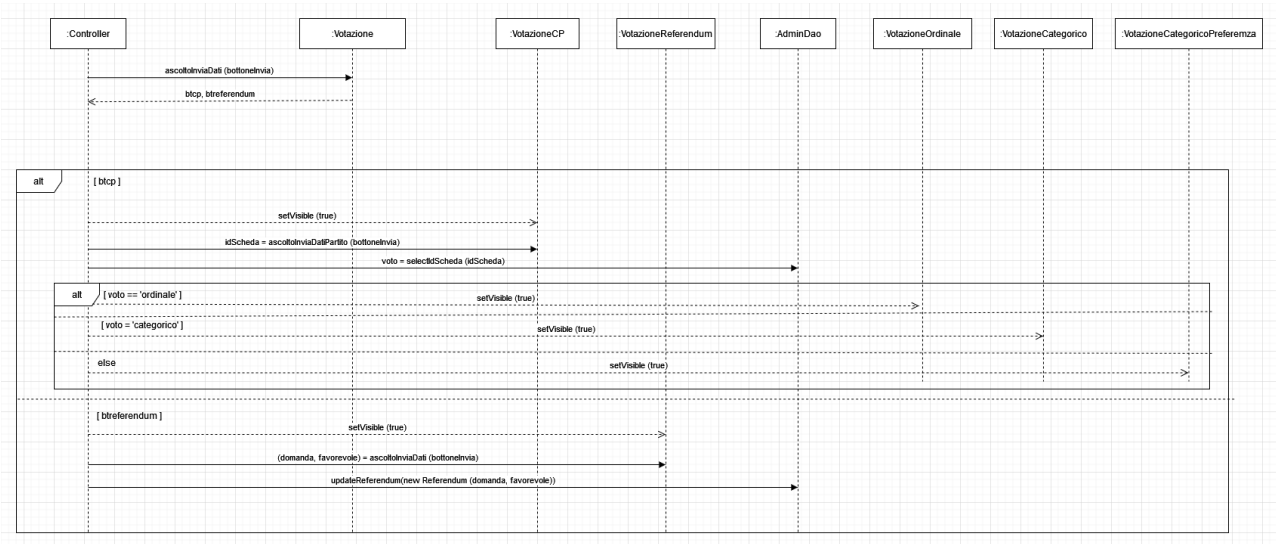
2.4.3 Autenticazione



2.4.4 Gestione Sistema di Voto



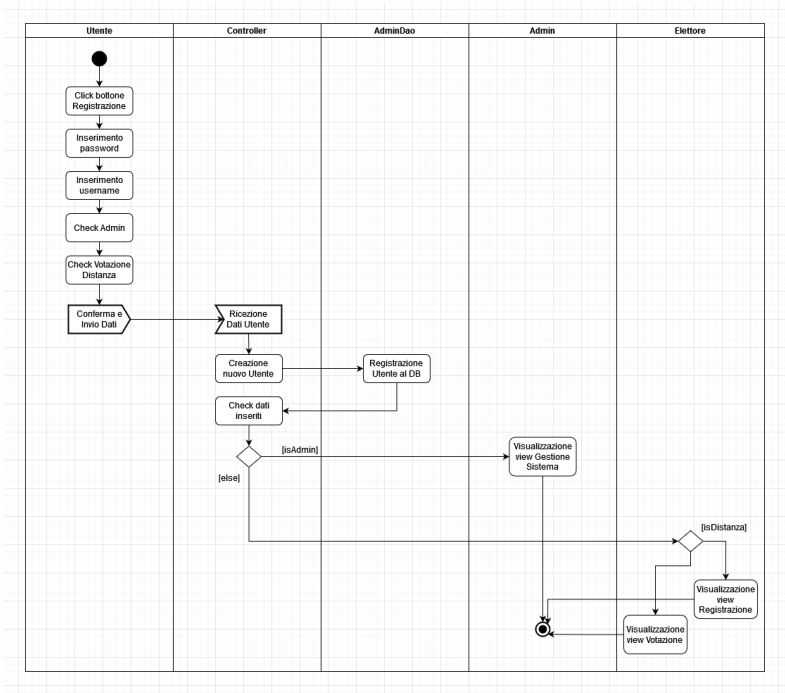
2.4.5 Votazione



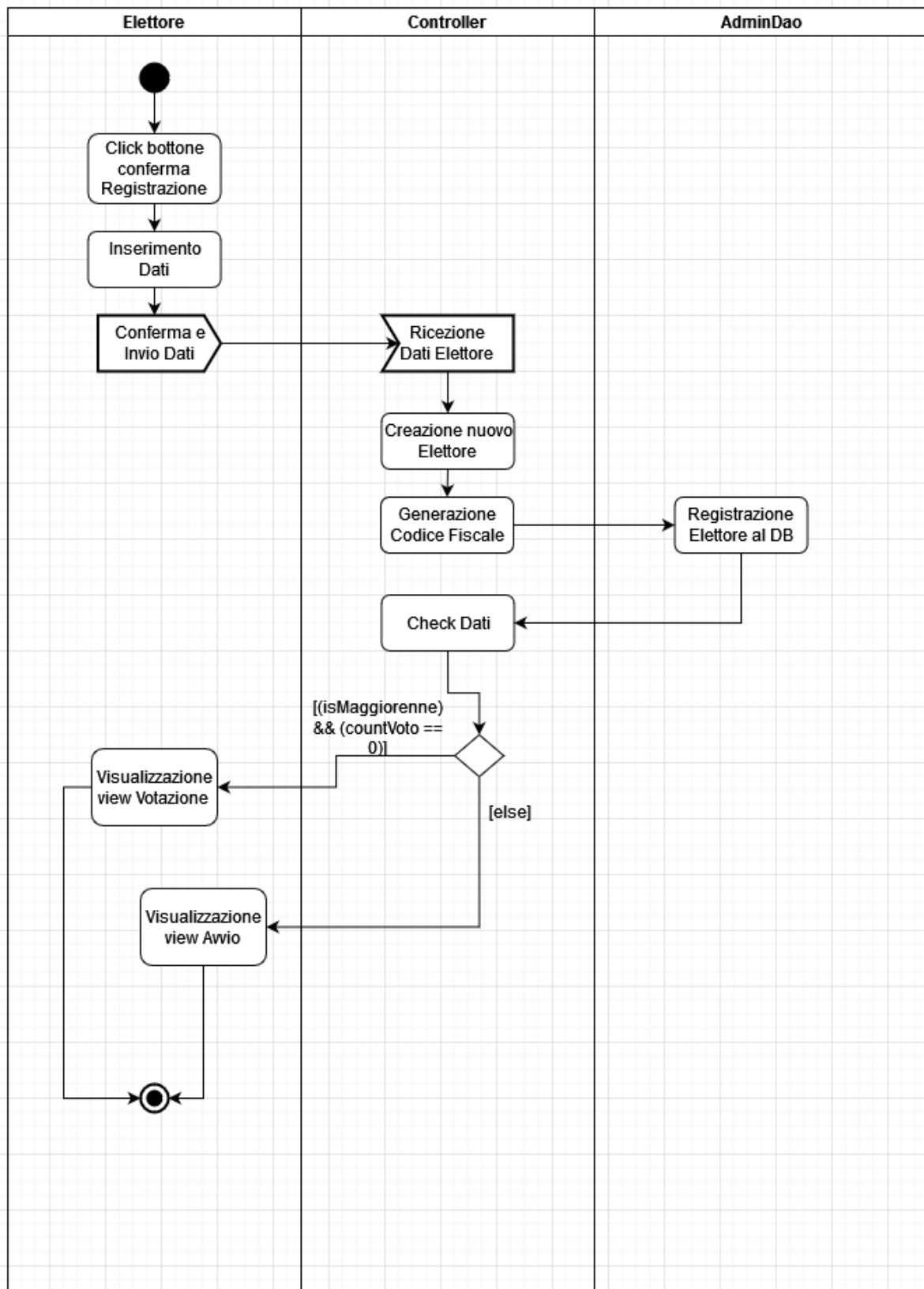
I sorgenti ad alta risoluzione si trovano nella cartella sequence diagram all’interno della cartella images diagram.

