



# **Modernized Elections System For the State of Washington**

**RFI No. 16-04**

**Attn: Stephanie Goebel**  
RFI Coordinator  
Project Manager

**The State of Washington  
Secretary of State**



**Submitted By:**

**ResQSoft Inc.,**  
8300 Boone Boulevard  
Suite 500 PMB5015  
Vienna, VA 22182  
(703) 821-6999

## Table of Contents

Executive Summary .....	2
1. Company Information .....	3
2. Scope of Work .....	7
i. Business Requirements .....	7
ii. Security Methods .....	8
iii. System Integration Approach and Methodology .....	9
iv. Project Management Approach and Methodology .....	10
v. Funding Approach and Cost Distribution .....	11
vi. Data Conversion and Migration .....	11
vii. User Experience Design Approach and Methodology .....	12
viii. System Support, Maintenance and SLA (Service Level Agreements) .....	14
ix. Contract Vehicles and Strategies .....	14
x. Testing .....	15
xi. Training .....	15
xii. Documentation .....	16
xiii. Voter Outreach .....	16
xiv. Timeline Estimate .....	17
xv. Cost Estimate .....	17
3. Summary .....	17

## Executive Summary

ResQSoft, Inc. (ResQSoft) is pleased to submit this Response to RFI No. 16-04 for Modernized Elections System for the State of Washington.

We are a small business with unique technology, system architecture planning and modernization project capabilities with a focus on Federal and State Government project and initiatives. With a single unified toolset, supported by sound project management principles and methodology, we can give you an automation assisted approach to:

- Building a modern elections system with high quality code that is SOA compliant and enterprise ready;
- New development, to the same platform that exists today with a consistent and maintainable body of source code that you own;
- Maintenance and enhancements, many of which can be accomplished in a declarative fashion with very little manual coding.

We support the best of the web platforms, with the ability to create code for the Java / J2EE platform or for the .NET platform, either one. Our .NET solution is based on the latest Microsoft recommendations (MVC and C#), and newer programming techniques.

In our experience, we have found that package-based implementations (COTS) are cumbersome. These projects require configuration and customization that, despite claims by vendors to be a minimum effort process, take longer than anticipated. Modifying existing functions, package definitions and building integration bridges to other interfaces requires extensive planning and testing.

ResQSoft, with our ResQSoft® Engineer technology, has an answer; and we believe it is probably the only efficient and effective way to get there. What we do is redevelop, or build the system based on your business requirements and utilizing our tool set can automate the code building process and demonstrate efficiency gains in terms of schedule and cost.

Our approach does provide:

- Application Architecture definitions that suite the department and county needs
- High quality source code, with comments, unit tests, and proper structure for the web;
- Section 508 compliant for accessibility required by the Americans with Disabilities Act;
- Service Oriented Architecture (SOA), for easy integration and with SOA interfaces created automatically; and
- Rapid delivery (less than 2 years to first deployment).

It's worth emphasizing that we automatically create SOA web services for system to system integration while the different systems are in different stages of development. Web services, of course, are easily used to allow separate modern Java or .Net applications to interoperate, as well. Our software also complies with Section 508 of the Americans with Disabilities Act;

And, because of the power of the ResQSoft® Engineer technology, we believe we can do all this far less expensively, and far faster, than anyone else. In fact, almost all of our engagements are done on a fixed price basis, and we are proud to say that we have never initiated a change control process.

We would also like to offer you a technical presentation and/or a demonstration of our technology, at your convenience, and we have Proof of Concept and Pilot programs that allow you to see the power of the technology we have applied across critical Federal and State programs.

## 1. Company Information

<b>1.1 Company Name and Address</b>	<b>ResQSoft Inc.,</b> 8300 Boone Boulevard Suite 500 PMB5015 Vienna, VA 22182 (703) 821-6999
<b>1.2 Contact Person</b>	Bill Cody Sales Account Executive ResQSoft Inc. 678-592-2600 <a href="mailto:bcody@resqsoft.com">bcody@resqsoft.com</a>

### About Us:

ResQSoft is what we sometimes call a “tool-driven services company,” specializing in software development by using very advanced automated tools to accelerate development and improve the quality of our code. This is an entirely different concept as opposed to using design or diagramming tools. We use those, of course, but our own tools are focused on actually writing code, because code, not diagrams, is what the business of computer software is really all about. We also act as a technology provider, working closely with top systems integrators.

Our core strengths are:

- ❑ **Modernizing legacy software applications** to new web-enabled systems (with SOA and high quality Java or .NET code); and
- ❑ **Developing new systems using web technologies** to meet customer requirements.

