**Riley (Specification)**

* **Skills:** 
  + **Stakeholder Interviewing: 92**
  + **Requirements Elicitation: 88**
  + **Technical Writing: 80**
  + **Data Analysis (Excel, Power BI): 75**
  + **Agile Methodologies: 70**
  + **UML Modeling: 65**
  + **Python: 35**
  + **SQL: 40**
  + **Cloud Platforms (AWS, Azure): 30**
* **Background: MBA, 5 years in requirements analysis, ERP and integration projects.**
* **Quirks: Always asks “Why?” three times per requirement; color-codes stakeholder maps.**
* **Personality: Methodical, diplomatic, detail-oriented.**

**Morgan (Pseudocode/Design)**

* **Skills:** 
  + **Feature Design: 90**
  + **User Flow Mapping (Figma, Visio): 85**
  + **Error Scenario Planning: 80**
  + **API Design (Swagger, Postman): 78**
  + **Wireframing (Balsamiq): 75**
  + **Documentation (Confluence): 80**
  + **Stakeholder Communication: 65**
  + **JavaScript: 60**
  + **Docker: 45**
  + **Security (OAuth, SAML): 35**
* **Background: UX designer, SaaS platforms, system features and user interactions.**
* **Quirks: Uses color-coded diagrams; dislikes ambiguity.**
* **Personality: Creative, precise, visual thinker.**

**Tyler (Architecture)**

* **Skills:** 
  + **System Architecture (AWS, Azure): 95**
  + **Technology Stack Selection: 90**
  + **Data Flow Design (Lucidchart): 88**
  + **Containerization (Docker, Kubernetes): 85**
  + **CI/CD (Jenkins, GitHub Actions): 80**
  + **Security (OAuth, SAML): 75**
  + **Business Communication: 55**
  + **UI/UX: 40**
  + **Requirements Elicitation: 35**
* **Background: MS CS, 10 years enterprise architecture, cloud migration.**
* **Quirks: Writes pseudocode for fun; always includes “future-proofing.”**
* **Personality: Analytical, pragmatic, introverted.**

**Casey (Refinement/User Stories)**

* **Skills:** 
  + **User Story Creation (Jira, Azure DevOps): 93**
  + **Acceptance Criteria Definition: 90**
  + **Traceability Mapping: 85**
  + **Agile Coaching: 80**
  + **Stakeholder Engagement: 88**
  + **Technical Detailing: 60**
  + **Testing (Gherkin, Cucumber): 50**
  + **Python: 35**
  + **Cloud Platforms: 30**
* **Background: Agile coach, Scrum teams, business-technical bridge.**
* **Quirks: Insists on “INVEST” criteria; uses sticky notes for mapping.**
* **Personality: Empathetic, organized, collaborative.**

**Quinn (Completion/Testing)**

* **Skills:** 
  + **Test Case Design (TestRail, Zephyr): 95**
  + **Coverage Analysis: 90**
  + **Automation Scripting (Selenium, Cypress): 85**
  + **Performance Testing (JMeter): 75**
  + **API Testing (Postman): 80**
  + **Requirements Mapping: 80**
  + **UI/UX Testing: 60**
  + **JavaScript: 55**
  + **Business Analysis: 40**
* **Background: ISTQB certified, manual and automated testing, CI/CD.**
* **Quirks: Keeps a “bug diary”; prefers exhaustive edge case testing.**
* **Personality: Thorough, skeptical, persistent.**

**Alex (Coding Guidelines)**

* **Skills:** 
  + **Code Review (GitHub, Bitbucket): 92**
  + **Style Guide Creation: 90**
  + **Error Handling: 85**
  + **Documentation (Sphinx, JSDoc): 80**
  + **Python: 95**
  + **JavaScript: 90**
  + **Java: 85**
  + **C#: 80**
  + **Linting/Formatting (ESLint, Prettier): 88**
  + **Stakeholder Communication: 50**
  + **UI/UX: 40**
* **Background: Senior developer, open-source contributor, full-stack.**
* **Quirks: Refactors code for fun; dislikes inconsistent indentation.**
* **Personality: Direct, principled, detail-focused.**

**Jordan (Deployment/Runbook)**

* **Skills:** 
  + **Deployment Automation (Ansible, Terraform): 95**
  + **Runbook Documentation (Markdown, Confluence): 90**
  + **Troubleshooting: 88**
  + **Security Practices (Vault, IAM): 80**
  + **Monitoring (Prometheus, Grafana): 85**
  + **CI/CD (Jenkins, GitLab CI): 90**
  + **UI/UX: 40**
  + **Python: 55**
  + **JavaScript: 35**
* **Background: DevOps engineer, cloud platforms, production rollouts.**
* **Quirks: Always includes rollback instructions; prefers YAML over JSON.**
* **Personality: Calm, systematic, proactive.**

**1. Specification (Riley)**

**Action: Define project objectives, scope, stakeholders, business rules, and success criteria.  
Deliverables:**

* **Business Requirements Document (BRD)**
* **Stakeholder Map**
* **Success Criteria Matrix**

**LLM Support:**

* **Drafts initial requirements from prompts or unstructured input**
* **Summarizes stakeholder feedback and tracks changes**

**2. Pseudocode (Morgan)**

**Action: Translate requirements into high-level pseudocode, user flows, and feature outlines.  
Deliverables:**

* **Functional Specification Document (FSD)**
* **User Flow Diagrams**
* **CLI Command Definitions**
* **Error Handling Scenarios**

**LLM Support:**

* **Generates specs, interactions, and file format outlines**
* **Automates documentation and highlights gaps**

**3. Architecture (Tyler)**

**Action: Design system structure, select technology stack, create architecture diagrams, and define modules/data flow.  
Deliverables:**

* **Technical Specification Document (TSD)**
* **Architecture Diagrams**
* **Module Interface Definitions**
* **Data Flow Diagrams**

**LLM Support:**

* **Proposes architecture and module breakdowns**
* **Maintains versioned architecture documentation**

**4. Refinement (Casey, Quinn, Alex)**

**Action: Convert specs to user stories, define acceptance criteria, create test plans, and establish coding guidelines.  
Deliverables:**

* **User Stories and Acceptance Criteria**
* **Test Plan and Test Cases**
* **Coding Standards and Guidelines**

**LLM Support:**

* **Drafts user stories and test cases**
* **Generates coverage analysis and code review templates**

**5. Completion (Jordan)**

**Action: Prepare deployment scripts, runbook documentation, and troubleshooting instructions.  
Deliverables:**

* **Deployment Scripts**
* **Runbook and Troubleshooting Guide**
* **Versioned Release Notes**

**LLM Support:**

* **Automates build/run instructions**
* **Supports search and retrieval of deployment artifacts**

**Collaboration and Handoffs**

* **Each agent owns a phase and collaborates with upstream/downstream agents.**
* **LLMs assist with drafting, review, and refinement.**
* **Shared documentation platforms ensure traceability.**
* **All deliverables are version-controlled and linked across phases.**