E-voting system for Ireland

13.11.2017

— **CS3500 Software Engineering**

Team L

Ács Dávid - 117106523

Gonzalo Carranza Pérez-Tinao - 117106527

Palcu Liana- Daniela - 117106643

Martinez de Rute Maria - 117106133

1. Overview

The purpose of the system will be to conduct all elections of Ireland. The system will allow citizens of the country to vote in a secure, anonymous and reliable way for the candidates of the ongoing election. The system will allow the introduction of the candidates and the rules of the election by authorized election officers.

The system will have several modules, notably the server side of the election system and the client side. Two or more instances of these modules will collaborate during the election. Please note that there can be more than one instance of the server part of the system, which must be able to communicate between them and to share the workload among them. Obviously there will be more than one client component in the system, all of which will connect to one of the servers available.

The server side will be operated by the system administrators and government officials of the system and they can set up the available choices during the election. The client side of the system will be operated by the voters of the country.

Correct functionality, availability, security and robustness are among the most important requirements of the system, since without these qualities the product is worthless in the eyes of the public, the government and practically all stakeholders of the system.

1. 1. Server side of the system

This part of the system is responsible for authenticating the voter, validating and counting all the votes of the election. The server side of the system must be deployed to a physically secure location and on a private network.

This part of system is made of up multiple modules: a module which stores data, such as a database management software, the email subsystem, and another application which enables the authorized users to perform operations on the system and to view statistics, logs and perform audits. The requirements of the server side of the system are specifically tailored for stakeholders that interact with this part of the system: system administrators and government officials.

1. 2. Client side of the system

The client side of the system will be a piece of software which will communicate with the server side of the system, in order to convey the choice of the user in a structured way. The client side can be any piece of software that respects the interface of the server side of the system. This will allow us to extend the system by creating new types of clients, which can easily be incorporated into the existing system. The requirements of the client side should reflect the needs of the voters regarding interaction with the client side of the system.