Apart from building new systems based on business requirements and stakeholder needs, we can bring your legacy systems forward to a common platform, and make the architectural and functional modifications needed to implement common / shared services and the new functionality that comes from changed requirements, regulatory demands, or simply the features and functions that could not be implemented because of the restrictions of the legacy environment. At the end, the result is a streamlined body of application software, well-structured and well commented, with standardized structures and excellent performance in production.

ResQSoft has grown steadily since its inception, building a first-rate team of architects, solution consultants and developers who are experienced with modern and legacy platforms.

### Value Proposition:

ResQSoft is in the forefront of modernizing legacy applications with the objective of helping customers transition to a modern platform within a short duration. Our comprehensive strategy, approach and methodology combined with robust tools paves the way for state agencies to:

- ❖ Transform their business processes “as is” to a newer platform (Java, .NET)
- ❖ Build SOA interfaces automatically that can be integrated to existing enterprise solutions
- ❖ Continue system modifications and release management without having to freeze the legacy code
- ❖ Manage budget and schedule leveraging our Fixed Price engagement model.

- ❖ Allows state agencies to continue with the 'in-house' maintenance and operations as training and documentation are by-products of our methodology and tools.

- **Faster Delivery**

Because we create the entire structure and architecture for your application with automatically written code, we gain the ability to have more developers working in parallel without stepping on each other. Booz Allen found a 4X productivity improvement from our processes and technology, and no code is created faster than automatically written code.

- **Code Quality**

Although we use automatic code generation, the code looks like human programmers wrote it. Other code generator technology may have cryptic variables and procedure names, or have a "one size fits all" structure and set of frameworks that cannot be changed. In contrast, we have been evaluated by the U.S. Air Force, by CAST Software, and by others as having excellent code quality.

- **Code and Frameworks per your Standards**

We routinely update our templates and generators at the start of the project to meet your desired program architecture, security requirements and enterprise standards. We document the solution in an Architecture Document, typically a deliverable, and we prove it with a sample application that can be tested and evaluated within the first 60 to 90 days after project initiation.

- **Legacy Proofing**

One thing is certain in the computer applications industry, and that is change. If you have a Visual Studio 2008 application, it is legacy by the time Visual Studio 2010, 2012 and 2015 have come and gone. Older Java applications have the same problem. But, because we generate the routine code in the application automatically, from the existing metadata, we can switch templates and generate code for a more modern platform. This greatly reduces the cost and time to stay current with the industry. Our solutions cater to the mainframe legacy applications as well.

- **Agile, Frequent Visible Results**

We work in one month iterations, with (optionally) 2 sprints in each. At the end of each iteration there is a baseline code drop, so the evolving application is visible and can be reviewed for customer input. We don't charge you for 2 years of writing documents before you see any code.

- **SOA Enabled, Cloud Ready**

Our customers normally want a Service Oriented Architecture (SOA) and they want to be able to easily and securely deploy the application on servers for access by browser. We deliver! And our software will run on the Mil Cloud, on GCSS-AF, on Amazon AWS, and on Microsoft Azure. SOA interfaces for external systems can be created also, with WSDL or JSON and REST.

- **Fewer Defects**

Because we build the applications from reusable code, the code in our libraries undergoes exhaustive testing and benefits from improvements made in prior projects. When we do find a defect, fixing it once in a template typically fixes a dozen undetected occurrences of the same problem. We also use unit testing, regression testing, and coding best practices to target common programming errors, along with Continuous Integration and tools like PMD and CAST.

**Ease of Enhancement / Maintenance**

Because the code is standardized, with meaningful variable names, and internal comments, it is easy to enhance and maintain. Our tools handle new development as well as modernization, so we can add enhancements to your code as we rewrite it, if desired.

**Scalability**

We follow industry best practice design patterns, such as Model View Controller (MVC). Our design patterns emphasize performance, and we are happy to load test any time. We have developed applications with up to 122,000 registered users.

**Security**

Our code is secure, and it is routinely tested to be sure that it remains so. We can also integrate your own security framework if desired.

**Flexible Business Model**

We will do the work turn-key, or as a subcontractor to your favorite systems integrator. We will also form an Integrated Product Team and partner with your own in-house IT staff if desired.

**Fixed Price Engagement**

We are willing to work fixed price, because our sophisticated estimating model and documented processes allow us to do so. Most firms will not do this.

**POC (Proof of Concept)**

Our tools and methodologies allow completion of a low cost, efficient and “non-invasive” proof of concept. The POC demonstrates the Return on Investment (ROI), and allows our customers to engage with virtually no disruption to their day to day business operations.

