|  |  |  |
| --- | --- | --- |
| **T4 TECHNICAL EVALUATION FORM – FIRM FIXED PRICE & TIME-AND-MATERIALS** | | |
| T4 Number  T4-0250 | Task Title  VistA Adaptive Maintenance | |
| Name of Offeror  Offeror C | | Date of Proposal  August 17, 2017 |
| **1. Technical Evaluation Criteria:**  TECHNICAL: The evaluation of the technical proposal considered the following:  (1) Understanding of the Problem – The Technical Volume of the Task Execution Plan (TEP) was evaluated to determine the extent to which it demonstrates a clear understanding of all features involved in solving the problems and meeting and/or exceeding the requirements presented in the task and the extent to which uncertainties are identified and resolutions proposed.  (2) Feasibility of Approach –The Technical Volume was evaluated to determine the extent to which the proposed approach is workable and the end results achievable. The Technical Volume was evaluated to determine the level of confidence provided the Government with respect to the Offeror’s methods and approach in successfully meeting and/or exceeding the requirements in a timely manner.  **2. Proposal Summary:**  The Offeror provided a technical and management approach to create a Veteran Integrated Care Service (VICS) that surfaces and migrates clinical functionality and business logic from the VA’s Veterans Information System Technology Architecture (VistA) Electronic Health Record (EHR) into a data model-driven, Node Package Manager (NPM)-installable, Node.js-based national service with no legacy Massachusetts General Hospital Utility Multi-Programming System (MUMPS) code dependencies. The Offeror’s response described the technical and management solution to the problem identified in the Performance Work Statement (PWS).  The Offeror’s overall solution architecture was provided, showing the existing architecture, and their planned “to-be” architecture, and their interdependencies (Exhibit 3). This proposed architecture described all components required for the desired solution, with the required functionality including CPRS running as-is, unchanged over both the existing VISTA architecture and the to-be Centralized Model-driven VICS Services architecture via an RPC emulation layer. The Offeror’s Model-driven VICS Service runs on top of a centralized, industry-standard NoSQL Data Store (Microsoft Azure, Amazon CosmosDB, or MongoDB) while at the same time providing backwards compatibility and synchronization to the individual legacy VISTA systems, allowing migration off the VISTA architecture without interruption in services to CPRS or VPR clients.  In addition to the overall architecture, the Offeror provided supporting descriptions of the specific components the Offeror will deliver including a RPC Router, RPC Emulator, Virtual Patient Record (VPR) Emulator, a Representational state transfer (REST) interface, a service interface, and the VICS service, delivered in the VA-specified Node.js-based, NPM-installable, JavaScript form with no MUMPS dependencies – each of which is common across all four (4) VICS specified by VA.  The Offeror’s proposal included details on how they would develop each of the architectural components, why they are necessary, how they work, and features of each. Their initial task was the generation of the Native Data Model from the as-is VISTA Fileman data dictionaries, which is then refined to a common subset of across all 130 VISTAs (Common Native Data Model) as the foundation model to provide national VICS Services. Details on what, why, and how the Common Native Model would be refined, and how this would be automatically translated backwards and forwards from the Native Model to support backwards compatibility as well as new features and functionality was described (Exhibit 4). VICS Data Model categorization were described with examples and features (Exhibit 5). Common VICS Utility Services and their features and advantages were described (Exhibit 6). Portable VICS Interfaces and their benefits were described (section 2.4). Emulation of Remote Procedure Calls (RPCs) and emulation of the Virtual Patient Record (VPR) was described in step-by-step detail. The use of Node.js compatible, cloud-based NoSQL data storage was described, including use of MongoDB, Microsoft's CosmosDB, and Amazon's DynamoDB. Development of a Node Package Manager (NPM)-installable RPC Router and its features was described (section 2.8). Finally, implementation of Backward Synchronization Service (section 2.9) was described to account for the interdependencies of the existing MUMPS VISTA functionality in parallel with the new VICS Services. The Offeror stated that all components of the above to-be architecture would be developed in pure JavaScript, Node.js-based, Node Package Manager (NPM)-installable plugins.  The proposal describes how the Offeror will utilize Remote Procedure Call “(RPC) sniffing” to assess RPC traffic between the Computerized Patient Record System (CPRS) and VistA and the impact to the data store of the RPC execution sequences from which to develop Node.js-based Data models to fully and formally capture VistA behavior and which will be reusable to build the national Veteran Integrated Care Services (VICS) and automated testing suite. Finally, the proposal outlines the staffing levels by labor category and PWS functional area for the entirety of the project as allocated between the prime and its subcontractors.  