Johns Hopkins
Engineering for Professionals

SOVOS

Secure Online Voting System

Systems Engineering Project
Oral Presentation

Haijing ‘Henry’ Chen

May 1, 2018

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Agenda

SOCOS

- Introduction
- Proposed System
- Deliverables
  - Requirement Analysis and CONOPS
  - Functional Analysis
  - Trade Study
  - Conceptual Design
  - System Specification
  - Test and Evaluation
  - Risk Management
- Future Developments
- Project Wrap-up

About Me

SOMOS

- Background
  - Software Engineer – KEYW Corp. (Current)
  - Software Engineer – Northrop Grumman
  - B.S. in Electrical Engineering – University of Maryland
- Experience – Full-Stack software developer
  - Web development
  - Server-side software development
  - Database management
  - Linux system management
- Interests
  - Web application technologies
  - Online gaming

Project Proposal

SOVOS

- Secure Online Voting System (SOVOS)
- Software application to allow online voting that is accessible, secure, reliable, and auditable.
- Secured from vulnerabilities through rigorous analysis and implementation of modern security technology to protect the software, data, hardware, and network.
- Work in concert with existing voting and voter registration systems in the state of Maryland, including auditing and recounting.

New System for an Online Age

SOVOS

- Evolution of legacy non-digital systems into online software applications
- Availability of mobile, hand-held computing devices
- Availability of Internet connectivity
- Increased industry focus on software security

- Increased voter turnout and participation
- Ease of use
- Less time spent
- Decrease stress on physical polling places and election volunteers

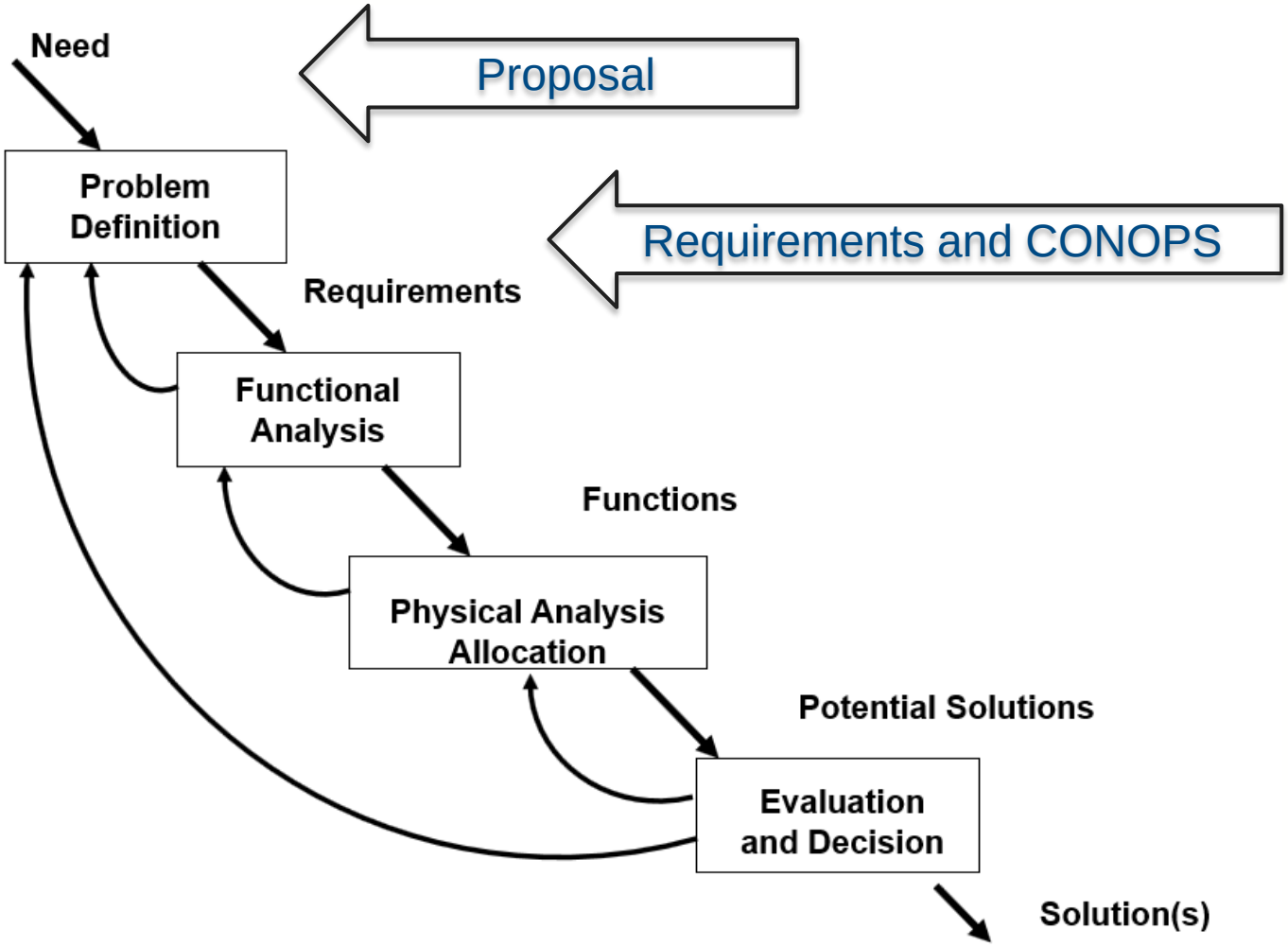
Consider the Limitations

SOVOS

- Voter confidence – counteract ineligible voting, ballot stuffing, integrity of ballots cast
- Security of online systems
  - Existing security technology
  - Misuse or incorrect implementation of security
  - Unknown security threats
- Each state has different voting systems
- Integration with existing voting systems - Maryland

System Engineering Approach

SOSOS



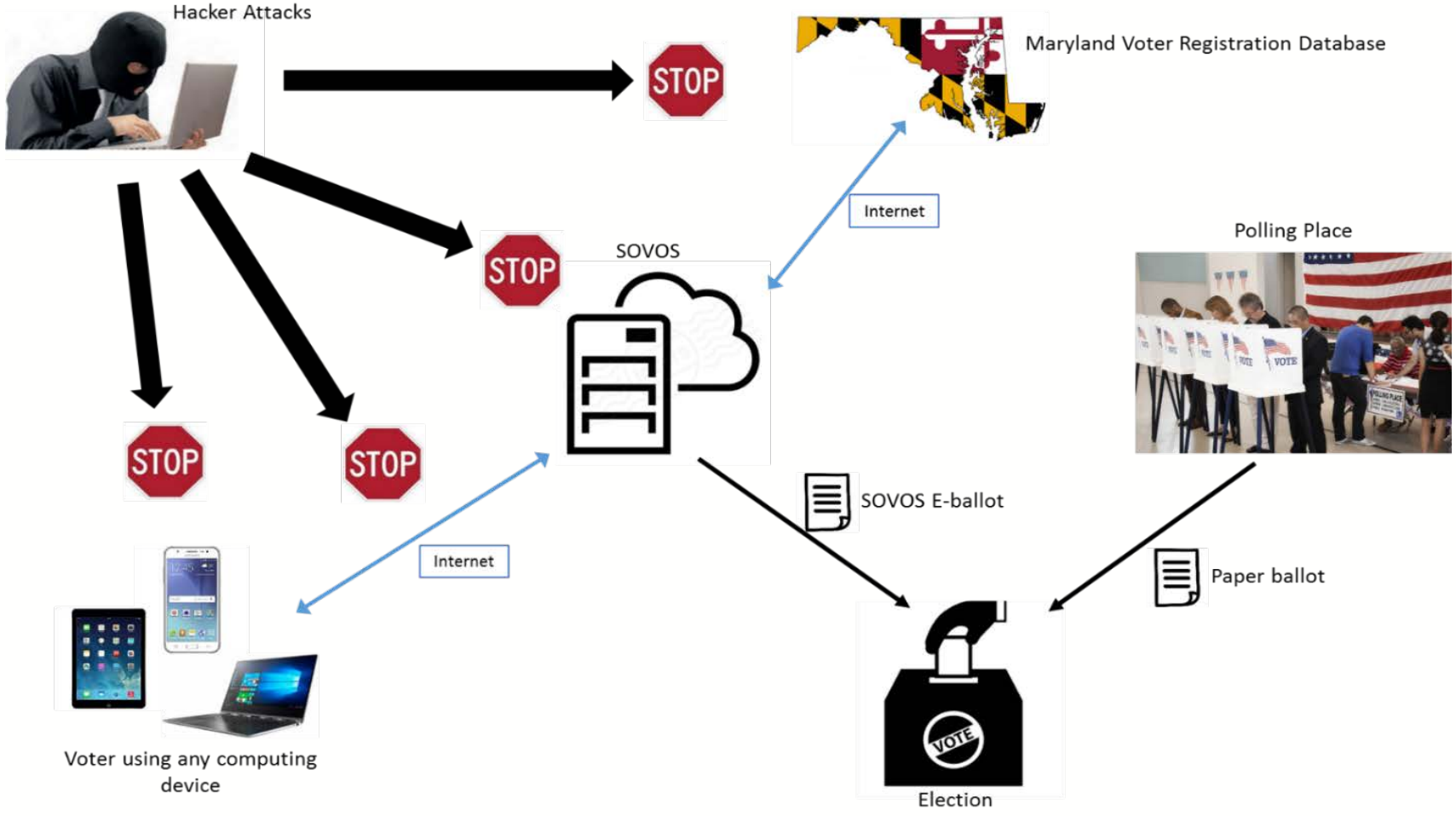
Requirements and CONOPS

SOSOS

- CONOPS
- Requirement organization
- Methods used to gather requirements
- Operational Requirements
- Requirement metrics

Concept of Operations

SOVOS



Requirement Organization

SOSOS

- Traceability
  - x.x.x.x.x numbering scheme. 1.1.1 is a child requirement of 1.1.
  - Track the source of requirements to project deliverable
- Metadata
  - KPP – Key Performance Parameters
  - Potential trade study
  - Quantitative?
  - Verification Method
    - *Inspection*
    - *Analysis*
    - *Demonstration*
    - *Test*

Requirement Gathering

SOSOS

- Project proposal to CONOPS to operational requirements
- Needs Analysis
  - Technological Opportunity
  - Operational Deficiencies
  - Operational Analysis
- Contact key stakeholders
  - Local politicians
  - Local political parties
  - Contact SMEs
- Subject area research
  - Analysis of existing online voting systems: Australian iVote, Estonia, Voting in Kenya, Helios Voting

Operational Requirements

SOSOS

Req #	KPP	Requirement
1	y	System shall allow voters to vote online using a computer device connected to the internet.
2	y	System shall be able to simultaneously collect votes from all voters attempting to vote online.
3	y	System shall tabulate votes collected and produce voting results for online votes.
4	y	System shall leverage the existing Maryland state voter registration database, which will be the source system of record for voter registration information.
5	y	System shall work in conjunction with existing voting system in the state of Maryland and shall not exclude voters from using existing voting methods.
6	y	System shall protect against hacker attacks on all system components.
7	y	System shall adhere to U.S. federal laws and regulations on voting and elections.
8	y	System shall adhere to Maryland state laws and regulations on voting and elections.