**No Change Orders**

Since our company was founded, we have never asked a customer for a change order. We have had a few situations where a customer asked us to do some additional work, and offered us one. Can our large competitors say the same?

**No Vendor Lock-in or Holdbacks**

We give you a copy of all the code, so you can maintain your system with no further license fees. You get the source code for everything, our own API libraries (if applicable) included. You can maintain your own system, or have your integrator maintain it - and we can provide training if needed, as well.

### **What we do from a business perspective for our clients:**

ResQSoft, with our ResQSoft® Engineer technology, has an answer; and we believe it is probably the only efficient and effective way to get there. What we do is redevelop, not translate, your existing applications, and we can then rapidly modify and enhance them to meet your exact requirements.

We deliver:

- High quality source code, with comments, unit tests, and proper structure for the web;

- Section 508 compliant for accessibility required by the Americans with Disabilities Act;
- Service Oriented Architecture (SOA), for easy integration and with SOA interfaces created automatically; and
- Rapid delivery (less than 2 years to first deployment).

Our code looks like it was hand-written for you, except that it has far fewer defects and costs much less than the \$10 - \$20 per line (or far more, in troubled projects) that others wind up paying in the end for custom work.

It's worth emphasizing that we automatically create SOA web services for system to system integration while the different systems are in different stages of development. Web services, of course, are easily used to allow separate modern Java or .Net applications to interoperate, as well. Our software also complies with Section 508 of the Americans with Disabilities Act;

And, because of the power of the ResQSoft® Engineer technology, we believe we can do all this far less expensively, and far faster, than anyone else. In fact, almost all of our engagements are done on a fixed price basis, although we also work T&M when the requirements are open-ended.

We will answer the RFI questions, and provide information on the underlying approach, herewith. We would also like to offer you a technical presentation and/or a demonstration of our technology, at your convenience, and we have Proof of Concept and Pilot programs that allow you to see the power of the technology applied to your own source code and requirements for a very minimal investment.

As requested in the RFI, the subsequent sections provide detailed responses to the questions.

## 2. Scope of Work

### i. Business Requirements

1. Exhibit B contains business requirements for the Washington State Modernized Elections System. (Note the scope of requirements excludes ballot creation and Tabulation.) Vendors are requested to validate and proof the business requirements to identify any requirements they believe have overlooked. Please provide a list of additional business requirements you recommend we consider for inclusion in a future RFP.

ResQSoft has reviewed the list of 463 business requirements and would like to recommend the following additions and changes to the list:

- Requirement 2A : Find Voter Information
  - In addition to the requirements listed under 2A, we recommend that the state incorporate phonetic search as an option for searching voters based on Last Name and First Name.
- Requirement # 9: System must support current versions of major modern web browsers in use at the time of system delivery. Vendor support must include ability to keep current with major browser enhancements.
  - In addition, we recommend that the system allows upward and downward compatibility of all supported browsers up to 2 versions.
- Requirement # 2H : Duplicate Management
  - When duplicates are identified by the system, there must be a provision to identify, rank and sort the duplicates based on the critical data elements (Name, DOB, Social Security Number) and allowing system users to choose the record for merge and delete.

2. Also pertaining to business requirements in Exhibit B, please identify any requirements you believe to be exotic. In other words, identify any requirements that you believe are uncommon, difficult to fulfill, or for any other reason contribute significant cost and/or time to the Modernized Elections System? Please identify which, if any, of the identified requirements are exotic and why.

ResQSoft has reviewed the list of 463 business requirements and would like to recommend the following:

- Requirement 2S: Scan Documents
  - Requirement 192: System must allow a scan document to be updated.  
This requirement would require specific details with regards to the document being updated.  
What does the update function entail?  
How often can the document be updated?  
What would be source of verification prior to allowing an update of the scanned document?  
Does update mean, replacing a new version or adding information to an already existing scanned document?
  - Requirement 197: System must allow for signatures to be clipped on a scanned document.  
Purpose of this requirement remains unclear and introduces security and data risk as the clipped signature might still be available on the clipboard and could potentially cause identity theft.

## ii. Security Methods

3. Exhibit A contains the WA OCIO IT Security policies. Within Exhibit B, there is a worksheet titled "Critical Election Periods". Washington State Elections Officials desire a solution that balances the provision of uninterrupted services during critical election periods with cost. Please provide a recommendation for **high availability**.

In order to achieve "high availability" to provide uninterrupted services during critical election periods, it is imperative that the underlying architecture of the application is assessed for risks and mitigation strategies are defined.

