Renkemet Dev Agile-Scrum Charter

1. Team Name:

Renkemet Dev

2. Vision:

To provide a one-stop-shop or linguistic full-service for ancient Egyptian linguistics and related primary sources.

3. Mission:

To develop a scalable, cloud-native language learning and management platform that delivers personalized experiences, ensures security, and leverages AWS for high availability and performance.

4. Success Criteria

- MVP deployment by December 2025 with authentication and real-time analytics.
- Maintain 99.9% uptime for backend and frontend services.
- Keep API response time below 200ms, leveraging caching and optimized queries.
- CI/CD automation for backend, frontend, and infrastructure with rollback safety.
- Implement secure authentication & authorization via AWS Cognito/Auth0.
- Enable real-time analytics & monitoring with AWS Kinesis and OpenTelemetry.
- Ensure automatic rollback on failed deployments and error rate thresholds.

5. Guiding Principles:

- Strictly adhere to the scrum manifesto: Individuals over processes, working code over robust documentation, collaboration over negotiation and high adaptability.
- Cloud-first approach leveraging AWS-native services.
- Deliver working software in short, iterative cycles.
- Focus on API-first development with microservices.
- Infrastructure as Code (IaC) for consistency and automation.
- Ensure observability and monitoring using AWS CloudWatch and logs.
- Multi-region failover strategies for high availability.
- Advanced security policies (IAM best practices, API security, data encryption).
- Cost optimization techniques (compute, storage, and network efficiency).

6. Team:

Name	Role	Responsibility	Contact
**	Product Owner	Defines backlog priorities, aligns with business goals.	**
**	Scrum Master	Facilitates Scrum events, removes impediments, ensures Agile adoption.	**
**	Backend Engineer	Develop and maintain Express.js microservices, integrate MongoDB Atlas, Redis, and authentication services	**
**	Frontend Engineer	Build a responsive UI (React), integrate APIs, optimize caching & performance.	**
**	DevOps Engineer	Manages CI/CD pipeline, Terraform IaC, AWS ECS, logging, monitoring, and security.	**
**	QA Engineer	Automates API/UI testing, ensures performance, security, and integration testing.	**
**	Data Engineer	Implements user behavior tracking, analytics pipelines using AWS Kinesis & OpenTelemetry.	**

** teammates pending

7. Scope & Constraints:

✅ In Scope:	❌ Out of Scope:
• Authentication & Authorization via AWS Cognito/Auth0 (OAuth, JWT, RBAC). • User behavior tracking & analytics (AWS Kinesis, Google Analytics). • Observability & auto-scaling policies (AWS CloudWatch, Prometheus, OpenTelemetry). • Error handling with automated alerts & rollback triggers. • Microservices-based backend with Express.js, MongoDB Atlas, and Redis caches. • AWS API Gateway & ALB for load balancing and routing. • React frontend deployed on S3 & served via AWS CloudFront. • AWS ECS (Fargate) auto-scaling for backend services. • CI/CD pipeline automation using GitHub Actions & Terraform. • S3 rollback logs, automated cleanup & CloudWatch monitoring.	• Mobile app development (Current focus: PWA). • AI-driven artifact recognition and resource recommendations (Future phase). • Multi-tenant (User Groups) enterprise features (Future phase).

8. Ways of Working:

- Sprint Duration: 2 weeks.
- Sprint Planning: Every Monday at 10 AM.
- Daily Standup: 15-minute sync at 9:30 AM.
- Sprint Review & Demo: Every Friday at 3 PM.
- Sprint Retrospective: Every Friday at 4 PM.
- Code Reviews: Mandatory for all pull-requests before merging.
- Security Audits: Monthly IAM & API security reviews.

9. Definition of Done (DoD):

A user story is considered Done when:

- ✅ Code is written, reviewed, and merged.
- ✅ Express.js microservice is deployed via ECS (Fargate) and is accessible via API Gateway/ALB.
- ✅ Authentication & role-based access control (RBAC) is implemented using AWS Cognito/Auth0.
- ✅ Automated tests (unit, integration, security) pass with 80%+ coverage.
- ✅ MongoDB Atlas & Redis caching optimizations are in place.
- ✅ React frontend updates are deployed to S3 & CloudFront with cache invalidation.
- ✅ AWS CloudWatch, Prometheus, and OpenTelemetry monitoring is active.
- ✅ Real-time user behavior tracking is enabled via AWS Kinesis & Google Analytics.
- ✅ AWS security & IAM policies are reviewed for compliance.
- ✅ Documentation (API, Infra, Security, DevOps) is updated.

10.Risks & Assumptions:

Risks:	Assumptions:
• Authentication failures due to misconfigured OAuth/JWT policies. • Latency issues due to MongoDB Atlas multi-region setups. • Scaling issues with ECS auto-scaling during peak traffic. • Security vulnerabilities due to misconfigured IAM roles or API Gateway.	• AWS infrastructure budget is approved and available. • IAM roles and policies are properly configured for AWS service integrations. • Developers are familiar with Express.js, MongoDB Atlas, Redis, and AWS. • Security best practices (e.g., least privilege IAM policies, API rate limiting) are followed.

Approved By:

_____ Date: Feb 2, 2025

[Approver Name] (CTO, Selfpacer Technologies)