MODEL:

Person

#firstName: String
#lastName: String
#SIN: int

Address

-streetNumber: String
-street: String
-city: String
-postalCode: String

#address
1

-province
1

<<Enumeration>>
Province

+getProvinceList(): Collection<String>
+getProvinceName(in province:Province): String
+getProvinceFromName(in provinceName:String): Province

-province
1

Voter

#username: String
#password: String
#hasVoted: boolean

+vote(): void

#district
1

-voter
1

District

-name: String
-id: int

+addCandidate(in candidate:Candidate): void
+getCandidate(in party:Party): Candidate
+getAllCandidates(): Collection<Candidate>

Vote

-candidate
+candidates
0..*

Candidate

+runFor(in party:Party): void

-leader
1

-party
1

Party

-name: String

ResultSet

+DISTRICT: final int = 0
+TOTAL: final int = 1
-type: int
-districtVotes: Collection<Integer>
-totalVotes: Collection<Integer>

NETWORK:

```
┌─────────────────────────────────────────────────────┐
│                      Message                        │
├─────────────────────────────────────────────────────┤
│+Method: enum = {GET, POST, PUT, DELETE, CONNECT}    │
│-data: Object                                        │
│-method: Method                                      │
│-checksum: long                                      │
│-length: int                                         │
│-senderPort: int                                     │
│-type: Type                                          │
│-senderAddr: InetAddress                             │
├─────────────────────────────────────────────────────┤
│+calculateChecksum(in data:Object): long             │
│+getBytes(data:Object): byte[]                       │
│+getBytes(): byte[]                                  │
│-calculateLength(): void                             │
└─────────────────────────────────────────────────────┘
```