Architectural risk assessment is a risk management process that identifies flaws in a software architecture and determines risks to business information assets that result from those flaws. Through the process of architectural risk assessment, flaws are found that expose information assets to risk, risks are prioritized based on their impact to the business, mitigations for those risks are developed and implemented, and the software is reassessed to determine the efficacy of the mitigations.

Three activities can guide architectural risk analysis: known vulnerability analysis, ambiguity analysis, and underlying platform vulnerability analysis. Architectural risk analysis examines the preconditions that must be present for vulnerabilities to be exploited and assesses the states that the system may enter upon exploitation. As with any quality assurance process, risk analysis testing can only prove the presence, not the absence, of flaws. Architectural risk analysis studies vulnerabilities and threats that may be malicious or non-malicious in nature. Whether the vulnerabilities are exploited intentionally (malicious) or unintentionally (non-malicious) the net result is that the confidentiality, integrity, and/or availability of the organization's assets may be impacted. Risk management and risk transfer instruments deal with unmitigated vulnerabilities.

One of the strengths of conducting risk analysis at the architectural level is to see the relationships and impacts at a system level. In practice, this means assessing vulnerabilities not just at a component or function level, but also at interaction points. Use case models help to illustrate the relationships among system components. The architecture risk analysis would factor these relationships into the vulnerabilities analysis and consider vulnerabilities that may emerge from these combinations.

Threats are agents that violate the protection of information assets and application's site security policy. Threat analysis identifies for a specific architecture, functionality and configuration. Threats may be mapped to vulnerabilities to understand how the system may be exploited. A mitigation plan is composed of countermeasures that are considered to be effective against the identified vulnerabilities that the threats exploit.

Having determined what threats are important and what vulnerabilities might exist to be exploited, it can be useful to estimate the likelihood of the various possible risks. In software security, "likelihood" is a qualitative estimate of how likely a successful attack will be, based on analysis and past experience. Since it is based on past experience, this likelihood cannot account for new types of attacks or vulnerabilities that have not yet been discovered. It might not accurately reflect the probability of a successful attack. Nonetheless, the concept of likelihood can be useful when prioritizing risks and evaluating the effectiveness of potential mitigations.

The various risks that have been identified and characterized through the process of risk analysis must be considered for mitigation. Mitigation of a risk means to change the architecture of the software or the business in one or more ways to reduce the likelihood or the impact of the risk. Formal and informal testing, such as penetration testing, may be used to test the effectiveness of the mitigations.

4. Exhibit A contains the WA OCIO IT Security policies. Within Exhibit B, there is a worksheet titled “Critical Election Periods”. Washington State Elections Officials desire a solution that balances the provision of uninterrupted services during critical election periods with cost. Please provide a recommendation for **disaster recovery**.

There is no question that State of Washington’s Election System wants to protect their operations from downtime and loss of data. Traditional disaster recovery solutions typically cost too much; they’re too complex; and they are not always reliable.

ResQSoft’s recommendation would be to research and choose a solution that supports native cloud-based disaster recovery capabilities. Cloud based DR (disaster recovery) provides simple and secure asynchronous replication and failover. Consider these reasons as to why you should look to the cloud for your disaster recovery needs.

- Avoid Building Secondary Disaster Recovery Sites:** Cloud-based disaster recovery allows you to protect your production environment without the expense and management burden of replicating it to a secondary site operated by core IT staff. It’s a great way to improve upon existing disaster recovery plans—or get a new plan off the drawing board and into operation— with minimal cost and resources.
- Replace or Enhance Traditional Disaster Recovery Solution:** With cloud-based disaster recovery you can replace or enhance a traditional in-house or secondary site disaster recovery solution with newer functionality without capital investment, with elastic scaling and flexible subscription terms.
- Deliver Disaster Recovery Solution for Remote Offices:** Cloud-based disaster recovery makes it possible to protect remote office sites without additional investments and with lower operational expenses. You can extend your current disaster recovery plan to include satellite offices with no additional recruiting, hiring, and training expenses.

### iii. System Integration Approach and Methodology

5. Please provide a recommendation for system integration approach and methodology, which most effectively supports the specified business requirements and other concerns mentioned in the Background and Objective section.