Operational Requirements

SOVOS

Req #	KPP	Requirement
9	y	System's online voting process shall be easy to learn, accessible, and have a time commitment of less than in-person voting.
10	y	System shall be auditable to verify the integrity of the system functions and requirements.
11	y	System shall support election recounts.
12	y	System shall provide evidence to voters and election officials that votes are recorded as they intended and not tampered.
13	y	System shall protect the privacy of the voter and the anonymity of the vote.
14		System shall have redundancy and recovery procedure in place to protect all data produced.
15		System shall be able to automatically diagnose and indicate that the system is operating correctly.
16		System shall have complete and comprehensive documentation and training for system operation.

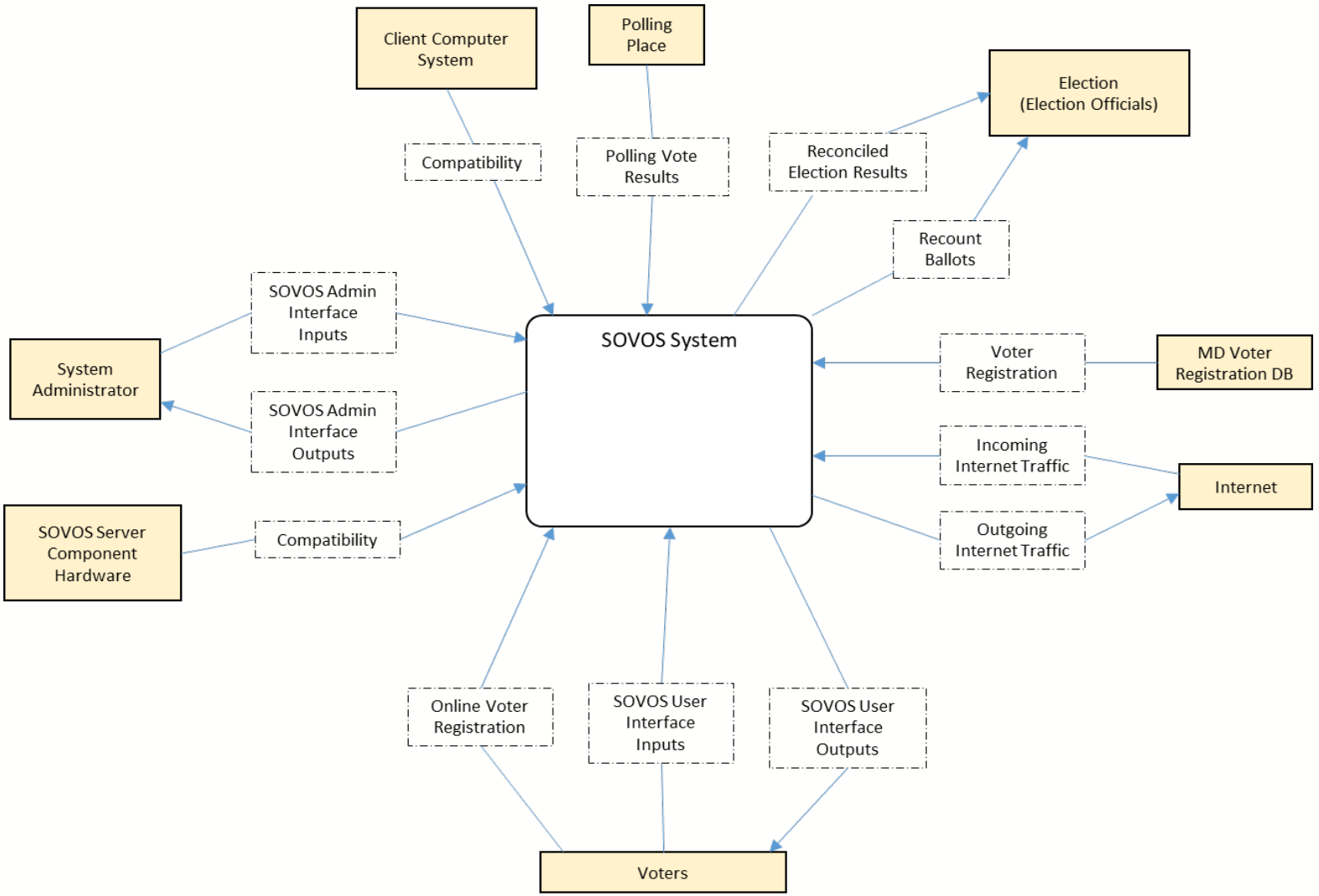
Concept Exploration

SOVOS

- Deep dive into the various subject areas for SOVOS
  - Operational concepts of existing voting systems
  - Maryland election laws and governing bodies
  - Federal election laws and governing bodies
  - Industry standard best practices for software and computer security
  - Data redundancy
  - Protecting system network
  - Vote recount
  - Auditing a software system
  - Anonymity of the vote
  - Encryption

System Context Diagram

SOVOS



Requirement Metrics

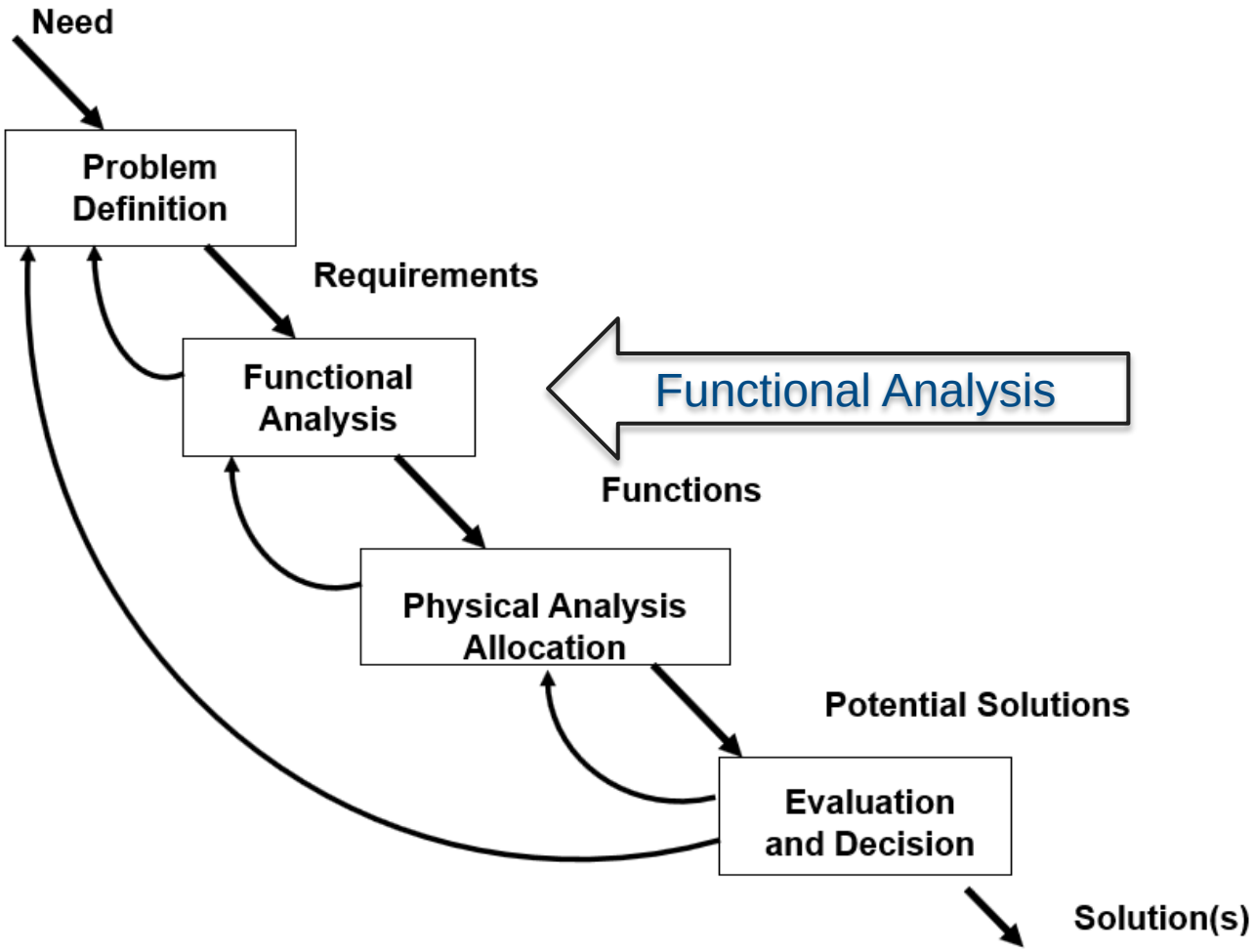
SOSOS

- Total System Requirements: 96
  - Operational Requirements: 16
  - Trade Study Requirements: 6
  - KPP Requirements: 61
  - Quantifiable Requirements: 4

  - Inspection Requirements: 32
  - Analysis Requirements: 17
  - Demonstration Requirements: 33
  - Test Requirements: 14

System Engineering Approach

SOSOS



Functional Analysis

SOSOS

- Logical generation of subsystems based on grouping system requirements
- Analyze system requirements for functions
- Decompose top level functions into lower level functions
  - Functional flow and sequence
  - Functional hierarchy
  - Functional interfaces
- Functional system context

