**Agile Framework for RegReach Data Extraction Tool**

**Project Overview:**

The RegReach Data Extraction Tool is an internal tool designed for the Operations Regulator Outreach department. The objective is to develop a web-based ETL application to extract and process questionnaire information sent to clients and regulators. The tool will accept various file types, recognize and extract content, and allow the team to review, edit, and export the data. The application will feature a React.js front-end and a Python backend, leveraging technologies in computer vision, data mining, and PDF operations.

**Agile Framework: Scrum**

**Roles:**

* **Product Owner:** [Your Name or Designated Stakeholder]
* **Scrum Master:** [Your Name or Designated Facilitator]
* **Development Team:** [List of Developers, QA, and any other team members]

**Artifacts:**

1. **Product Backlog:**
   * A prioritized list of all features, enhancements, and bug fixes required in the product.
2. **Sprint Backlog:**
   * A set of Product Backlog items selected for the Sprint, plus a plan for delivering the product increment.
3. **Increment:**
   * The sum of all the Product Backlog items completed during a Sprint and all previous Sprints.

**Ceremonies:**

1. **Sprint Planning:**
   * Define what can be delivered in the Sprint and how that work will be achieved.
2. **Daily Stand-up:**
   * Short, daily meetings to synchronize activities and create a plan for the next 24 hours.
3. **Sprint Review:**
   * Meeting at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed.
4. **Sprint Retrospective:**
   * Reflect on the past Sprint and identify improvements for the next Sprint.

**Project Outline**

**Product Backlog:**

1. **User Authentication:**
   * Implement user login and access control.
2. **File Upload and Handling:**
   * Allow users to upload various file types (PDF, DOCX, XLSX).
3. **Content Detection and Recognition:**
   * Develop algorithms for detecting and recognizing contents in uploaded files using computer vision and data mining techniques.
4. **Data Extraction:**
   * Extract filled questionnaire data from different file formats.
5. **Data Summarization:**
   * Summarize extracted data and display it in an intuitive interface.
6. **Review and Edit Interface:**
   * Enable users to review, edit, and reject extracted data.
7. **Export Functionality:**
   * Allow exporting of processed data to various file types (CSV, JSON, XML).
8. **Error Handling and Validation:**
   * Implement error handling and validation for file uploads and data extraction.
9. **Integration with Existing Systems:**
   * Integrate the tool with existing firm systems and databases.
10. **Performance Optimization:**
    * Ensure the application performs efficiently with large datasets.
11. **User Interface Design:**
    * Develop a user-friendly interface using React.js.
12. **Backend Development:**
    * Implement the backend logic using Python.
13. **Testing:**
    * Conduct unit tests, integration tests, and user acceptance tests.

**Sprint Backlog for Sprint 1:**

1. User Authentication
   * Implement basic user login.
   * Set up access control mechanisms.
2. File Upload and Handling
   * Develop the file upload interface.
   * Support for uploading PDF and DOCX files.

**Sprint Backlog for Sprint 2:**

1. Content Detection and Recognition
   * Develop and integrate initial content recognition algorithms.
   * Test with sample files.

**Sprint Backlog for Sprint 3:**

1. Data Extraction
   * Implement data extraction for PDF files.
   * Begin development for DOCX extraction.

**Sprint Backlog for Sprint 4:**

1. Data Summarization
   * Develop the data summarization module.
   * Display extracted data in a user-friendly format.

**Sprint Backlog for Sprint 5:**

1. Review and Edit Interface
   * Create the interface for reviewing and editing extracted data.
   * Implement reject functionality.

**Sprint Backlog for Sprint 6:**

1. Export Functionality
   * Enable exporting of data to CSV and JSON formats.
   * Validate the export functionality with sample data.

**Sprint Backlog for Sprint 7:**

1. Error Handling and Validation
   * Add error handling for file uploads.
   * Validate extracted data for consistency.

**Sprint Backlog for Sprint 8:**

1. Integration with Existing Systems
   * Connect the tool to existing firm databases.
   * Ensure data flows correctly between systems.

**Sprint Backlog for Sprint 9:**

1. Performance Optimization
   * Optimize data processing algorithms.
   * Test performance with large datasets.

**Sprint Backlog for Sprint 10:**

1. User Interface Design
   * Refine the React.js front-end.
   * Ensure a consistent and intuitive user experience.

**Sprint Backlog for Sprint 11:**

1. Backend Development
   * Finalize backend logic using Python.
   * Ensure seamless integration with front-end.

**Sprint Backlog for Sprint 12:**

1. Testing
   * Conduct unit tests and integration tests.
   * Perform user acceptance testing with stakeholders.

**Release Plan:**

* **Release 1.0 (End of Sprint 6):**
  + Basic functionality including user authentication, file upload, content detection, data extraction for PDFs, and data summarization.
* **Release 2.0 (End of Sprint 9):**
  + Full functionality including review and edit interface, export functionality, error handling, and validation.
* **Release 3.0 (End of Sprint 12):**
  + Performance optimizations, integration with existing systems, refined UI, final backend implementation, and thorough testing.

**Definition of Done:**

1. Code is written, tested, and reviewed.
2. Features are fully documented.
3. User stories are accepted by the Product Owner.
4. No critical bugs are present.
5. The Increment is integrated into the main branch without issues.
6. The application is deployable.