2. Allocation view

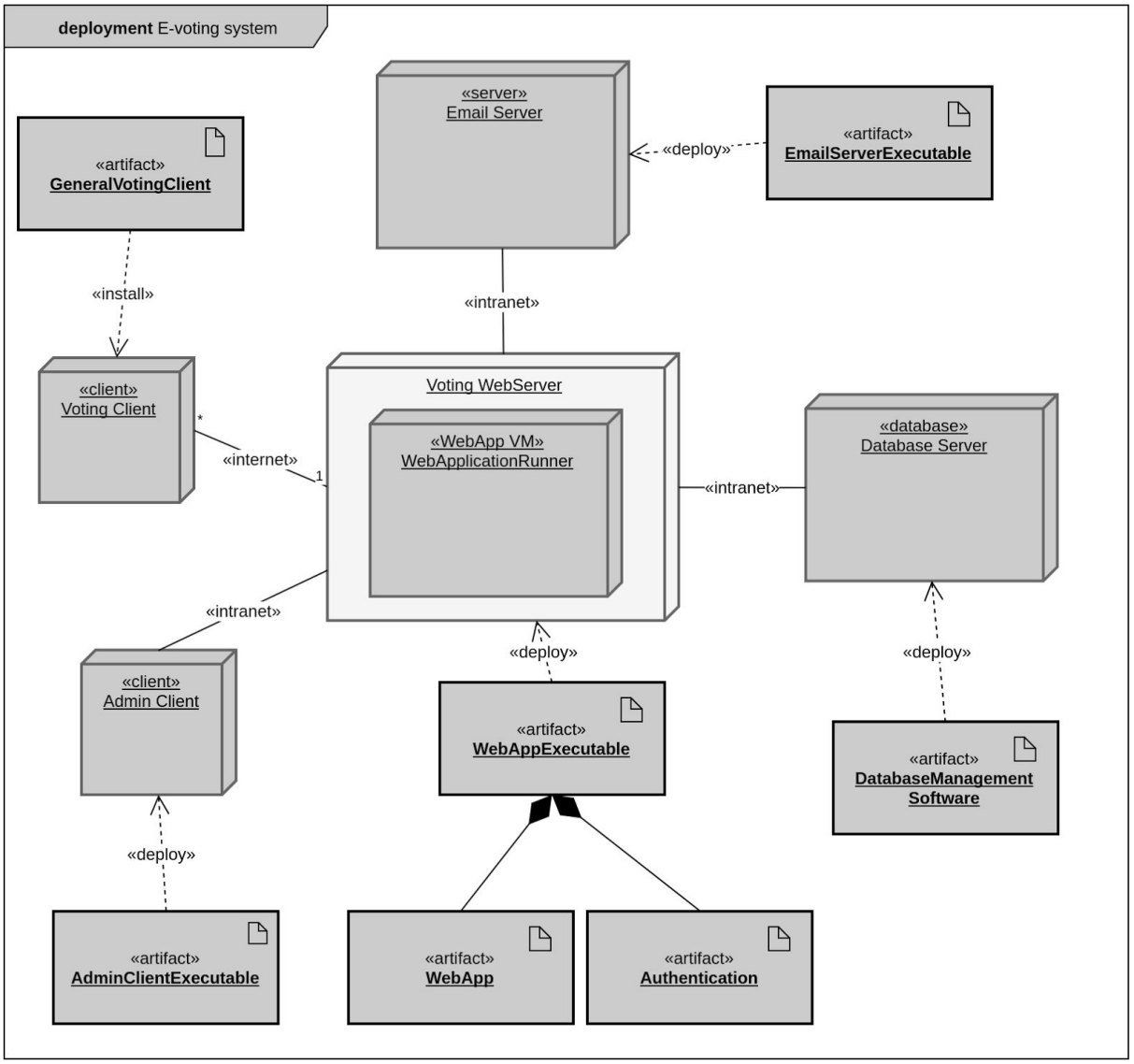

Figure 1. Allocation view

In the deployment diagram presented above (Figure 1.) we can see the main components and where they are deployed. There are two main categories of nodes present: Clients and Servers. The WebServer is communicating with multiple other servers in the intranet, it is useful to separate them, since it will make the system more secure.

The Voting Client is using the Internet to communicate with the WebServer.

The Admin Client can only connect through the intranet.

All nodes have an associated artifact, which is deployed/installed on them. In case of the WebApp Executable the deployed artifact is made up of two other artifacts: WebApp and Authentication, each having different responsibilities.