Based on our experience managing and delivering critical engagements with Federal and State government entities, we would like to recommend the following key integration steps and methods:

- Initial Analysis and Requirements Documentation:** Assessing the business requirements, validating and mapping them to business needs with priorities would be a critical first step
- Group estimating and consensus workload estimating:** Estimating the work and planning accordingly allows the team to manage the schedule and mitigate schedule risks.
- Pair programming for complex code and debugging** – This technique guarantees programmer efficiency and brings in the added benefit of testing early and identifying business rule complexities and dependencies
- Group design process with lightweight documentation:** Rather than generate large volumes of documentation that increase the likelihood of cumbersome deliverable review processes and timelines, group design and relevant documentation keep the project on track and allows management of the overall schedule.

- **Incremental development with frequent visible results:** An incremental approach with frequently viewed results gains stakeholder confidence and creates the process for early detection of issues and functional discrepancies.
- **Options for customer involvement in the development team:** When customers are involved with the team during the build phase, it allows great visibility and generates early buy-in” from the user base thereby significantly increasing the chances of early acceptance and approval.
- **Rapid requirements definition using prototyping and other approaches;** and
- **Maximum code reuse, preferably at the requirements level.**

We don't use every technique on every project, but a combination of these techniques provides transparency for you, as well as an Agile and modern software development methodology backed by solid engineering.

#### iv. Project Management Approach and Methodology

6. Please provide a recommendation for project management approach and methodology, which most effectively supports the specified business requirements, other concerns mentioned in the Background and Objective section and project values of transparency and collaboration amongst the state's 40 separately elected Elections Officials..

ResQSoft recommends an iterative approach and methodology that will allow the State of Washington to initiate, manage, control and deploy the solution within the prescribed window and timelines.

- **Planning:** Initial planning before the project initiation phase is of paramount importance to the stakeholders and the vendor. During the planning phase, identifying the organization structure, execution methodology, issue escalation protocols and well defined acceptance criteria are key to the success of the project. Vendors must provide a clear and detailed project management approach that eliminates standard boiler plate templates and charts. The project management approach must be supported with a well-balanced organization chart that identifies roles, responsibilities and channels for communication.
- **Communication Plan:** As we have all learned over the years, that the critical success factor for any project is communication at all levels for both parties. Often, lack of guidelines and escalation mechanisms turn out to be the single point of failure. We believe the communication plan is a living document that is continuously maintained during the course of the project and serves as one of the key ingredients of a solid project management plan.
- **Requirements and Prototypes:** While compiling and building the requirements repository, the emphasis should be on the depth of the functional and business aspects with the objective of satisfying the end user needs. In order to avoid ambiguity and stay within the parameters of the business requirements, building prototypes that encompass functional, technical and architectural features is an absolute necessity. Prototypes when built, reviewed and discussed creates a visual medium for users to foresee the solution before the testing phase begins.
- **Test Planning:** Another critical component of the methodology is advance planning for the test phases – Integration Testing, Technical Testing and User Acceptance Testing. When the details are captured and shared, the testing team along with the OCM (Organization Change Management) team and Outreach team can plan their activities, risks and mitigation strategies.
- **Architecture:** Offerings and capabilities that vendors propose must cater to the immediate needs of the agency and satisfy future enterprise level initiatives and needs. For example, our solution will build the required framework on the SOA platform and provide extensions that may be required to

interface with other agencies whenever needed. Architectural planning and capability assessments are vital to the overall solution and expedites the build and deployment processes.

- **Deliverable Management:** Typically, deliverables tied to milestones are the industry norm for fixed price contracts. However, the process of creating the deliverables, submission criteria and acceptance timeframes introduce delays and miscommunication. To mitigate these schedule risks, our recommendation is to adhere to a collaborative plan that provides visibility to the users and stakeholders as and when the deliverable's building blocks are created. Iterative development, planning, prototyping and information sharing at every level improves the overall deliverable management process.
- **Monitoring, Decision Trees and Empowerment:** Identify a team of 10 members to represent the Election Officials. Each member represents 4 counties and should be empowered to make decisions on behalf of his group. This committee will play a very critical role in the success and delivery of the project.

Our project management approach is documented in a series of Standard Operating Procedures and is backed by key practices of the CMMI and PMBOK.

#### v. Funding Approach and Cost Distribution

7. Please provide a recommendation for funding approach and cost distribution, which most effectively supports the specified business requirements, other concerns mentioned in the Background and Objective section and project values of transparency and collaboration amongst the state's 40 separately elected Elections Officials. Please include citations of the recommended approach in place throughout state and local governments.

We recommend that funding for the projects and programs, reside at the State Level, and managed by the PMO team. The allocation schema to each county could be based upon users, voters and/or size of code base defined by the results of Section IV above, cognizant of "Monitoring, Decision Trees, and Empowerment".