Subsystems

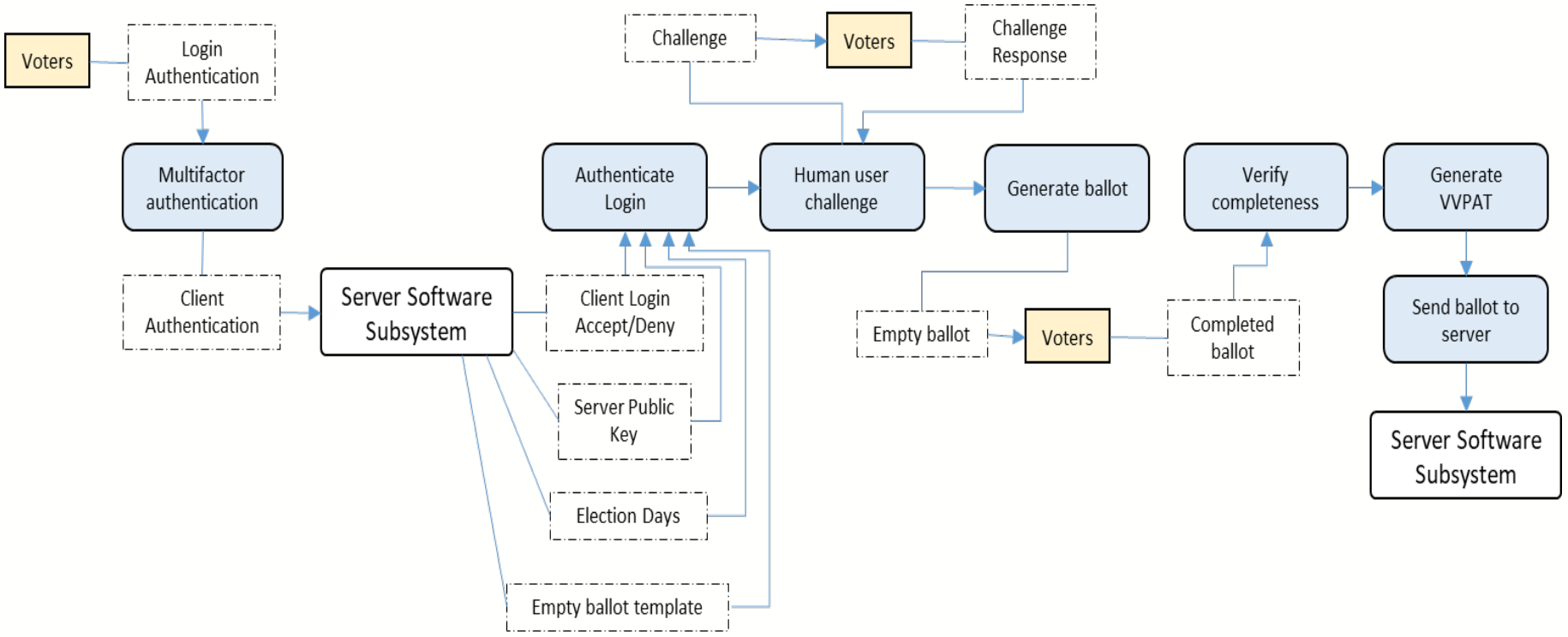
SOSOS

- Software Security
- Voter Registration
- Vote Reconciliation
- Client Software
- Server Management
- Server Software
- Database Management
- Network Security

Functional Requirements

SOSOS

- Client Software subsystem functional flow



Generating Functions

SOVOS

Req #	Requirement	Function?	Subsystem
1.5	System shall allow users to fill out and submit their ballot for an election online.		Client Software
1.5.1	System client component shall present to the user a ballot user interface that allows user the fill out the ballot that is specified by the SOVOS system.	Yes	Client Software
1.5.2	System ballot user interface shall accept user inputs for filling out the ballot.	Yes	Client Software
1.5.3	System ballot user interface shall verify the ballot is filled out correctly.	Yes	Client Software
1.5.4	System ballot user interface shall allow the user to submit a valid and completed ballot.	Yes	Client Software
1.5.5	System ballot user interface shall alert the user if the current date is not one of the election dates and prevent user from filling out the ballot or viewing the empty ballot.	Yes	Client Software

Function Interface Traceability

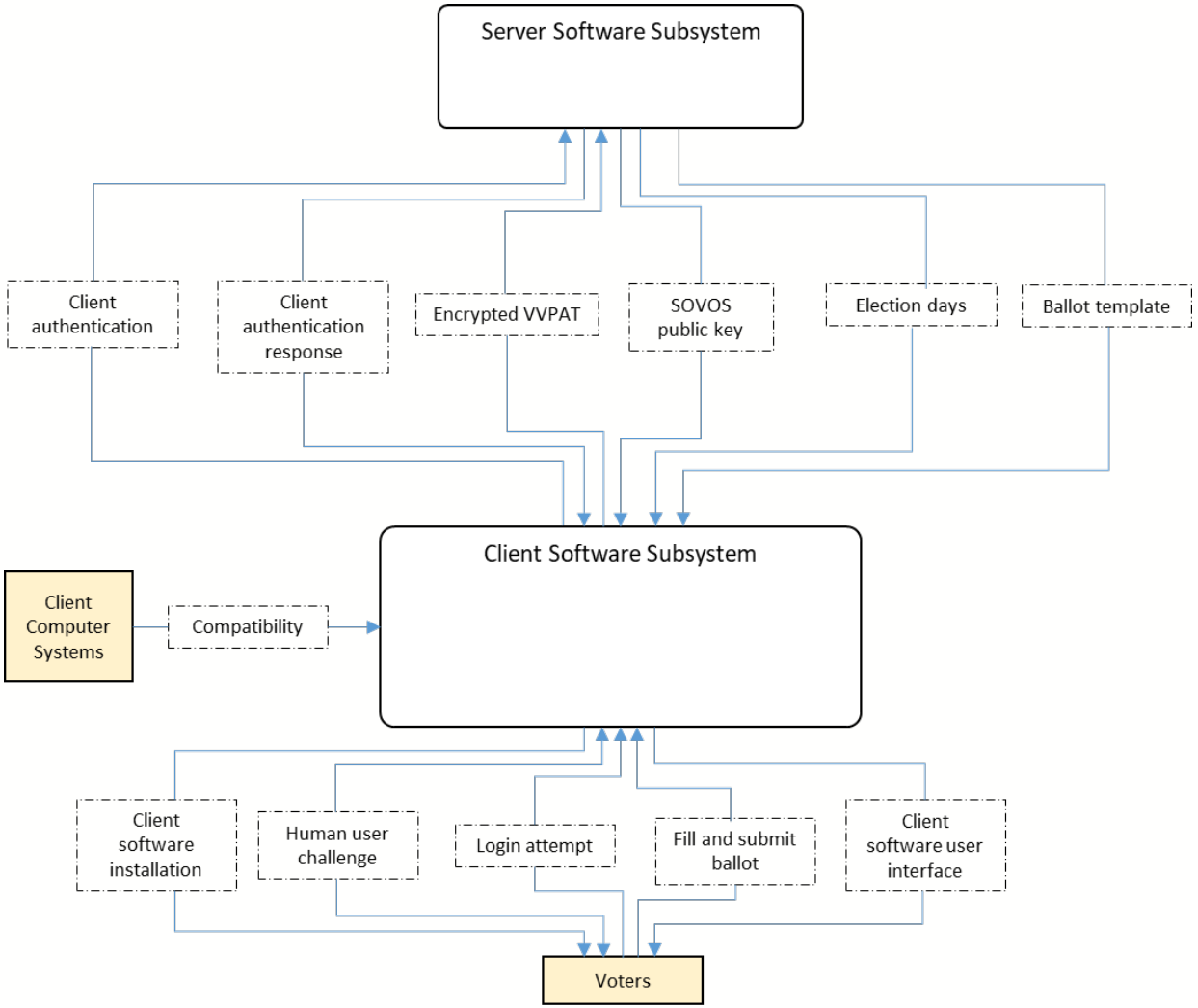
SOVOS

- Interface directionality; source, destination, data transferred
- Traceability to function and report section

ID	From Key	From Item Name	To Key	To Item Name	Data transferred	Function(s)	FAR Section	Subsystem
2001	SOVOS	Software Security	External	Voters	SOVOS client software	6.5.1	2.1	Software Security
2002	SOVOS	Software Security	SOVOS	Server Management	SOVOS server software	6.5.2	2.1	Software Security
2003	External	MD Voter Registration System	SOVOS	Voter Registration	Online voter registration	4.6	2.2	Voter Registration
2004	SOVOS	Voter Registration	External	Voters	Login authentication information	4.1.1	2.2	Voter Registration

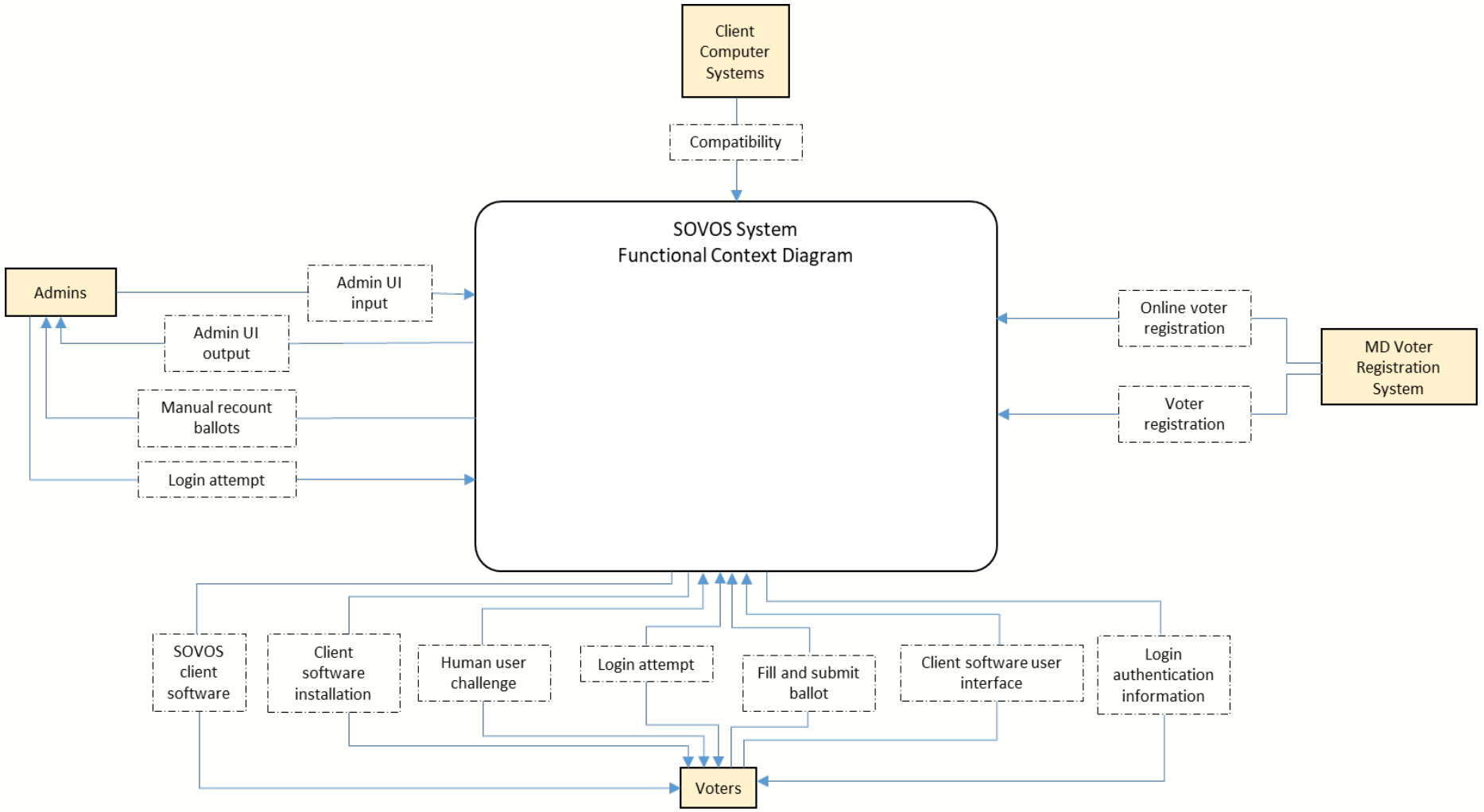
Subsystem Functional Context

SOVOS



System Functional Context

SOVOS



Trade Study

SOSOS

- Analysis of alternatives on system requirements
  - Rejected trade studies
  - Informal trade studies
- Formal trade study

