Chapter 1

Requirements

The requirements are categorized according to the FURPS+ model.

1.1 Functional requirements

Identifier	Requirement summary
Features	
FR 1.1	The system must allow users to vote.
FR 1.2	The system must allow eligible users to configure an election prior to the start of it.
FR 1.3	The system must allow eligible users to start and stop an election.
FR 1.4	The system must allow eligible users to obtain the result of an election.
FR 1.5	The system must allow users to confirm that their vote has been registered by the bulletin boa
Interoperability	
FR 2.1	The system's source must be easily understandable to wide amount of software porfessionals.
FR 2.2	All system modules require Jolie to run?.
Extendability	
FR 3.1	Each system module must be replaceable without changing the other modules.
FR 3.2	System modules must communicate across well defined interfaces.
Composability	
FR 4.1	All communication between system modules must be verifiable.
Manageability	
FR 5.1	The system must be administrated by a set of authorities.
Maintainability	
FR 6.1	The system source code must be well commented and documented.
Security	
FR 7.1	The system must take measures to ensure that ones ballot is always anonymous.
FR 7.2	The system must be correlation resistant.
FR 7.3	Communication between the bulletinboard and the client, as well as the bulletinboard and the

1.2 Detailed Requirements

FR 1.1 - The system must allow users to vote

The system must be able to capture votes for all eligible people, such that it is taken into account when the ballots are counted. It must be possible to overwrite ones ballot as long as the election is still in progress, by the exactly same process as voting without overwriting.
It requires the following input to vote:

- The identifier of the election option.

- The personal identifier of the voter.
- The password of the user.

FR 1.2 - The system must allow eligible users to configure an election prior to the beginning of it

The system must allow the authorities to configure certain options in the system. These options include, but are not limited to, the election options, the keys used for the cryptographic parts of the system and when the elections begins and ends.

FR 1.3 - The system must allow eligible users to start and stop an election

FR 1.4 - The system must allow eligible users to obtain the result of an election

The system must allow the authorities to obtain the result of an election. This can be done only after an election has finished and only by providing the decryption key. The system must not in any way allow the holder of the key to decrypt anything less than the sum of all votes.

FR 1.5 - The system must allow users to confirm that their vote has been registered by the bulletin board server.

The system must not allow others to confirm whether a certain user, other than themselves, has voted. The user must have knowledge of the ciphertext sent to the bulletinboard to check whether he has voted. If the

FR 3.1 - The system's source must be easily understandable to wide amount of software professionals

The system must be written in a way which the average software professional can easily read and understand, in order to make the system more transparent. A more transparent system is, easier to verify sine more people will be able to understand the how the system work and thus be able to verify it.

FR 4.1 - All communication between system modules must be verifiable

Since the system shall be used to decide how many people has voted for the different candidates and/or parties, an error in the system can have a very large impact. To increase the likelihood of the software developers catching any mistake the system must be as transparent as possible. Furthermore it should be possible to verify the communication between the modules. It is important that if one module sends a message to another module, the message is the same when received.