Any funding and allocations processes should incorporate the following:

- OMB Circular A-122, Cost Principles for Non-Profit Organizations
- Utilize current funds allocated for the program
- Explore options for counties to share the funding needs
- Cost distribution could be based on user base and size of county

A key advantage of our approach is that the implementation can often be done within a single year's funding — or at least few "trips to the well" than are needed for other, slower approaches.

#### vi. Data Conversion and Migration

8. Please provide a recommendation for data conversion and migration, which most effectively supports the specified business requirements, other concerns mentioned in the Background and Objective section and project values of transparency and collaboration amongst the state's 40 separately elected Elections Officials..

Our recommendation for data conversion and migration follows a sequence of best practices we have applied across our projects and engagements:

- **Determine the purpose of the data:** Analyze the data environment, where and how it will be leveraged and who will use the data, while thoroughly assessing the technical environment. Also, determine if the data would be used for analytics in the future.
- **“As-is” Assessment before migrating data:** Since data is dynamic and changes continue to happen while the system is in use, it is tied directly to the business processes and users. It will be prudent to establish data standards, business rules for migration, exception handling methods and data cleansing strategies. Soliciting input from super users, IT staff and county workers who are subject matter experts would be critical to the process of assessment.
- **Data Quality:** Perform a complete quality assessment to ensure standardization exists across the new system features and would support the business requirements – both existing and future legislative and federal mandates. While measuring quality of data, consider removing duplicate data sources, redundant data elements and a master data management approach.
- **Business Rules:** Review existing business rules, compatibility to the new solution or package, and identify rules that may apply to one time data conversion files and or elements.
- **Data Ownership:** Determine data owners, frequency, sources of data and establish a decision making process for what to migrate and what to either archive or ignore.
- **Identify Data Migration Lead:** Identifying a key business and technical lead (prior to beginning the project) to represent and manage the effort on behalf of the state would benefit the state in terms of schedule management, risk mitigation and overall data quality.
- **Continuous Monitoring and Process Improvement:** Ongoing management of the data migration effort and improving the processes to suit the application and requirements is recommended.
- **Measure, Measure and Measure:** Tracking data migration efforts, planning for mock conversions and data cleansing are critical techniques that must be in place. Planning for data anomalies after the final run and adhering to well defined post conversion clean up methods guarantees user satisfaction.

## vii. User Experience Design Approach and Methodology

9. Please provide a recommendation for user experience design approach and methodology, which most effectively supports the specified business requirements, maximum stakeholder usability and adoption and project values of transparency and collaboration amongst the state’s 40 separately elected Elections Officials.

UX experience refers to the end-user experience experienced by customers at their end with regards to what the application delivers as a solution to meet their daily business needs, UX is different from UI or User interface, which is the portal, which is design interface of the website through which the user interacts and experiences the digital world. UX design is about creating the website design which would deliver the best possible user experience while at the same time, successfully achieving the business goals.

**User Testing:** One of the must-do UX design practice is to ensure that user testing is carried out as rigorously as possible. This is where the effectiveness, usefulness of the design would be realized. Conducting many iterations to optimize the UX design would be necessary. Balancing attractive design, resources, loading data and internet bandwidth must be key factors to consider during UX design and supported by the methodology.

**Mobile Usage:** The UX design methodology must cater to the needs of the mobile devices as trends indicate that desktop internet usage has fallen from 90 to 60 % and is bound to reduce over years as mobile usage goes up. This may result in a reversal of design strategies. Prioritizing mobile usage is critical before other access methods are taken into consideration.

**Devices and Tools:** The usage of different kind of devices to access the internet makes it necessary for designers to focus on delivering responsive designs. Responsive design with matching standards is required and the design methodology must support this requirement.

**Sketches:** A quick way of visualizing a new interface by using paper and pen. Sketches are useful to validate product concepts and design approaches both with team members and users.

**Wireframes:** A visual guide that represents the page structure, as well as its hierarchy and key elements. Useful to discuss ideas with team members and clients, and to assist the work of designers and developers.

**Prototype:** A prototype is a simulation of the website navigation and features, commonly using clickable wireframes or layouts. It's a quick and dirty way to test and validate a product before fully developing it.