2.5 Diagrammi delle attività

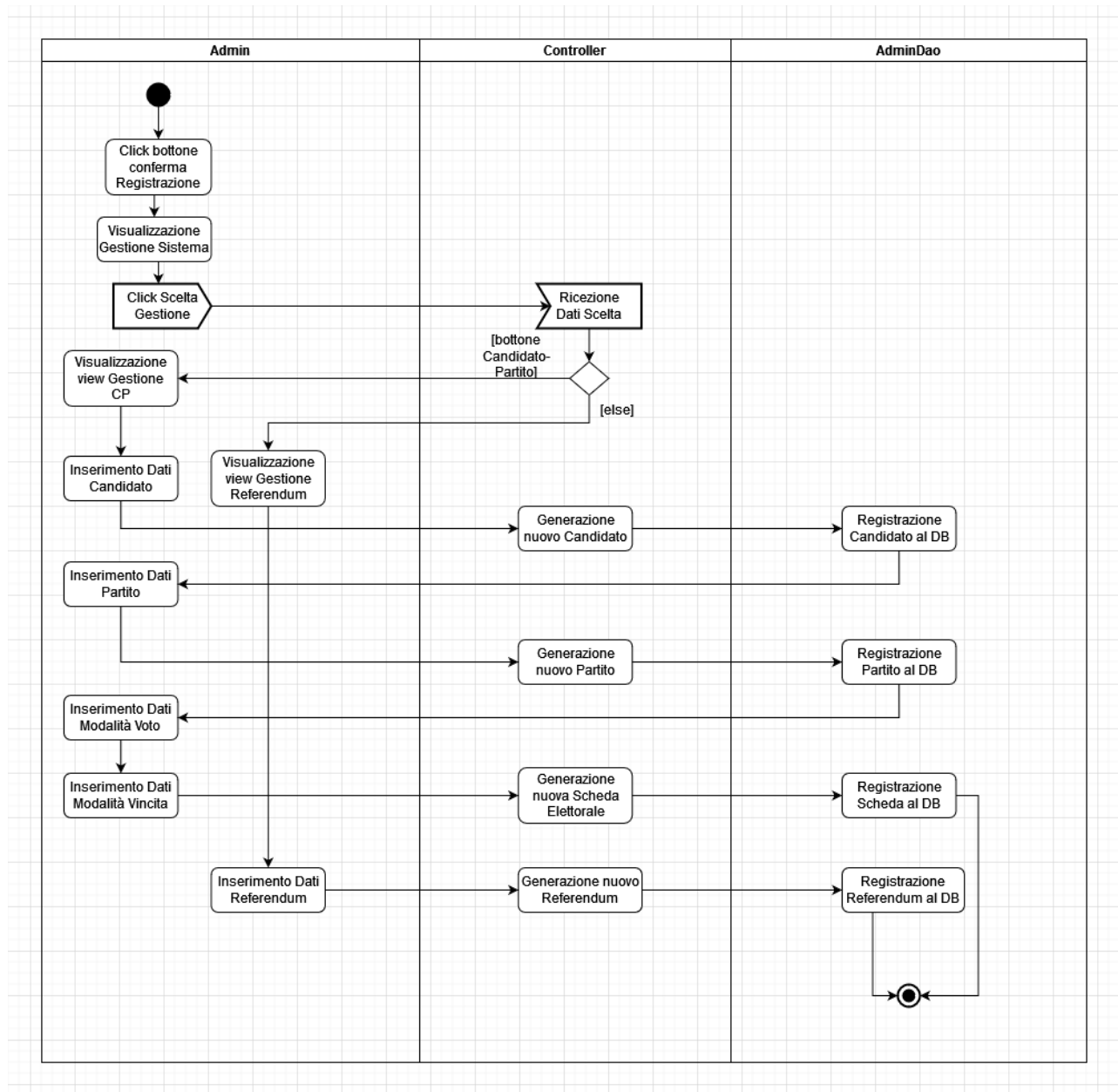
2.5.1 Registrazione



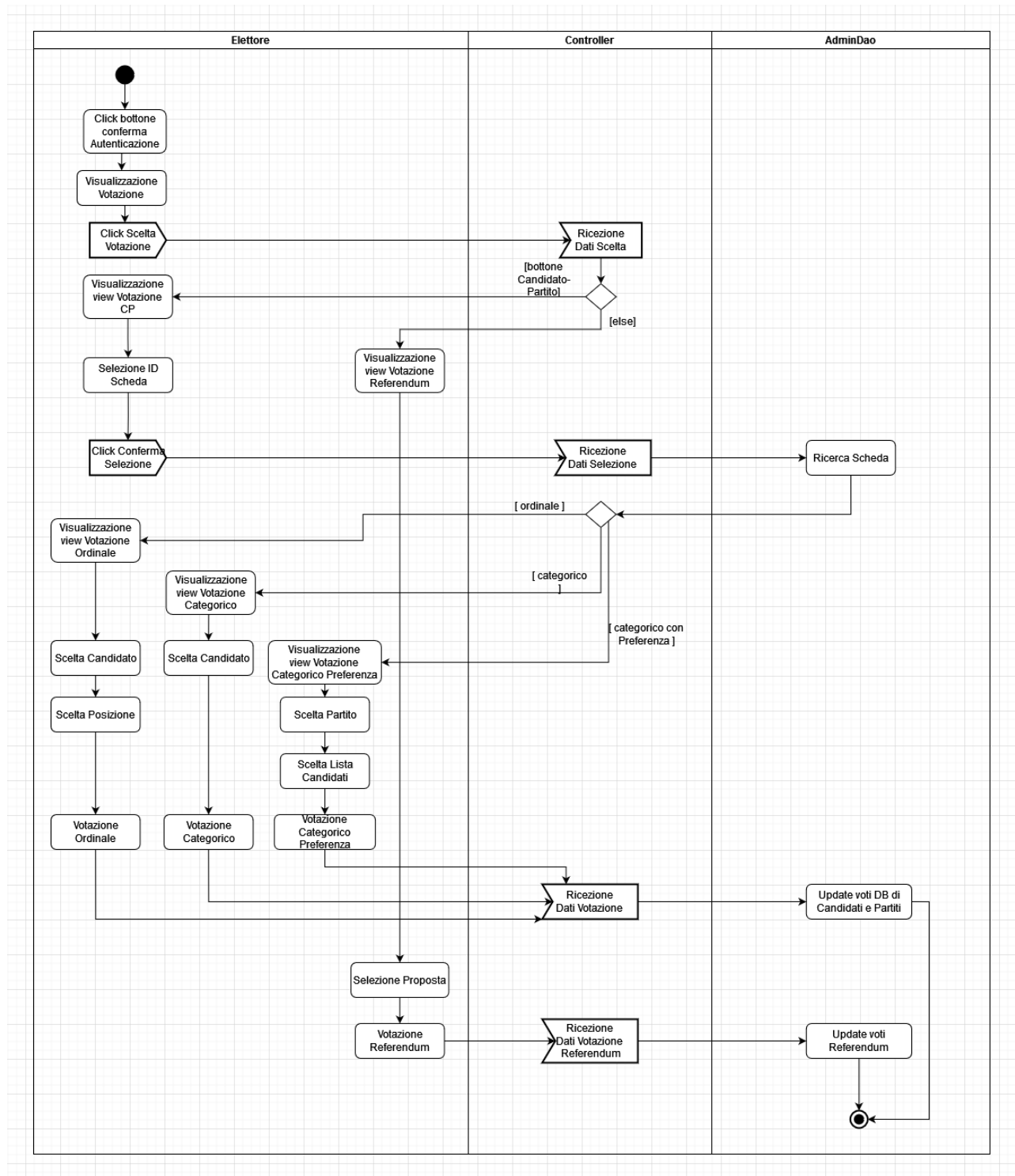
2.5.2 Autenticazione



2.5.3 Gestione Sessione di Voto