Formal Trade Study

SOSOS

- Methodology
  - Select topic
  - Define alternatives
  - Define criteria
  - Conduct SME interviews
  - Define criteria weights
  - Define criteria utility functions and utility of each alternative
  - Generate results
  - Perform sensitivity analysis

Trade Study

SOVOS

- Topic - Determine best design approach to multifactor authentication for the SOVOS client user interface

- Types of multifactor authentication
  - Basic – username, password, PIN, security question, text message
  - Token – physical or software token with automatically generated login key
  - Biometrics – fingerprint, palm scan, facial recognition, voice recognition

Alternatives

SOSOS

- 1. “All of the above” “3-factor” authentication – uses all 3 types
- 2. Simple authentication – uses only basic authentication
- 3. Basic and biometrics – uses everything except tokens
- 4. Basic and tokens – uses everything except biometrics

Criteria

SOMOS

- A. Usability
- B. Privacy Concerns
- C. Security
- D. Accessibility
- E. Implementation Complexity
- F. Maintainability

SME Interviews

SOSOS

- SMEs
  - Computer security expert
  - Senior software engineer
  - Experience software tester
- Questions
  - Rank relative importance of each criteria to each other
    - *Used to generate the pair-wise comparison matrix*
  - Estimate the effectiveness of each alternative for each criteria
    - *Used to generate the utility of alternatives*
  - Additional feedback and subject matter research used to generate utility functions

Results

SOCOS

- Multifactor authentication using basic authentication and token.

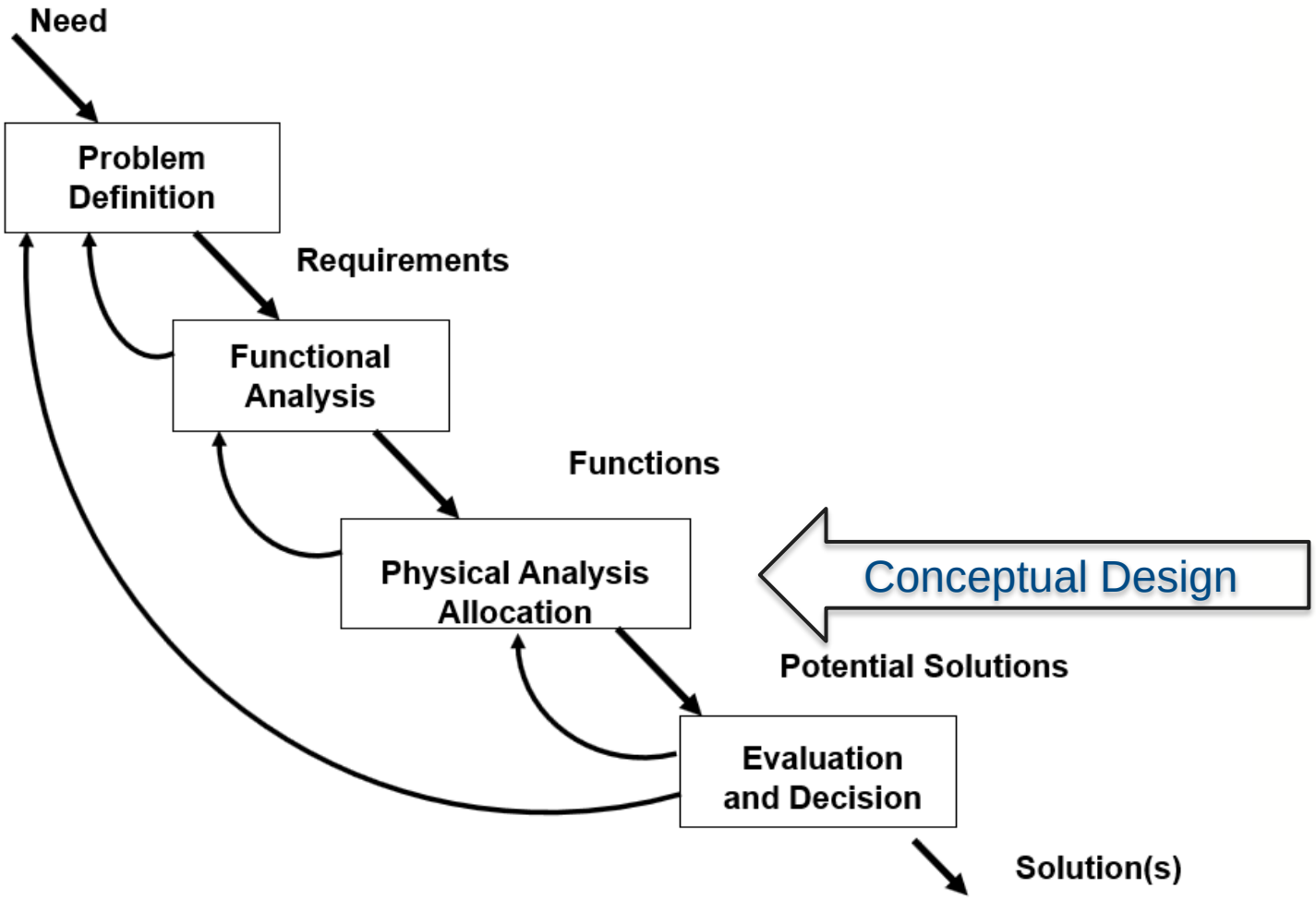
Alternatives	Criteria A	Criteria B	Criteria C	Criteria D	Criteria E	Criteria F	Criteria Weights	Results
1	0.8000	0.4000	9.5000	6.5000	0.8000	2.5000	0.3262	4.35482
2	1.0000	1.0000	6.0000	1.0000	9.2000	9.0000	0.1423	3.53356
3	0.9000	0.4000	9.0000	6.5000	3.0000	4.5000	0.3531	4.41155
4	0.9000	1.0000	9.0000	8.4000	4.8000	4.5000	0.0836	4.75891
							0.0573	
							0.0373	

- Sensitivity analysis did not affect the final result.

