**ReqVista - Detailed Deliverables**

**1. Software Components**

**1.1 Database and Data Access**

* **Azure SQL Database**: Fully implemented cloud database based on the ER diagram
* **Entity Framework Core Models**: C# entity models derived from ER diagram with ObservableProperty attributes
* **Database Migration Scripts**: Scripts for creating and updating database schema
* **Data Access Layer**: Repository pattern implementation with Entity Framework Core
* **Query Optimization**: Optimized queries and indexes for performance
* **Data Seeding Scripts**: Initial data for testing and development environments

**1.2 API Services**

* **RESTful API**: Complete set of API endpoints for all system entities
* **API Documentation**: Swagger/OpenAPI documentation
* **Authentication Middleware**: JWT-based authentication implementation
* **Authorization Policies**: Role-based access control for all endpoints
* **Validation Logic**: Request validation with error handling
* **API Versioning**: Implementation of API versioning strategy
* **Integration Endpoints**: Endpoints specifically for external system integration
* **Background Services**: Services for asynchronous processing and scheduled tasks

**1.3 Blazor Web Application**

* **Component Library**: Reusable UI components based on design system
* **Page Implementations**: Complete set of pages for all system functionality
* **Authentication UI**: User login, registration, and profile management
* **Authorization UI**: Role and permission management interface
* **Responsive Layouts**: Layouts optimized for various screen sizes
* **Form Implementations**: Forms for all data entry scenarios
* **Client-side Validation**: Input validation implementation
* **State Management**: State management solution for the application
* **Reporting UI**: Interface for generating and viewing reports
* **Administration Console**: System administration interface

**1.4 .NET MAUI Mobile Application**

* **UI Components**: Native mobile UI components
* **Offline Data Store**: LiteDB implementation for offline storage
* **Synchronization Service**: Bidirectional sync between mobile and backend
* **Mobile-specific Views**: Optimized views for mobile experience
* **Native Integration**: Integration with device capabilities (camera, notifications, etc.)
* **Authentication Implementation**: Secure authentication for mobile users
* **Offline Workflow Support**: Support for working entirely offline when needed
* **Push Notification Integration**: Push notification implementation for mobile users

**1.5 Shared Component Libraries**

* **View Models**: Shared ViewModels between web and mobile
* **Business Logic**: Shared business logic components
* **Validation Rules**: Shared validation logic
* **Helper Utilities**: Common utility functions and extensions
* **Theme Components**: Shared styling and theming components
* **API Client**: Shared API access client
* **Authentication Components**: Shared authentication logic

**1.6 Integration Components**

* **External API Clients**: Client implementations for external systems
* **Webhook Implementation**: Webhook processing for external notifications
* **Import/Export Services**: Services for importing and exporting data
* **File Processing**: Components for processing uploaded files
* **Email Integration**: Services for email notifications
* **Reporting Engine**: Integration with reporting tools

**2. Documentation**

**2.1 System Documentation**

* **System Architecture Document**: Overall architecture diagrams and descriptions
* **Component Design Documents**: Detailed design for each major component
* **API Reference**: Comprehensive API documentation with examples
* **Data Dictionary**: Definitions of all data elements in the system
* **Security Design Document**: Security architecture and implementation details
* **Integration Guide**: Guide for integrating with external systems
* **Performance Optimization Guide**: Strategies for optimizing system performance

**2.2 User Documentation**

* **User Manual - Web Application**: Complete user guide for web application
* **User Manual - Mobile Application**: Complete user guide for mobile application
* **Quick Start Guides**: Task-oriented guides for common scenarios
* **Video Tutorials**: Instructional videos for key features
* **FAQ Document**: Answers to frequently asked questions
* **Troubleshooting Guide**: Solutions for common issues

**2.3 Administrative Documentation**

* **Installation Guide**: Instructions for system installation and configuration
* **Upgrade Guide**: Procedures for upgrading the system
* **Backup and Recovery Guide**: Procedures for backup and disaster recovery
* **Security Administration Guide**: Guide for security configuration and management
* **Performance Tuning Guide**: Instructions for optimizing system performance
* **Monitoring Guide**: Procedures for system monitoring and alerting