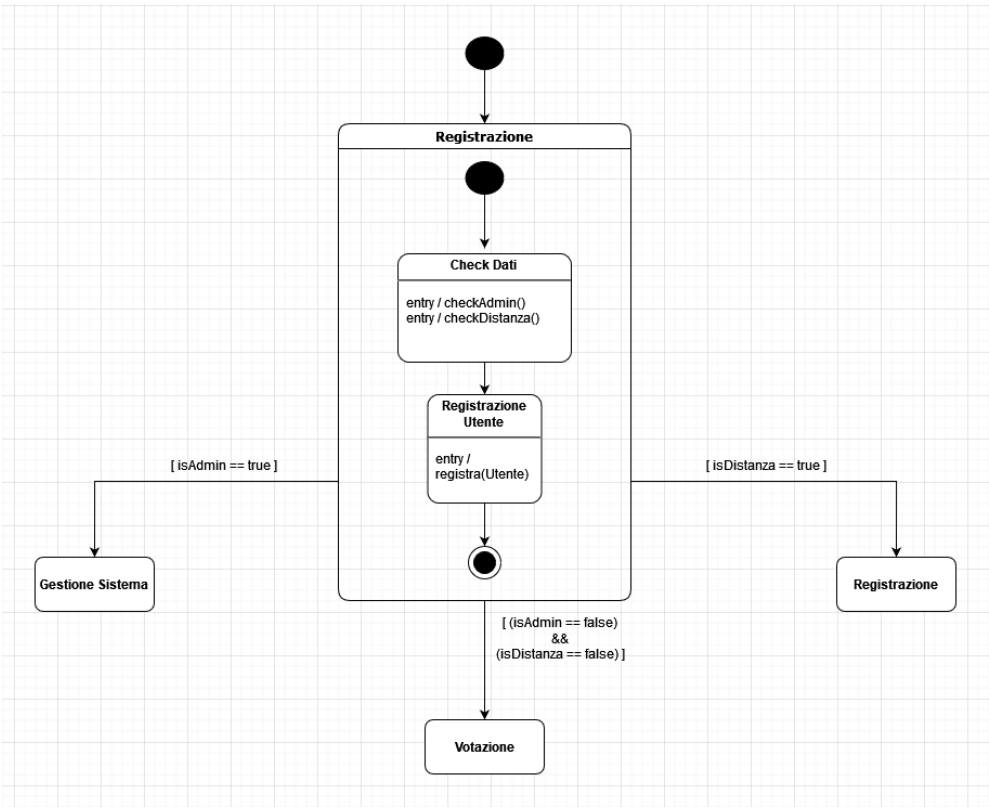
2.5.4 Votazione



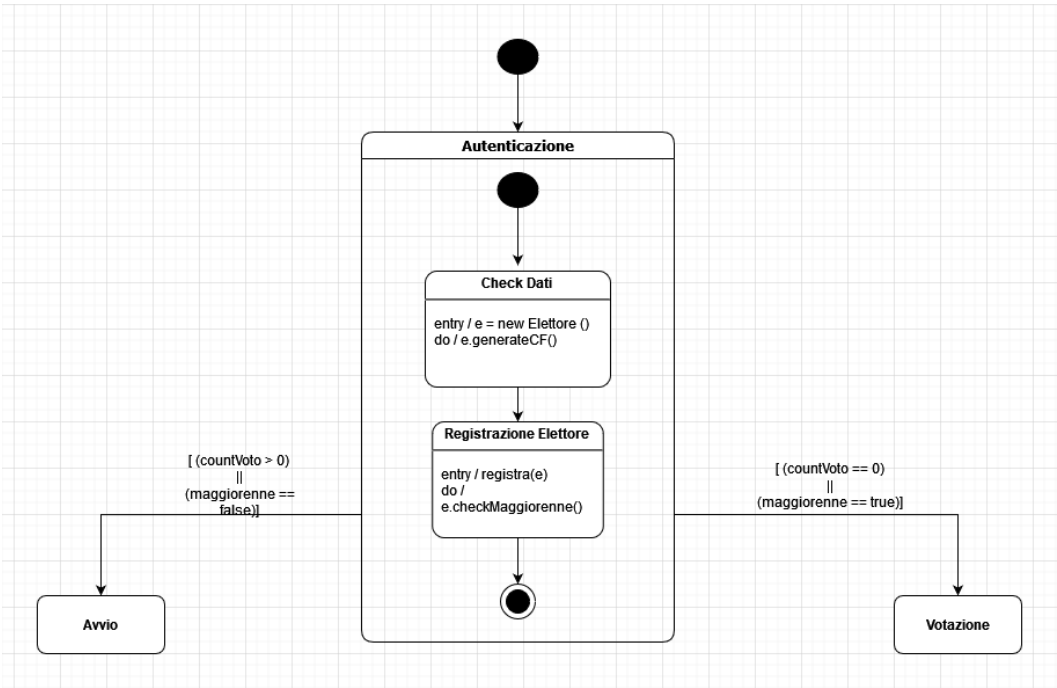
I sorgenti ad alta risoluzione si trovano nella cartella activity diagram all'interno della cartella images diagram.