- Updates to requirements and system design

System Engineering Approach

SOSOS

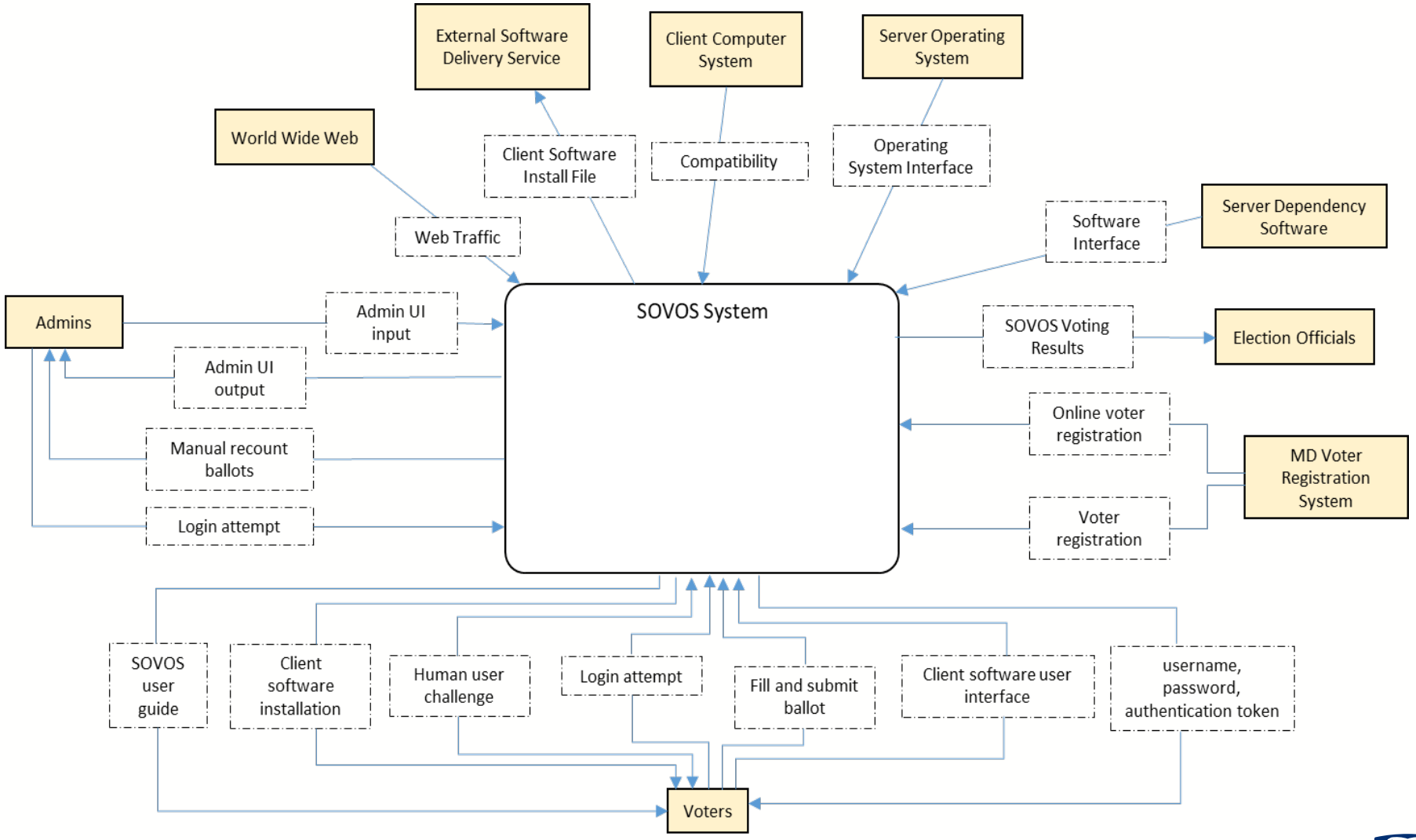


Conceptual Design

SOSOS

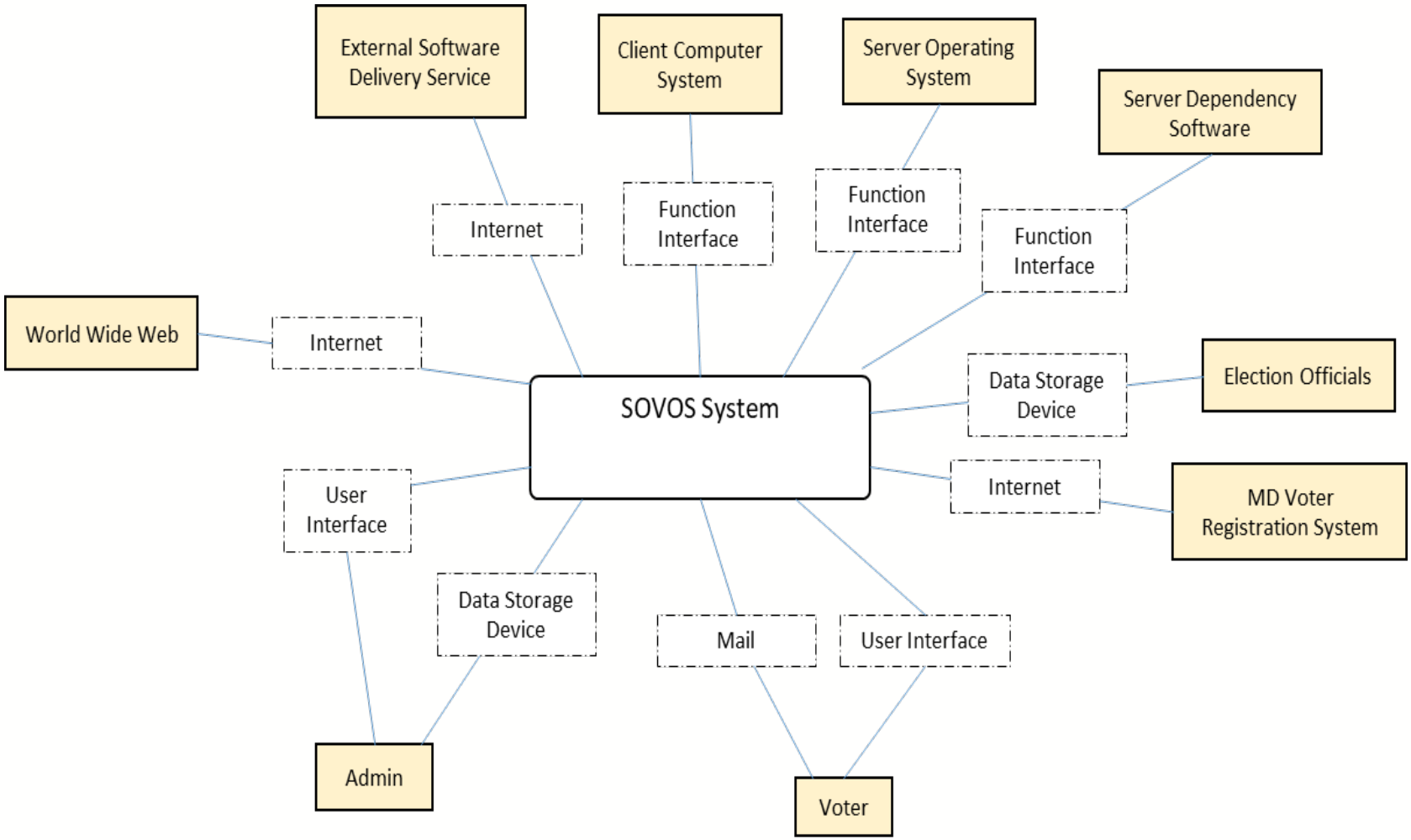
- Initial physical architecture and conceptual design
- Functions translated to physical components
- Functional system context and interfaces translated to physical context and interfaces
- Traceability of physical components to functions and interfaces
- Conceptual design of subsystems based on functional context of subsystems