**2.4 Development Documentation**

* **Source Code Documentation**: Inline code documentation and generated reference
* **Development Setup Guide**: Instructions for setting up development environment
* **Coding Standards**: Coding conventions and best practices
* **Extension Guide**: Guide for extending and customizing the system
* **Test Documentation**: Documentation of testing approach and procedures
* **Release Notes**: Detailed notes for each system release

**3. Infrastructure**

**3.1 Azure Environment**

* **Resource Group Configuration**: Organized Azure resource groups
* **Azure SQL Database**: Configured production database
* **App Service Plans**: Appropriate service plans for web and API hosting
* **API Management**: API management configuration if required
* **Storage Accounts**: Configured storage for files and backups
* **CDN Configuration**: Content delivery network for static assets
* **Azure Monitor**: Monitoring and alerting configuration
* **Application Insights**: Application performance monitoring setup
* **Key Vault**: Secure storage for secrets and certificates
* **Azure Active Directory**: Authentication and identity configuration

**3.2 Containerization**

* **Docker/Podman Files**: Container definitions for all components
* **Container Registry**: Private registry for container images
* **Orchestration Configuration**: Kubernetes or Container Apps configuration
* **Container Security**: Secured container configuration
* **Container Networking**: Networking configuration for containers
* **Storage Integration**: Persistent storage configuration for containers

**3.3 CI/CD Pipeline**

* **Build Pipelines**: Automated build pipelines for all components
* **Test Pipelines**: Automated test execution pipelines
* **Deployment Pipelines**: Automated deployment pipelines for all environments
* **Quality Gates**: Quality checks integrated into pipelines
* **Environment Configurations**: Configuration for development, testing, staging, and production
* **Infrastructure as Code**: ARM templates or Terraform scripts for infrastructure

**3.4 Development Environment**

* **Developer VM Images**: Configured virtual machine images for development
* **Local Development Tools**: Configuration for local development environment
* **Podman Development Setup**: Configuration for container-based development
* **Mock Services**: Mock implementations of external services for development
* **Test Data Generation**: Tools for generating test data

**4. Testing Assets**

**4.1 Test Plans and Cases**

* **Test Strategy Document**: Overall testing approach and methodology
* **Functional Test Cases**: Test cases for all functional requirements
* **Non-functional Test Cases**: Test cases for performance, security, etc.
* **Regression Test Suite**: Test cases for regression testing
* **Acceptance Test Criteria**: Defined criteria for user acceptance testing
* **Defect Tracking Process**: Process for tracking and resolving defects

**4.2 Automated Testing**

* **Unit Test Suite**: Automated unit tests for all components
* **Integration Test Suite**: Automated integration tests
* **API Test Suite**: Automated tests for API endpoints
* **UI Test Suite**: Automated tests for web and mobile UI
* **Load Test Scripts**: Scripts for performance and load testing
* **Security Test Scripts**: Automated security testing scripts

**4.3 Test Environments**

* **Test Database**: Configured database for testing
* **Test API Services**: Deployed API services for testing
* **Test Web Application**: Deployed web application for testing
* **Test Mobile Application**: Configured mobile application for testing
* **Test User Accounts**: Configured test user accounts with various roles
* **Data Reset Scripts**: Scripts for resetting test environments

**4.4 Test Results and Reports**

* **Test Execution Reports**: Reports of test execution results
* **Defect Reports**: Documentation of identified defects
* **Performance Test Results**: Results of performance testing
* **Security Scan Reports**: Results of security scans
* **Code Coverage Reports**: Reports of code coverage from automated tests
* **User Acceptance Test Results**: Results of user acceptance testing

**5. Training Materials**

**5.1 End User Training**

* **Training Curriculum**: Structured curriculum for user training
* **Training Manuals**: Comprehensive training materials
* **Training Exercises**: Hands-on exercises for users
* **Training Videos**: Instructional videos for key features
* **Quick Reference Guides**: Brief guides for common tasks
* **Online Help System**: Context-sensitive help integrated into the application