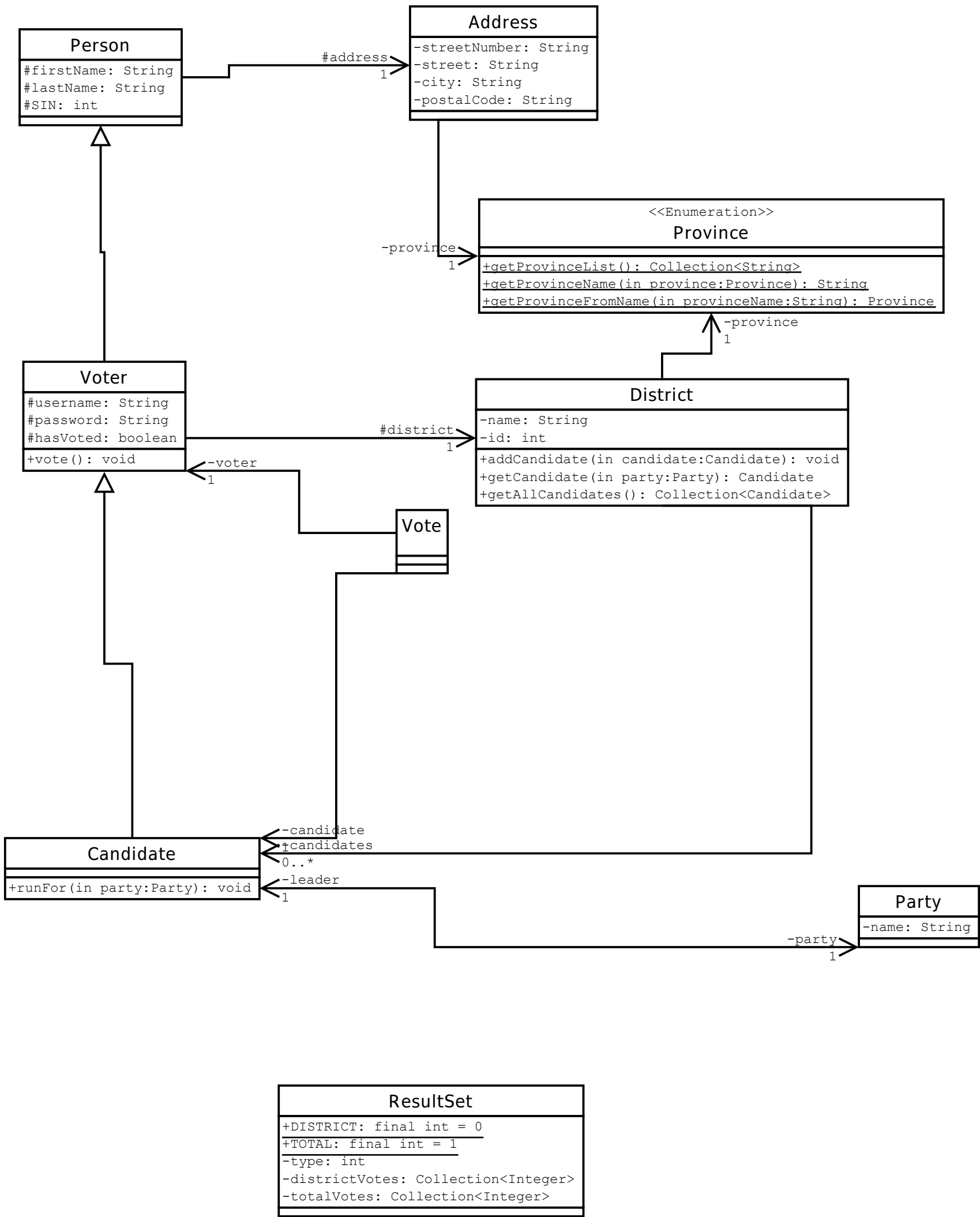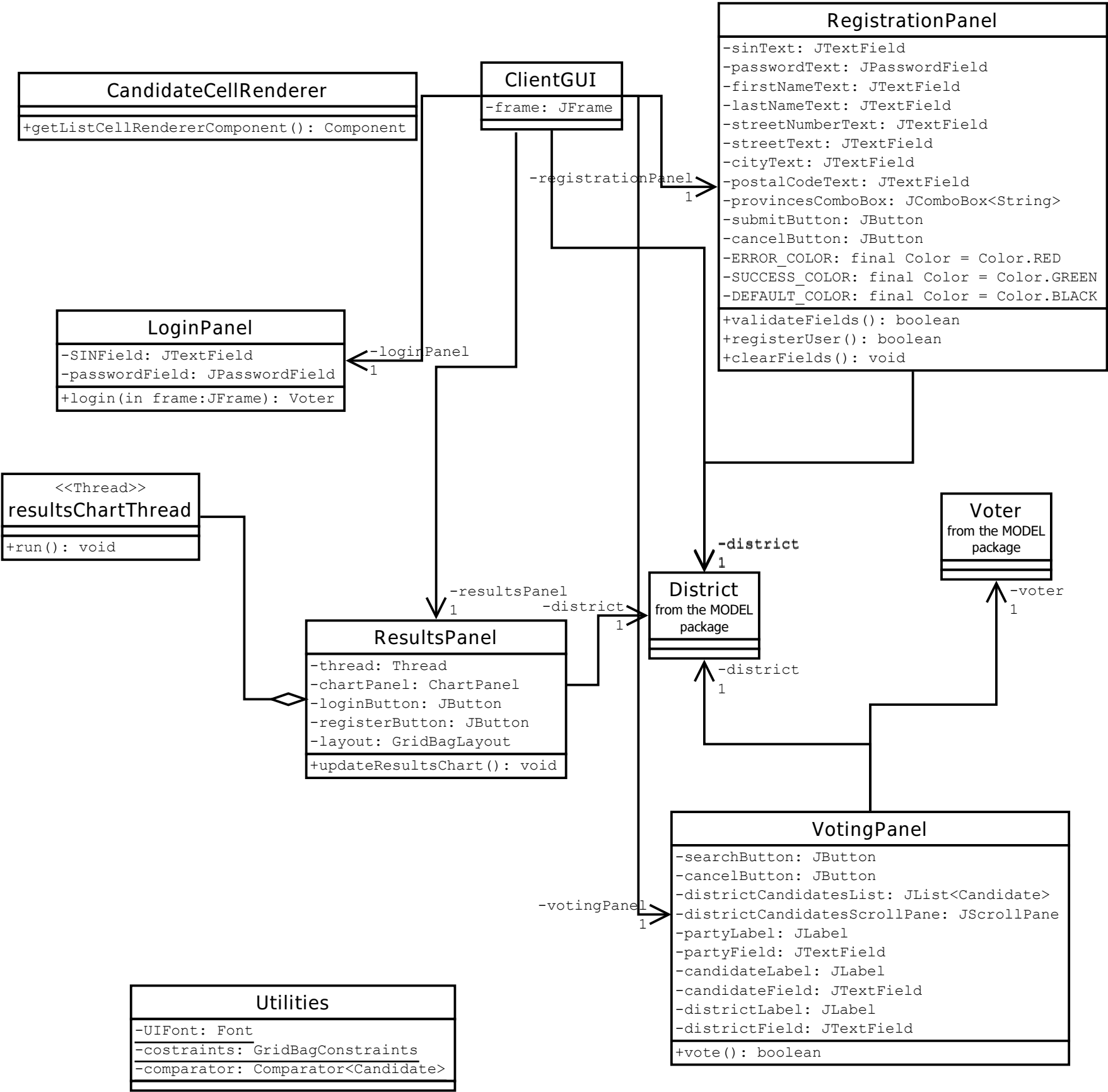
```
┌───────────────────────────────┐
│        <<Exception>>          │
│    MessageCorruptException    │
├───────────────────────────────┤
├───────────────────────────────┤
└───────────────────────────────┘
```

```
┌─────────────────────────────────────────────────────────┐
│                        WSocket                          │
├─────────────────────────────────────────────────────────┤
│+PACKET_LEN: final int = 500                             │
│+FRAG_LEN: final int = 200                               │
│-TIMEOUT: int = 10                                       │
│-socket: DatagramSocket                                  │
│-addr: InetAddress                                       │
│-port: int                                               │
├─────────────────────────────────────────────────────────┤
│+listen(in port:int,in host:String): WSocket            │
│+listen(in port:int): WSocket                           │
│+connect(in port:int,in host:String): WSocket           │
│+connect(,in port:int): WSocket                         │
│+close(): void                                          │
│+receive(): Message                                     │
│-sendConfirmation(in port:int,in host:InetAddress): void │
│+send(in msg:Message): void                             │
│+sendTo(in msg:Message,in port:int): void               │
│+sendReceive(in msg:Message,in port:int,               │
│              in host:InetAddress): Message             │
│+sendReceive(in msg:Message): Message                   │
│-receiveFragments(): Message                            │
│-sendFragments(in msg:Message,in port:int,             │
│              in host:InetAddress): void               │
└─────────────────────────────────────────────────────────┘
```