System Context



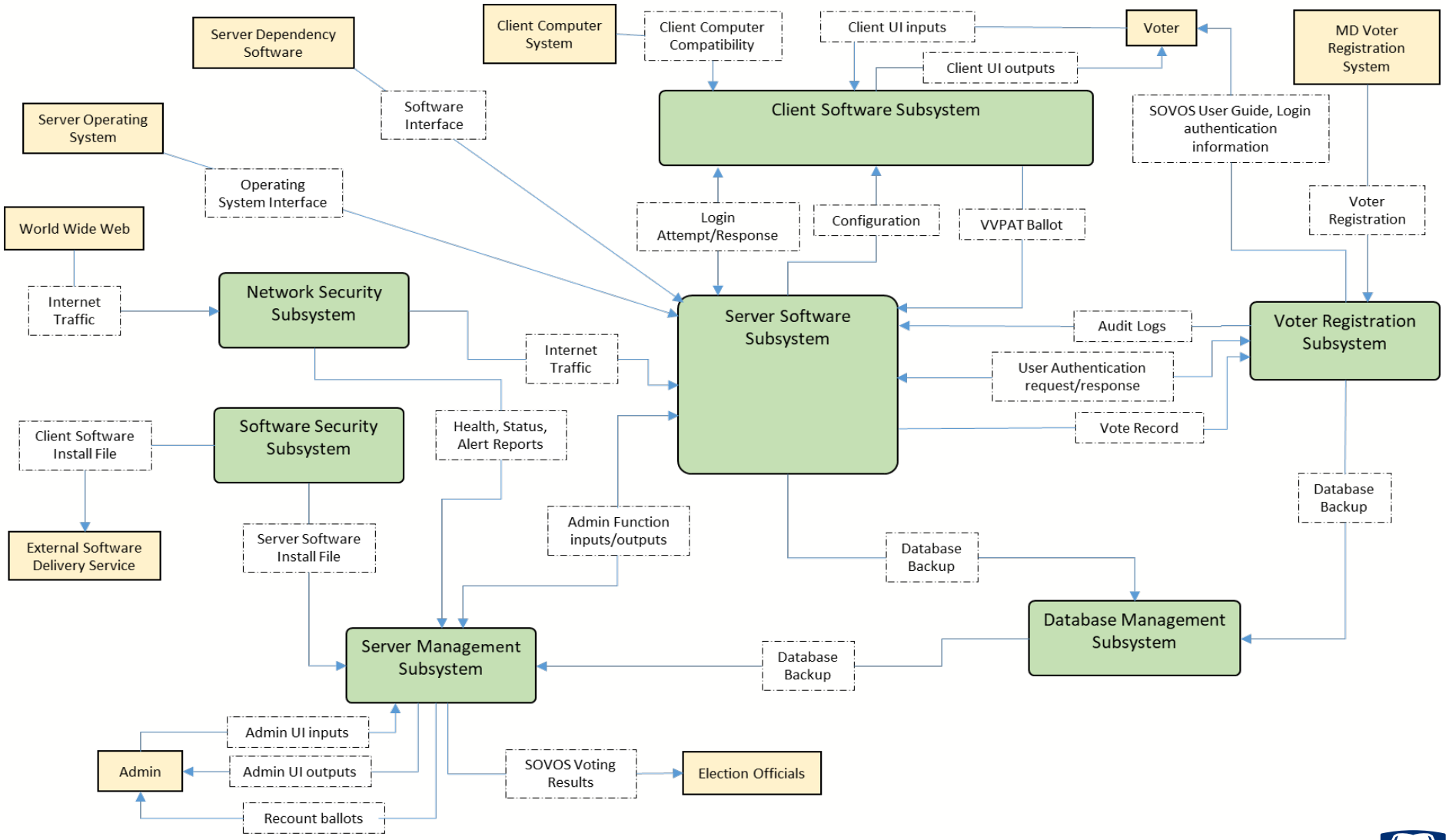
Physical Linkage Context

SOVOS



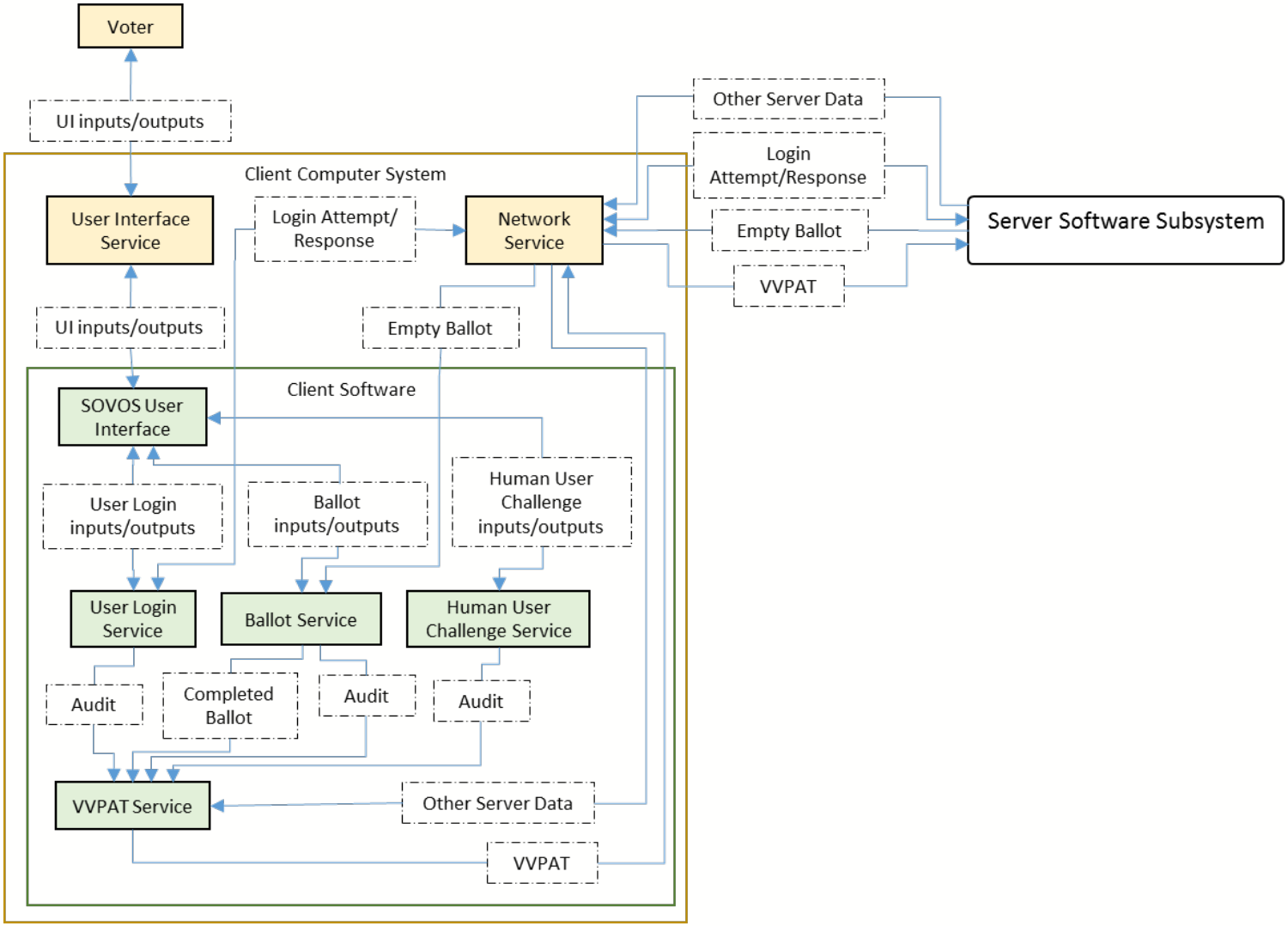
System Concept Block Diagram

SOVOS



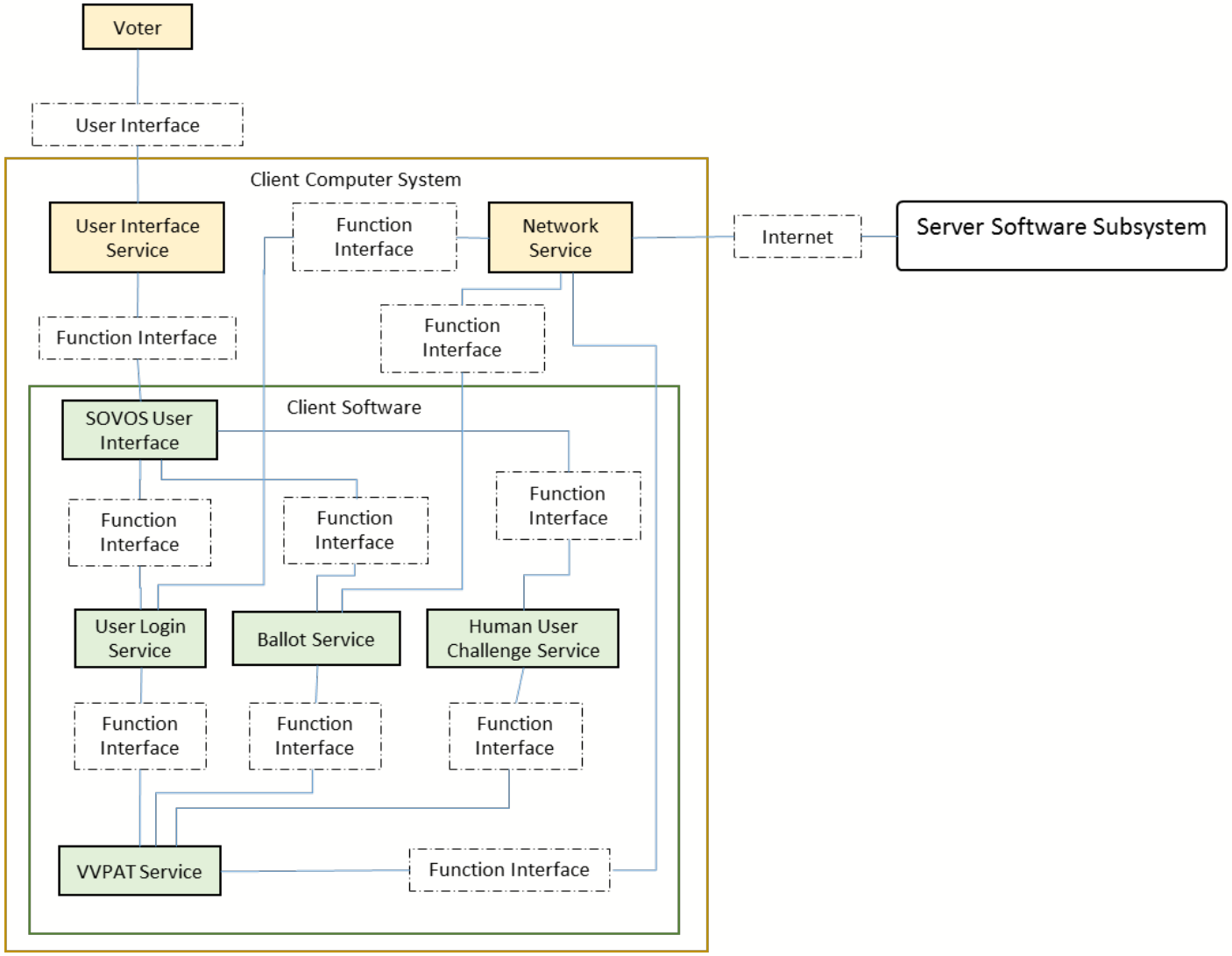
Physical Block Diagrams

SOVOS



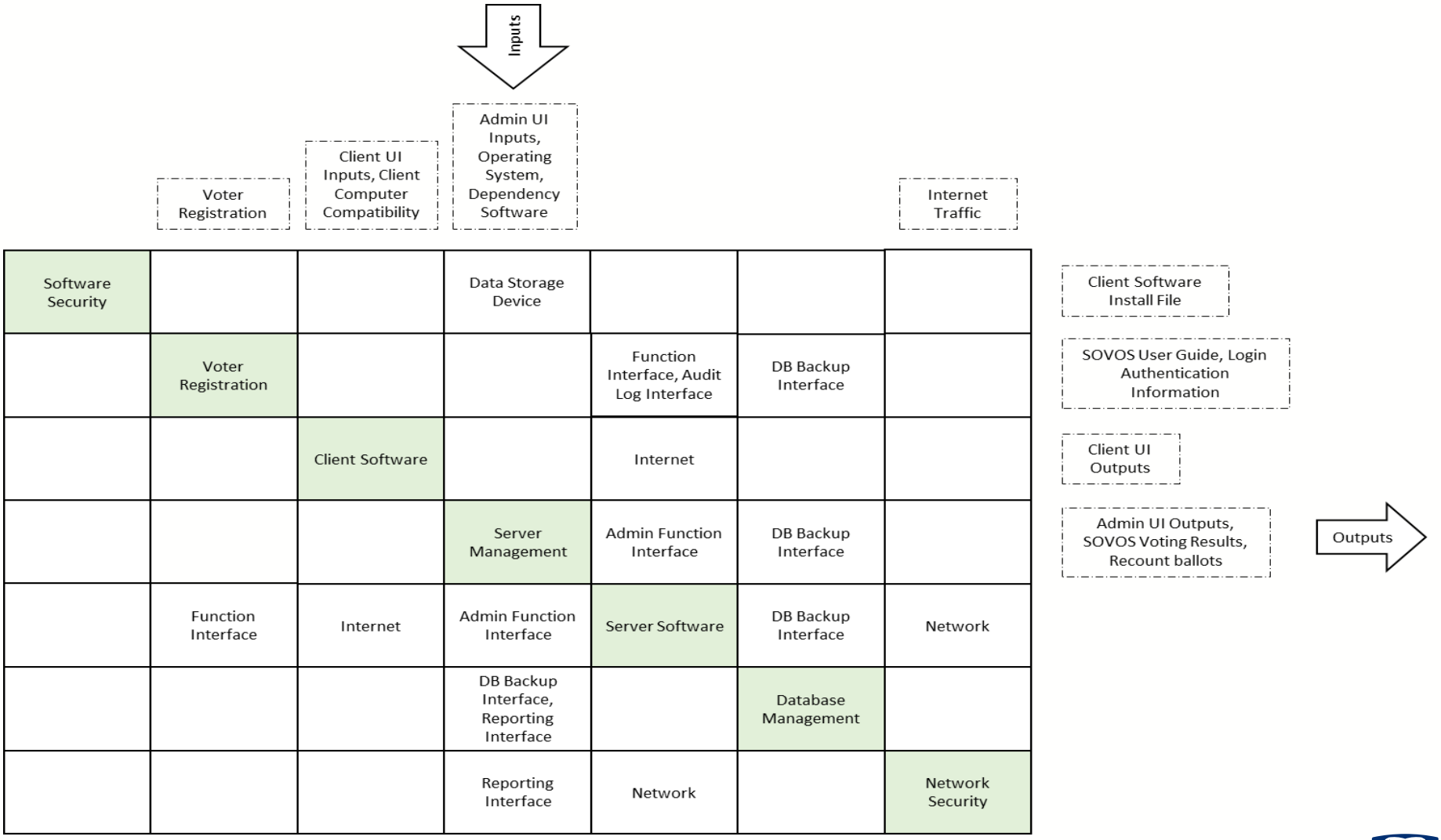
Conceptual Block Diagrams

SOVOS



Physical N2 Diagram

SOVOS



Component Traceability

SOVOS

ID	Subsystem	Item Name	Item Type	Requirement Traceability
1033	Client Software	Client Computer System	External Component	
1034	Client Software	Client Software	Internal Component	7.3.1, 7.3.2, 7.4.1, 7.4.2, 9.2.1, 9.2.2, 9.2.3, 9.2.4, 13.1.1
1035	Client Software	Voter	External Component	
1036	Client Software	Internet	Interface	
1037	Client Software	User Interface Service	External Component	
1038	Client Software	Network Service	External Component	
1039	Client Software	Function Interface	Interface	
1040	Client Software	User Login Service	Internal Component	1.1.1.1, 1.6.1.1, 1.6.1.2, 1.6.1.3, 1.6.1.4, 1.6.1.5, 1.6.1.7, 9.4.1, 10.6.1
1041	Client Software	Human User Challenge Service	Internal Component	6.9.1, 6.9.2, 6.9.3, 6.9.4, 9.4.2, 10.6.2
1042	Client Software	Ballot Service	Internal Component	1.5.1, 1.5.2, 1.5.3, 1.5.4, 1.5.5, 9.4.3, 10.6.3, 12.2.1
1043	Client Software	VVPAT Service	Internal Component	12.1.1.1, 12.1.1.2, 12.1.1.3, 12.2.2, 12.3.1, 13.1.2
1044	Client Software	SOVOS User Interface	Internal Component	1.1.1.2, 1.1.1.3, 1.2.1, 1.2.2, 1.2.3, 1.2.4

System Specification (A-Spec)

SOVOS

- Generated the complete system specification for SOVOS
- Methodology
  - Review traceability of physical components and interfaces to functional requirements and interfaces.
  - Review all existing requirements for clarity, unambiguity, and traceability
  - Review of all requirement metadata for accuracy and applicability
    - *KPPs*
    - *Trade studies*
    - *Verification methods*
  - Review of Quantitative Requirements for performance metrics
  - Review remaining system concerns

Requirement Metrics

SOSOS

Total Requirements:	251
Operational Requirements:	16
Trade Study Requirements:	6
KPP Requirements:	62
Quantifiable Requirements:	37
Functional Requirements:	139
Inspection Requirements:	36
Analysis Requirements:	45
Demonstration Requirements:	105
Test Requirements:	63
Demonstration and Test:	2

	RTM Total	New Requirements	Modified
RAR	96	96	
FAR	227	131	10
TSR	233	6	2
CDR	235	2	2
A-Spec	251	16	5

A-Spec Products

SOMOS

- Requirements Traceability Matrix
- Functional Interface Traceability Matrix
- Physical Component Traceability Matrix

System Concerns

SOSOS

- Software performance, failure, and reliability
  - Software response time and user perception
  - Mean time to repair (MTTR)
- Approximation of system usage
  - Server architecture
  - Network architecture
- System assumptions
  - Leverage existing Maryland Voter Registration Database
  - Using US Postal Service
- Cost and budget
- Security concerns

Test and Evaluation (TER)

SOSOS

- Initial TER Scope
  - Client Component
    - *Client Software subsystem (whole)*
    - *Software Security subsystem (part)*
- Methodology
  - Analyze System Specification requirements for verifiability.
  - Generate Integration Approach, the approach and plan for how the system will be integrated and tested to verify functionality and performance requirements.
  - Generate Qualification Approach, the approach and plan to subject the completed system to system qualification tests
  - Generate the Verification Cross-Reference Matrix (VCRM), correlating Integration & Qualification tests to requirements.