The Offeror has proposed to team with 2 subcontractors.  After review of the entire proposal, it was determined that the Offeror’s approach contained the Significant Strengths detailed below. The remainder of the VistA Adaptive Maintenance requirements was adequately addressed.  **3. Summary of Significant Strengths and Strengths:**  **Significant Strength #1:** **(TEP pp4-6 and 8 – 11, Sections 2.1-2.4 and 3.0, Request for Task Execution Plan (RTEP) B.1.1.a and B.1.1.b, PWS 5.2.1):**  The Offeror provides a highly detailed explanation of its approach for creation of a “Native Model” that defines each VistA in standard JSON/Node.js, and which includes all classes, triggers and indexing. (**What is proposed**) Defining and establishing a Common data model is one of the most complex tasks necessary for automatic translation between any VICS object and the native objects it defines. The Offeror’s approach to establish the VICS data model in the same data model format as the Common Native Model will ensure backward compatibility between FileMan resident data (i.e., in legacy, MUMPS-based format) and equivalent VICS data (i.e., in JSON/JavaScript format), supporting selective write-back to VistA of VICS data. (**WHY this is important**) The Offeror provides significant elaboration of the guidelines by which it will refine relevant classes of the Common Native Model and the importance of each of these elements in the standardization of VistA data, many of which are not covered by industry standards and likely makes up more than half of VistA data. The Offeror’s approach shows an understanding of the unique data synchronization problems to be addressed within each of the Patient Data Entry (PDE) functions of Vitals, Allergy, and Patient Problems (PWS 5.2.1). This significantly increases the Government’s assurance that the Offeror’s solution will facilitate automatic translation of VistA FileMan data to a standard VICS data model while also identifying what data is Veteran-specific and not standard to industry (e.g., VA-specific eligibility and service records, stop codes, health factors, service-related disabilities, billing, and disease registries) and which are VA-specific. The ability to differentiate these non-standard data from those applicable for standardization will enable VA to make decisions regarding its data storage policies as it migrates to a commercial EHR system. (**HOW this approach benefits the G’vt**) The Offeror’s approach to creating a Common Native Model, coupled with its demonstrated expertise with the classes, triggers and indexing required to establish the data standardization (as well as the cases in which standardization is not possible) will significantly reduce the risk of schedule delay, inaccurate costing, underestimation of the effort needed for establishing the required VICS components, and decreases the potential requirement of data loss; all of which strongly increases the confidence in the Offeror’s ability to deliver the requirements of the PWS. **(IMPACT Statement)**  **Significant Strength #2:** **(TEP pp4-6 Section 2.1, Request for Task Execution Plan (RTEP) B.1.1.c and B.1.2.c, PWS 5.2.2):**  The Offeror provides a highly detailed explanation of its approach for creation of a “Native Model,” consistent for that proposed for the prior-referenced RTEP requirements, that defines each VistA in standard JSON/Node.js, and which includes all classes, triggers and indexing relative to the Pharmacy applications functionality under CPRS as well as prompt and scroll control. (**What is proposed**) Defining and establishing a Common data model is one of the most complex tasks necessary for automatic translation between any VICS object and the native objects it defines, and this complexity is one of the main reasons the legacy VistA system cannot be ripped and replaced. The Offeror’s approach to establish the VICS data model in the same data model format as the Common Native Model will ensure backward compatibility between VistA FileMan’s resident data (i.e., in legacy, MUMPS-based format) and equivalent VICS data (i.e., in JSON/JavaScript format), supporting selective write-back to VistA of VICS data. (**WHY this is important**) The Offeror provides significant elaboration of the guidelines by which it will refine relevant classes of the Common Native Model and the importance of each of these elements in the standardization of VistA data, many of which are not covered by industry standards and likely makes up more than half of VistA data. The Offeror’s approach shows an understanding of the unique data synchronization problems to be addressed within the functions representing Outpatient Pharmacy Computerized Physician Order Entry (CPOE) PWS 5.2.2) applications. This significantly increases the Government’s assurance that the Offeror’s solution will facilitate automatic translation of VistA FileMan data to a standard VICS data model while also identifying what data is Veteran-specific and not standard to industry (e.g., VA-specific eligibility and service records, enabling VA to make decisions regarding its data storage policies as it migrates to a commercial EHR system. (**HOW this approach benefits the G’vt**) The Offeror’s approach to categorizing these unique, VA-specific data will enable VA to identify what data elements are not covered by commercial EHRs and/or commercial standards like Fast Healthcare Interoperability Resources (FHIR), and thus helps to identify what can and cannot be migrated to a commercial system. It also highlights the data elements VA must maintain, for which it will require a revised data storage policy and process, so that it can keep critical Veteran’s data available and accessible consistent with the lifetime records retention schedules applicable to the Veterans under VA’s care without jeopardy. **(IMPACT Statement)**  **Significant Strength #3:** **(TEP pp6-7, RTEP B.1.1.c and B.1.2.c, PWS 5.2.1 and 5.2.2):** The Offeror provides a highly detailed approach to MUMPS RPC emulation in javascript/Node.js. They first intend to identify all relevant RPCs by employing “RPC Sniffing” to capture sequences of RPC traffic between CPRS and VistA, and then employ “Snapshotting” to establish changes in the data store (FileMan) and process stack of VistA caused by these sequences. Their approach includes the development of Common Utility Services as elaborated in detail in Exhibit 6 spanning Change Events through Parameter Access and then porting and routing those elements through a VICS Interface available over REST using RPC Emulation. (**What is proposed**) Identifying all relevant RPCs to the PDE and Pharmacy CPOE clinical functions and then making them a common utility via a national/common utility service is a huge undertaking, and has not been accomplished before. Tracing RPC functions has cyclical consequences, as the call of one RPC may have corresponding calls and implications to other applications. The Offeror’s automation approach to identify the RPCs and then trace their consequences to the FileMan data store will facilitate the isolation of the output produced by these calls and facilitate the refinement of a Common Data Model from which to establish web-based, national services. (**WHY this is important**) The Offeror provides significant detail of its approach to establish the national services and proposed implementing an RPC Router to seamlessly and securely redirect some CPRS RPCs to national services and pass the remainder on to the appropriate VistA. Model-backed, NPM-installable, Node.js-based national services will support relevant RPCs over their interfaces and synchronize with functionality still running in VistA. Routing RPCs will (1) introduce precise auditing of all VistA RPC traffic for the first time, while re-implementation will (2) remove vulnerabilities from all emulated RPCs and (3) enable elevated access control. End-to-end (4) encryption of RPC traffic will guard against traffic interception. This significantly increases the Government’s assurance that the Offeror’s solution will not only address the desired migration off of VistA MUMPS through emulation, but will resolve many of VistA’s known security vulnerabilities, which has been a persistent, well-documented, but unresolved issue for years due to the complexity and dependencies of the RPCs in place today. (**HOW this approach benefits the G’vt**) Overall, the Offeror’s approach to MUMPS RPC emulation will create the desired PDE and Pharmacy CPOE National services in the desired format, providing a clear migration path to a commercial EHR by (1) centralizing VA operations and (2) isolating the Veteran-specific parts of VA care, which significantly reduces the risk to data loss or data synchronization issues post deployment.  **(IMPACT Statement)**  **Significant Strength #4:** **(TEP pp15-16, RTEP B.1.3, PWS 5.5.5):** The project calls for an automated Test VistA and a series of domain-specific and cross-domain regression test suites. The Offeror proposes to create a regression test suite (“VistA Domain Test Suite”) comprised primarily of “RPC Test Suites” that reproduce the ways CPRS and the Joint Legacy Viewer (JLV) use individual RPCs and sequences of RPCs. They propose creating the VistA Domain Test Suite before creating a VICS for a domain to formally capture that domain’s behavior so as to mitigate the risk and reproduction of largely untested, pre-existing functionality. They note an exception in which they will employ a combination of “terminal session”, API, and HL7 v2 tests interleaved with RPC tests to reproduce a pharmacist’s behavior to test how a pharmacist interacts with VistA, which, as the PWS points out, is neither through CPRS nor its RPCs. They also propose to test not only the functionality to be reproduced in a VICS, but running over a VICS to test the effectiveness of that VICS as a reproduction (e.g., as emulating the MUMPS functionality in the new, web-based service). (**What is proposed**) The Offeror’s proposed approach to “Dual Use” testing is a significant discriminator in that doing so enables testing to verify that the referenced VistA PDE and Pharmacy CPOE functions and their emulation ensure common behavior and verify that there are “no legacy MUMPS code dependencies” in the to-be-delivered national services solution, consistent with the most significant of requirements of the to-be-delivered solution. (**WHY this is important**) The Offeror’s testing approach and application of dual use testing will provide the ability to demonstrate safe and effective deployment of a VICS that should mirror a production install early on and consistent with project goals to demonstrate “no legacy MUMPS code dependencies,” which decreases Government risk early on in the project prior to Initial Operating Capability (IOC) and production migration. (**HOW this approach benefits the G’vt**) The Offeror’s dual use testing approach reduces the risk of schedule delays or of delivering inappropriate components to production, which appreciably increases the likelihood of successful completion of all the PWS requirements. **(IMPACT Statement)**  **Significant Strength #5:** **(TEP pp16-17, RTEP B.1.4, PWS 5.6):** The Offeror’s approach to IOC support focuses on producing a fully integrated demonstration (“IOC Demonstration”) in which their proposal specifies the acceptance criteria in detail of that IOC demonstration, and which is consistent with the stated goals and requirements of the project such that all clinical domains are emulated and delivered in the desired format. (**What is proposed**) The Offeror’s approach to the IOC demonstration significantly increases the assurance to VA that CPRS continues to operate unchanged after migration to VICS. (**WHY this is important**) The Offeror’s proposed use of an IOC demonstration to verify successful execution of all the stated acceptance criteria will provide the Government assurance of the performance of the centralized, national services and backwards compatibility to show successful “dual use” and verification that there are no legacy MUMPS code dependencies. (**HOW this approach benefits the G’vt**) By proving the successful operation of the emulated VICS at IOC, the Offeror reduces the risk of delivering inappropriate components to production, which appreciably increases the likelihood of successful and timely completion of all PWS requirements. (**IMPACT Statement)**  **Significant Strength #6:** **(TEP pp3-4, pp16-17, RTEP B.1.5, PWS 5.7):** The success of National Deployment relies ona solution architecture that has addressed all the components and applications involved when a VICS replaces the functionality of a VistA domain. The Offeror depicts a Solution Architecture in Exhibit 3 comprised of components including the RPC Router, RPC Emulator, VPR Emulator, a REST interface, a service interface, and the VICS Service, which the Offeror indicates explicitly that it will develop in the project in Node.js-based, NPM-installable JavaScript and developed without MUMPS. (**What is proposed**) The Offeror’s solution architecture addresses all requirements for model development, RPC emulation, and VistA Synchronization following common patterns and maximizing the use of shared utilities to create national services. Their proposed VICS architecture specifies the common structure and environment for every VICS which facilitates a consistent approach to national deployment of these services. (**WHY this is important**) The up-front development of the Offeror’s complete solution architecture, coupled with the successful results of IOC demonstration after development and delivery as indicated by Significant Strength #5, greatly increases the Government’s assurance that all required solution components have been identified and scoped within the Offeror’s Task Execution Plan (TEP), thus mitigating the potential for cost, schedule, or performance impacts. (**HOW this approach benefits the G’vt**) The successful deployment of the Offeror’s solution will facilitate ease of VA’s migration to a single, centralized commercial service and EHR. **(IMPACT Statement)**    **4. Summary of Significant Weaknesses and Weaknesses:**  None.  **5. Summary of Deficiencies**:  None.  **6. Special Terms and Conditions / Deviation / Critical Assumptions stated in TEP:**  None.  **7. Evaluation Criteria:**  **a. Understanding of the Problem**  Overall the Offeror demonstrates a X understanding of the requirements.  **b. Feasibility of Approach**  Overall the Offeror demonstrates an approach that is considered X feasible and is considered X risk.  **8. Rating:**  Outstanding- A TEP that meets or exceeds all of the Government’s requirements, demonstrates a thorough understanding of the problems, and is highly feasible (low risk). | | **Technical Rating:**  **Outstanding** |
|  | |  |
| **Evaluator Signature**  *Only one signature should be provided even if multiple technical evaluators participated. The lead technical evaluator should sign and date the technical reports.* | | **Date** |
| **Rafael M. Richards, MD MS, Physician Informaticist, Veterans Health Administration** | | |

*Contract Evaluation Form Rev 2.0 CAI 22 May 2009*