VIEW:

CandidateCellRenderer

+getListCellRendererComponent(): Component

ClientGUI

-frame: JFrame

RegistrationPanel

-sinText: JTextField
-passwordText: JPasswordField
-firstNameText: JTextField
-lastNameText: JTextField
-streetNumberText: JTextField
-streetText: JTextField
-cityText: JTextField
-postalCodeText: JTextField
-provincesComboBox: JComboBox<String>
-submitButton: JButton
-cancelButton: JButton
-ERROR_COLOR: final Color = Color.RED
-SUCCESS_COLOR: final Color = Color.GREEN
-DEFAULT_COLOR: final Color = Color.BLACK

+validateFields(): boolean
+registerUser(): boolean
+clearFields(): void

-registrationPanel
1

LoginPanel

-SINField: JTextField
-passwordField: JPasswordField

+login(in frame:JFrame): Voter

-loginPanel
1

resultsChartThread
<<Thread>>

+run(): void

District
from the MODEL
package

-district
1

Voter
from the MODEL
package

-voter
1

-resultsPanel
1

-district
1

ResultsPanel

-thread: Thread
-chartPanel: ChartPanel
-loginButton: JButton
-registerButton: JButton
-layout: GridBagLayout

+updateResultsChart(): void

-district
1

VotingPanel

-searchButton: JButton
-cancelButton: JButton
-districtCandidatesList: JList<Candidate>
-districtCandidatesScrollPane: JScrollPane
-partyLabel: JLabel
-partyField: JTextField
-candidateLabel: JLabel
-candidateField: JTextField
-districtLabel: JLabel
-districtField: JTextField

+vote(): boolean

-votingPanel
1

Utilities

-UIFont: Font
-costraints: GridBagConstraints
-comparator: Comparator<Candidate>

CONTROLLER:

CentralServer

+CENTRAL_SERVER_PORT: final int = 60001
+PERIOD: final int = 3000

ClientController

+SIMULATION_MODE: final int = 0
+USER_MODE: final int = 1
-districtServerPort: int

+closeSocket(): void
+registerUser(in voter:Voter): boolean
+getParties(): Collection<Party>
+getDistrictCandidates(in district:District): Collection<Candidate>
+vote(in candidate:Candidate,in voter:Voter): void
+getLocalResults (in district:District): Collection<Integer>
+getNationalResults(): Collection<Integer>
+updateCandidate(in candidate:Candidate,
in party:Party): boolean
+loginUser(in username:String,in password:String): Voter
+userHasVoted(voter:Voter): boolean
+simulate(in inputFolder:String,in outputFolder:String): void
+simulateFromFile(in inputFile:String,in outputFile:String): void
+startUI(): void

-socket
1

WSocket
from the
NETWORKING
package

-socket
*

DistrictServer

+populateParties(in inputFile:String): void
+receiveMessages(): void
+processMessages(): void

-district
1

District
from the MODEL
package

-voters
0..*

Voter
from the MODEL
package

-parties
0..*

Party
from the MODEL
package

-candidates
0..*

Candidate
from the MODEL
package

-votes
0..*

Vote
from the MODEL
package

<<Interface>>
RtvsType

+CANDIDATES: final String = CANDIDATES
+REGISTER: final String = REGISTER
+VOTE: final String = VOTE
+LOGIN: final String = LOGIN
+RESULTS: final String = RESULTS
+HAS_VOTED: final String = HAS_VOTED
+PARTIES: final String = PARTIES
+RUN: final String = RUN