Vision and Scope Document

for

ARIA

Advanced Requirements Intelligence & Analytics

Version 1.0 approved

Prepared by Crackers team

20.09.2025

Table of Contents

1. Business Requirements **1**
1.1. Background 1
1.2. Business Opportunity 1
1.3. Business Objectives 1
1.4. Success Metrics 2
1.5. Vision Statement 2
1.6. Business Risks 2
1.7. Business Assumptions and Dependencies 2
2. Scope and Limitations **3**
2.1. Major Features 3
2.2. Scope of Initial Release 3
2.3. Scope of Subsequent Releases 3
2.4. Limitations and Exclusions 3
3. Business Context **4**
3.1. Stakeholder Profiles 4
3.2. Project Priorities 5
3.3. Deployment Considerations 5

Revision History

Name	Date	Reason For Changes	Version
Initialization	20.09.2025	-	1.0

1. Business Requirements

1.1. Background

Traditional requirements prioritization is time-consuming and subjective, becoming impractical for projects with over 100 requirements. Manual methods lead to inconsistent priorities and delayed decisions. Our AI-powered tool automates multi-criteria analysis, enabling rapid, objective prioritization for mid-sized projects.

1.2. Business Opportunity

The global requirements management tools market is experiencing rapid growth, driven by increasing demand for automated, intelligent solutions. Current market solutions lack comprehensive AI-powered prioritization capabilities that integrate cost, risk, and business value analysis in a unified platform.

Key market gaps include:
- Limited AI integration in existing requirements management platforms
- Absence of comprehensive trade-off analysis tools
- Lack of automated priority scoring based on multiple criteria
- Insufficient visual representation of requirements priorities and dependencies
- Poor scalability for large-scale projects with hundreds of requirements

Organizations in software development, product management, and project management sectors require intelligent tools that can process qualitative and quantitative data to deliver objective prioritization recommendations, ultimately improving project success rates and resource utilization.

1.3. Business Objectives

- Deliver a functional MVP within 8 weeks that processes 50–100 requirements and generates prioritized output using weighted scoring and a simple ML model
- Achieve ≥70% agreement with expert manual prioritization on a test set of 30 requirements
- Ensure average processing time ≤5 seconds for 100 requirements
- Obtain ≥80% positive usability feedback from 10 student/instructor testers
- Document architecture, API, and model training process by week 10 for future handover

1.4. Success Metrics

Category	Metric	Target
Functionality	Demo with 50–100 requirements → priority list + chart	Achieved in week 8
Accuracy	Correlation with expert ranking	≥70% on 30-item test set
Performance	Processing time per run	≤5s for 100 requirements
Reliability	Critical bugs in final demo	<3
Usability	Positive feedback rate	≥80%
Documentation	Architecture diagram, README, API spec	Completed by week 10
Team Efficiency	Sprint milestone delivery	4 sprints completed on schedule

1.5. Vision Statement

Develop a proof-of-concept AI tool that automates requirements prioritization, demonstrating how lightweight ML can support decision-making in academic, early-stage startup projects and enterprise companies.

1.6. Business Risks

- Quality: Limited accuracy of understanding for different requirements
- Team expertise: Mixed student skill levels in MLOps
- Time Constraints: Semester timeline limits advanced features

1.7. Business Assumptions and Dependencies

- Students devote ~15 hours/week each
- Access to university servers for deployment
- Sample requirement datasets available
- Use of open-source LLM models

2. Scope and Limitations

2.1. Major Features

- Requirements Input: CSV/Excel upload; template form
- Prioritization Engine: Weighted scoring + decision tree model
- Visualization: Sorted list; bar chart of scores
- Export: CSV export; static HTML report

2.2. Scope of Initial Release

- Single-user web app with core prioritization workflow
- No authentication; English-only text
- Deployed on Heroku free tier or university server

2.3. Scope of Subsequent Releases

- Version 2.0: User login; support 100+ requirements; basic API
- Version 3.0 (future): Advanced models; multi-user collaboration; integrations

2.4. Limitations and Exclusions

- No real-time collaboration or mobile native app
- No enterprise-level security or integrations
- Maximum 100 requirements per session

3. Business Context

3.1. Stakeholder Profiles

Stakeholder	Major Value	Attitudes	Major Interests	Constraints
Course Instructor	*Clear demonstration of AI-driven requirements prioritization*	*Supportive, academically rigorous*	*Concept validation, teaching outcomes*	*Semester timeline, academic evaluation criteria*
Local Startups	*Lightweight prioritization tool for early-stage product development*	*Curious, cost-sensitive*	*Quick prototyping, actionable insights*	*MVP scope, limited support*
Enterprise Teams	*Scalable, objective prioritization for large requirement sets*	*Interested but cautious, security-focused*	*Integration with existing workflows, compliance*	*Integration complexity, data privacy and security standards*
Business Analysts	*Streamlined requirement analysis with traceability*	*Analytical, process-driven*	*Stakeholder alignment, documentation quality*	*Maintaining documentation standards, auditability*
Product Managers	*Data-driven feature prioritization*	*Outcome-focused, strategic*	*Business value maximization, roadmap alignment*	*Rapid decision cycles, stakeholder buy-in*

3.2. Project Priorities

Dimension	Driver (state objective)	Constraint (state limits)	Degree of Freedom (state allowable range)
Schedule	*Deliver MVP by end of week 8; full polish by week 10*	*10-week project period*	*±1 week buffer for testing and documentation*
Features	*Implement core prioritization workflow (input, scoring, output)*	*Advanced LLM/ML and integrations deferred*	*Can include one additional feature if core is stable*
Quality	*Ensure stable demo with ≤3 critical bugs*	*Zero tolerance for crashes during demo*	*Up to 5 minor bugs allowed, prioritized for post-demo fixes*

Performance	*Achieve ≤5s processing for 100 requirements*	*Heroku/free-tier deployment limits CPU/memory*	*Processing time ≤7s acceptable under high load*
Usability	*Obtain ≥80% positive usability feedback*	*Limited user testing pool (10 testers)*	*≥70% positive feedback acceptable if test pool expands*
Team Capacity	*Complete sprints on time with 15h/week per member*	*Part-time student involvement*	*Up to 20h/week if needed during critical sprints*

3.3. Deployment Considerations

- Environment: Docker containers deployed on Heroku free tier or Innopolis servers
- Monitoring: Basic logging; enterprise-grade monitoring deferred
- Security: Input validation for MVP; enterprise security (SSO, encryption) planned for post stages
- Backup: Weekly automated backups; enterprise backup strategy in future roadmap