3. Model View

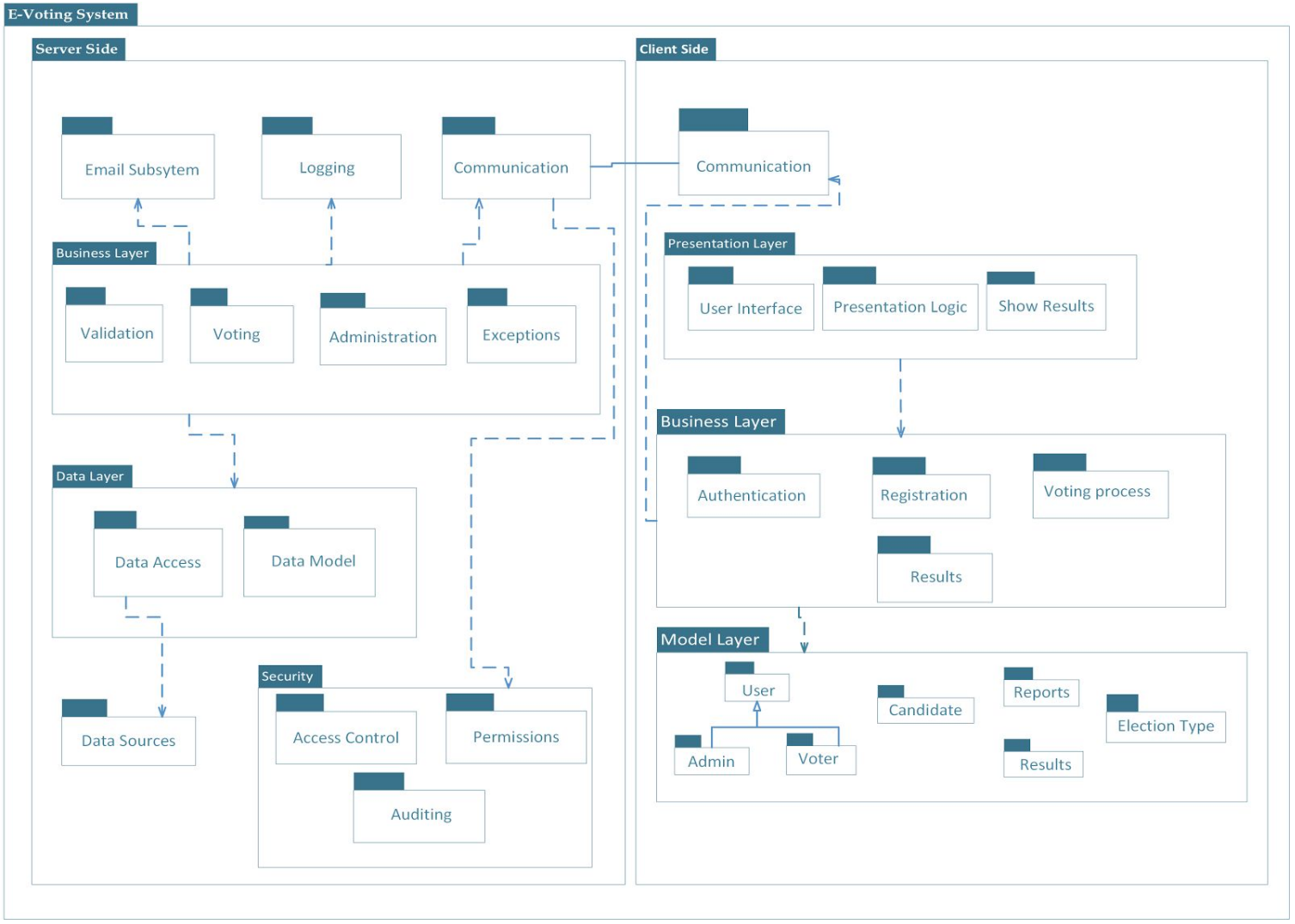

Figure 2. Model view

We divided the system in two parts:client-side and server-side.

The Server side part is composed by:

- **Email Subsystem** - it is used by Business Layer in the Validation process.
- **Logging** - it's a package used to log the events from the Business Layer
- **Communication** - it is used for exchanging information between client-side module and the server side module.
- **Business Layer -** it perform the operations provided by the server-side.
  - ➔ Validation
  - ➔ Voting - it counts the votes

    - ➔ Administration
    - ➔ Exceptions
- **Data Layer -** it keeps the model of data and access this from database
    - ➔ Data Access
    - ➔ Data Model
- **Security** - assure a secure communication between server side and client side.
    - ➔ Access Control
    - ➔ Permissions
    - ➔ Auditing
- **Data Sources** - it can be view as a package which make the connections with the database

The Client side is composed by:

- **Communication -** it is used for exchanging information between client-side module and the server side module.
- **Presentation Layer -** it presents the interface between the user and the effective implementation.
    - ➔ User Interface
    - ➔ Presentation Logic
    - ➔ Show Results
- **Model Layer** - it keeps the structure of the data, how it looks and how it can be modeled.
    - ➔ User
    - ➔ Admin
    - ➔ Voter
    - ➔ Candidate
    - ➔ Reports
    - ➔ Results
    - ➔ Election Type
- **Business Layer** - it performs the basic operations which are necessary for an e-voting system.
    - ➔ Authentication
    - ➔ Registration
    - ➔ Voting Process
    - ➔ Results - it processes the results.