#### **User Research and Validation Methods:**

- Focus Group** - A panel of people discussing a specific topic or question. Teaches about the users' feelings, opinions and even language. Useful when the target audience is new or unknown for the team.
- Quantitative Survey** - Questions that provide numbers as result. Quick and inexpensive way of measuring user satisfaction and collecting feedback about the product. It could indicate the need for a deeper qualitative test.
- Usability Test** - One-to-one interview research in which the user is asked to perform a series of tasks in a prototype or a product. Validates and collects feedback of flows, design and features.
- Card Sorting** - A technique that consists in asking users to group content and functionalities into open or closed categories. Gives you input on content hierarchy, organization and flow.
- A/B Test** - Offering alternative versions of your product to different users and comparing the results to find out which one performs better. Great for optimizing funnels and landing pages.
- Eye Tracking** - A technology that analyzes the user's eye movements across the interface. Provides data about what keeps users interested on the screen and how their reading flow could be optimized by design.

In addition, **Concepting Methods** would be a key ingredient to the UX Design Methodology:

- Brainstorming
- Mood boards
- Storyboards
- User Flow
- Task Analysis
- Taxonomies

These concepting methods are supported by the following **Product Planning Methods**:

- Content Audit
- Heuristic Analysis
- Site Map
- Features Road Map
- Use Case and scenarios
- Metrics Analysis

### viii. System Support, Maintenance and SLA (Service Level Agreements)

10. Please provide a recommendation for system support, including service and maintenance, service level agreements and help desk, which most effectively supports the specified business requirements, other concerns mentioned in the Background and Objective section and project values of transparency and collaboration amongst the state's 40 separately elected Elections Officials..

Our service level agreements are industry standard, and can be adjusted to comply with The Office of the Chief Information Officer (OCIO) who has authority for statewide technology policy and standards.(ESSB 5931). Polices are adopted by the OCIO and approved by the Technology Services Board.

These SLAs will be defined in section IV above and comply within OCIO "Statement of Performance Reports"

- Maintenance plans, approach and SLAs - Typically defined with the Project Management Plan with details documented in the System Management Plan. The State IT Department's input and ideas will be crucial in creating a plan that benefits the state and doable by the vendor.
- System Support defined by years – 1 to 2 years, 3 to 5 years etc., with appropriate staffing levels help desk and IT Support. The staggered plan allows the state to manage the cost of maintenance over a period of 1 to 5 years.
- Performance based SLAs – Defined by severity of issues, time taken to resolve them and the occurrence of the same or similar issues. The metrics are then compiled for the month and compared to established thresholds for delivery and maintenance.
- Metrics driven cost management

### ix. Contract Vehicles and Strategies

11. Please provide a recommendation for contract vehicles and strategies in support of your recommended approach to system support and system integration.

There are a number of different ways to approach this problem, and in many cases, acquisition is as hard or harder than doing the actual work (as we all know). Our normal approach would involve the following points:

- Review current contract / best practices
- Review any private industry contract mechanism that might generate interest
- Payments based upon milestone completions and or deliverables
- Vendor awards with potential set aside for MBEs (Minority Business Enterprises)
- We will follow any standard or policy currently in place with OCIO, as those listed below, or other relevant Policies:
  - Commonly Used Software Product Standard (186.10)
  - Established Enterprise Services (185)
  - Integration Services Governance Standards (183.30.10)
  - Securing Information Technology Assets Standards (141.10)

We have established minority owned business partners, and a Native American owned one as well. These are true partners, not pass-through, and the quality of the work is first rate. We can also work with your established successful systems integrators who may already have a suitable contract vehicle. We always prefer fixed price work, it's better for us and better for you, too!

## x. Testing

12. Please provide a recommendation for testing, complete through final acceptance testing and to include a mock election.

Numerous studies have shown that reusing code is the highest productivity & quality approach known for building quality software systems. We recommend that you conduct thorough and complete test programs that will verify that each of your requirements have been met. Test locations, tests complexity, number & types of tests and the test resources used impact testing costs. A quality test program could cost up to 45% of the total project cost or budget.

Software test programs should include:

- Design reviews—there should be a preliminary design review conducted after the completion and submission of high level design documents; plus a detailed design review conducted after submission of the detailed design documents.
- Development testing—development testing should include scripted unit testing, software build integration testing, full regression testing with scripted browser-based functional tests. Different test cases include identification and description of test cases, test steps & test results---including when the tests were run and the outcome through final acceptance
- System Integration and Technical Testing – this critical test phase allows the system components to be talk to each other, share data, services and perform the functional requirements as outlined within the prescribed performance windows.
- User Acceptance and Site Testing - site testing should include hardware/software integration testing, subsystem testing & system testing. Site testing should be completed at the final installation site with communications connectivity to the field devices.
- Mock Election: The approach to a mock election test is similar to a pilot phase, however the only difference is that the data is purged when the mock election phase is completed. Preparation includes site readiness, data conversion, user base identification, security protocols, communication and outreach and stakeholder involvement to mimic an election. Capturing the details while planning and during execution and close out are very important milestones that need to be managed by a special group of project executives.