**5.2 Administrator Training**

* **Administrator Training Curriculum**: Curriculum for system administrators
* **Installation and Configuration Training**: Training for system setup
* **Security Administration Training**: Training for security configuration
* **Troubleshooting Training**: Training for issue diagnosis and resolution
* **Performance Optimization Training**: Training for system optimization
* **Disaster Recovery Training**: Training for backup and recovery procedures

**5.3 Developer Training**

* **Developer Onboarding Guide**: Guide for new developers
* **Architecture Training**: Training on system architecture
* **API Usage Training**: Training for using and extending the API
* **Extension Development Guide**: Guide for developing system extensions
* **Development Best Practices**: Documentation of development best practices
* **Code Review Guidelines**: Standards for code review

**6. Project Management Deliverables**

**6.1 Project Planning**

* **Project Schedule**: Detailed project timeline and milestones
* **Resource Plan**: Allocation of resources to project activities
* **Risk Register**: Identification and mitigation of project risks
* **Communication Plan**: Plan for project communications
* **Quality Management Plan**: Plan for ensuring project quality
* **Change Management Process**: Process for handling change requests

**6.2 Project Tracking**

* **Status Reports**: Regular reports on project progress
* **Milestone Completion Reports**: Documentation of milestone completion
* **Issue Log**: Tracking of project issues and resolutions
* **Change Request Log**: Documentation of change requests and dispositions
* **Meeting Minutes**: Documentation of project meetings
* **Burndown/Burnup Charts**: Visual tracking of project progress

**6.3 Project Closure**

* **Project Completion Report**: Final report on project completion
* **Lessons Learned Document**: Documentation of project lessons learned
* **Transition Plan**: Plan for transitioning to operations
* **Support Handover Document**: Documentation for support team
* **Final Acceptance Document**: Documentation of final acceptance
* **Asset Inventory**: Inventory of project deliverables

**7. Operational Support**

**7.1 Support Documentation**

* **Support Procedures**: Documented support procedures
* **Escalation Paths**: Defined paths for issue escalation
* **SLA Documentation**: Definition of service level agreements
* **Known Issues and Workarounds**: Documentation of known issues
* **Troubleshooting Guide**: Guide for resolving common issues
* **System Monitoring Guide**: Guide for monitoring system health

**7.2 Maintenance Procedures**

* **Maintenance Schedule**: Schedule for routine maintenance
* **Patch Management Process**: Process for applying system patches
* **Database Maintenance Procedures**: Procedures for database maintenance
* **Backup Verification Procedures**: Procedures for verifying backups
* **Performance Tuning Procedures**: Procedures for ongoing performance optimization
* **Security Update Process**: Process for applying security updates

**7.3 Operational Tools**

* **Monitoring Dashboard**: Dashboard for system monitoring
* **Alert Configuration**: Configured alerts for system issues
* **Log Analysis Tools**: Tools for analyzing system logs
* **Performance Analysis Tools**: Tools for analyzing system performance
* **User Management Tools**: Tools for managing system users
* **Reporting Tools**: Tools for generating operational reports

**8. Compliance and Governance**

**8.1 Security Compliance**

* **Security Controls Documentation**: Documentation of implemented security controls
* **Vulnerability Assessment**: Assessment of system vulnerabilities
* **Penetration Test Report**: Results of security penetration testing
* **Data Protection Impact Assessment**: Assessment of data protection measures
* **Security Incident Response Plan**: Plan for responding to security incidents
* **Access Control Documentation**: Documentation of access control implementation

**8.2 Audit Support**

* **Audit Trail Implementation**: Implementation of comprehensive audit logging
* **Audit Report Templates**: Templates for generating audit reports
* **Compliance Checklist**: Checklist for verifying compliance
* **Data Retention Implementation**: Implementation of data retention policies
* **Data Classification Guide**: Guide for classifying system data
* **Regulatory Compliance Documentation**: Documentation of regulatory compliance

**8.3 Governance**

* **System Governance Framework**: Framework for ongoing system governance
* **Policy Documentation**: Documentation of system policies
* **Role and Responsibility Matrix**: Definition of roles and responsibilities
* **Decision-making Process**: Documented process for system decisions
* **Change Advisory Board Charter**: Charter for the change advisory board
* **Release Management Process**: Process for managing system releases