4. Use-case diagram

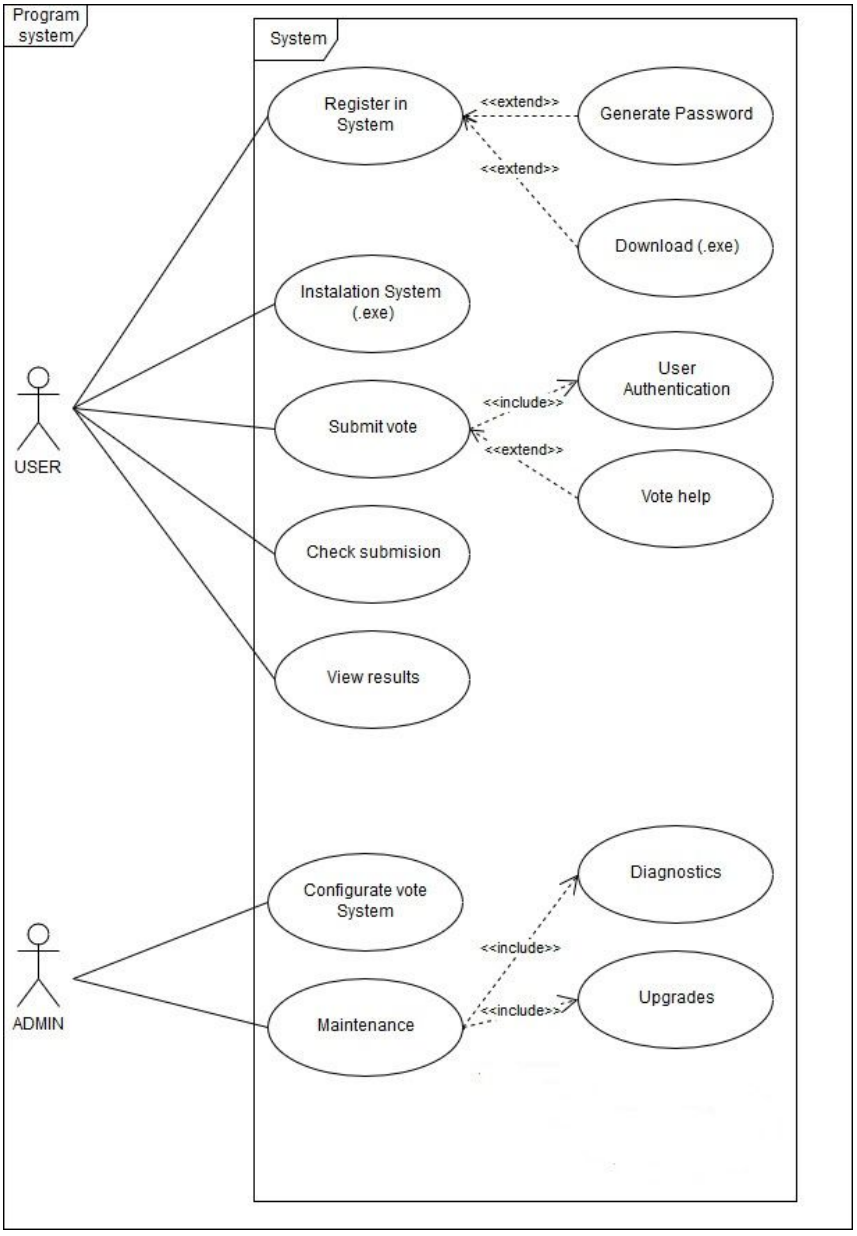

Figure 3. Use-case diagram Software System

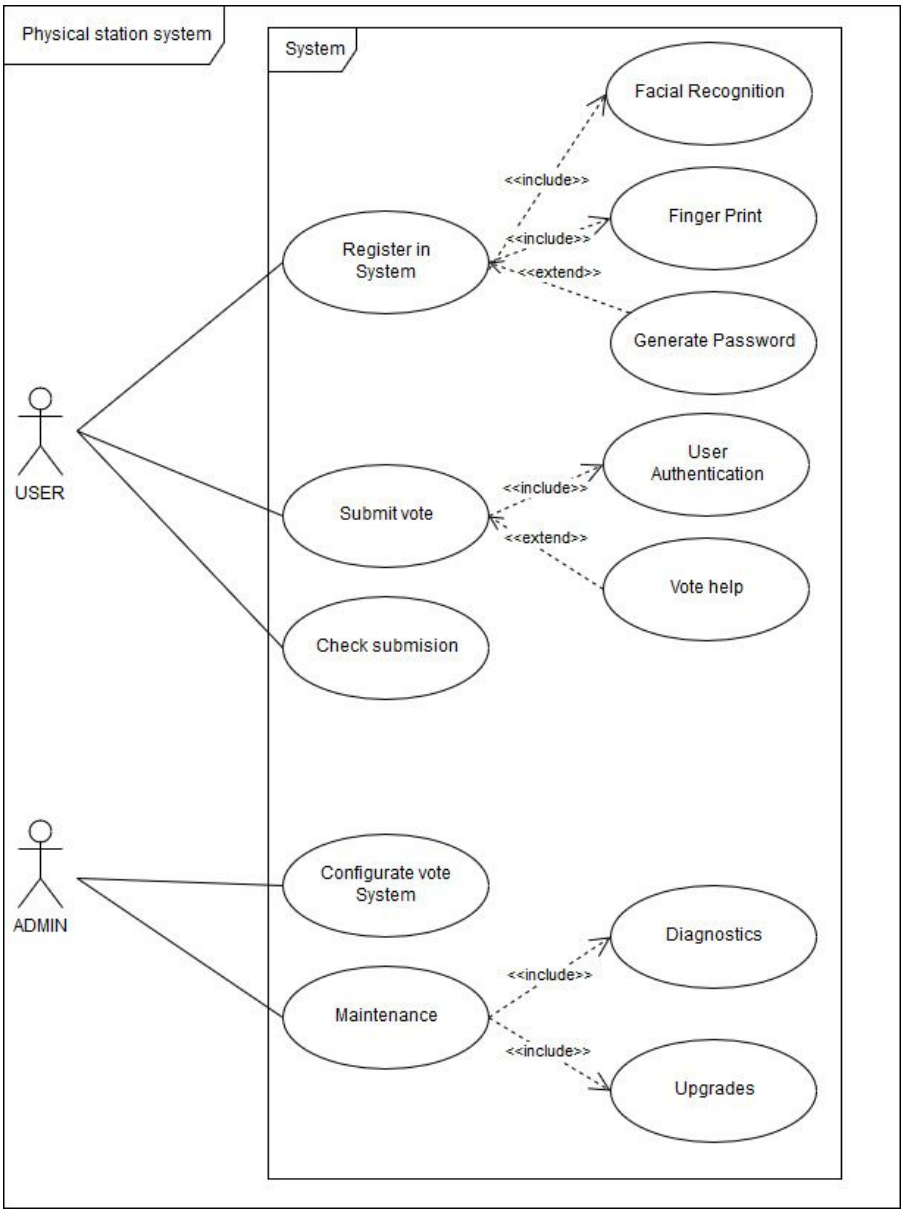

Figure 4. Use-case diagram Physical station

In the use-case diagram we can see how the user interacts with the system.

We divided in two parts, one is the program system that is the option to vote from home using an executable program. In here the user have to register in the page where later the user can download the executable program and the system gives a password to the user. After installing the system the user have to login in the system program and later can submit the vote. After submitting the vote he can check the submitted vote and later view the results of the hole votation.

The second one is the physical station system. This option is where the user can go to a specific place and vote there. Previously to the votation the user must register in the system, in this case it includes facial recognition and fingerprint.

After login in the system the user can submit the vote.

The admin is in charge of the configuration of the vote system and the maintenance of it making diagnostic and upgrades if it is necessary.

5. Component and Connector view

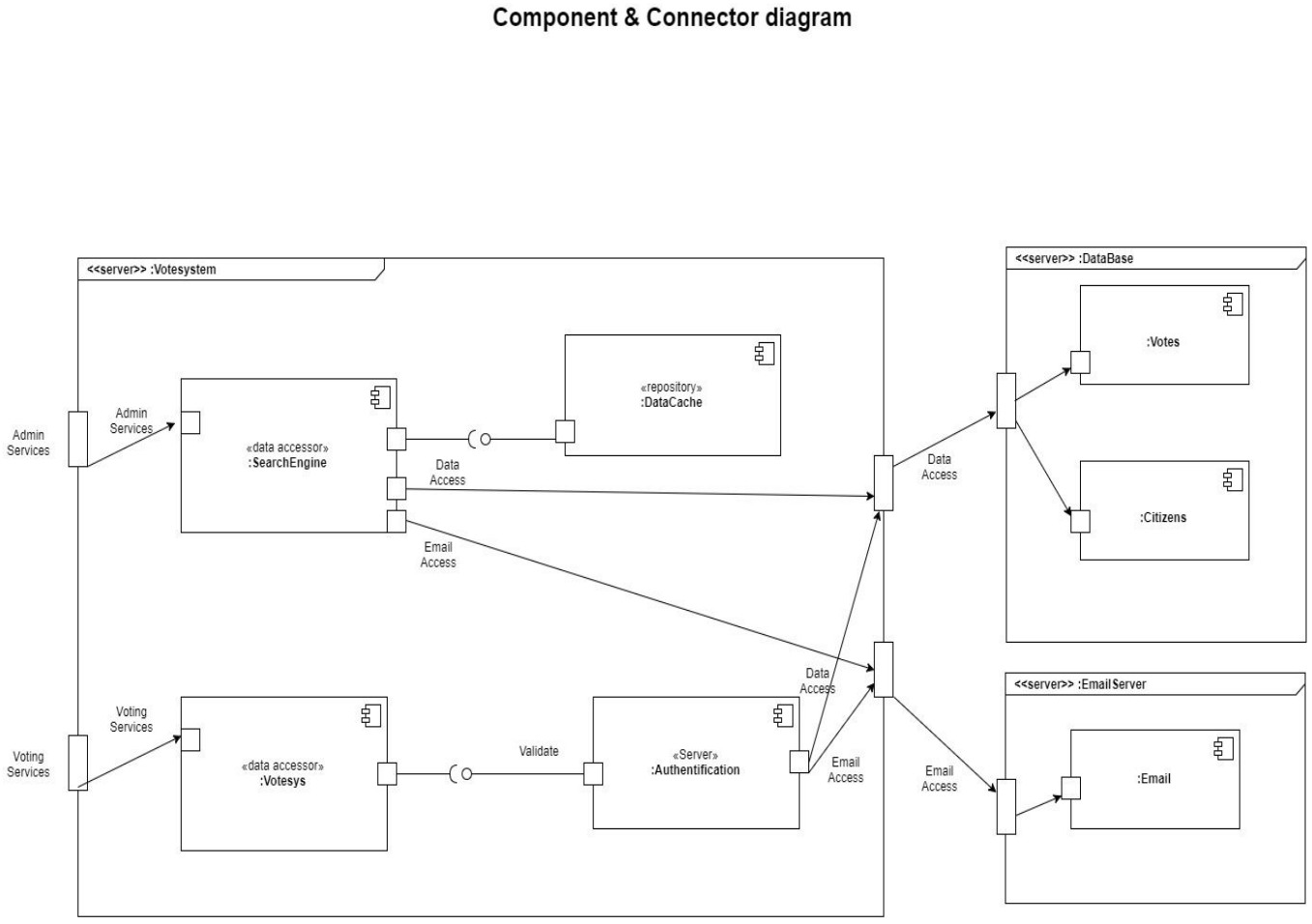

Figure 5. Component and Connectors diagram

The **Component & Connector diagram** shows the components of our system, ports, provided interface, required interface, connectors and the relationships between them.

The **Admin Client** has different services and it has access to the Database server and the Email server.

The **Voting Client** has to be authenticated and then it has access to the Database and the Email as the Admin Client.

The **Database** stores the information about the citizens who are going to vote and the votes that they make during the voting period.

The **Email Server** is very useful and it can be use for many things, its main use is to communicate information to voters. It is accessible by the Voting Client and the Admin Client.

6. Contributions

We distributed the work equally among ourselves. Each one of us created a diagram and the corresponding description.