Test Approach

SOVOS

- Modular approach to testing, “system builds”
- Integration Approach
  - Functional Test Environment – Test the functions of individual components
  - Integration Test Environment – Test the integration of components
- Qualification Approach
  - System Test Lab – Complete physical implementation in a test environment
  - SOVOS System (Beta Testing) – Production system prior to delivery

Test Plan Matrices

SOOS

- Test ID
- Test Name
- Verification method
- Objective and details
- Test environment
- Test input
- Test output
- Requirements tested
- Expected results or pass/fail criteria

VCRM

SOSOS

- Used to cross-reference tests and verification to the requirements.
- Tracks additional changes made to requirements to clarify tests
- Every requirement is tested by at least one test
- Verification method of the requirement should match the verification method of the test

Risk Management

SOVOS

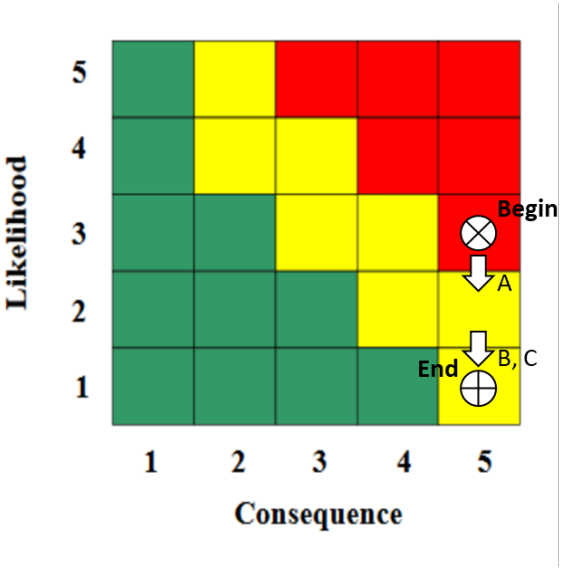
- Defined the Risk Management Plan for SOVOS
- Tracked progress of risk mitigation over the course of the project

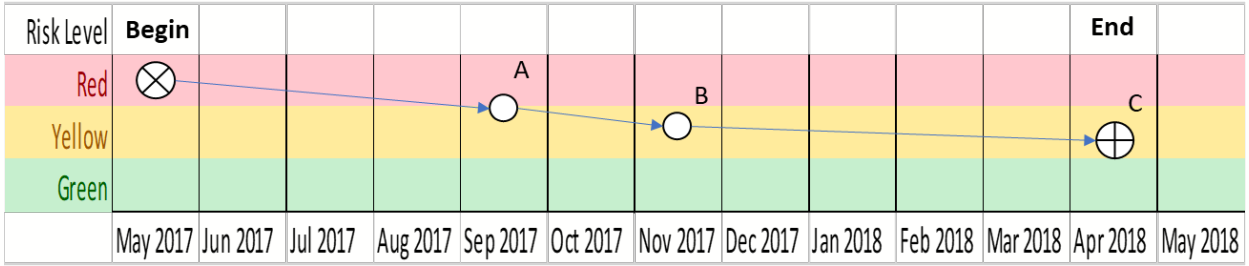
Risk	Initial Likelihood	Initial Consequence	Final Likelihood	Final Consequence
Failure to prevent known security threats	3	5	1	5
Unknown security threats	4	5	1	5
Error in tabulating vote results	3	4	1	3
Lack of stakeholder confidence	3	5	1	5
Lack of stakeholder feedback and SME input	3	4	1	4
Lack of domain knowledge	3	5	1	5
Unforeseen schedule risk	4	4	1	4

Risk Management Results

SOSOS

- Risk 1: Failure to prevent known security threats
- Mitigation:
  - A. Subject matter research
  - B. RAR
  - C. TER

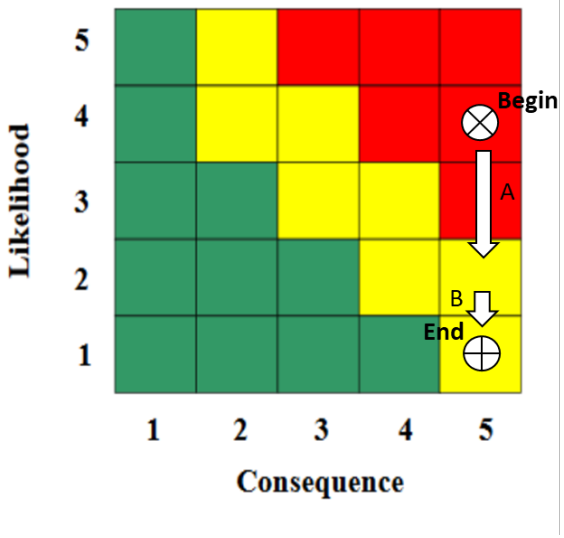


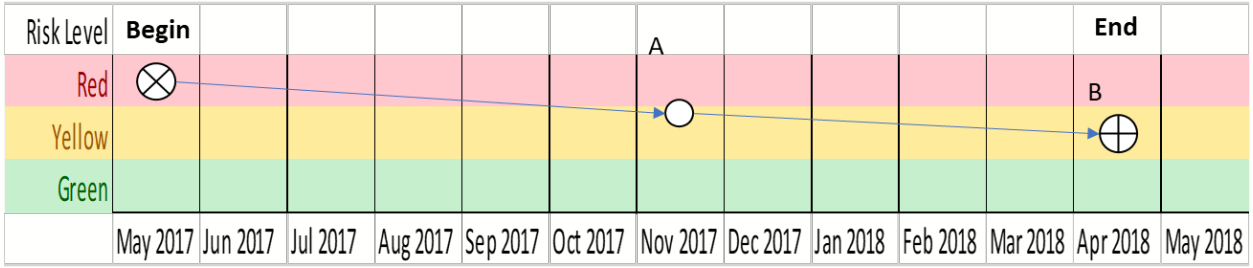


Risk Management Results

SOMOS

- Risk 2: Unknown security threats
- Mitigation:
  - A. RAR
  - B. TER

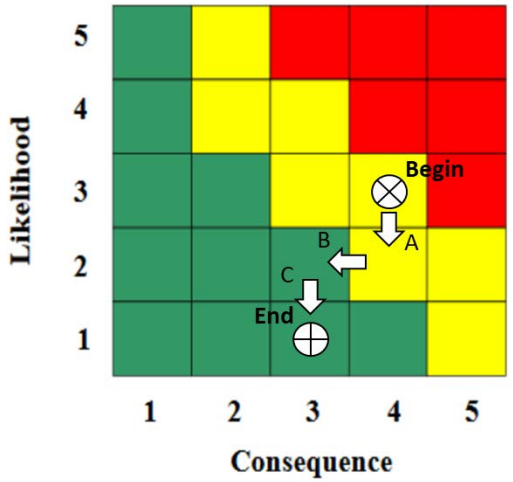


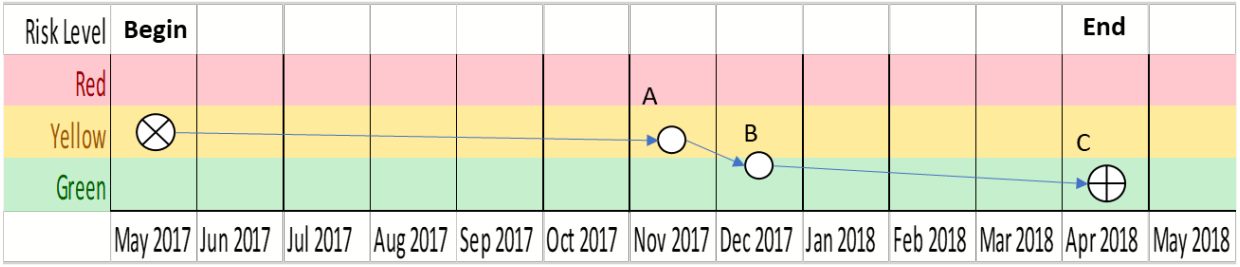


Risk Management Results

SOSOS

- Risk 3: Error in tabulating vote results
- Mitigation:
  - A. RAR
  - B. TSR
  - C. TER

