Functional Requirements

1. System Reliability: The whole system should work even if there are many single point failures. System degeneration should be slow in the event of failures. It should never halt completely.
2. Correct Registration: This requires correct voter identification and eligibility. The system should guard against duplicate registration or fraudulent registration of phantom voters. Given the state of technology, we believe that this will be the hardest of all requirements to meet. It is best done by paper, the way it is in effect in many counties today. The aim should be to avoid large scale automated fraud of fraudulent registration as opposed to completely eliminating the problem.
3. Vote Confirmation: The voter should have some feedback of the vote he/she cast. In no event should the voter be told that his/her vote was counted even though the system failed to register it. This might be a tricky requirement because the system as a whole might fail resulting in many lost votes and it may be impossible to tell the voter of such a failure. One way to alleviate this problem is to count and make an audit trail of the votes as the voters cast them. The audit trail will allow for an alternative way of knowing what the votes were incase of system failure.
4. Provisional Voting: The voters should be able to vote provisionally incase some sort of human or system error did not record them as valid voters.
5. Vote auditing, recount or contest should be possible.
6. Incremental Deployment: The new system should be retrofitted with the old system for comparison. Two separate counts should be generated with a way of tying one persons' vote on both machines by means of something like timestamps. We do this in order to identify the errors in the new – or even old – system.
7. Voting Software and Platform: The operating system should be dedicated and lightweight, including only things necessary for voting. Software should be small and simple to keep security bugs minimum. It should also be open source (debatable).
8. System Testability: Testing should be able without having a separate test mode. In this way, the system can truly be checked. Therefore, there should be some way of distinguishing test ballots from real ballots.
9. Simple User Interface: This should require bare minimum user training for him/her to cast his/her vote. It should minimize user rejection of the system. We do not want the user to be frustrated. There should be sufficient user education tools in the form of pamphlets or dummy systems.
10. Vote Selling: The system should not allow any form of vote confirmation that will encourage vote selling. Therefore, receipts or any transportable confirmation of the vote should not be permitted.

11. Maximizing Throughput: The amount of time spent on the system by a voter should be minimized. This will not be an area of high focus for our project.

Security Requirements

1. Voter Authentication: Verifying that the voter is registered to vote and is voting for the county he/she is registered in.
2. Vote Integrity: The electronic ballot cannot be modified.
3. Vote Privacy: No one can learn how an individual person has voted.
4. Vote Duplication: No one should be able to vote more than once.
5. Vote Reliability: No vote should be lost. We should aim for zero residual votes.
6. Vote Verifiability: Voters can verify that their ballots were counted correctly and protest anonymously. Anonymous protesting may be an optional feature because that would require a sophisticated cryptographic scheme.
7. Software Certification: The software should come from a reliable source. It is not easy to verify this requirement.
8. Software Security: It should also not be possible to modify or corrupt it.
9. Secure Information Exchange: User authentication or vote information transmitted between the client and the server should be secure.
10. Safe Vote Storage: After the election period, the votes should be stored for auditing purposes.

The above are guidelines that we will adhere to but our focus will be on the following issues.

- Voter registration
- A way to incrementally deploy voting systems
- Generate audit trails