2.6 Macchine di stato

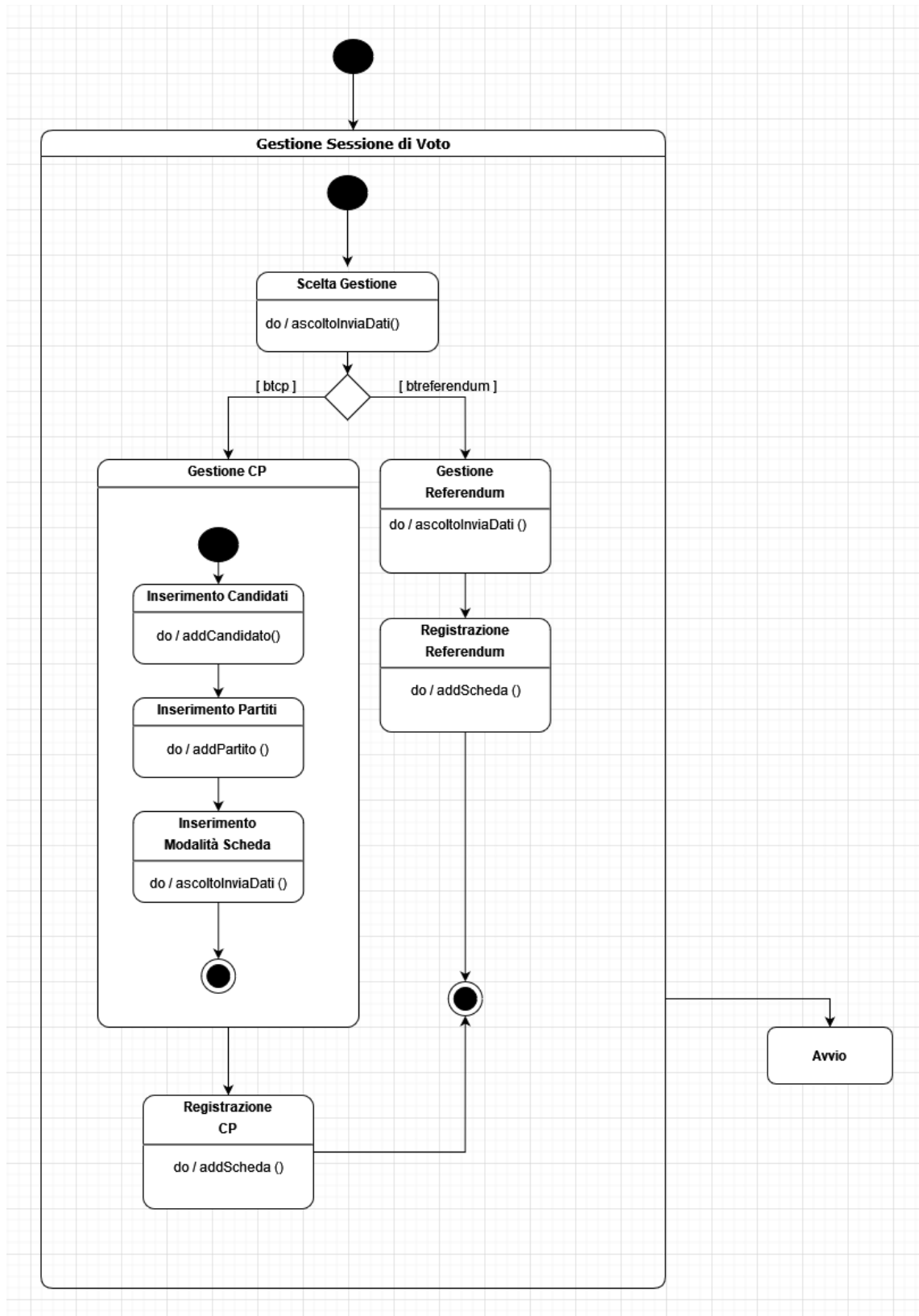
2.6.1 Registrazione



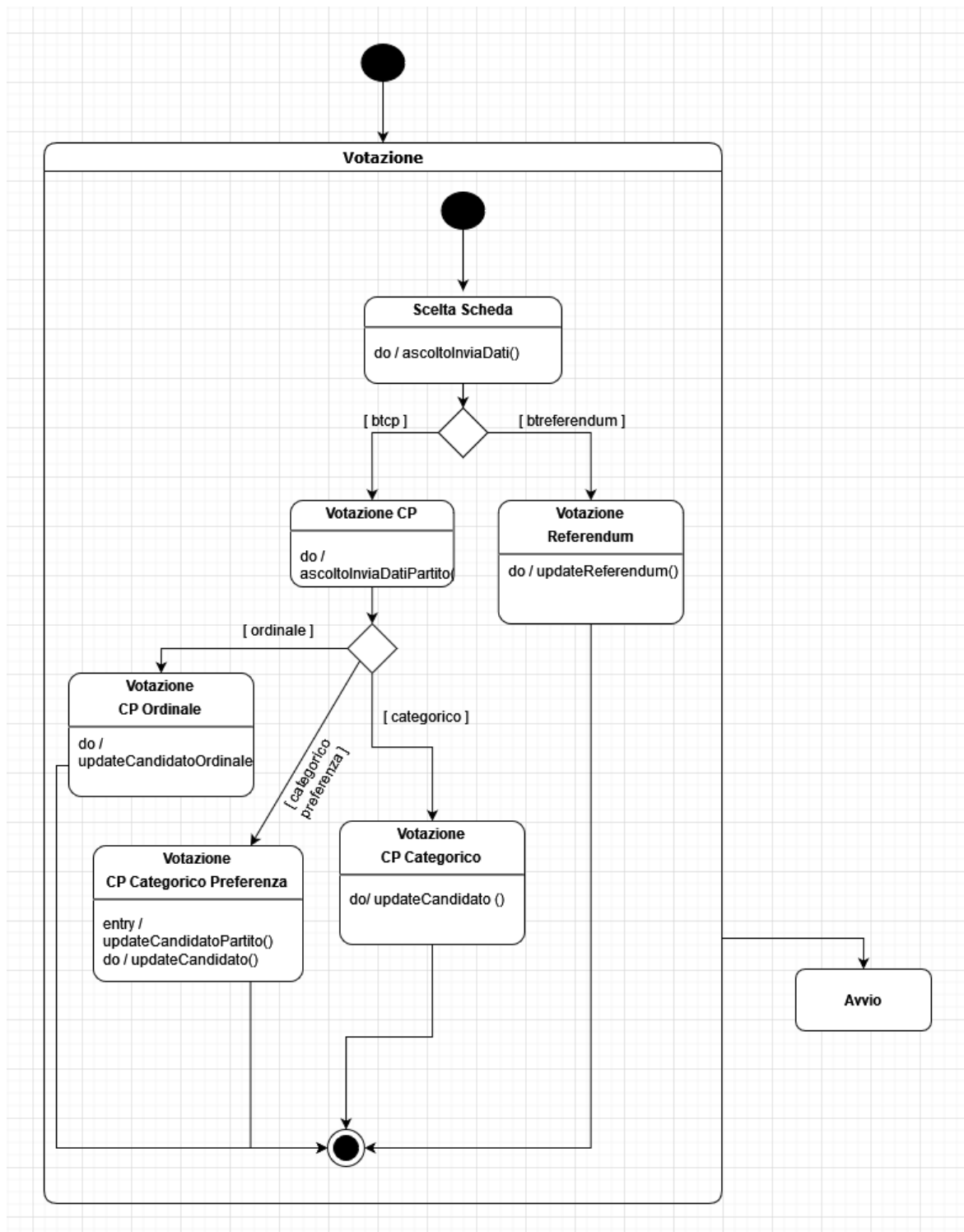
2.6.2 Autenticazione



2.6.3 Gestione Sessione di Voto

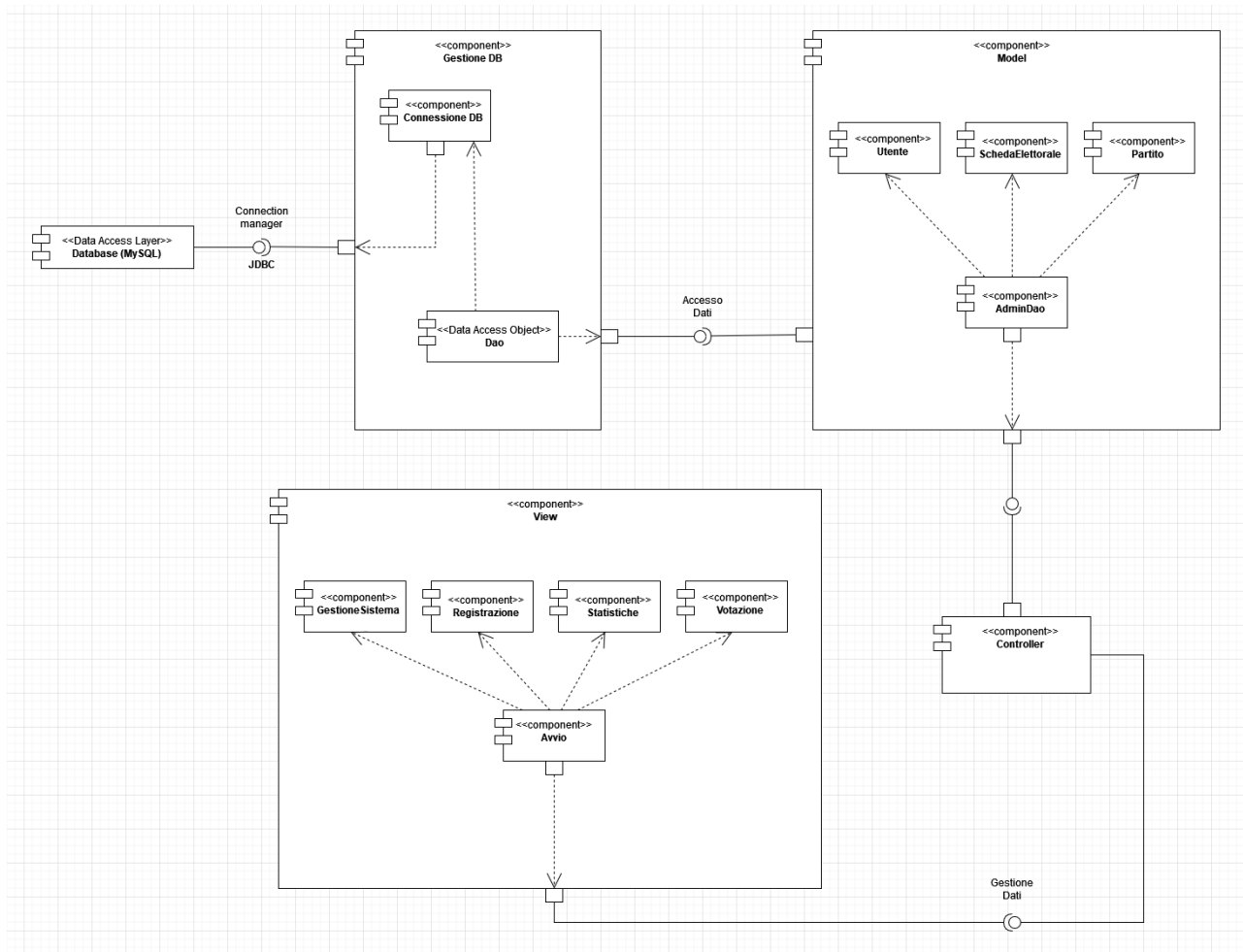


2.6.4 votazione



I sorgenti ad alta risoluzione si trovano nella cartella fsm diagram all'interno della cartella images diagram.

2.7 Diagramma dei componenti



E' possibile vedere come il diagramma delle componenti sia suddiviso in 4 macro sezioni:

- La prima è quella relativa alla gestione del Database che permette ad AdminDao di gestire, salvare, eliminare e accedere ai dati salvati in modo persistente.
- C'è poi la sezione Model che grazie alla componente precedente usa l'interfaccia per accedere ai dati e invia inoltre i risultati delle operazioni riguardanti i dati alla componente Controller.
- La sezione Controller prende i dati da Model restituendoli alla componente View
- La componente View sarà aggiornata dal Controller in funzione dello scenario in cui si trova il sistema o in base ai dati ricevuti dai modelli

Il sorgente ad alta risoluzione si trova nella cartella component diagram all'interno della cartella images diagram.

3 Implementazione del sistema

3.1 Discussione dei Design Pattern utilizzati

MVC: il design pattern più incisivo è sicuramente il model view controller.

Infatti l'intera applicazione è stata divisa in tre packages, contenenti i modelli, il controller e le views.

Tutto il progetto è stato costruito in base a questo pattern, in quanto si ha in ogni momento una dipendenza uno-a-molti tra model e view e tra view e controller.

L'idea è proprio quella di dividere la dichiarazione della UI dalla manipolazione di essa e dal dominio, che può essere riutilizzato non avendo dipendenze.

DAO: il secondo pattern più influente è il DAO, Data Access Object, che astrae il database.

Grazie al DAO mi è stato possibile mantenere le interazioni col Database in una classe distinta e indipendente dal resto del sistema.

Infatti, tutte le query per recuperare i dati dal DB sono incapsulate in metodi specifici all'interno della classe AdminDao.

Singleton: Un altro pattern utilizzato è il singleton, utilizzato per la classe Logger.

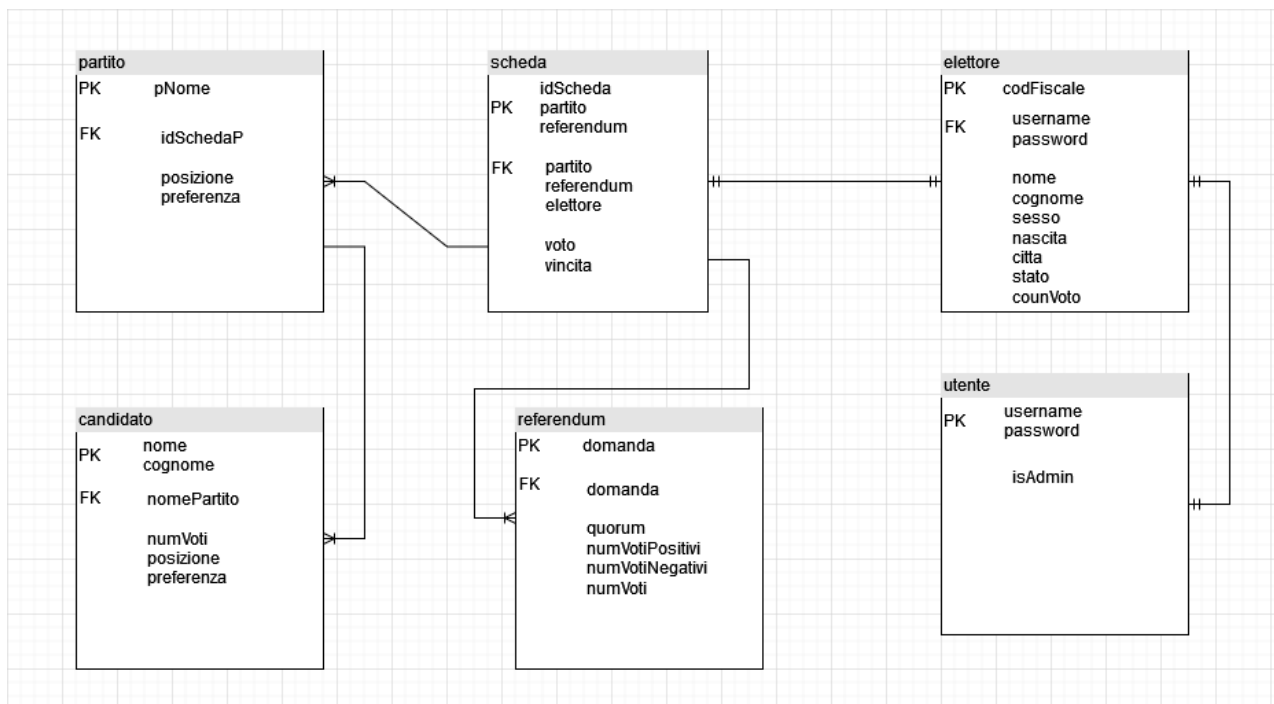
La classe Logger è utilizzata per il sistema di auditing ed è stata implementata come singleton per evitare la costruzione di multiple istanze a ogni esecuzione del programma.

E' stato quindi reso privato il costruttore e la singola istanza è accessibile tramite il metodo statico getInstance().

Questa classe è usata all'interno del controller per tenere traccia di tutte le operazioni eseguite nell'applicativo.

Observer: Swing utilizza un pattern observable per la gestione degli eventi, che vengono registrati e sollevati in caso di interazione con la UI.

3.2 Gestione dei dati persistenti



Il DBMS utilizzato è stato **MySQL**.

Dall'immagine è possibile visionare la struttura del database costruito a supporto del sistema per il salvataggio di dati persistenti.

Il Database è composto da 6 tabelle (e rispettive classi):

- **Utente**: è identificata dalla chiave primaria username e password ed è composta dal booleano isAdmin per capire se si tratta di un elettore o di un Admin.
- **Elettore**: è identificata dalla chiave primaria codFiscale, che rappresenta il codice fiscale di un elettore ed è un campo univoco. Ha inoltre la coppia username, password utilizzata come chiave esterna per riferirsi all'Utente.
- **Scheda**: è identificata dalla chiave primaria idScheda, partito e referendum. A ogni scheda sono associati più partiti e più referendum, tramite la chiave esterna relativa e l'associazione 1-N. Inoltre a ogni scheda è associato un elettore, tramite la chiave esterna relativa e l'associazione 1-1. La scheda è infine composta dai campi voto e vincita che rappresentano rispettivamente le modalità per la votazione e per la vincita della scheda elettorale.
- **Referendum**: è identificata dalla chiave primaria domanda, che rappresenta la proposta di legge. Lo stesso campo è anche la chiave esterna per riferirsi alla scheda. E' composta dai campi quorum, che è un campo double da 0 a 1, e dai campi relativi ai numeri di voti che serviranno per tenere i conti sulle votazioni e per poter capire se la proposta di legge verrà approvata o meno.

- **Partito:** è identificata dalla chiave primaria pNome, che rappresenta il nome del partito.
E' associata alla scheda tramite la chiave esterna idSchedaP, che è l'id della scheda, ed è composta dai campi relativi alle votazioni.
- **Candidato:** è identificata dalla coppia di chiave primaria nome,cognome.
Più candidati possono appartenere a un partito, e infatti esiste un'associazione 1-N tra partito e candidato, tramite la chiave esterna nomePartito.
E' composta da campi relativi alle votazioni.

Il sorgente ad alta risoluzione si trova nella cartella database diagram all'interno della cartella images diagram.

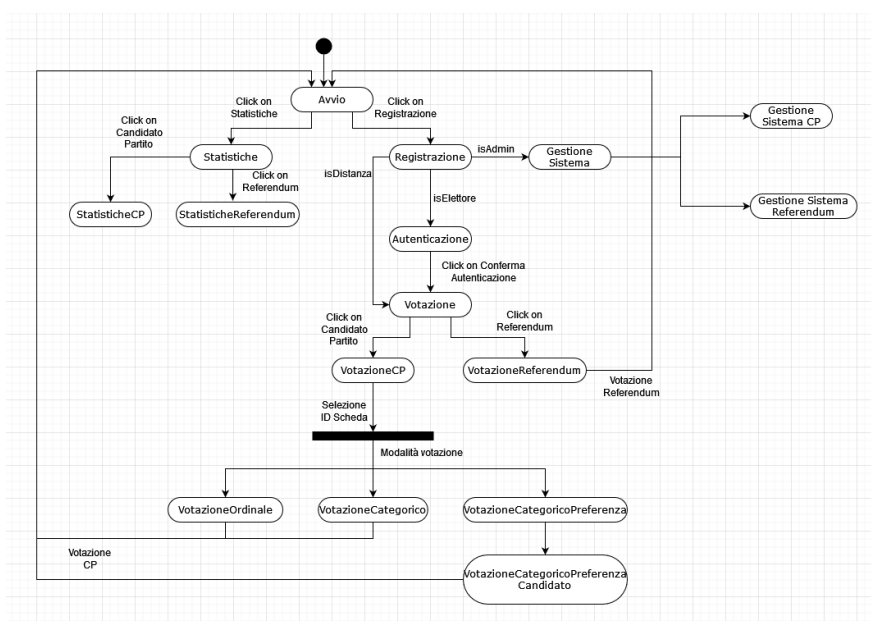
3.3 Descrizione dell'Interfaccia Grafica

La UI è composta da diverse pagine, ciascuna implementata manualmente attraverso le componenti visuali di Swing e ognuna gestita dal Controller.

La validità degli input è sempre controllata e, in caso di errori interni o dovuti a dati errati, verrà mostrato un alert di errore.

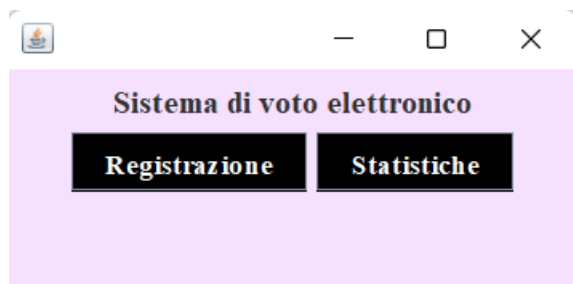
Sono inoltre visibili dei message dialog per la visualizzazione dei risultati delle votazioni e utilizzati anche per dare il benvenuto a nuovi utenti o confermare l'avvenuta registrazione dei dati o la conferma di votazione.

La navigazione è descritta dalla Navigation Map seguente:



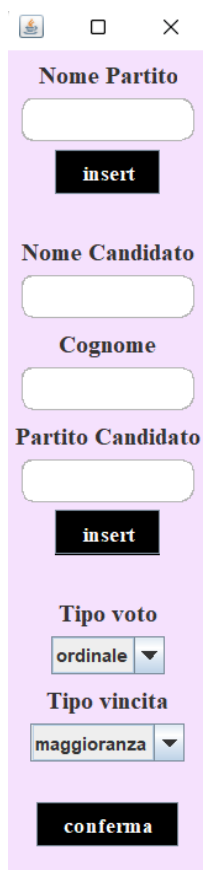
Di seguito saranno mostrati gli screenshot relativi ad alcune pagine implementate:

3.3.1 Avvio



La schermata di avvio è composta da due bottoni con cui è possibile interagire rispettivamente per registrarsi o per visualizzare l'andamento delle votazioni

3.3.2 GestioneCP



Questa è la schermata relativa alla gestione della sessione di voto per candidati e partiti.

E' possibile inserire più partiti tramite il bottone insert sotto Nome Partito.

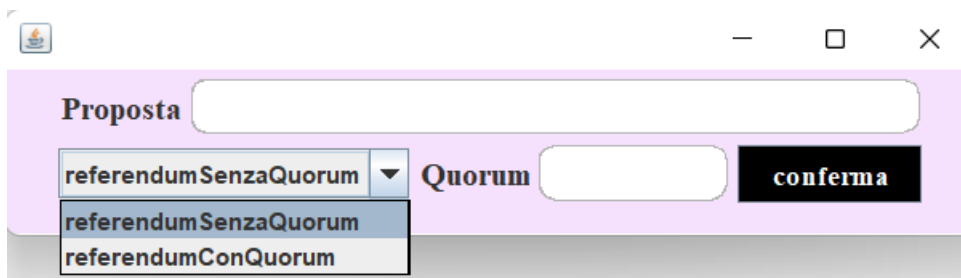
E' possibile inserire più candidati tramite il bottone insert sotto Partito

Candidato, di cui vengono ricavate le informazioni riguardo al nome, cognome e il partito a cui fa parte.

Infine è possibile inserire la modalità di voto e di vincita.

Cliccando il bottone conferma si registrerà la nuova scheda elettorale e si verrà reindirizzati alla pagina di Avvio


3.3.3 GestioneReferendum



The screenshot shows a web form titled "GestioneReferendum". It features a text input field labeled "Proposta". Below it is a dropdown menu with the selected option "referendumSenzaQuorum". To the right of the dropdown is a text input field labeled "Quorum". A black button labeled "conferma" is positioned to the right of the "Quorum" field. A small icon of a person is visible in the top left corner of the form area.

Questa è la schermata relativa alla gestione della sessione di voto per il referendum. E' possibile inserire la proposta di legge, la modalità di vincita e l'eventuale quorum. Cliccando sul bottone conferma verrà registrata la nuova scheda elettorale e si verrà reindirizzati alla pagina di Avvio.

3.3.4 RegistrazioneElettore



The screenshot shows a web form titled "RegistrazioneElettore". It contains several input fields: "Nome", "Cognome", "Sesso", "Nascita" (with a calendar icon), "Città", and "Stato" (with a dropdown menu showing "Italia"). A black button labeled "Conferma" is located at the bottom right. A small icon of a person is visible in the top left corner of the form area.

Questa è la schermata relativa alla registrazione dell'elettore. E' possibile inserire i vari dati dell'elettore. La data di nascita verrà poi scorporata in giorno, mese, anno per poter generare il codice fiscale. Si può scegliere lo stato attraverso una lista dei vari stati presi dalla classe StateCodes del package Document. Cliccando sul bottone conferma verrà registrato il nuovo utente e si verrà reindirizzati alla pagina di Votazione.

3.3.5 RegistrazioneUtente



A screenshot of a web application window titled "RegistrazioneUtente". The window has a light purple background. It contains two text input fields labeled "username" and "password". To the right of the "password" field are two checkboxes: "Admin?" and "Votazione a Distanza?". Below these fields is a black button with the text "conferma" in white.

Questa è la schermata relativa alla registrazione dell'utente.

E' possibile inserire username e password e selezionare la checkbox per admin o votazione a distanza.

Se viene selezionata la checkbox admin e sono stati inseriti i corretti username e password si verrà reindirizzati alla pagina di gestione del sistema, una volta cliccato il bottone conferma.

Se viene selezionata la checkbox Votazione a Distanza si verrà reindirizzati alla pagina di Autenticazione (RegistrazioneElettore).

Altrimenti si assume che la votazione avvenga in presenza e quindi si andrà direttamente alla pagina di Votazione.

3.3.6 Votazione

Votazione Categorico



A screenshot of a web application window titled "Votazione Categorico". The window has a light purple background. It contains a label "Scegli il candidato" followed by a dropdown menu. The dropdown menu is open, showing a list of candidates: "Alessandro Rossi Lega", "Alessandro Rossi Lega", "Emanuele Sara M5S", "Gianfranco Bianco PD", and "Paolo Verdi M5S". To the right of the dropdown menu is a black button with the text "conferma" in white.

Votazione Referendum



A screenshot of a web application window titled "Votazione Referendum". The window has a light purple background. It contains a label "Proposta:" followed by a dropdown menu. The dropdown menu is open, showing a list of proposals: "Mascherine obbligatorie?", "Mascherine obbligatorie?", and "prova". To the right of the dropdown menu is a checkbox labeled "Favorevole?" and a black button with the text "conferma" in white.

Questi sono esempi di schermate di votazione.