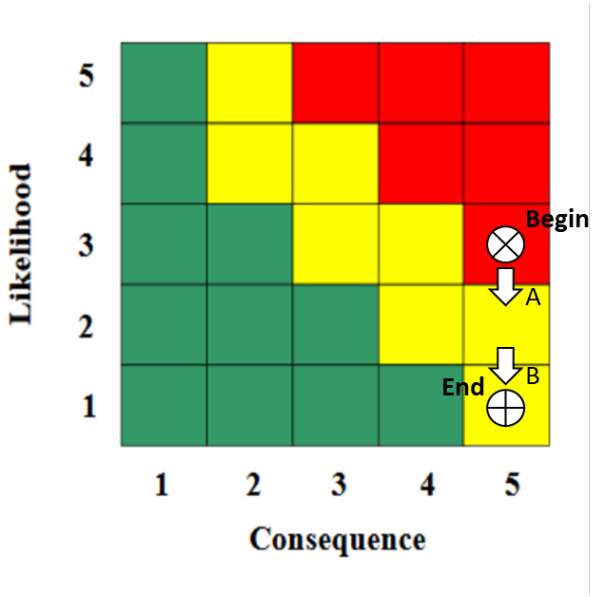


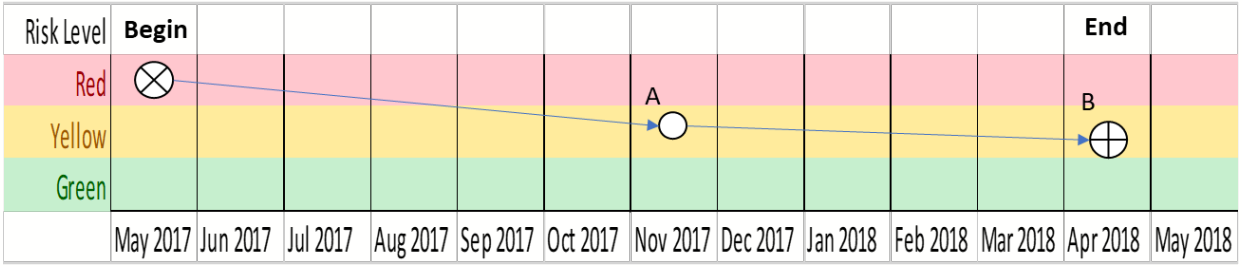


Risk Management Results

SOSOS

- Risk 4: Lack of stakeholder confidence
- Mitigation:
  - A. RAR
  - B. TER

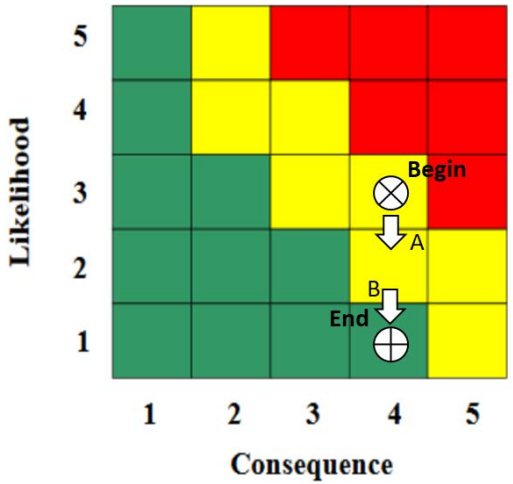


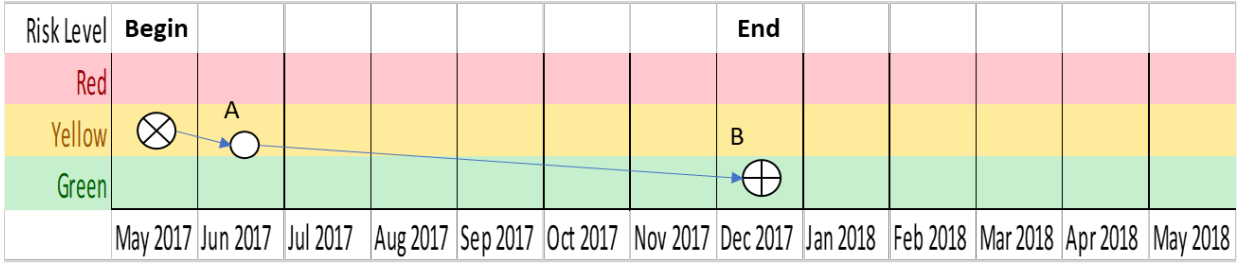


Risk Management Results

SOSOS

- Risk 5: Lack of stakeholder feedback and SME input
- Mitigation:
  - A. Quickly reach out to key stakeholders
  - B. TSR

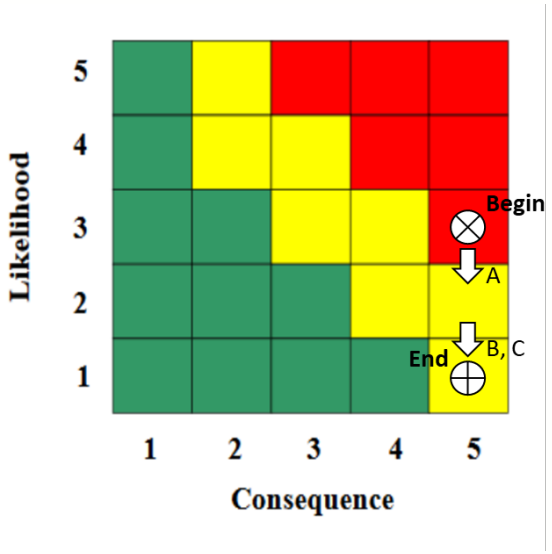


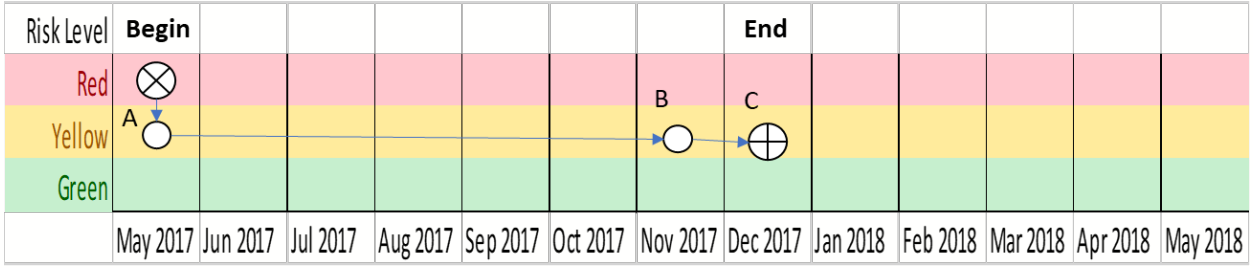


Risk Management Results

SOSOS

- Risk 6: Lack of domain knowledge
- Mitigation:
  - A. Proposal
  - B. RAR
  - C. TSR

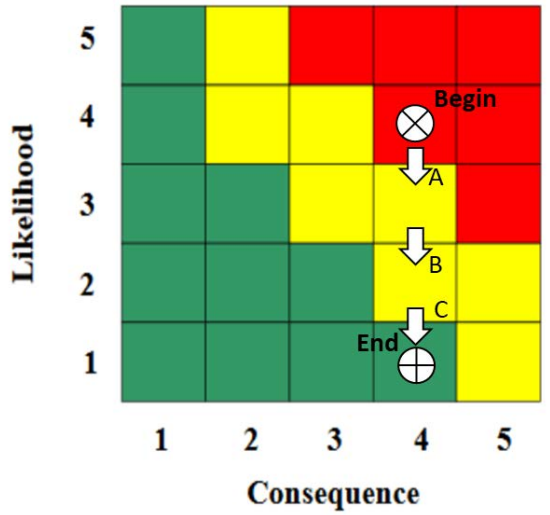


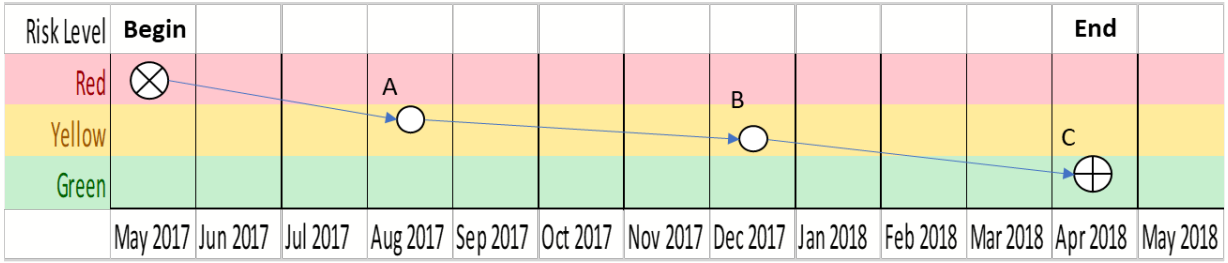


Risk Management Results

SOSOS

- Risk 6: Unforeseen schedule risk
- Mitigation:
  - A. Semester Extension
  - B. Semester Extension #2
  - C. Completion of all deliverables





SOVOS System Summary

SOVOS

- A security focused system engineering design for a modern online software application that allows voters to quickly and easily vote in state and federal elections.

- Complete SOVOS System implementation will need to consider additional security factors:
  - Background investigation for employees
  - Security analysis of software development process
  - More rigorous security testing and feedback loop
    - *NSA*
    - *FBI*
    - *Professional hackers*

SOVOS is Secure!

SOVOS

- Security in people
  - Background check and security vetting for everybody involved in SOVOS
- Security in process
  - Security focused development
  - Software dependency security analysis and management
  - Security analysis of design and implementation
- Security in product
  - Security focused testing and validation
  - System penetration testing with security experts
  - Secure and validated delivery of software to users

SOVOS is Secure!

SOVOS

- Security in registration
  - User registration through existing Maryland Voter Registration system
  - First-time login information physically delivered to user
- Security in login
  - User login using multifactor authentication including physical authentication token
- Security during use
  - End-to-end encryption of communication between client and sever
  - Human verification, challenge questions, CAPTCHA
  - Vote-verified “paper” audit trail (VVPAT)

SOVOS is Secure!

SOVOS

- Security in setup
  - Check list for server setup and configuration
  - Automated diagnostic for server setup
- Security in operation
  - Health and status monitoring
  - Database backups
  - Network intrusion detection and mitigation
- Security in management
  - Background check and security vetting for everybody involved in managing SOVOS servers
  - Two-person rule for admin access
  - Auditing of all admin functions performed

SOVOS is Secure!

SOVOS

- Security of the vote
  - VVPAT
  - Auditing
    - *Audit all key SOVOS functions, including client and server functions*
    - *Audit all admin functions*
    - *Database backups for audit logs*
  - Recount
    - *Generate paper ballot from VVPAT to facilitate manual recount*

Schedule Evaluation

SOCOS

WBS Item	Task Title	Expected Date	Expected (hours)	Actual Date	Actual (hours)
1	Project Conception and Approval	5/14/2017	66	5/14/2017	66
2.1	Requirements Analysis and CONOP	5/28/2017	60	11/16/2017	80
2.2	Functional Analysis	6/11/2017	70	12/3/2017	80
2.3	Trade Study	6/18/2017	45	12/7/2017	65
2.4	Conceptual Design	7/2/2017	70	1/28/2018	90
2.5	System Specification	7/16/2017	60	4/12/2018	75
2.6	Test and Evaluation	7/30/2017	65	4/14/2018	65
2.7	Risk Management	8/6/2017	40	4/12/2018	55
3.1	Final Report	8/20/2017	81	4/27/2018	60
3.2	Oral Presentation	8/24/2017	32	5/1/2018	22
	Total		589		658

Lessons Learned

SOSOS

- Importance of traceability of requirements, interfaces, and components
- Importance of staying on schedule
- Proper use of Word
- Importance of subject matter expertise and experience
- Scoping the project early
- Schedule more time for review of deliverables

Program Recommendations

SOSOS

- Earlier stimulation of project ideas
- Better Blackboard notifications
- Better notification for updates to SE project guidelines

Johns Hopkins
Engineering for Professionals

SOOS

Secure Online Voting System

Any Questions?

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING