## Project Statement of Work: Requirements Gathering Agent Enhancement and Word Export

**Prepared for:** [Stakeholder Name/Organization] **Prepared by:** [Your Name/Organization] **Date:** October 26, 2023 **Version:** 1.0

### 1. Project Description

This project focuses on enhancing the existing Requirements Gathering Agent (RGA) application and adding a critical new feature: professional Microsoft Word (.docx) document export. The RGA is an AI-powered tool that generates comprehensive PMBOK-compliant project documentation from existing project artifacts. This enhancement will significantly improve the usability and value of the RGA, particularly for enterprise clients requiring formal, easily shareable documentation in standard corporate formats. The current RGA suffers from limited output formats and a lack of professional formatting capabilities, hindering its adoption in corporate environments. This project directly addresses these limitations.

The project will leverage the existing RGA codebase and integrate a robust Word document generation engine. This new functionality will enable users to export all generated PMBOK documents in a professional, formatted .docx format, including metadata and proper styling.

### 2. Project Scope

**Included:**

* Enhancement of the existing Requirements Gathering Agent (RGA) application.
* Development and integration of a Microsoft Word (.docx) export feature.
* Thorough testing of the Word export functionality, including unit, integration, and user acceptance testing (UAT).
* Comprehensive documentation updates to reflect the new feature.
* Deployment of the updated RGA application to npm.

**Excluded:**

* Development of new PMBOK document templates beyond those currently supported.
* Integration with third-party project management software.
* Development of a PowerPoint (.pptx) export feature (planned for a future phase).
* Extensive UI/UX redesign of the existing CLI interface.

**Key Deliverables:**

* Updated RGA application with Word export functionality.
* Comprehensive test reports demonstrating the functionality and stability of the new feature.
* Updated user documentation including a dedicated section on Word export.
* Deployed RGA application on npm.

**Performance Requirements:**

* Word document generation should complete within a reasonable timeframe (to be defined based on testing).
* Generated Word documents should be compatible with Microsoft Word 2016 and later.
* The updated RGA application should maintain or improve upon the performance of the existing application.

**Constraints:**

* Project timeline: [Specify timeline - e.g., 4 weeks]
* Budget: [Specify budget]
* Resource availability: [Specify resource constraints, e.g., developer availability]

### 3. Objectives and Success Criteria

**Primary Objectives:**

* Successfully develop and integrate a Word (.docx) export feature into the RGA.
* Achieve 95% test coverage for the new Word export functionality.
* Successfully deploy the updated RGA application to npm.

**Secondary Objectives:**

* Gather user feedback on the Word export feature through UAT.
* Improve the overall user experience of the RGA.

**Success Metrics:**

* Successful completion of all planned testing phases with acceptable results.
* Positive user feedback on the Word export feature.
* Successful deployment of the updated RGA application to npm without critical errors.

**Acceptance Criteria:**

* The Word export feature generates correctly formatted .docx files for all supported document types.
* The generated .docx files meet the defined quality standards (e.g., proper formatting, metadata, compatibility).
* The updated RGA application passes all defined test cases.
* The updated RGA application is successfully deployed to npm.

### 4. Deliverables

| Deliverable | Description | Quality Standard | Schedule/Milestone | Dependencies |
| --- | --- | --- | --- | --- |
| Updated RGA Application | The core RGA application with the integrated Word export feature. | Functional, robust, and compliant with existing code standards. | End of Week 2 | Completion of Word Export Module |
| Word Export Module | The code module responsible for generating Word documents. | Unit tested, integration tested, and meets performance requirements. | End of Week 1 | N/A |
| Unit Test Reports | Comprehensive reports detailing unit testing results for the Word export module. | 100% code coverage for critical functions. | End of Week 1 | Completion of Word Export Module |
| Integration Test Reports | Reports on the integration testing of the Word export module within the RGA application. | All integrated components function correctly. | End of Week 2 | Completion of Updated RGA Application |
| User Acceptance Test (UAT) | Results from user testing of the Word export feature. | Positive feedback from a minimum number of testers (to be defined). | End of Week 3 | Completion of Updated RGA Application |
| Updated User Documentation | Updated documentation reflecting the new Word export functionality. | Clear, concise, and easy to understand. | End of Week 3 | Completion of UAT |
| Deployed RGA Application | The updated RGA application deployed to npm. | Successful deployment and accessible via npm. | End of Week 4 | Completion of all testing and documentation |

### 5. Approach and Methodology

* **Project Management Methodology:** Agile (Scrum or Kanban, to be specified)
* **Development Approach:** Iterative development with continuous integration and continuous delivery (CI/CD).
* **Quality Assurance:** Unit testing, integration testing, and UAT will be conducted throughout the project lifecycle. Code reviews will be performed.
* **Risk Management:** A risk register will be maintained to identify, analyze, and mitigate potential risks. Regular risk assessment meetings will be held.

### 6. Assumptions and Constraints

* **Assumptions:** The existing RGA codebase is stable and well-documented. Necessary resources (developers, testing environments) will be available as planned.
* **Technical Constraints:** The chosen Word generation library must be compatible with the existing RGA architecture.
* **Resource Constraints:** Developer time is limited to [Number] hours per week.
* **Timeline Constraints:** The project must be completed within [Number] weeks.

### 7. Roles and Responsibilities

* **Project Manager:** [Name] – Overall project management, risk management, communication.
* **Developer:** [Name] – Development of the Word export module and integration into the RGA.
* **Tester:** [Name] – Execution of unit, integration, and UAT testing.
* **Documentation Writer:** [Name] – Update of user documentation.

**Decision-Making Authority:** The Project Manager has the authority to make decisions related to the project scope, schedule, and budget, within the defined constraints.

**Communication Protocols:** Regular stand-up meetings (daily), weekly progress reports, and ad-hoc communication as needed.

This Statement of Work serves as a preliminary agreement and is subject to final review and approval by all stakeholders.