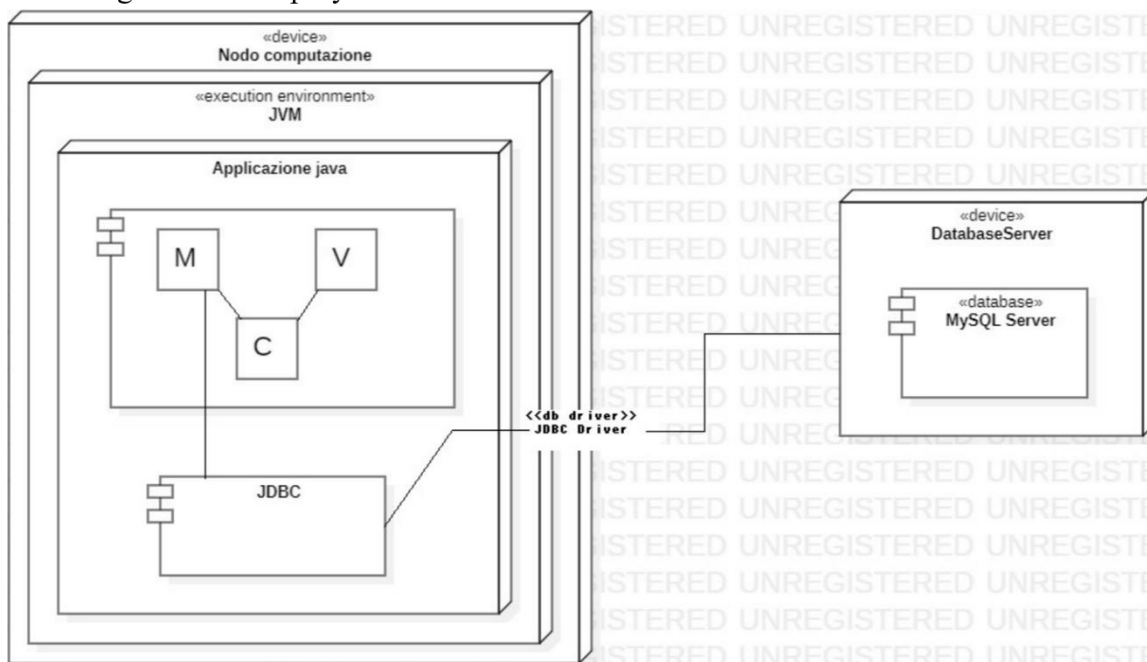
In alto si può vedere la schermata di Voto Categorico, in cui si può scegliere il candidato attraverso una lista di candidati che viene estrapolata dal Database.
In basso è possibile vedere invece la schermata di Voto del Referendum, in cui si può scegliere la proposta di legge e scegliere se si è favorevoli o meno.
Una volta cliccato il bottone di conferma saranno registrati i voti e modificati i numeri di voti nel database.

Le altre schermate, ad esempio relative alle altre modalità di votazione, è possibile visualizzarle nella cartella images gui.

Tutti i sorgenti inseriti nel documento, tra cui la navigation map e le interfacce grafiche, si trovano nella cartella images gui.

3.4 Diagramma di deployment

3.4 Diagramma di deployment



Come si può notare l'applicazione ha come unica interazione esterna quella con il Database tramite l'uso di **JDBC**.

3.5 Specifica e verifica dei vincoli

Di seguito sono mostrati i vincoli OCL di alcune funzioni.

Il corrispettivo in JML si trova direttamente applicato alle stesse sul codice sorgente sotto forma di commento.

Elettore

```
{  
context Elettore  
Inv: self.code.length == 16  
Inv: (self.nome != null) && (self.cognome != null)  
}
```

```
{  
context Elettore :: getCF ()  
post: self.code.length == 16  
}
```

Candidato

```
{  
context Candidato  
Inv: self.posizione >= 0  
}
```

Partito

```
{  
context Partito :: getPosizione ()  
post: self.posizione >= 0  
}
```

SchedaElettorale

```
{  
context SchedaElettorale  
Inv: SchedaElettorale.allInstances.isUnique (idScheda)  
}
```

Admin

```
{  
context Admin :: checkAdmin ()  
post: self.isAdmin == true  
}
```

3.6 Descrizione del testing

I test consistono in delle classi istanziate da JUnit e si trovano nel package org.alessandrofare nella cartella relativa ai test del progetto.

Sono state implementate tre classi di test:

- **TestElettore:**

- Test sul codice fiscale.

Viene istanziato un elettore coi relativi dati, viene generato il codice fiscale relativo e si testa se il codice generato è quello giusto

- Test Admin.

Viene istanziato un elettore coi relativi dati, e si settano anche username e password.

Una volta si settano dati non di Admin e si testa che il metodo checkAdmin dia risultato falso.

Infine si settano i giusti dati e si testa che il risultato dia esito positivo.

- Test sui campi

Viene istanziato un elettore coi relativi dati e si testa che i metodi getter siano corretti e in particolare che dalla data di nascita vengano estrapolati giorno, mese e anno.

- **TestScheda:**

- Test candidato-partito

Viene istanziata una scheda elettorale con liste di candidati e partiti, viene settata la modalità di vincita e di voto e si testa che il candidato vincitore e il partito politico a cui appartiene siano quelli corretti.

- Test referendum

Viene istanziata una scheda elettorale con una proposta di legge, il quorum e vengono settati la modalità di vincita e un numero di voti.

Prima si setta un numero di voti tali che la proposta di legge non venga approvata e si testa che ciò avvenga.

Poi viene aumentato di uno il numero di voti, attraverso il metodo setFavorevole (), e questa volta la proposta di legge verrà approvata e si testa che ciò avvenga.

- **TestUtente:**

- Test Admin e Campi

Viene istanziato un Admin e si testa che i relativi campi a lui associati siano stati correttamente settati.

3.7 Note per l'installazione e l'utilizzo

- L'ambiente di sviluppo utilizzato è Eclipse (ultima versione stabile).
Il progetto è stato implementato interamente utilizzando il linguaggio Java, in particolare la versione consigliata è Java 11
- Per poter visualizzare meglio il codice occorre aprire su Eclipse la cartella Voto-Elettronico come progetto
- Nella cartella libs è possibile trovare il file .jar necessario per collegarsi al database con il DBMS usato in questo progetto (MySQL). Il file si chiama mysql-connector-java-8.0.27.jar.
E' inoltre presente il file jar utilizzato per il testing, ovvero junit-4.13.jar.
Per poterli utilizzare su Eclipse occorre cliccare col tasto destro sul progetto, andare su properties > java build path e poi cliccare su Add External Jars per poter importare le librerie necessarie
- Nella cartella dumpDB si possono trovare due file dump per ricostruire lo schema usato.
Vanno quindi importati dentro il DBMS per poter essere utilizzati.
Le credenziali usate per accedere al database si trovano nel file ConnectionFactory all'interno del package Connection.
L'username è root, la password è alefabiofare98.
L'host è localhost e il database è chiamato votoelettronico
- Il file VotoElettronicoDUMP.sql serve per creare un database vuoto che si potrà riempire avviando l'applicazione e registrandosi.
Per poter accedere alle funzionalità di gestione del sistema bisogna inserire nella fase di registrazione le seguenti credenziali: username = ale, password = 00.
Una volta inserite sarà possibile inserire nel database le schede elettorali e far partire la sessione di votazione.
Quindi si potrà accedere come elettori e votare la scheda elettorale scelta
- Il file VotoElettronicoDUMP-1.sql serve per creare un database con già dei dati creati all'interno.
Sono stati creati vari utenti, in particolare l'utente di Admin con il campo isAdmin a 1.
Sono stati creati diversi elettori, ciascuno coi dati relativi.
Sono state create due schede elettorali, una relativa alle elezioni e una relativa al referendum. Ogni scheda ha un proprio id.
Sono state inserite liste di candidati, partiti e un referendum.
E' stato votato una volta il referendum, con voto positivo, ed essendo un referendum senza quorum la proposta verrà approvata (si può vedere l'esito dalle statistiche).
E' stato votato un candidato, con voto positivo, ed essendo una scheda con vincita di maggioranza il candidato votato sarà il vincitore

- Per poter lanciare l'applicazione sarà necessario runnare la classe `VotoElettronico.java` contenuta nel package `com.alessandrofare` che si trova nella cartella `src/main/java`