## xi. Training

13. Please provide a recommendation for training. Elections Administrators and Staff around the state possess an intimate familiarity with their existing systems. We will require a training plan that enables county and state users to develop a high degree of comfort with the replacement system(s) in advance of go-live in order to support a seamless implementation for all Washington State elections stakeholders. Training to include internal users and administrators/IT support staff.

- Identify key business leads across all counties
- Leverage IT Department, and any existing training programs or incentive programs
- Web based training, Train the trainer, Help Guides, How to videos, Library of procedures stored on the state Intranet.
- As part of our methodology during our projects and implementations, we offer as part of our delivery a training program for relevant staff, and program managers

- It is imperative that key leads and stakeholder of the client technical team, get summary and specific training of the ResQSoft software stack. Our software solutions are essentially comprised of two areas for production and testing.
- Upon definition of the user community needing the training courses, we can hold training seminars online, and also as part of closed community thru portals, and learning sessions with online testing, etc.
- While we are always open to post production and post implementation support, we also realize the benefit of training key team members of the client team, to handle any specific knowledge base of the software for efficient support and migration of minor technologies.

## xii. Documentation

14. Please provide a recommendation for documentation, including internal, external, and administrator.

- Identify key business leads across all counties: When county representatives are invited and encouraged to be part of the project and allowed to be an integral part of the overall strategy, execution and delivery of the solution, it expedites the process of documenting the business processes and workflows. The chosen vendor can collaborate with the county representatives to define a framework for documentation and identify the key features, expectations and usability of the artifacts.
- Leverage IT Department: The state IT department will play a pivotal role in the devising a strategy for documenting the needs of the users, IT Staff and methods to keep the documentation current and relevant. The needs of the user base, their patterns of information requests and the methods of communicating the responses are valuable lessons that can be leveraged when the vendor builds the documentation repository.
- Web based, Help Guides, How to videos, Library of procedures stored on the state Intranet: Traditional methods of help guides, web based tools, videos and a Hyperlinked User Guide are several methods that may be considered for documenting the overall application.
- System Management Guides, Technical, Functional, Policy, Super user tips are additional guides that may be required that we recommend.

## xiii. Voter Outreach

15. Please provide a recommendation of voter outreach requirements for the Modernized Elections System.

- Planned outreach strategy with defined milestones
- Leverage county representatives for communication
- Leverage existing Newsletters, Intranet, Email, Local Newspapers and employee portals of the State
- Define "Canvassers" needed based upon residential metrics county by county
- Ensure "non-technical" areas within the State have access to more traditional means of information, whereas modern online and internet access may not be available
- Ensure that traditional existing phone banks or phone outreach, are carried out towards any new modernized elections systems
- Ensure that new modernized systems follows non-partisan process and procedures, to result in objective and non-invasive processes
- Leverage any leading edge Political Science projects that may exist throughout leading Universities or Colleges throughout the State of Washington

**xiv. Timeline Estimate**

16. Please provide a timeline estimate for implementation of your envisioned solution in response to business requirements in Exhibit B and your responses to items 1 – 14 above.

Based on our experience delivering similar sized systems and reviewing the business requirements in Exhibit B, we believe the system implementation timeline would be between 14 and 18 months. Our projects in the Federal Government domain with complex coding requirements have been completed on time.

**xv. Cost Estimate**

17. Please provide a cost estimate for implementation of your envisioned solution in response to business requirements in Exhibit B and your responses to items 1 – 15 above.

Unfortunately, we are unable to provide a cost estimate without specifics on the choice of technology, System / application architecture and the current state of the existing application's data and operational Issues.

That said, if we were to build this system for you ourselves, using our normal process and resources, the cost would be less than \$10 million, fixed price, with no recurring future payments and full source code provided. If we provide technology to an integrator, our experience is that the cost always increases, reflecting the higher overhead of a larger company.

We look forward to the RFP and more information regarding the Election System in the near future.

**3. Summary**

We appreciate the opportunity to respond to this thoughtful RFI. We can summarize our recommendation by saying that we believe this work can be done fixed price, while maintaining the flexibility that you will need to ensure buy-in from the individual election operators within the state.

We offer the most rapid implementation approach: no package vendor can avoid substantial modifications and customization given your diverse requirements, and almost all packages are built with legacy code that is difficult to modify in ways that also deliver substantial performance.

We are flexible in terms of contract vehicles and welcome partners large and small.

We'd like to do a great system for you, and if given the chance, we will!