**Deployment Model**

**Overview:**

The system will be deployed in a cloud-based environment(AWS) that allows for availability scalability and maintainability. We will be making use of containerization and orchestration tools to manage efficiency. The deployment model will separate multi-tier deployment of the frontend, API, database, and AI agent services for modularity.

**Target Environments**

* Frontend: Vercel
* Backend: AWS ECS
* Database: AWS RDS
* Load Balancing: AWS ELB
* Media Storage: AWS S3
* Monitoring AWS CloudWatch

**Deployment Topology**

Frontend

* Deployed as a serverless web app
* Connect to backend over HTTPS requests

Backend API

* Containerized with Docker
* Ochestrated with AWS ECS
* Scaled horizontally with a Load balancer provided with AWS ELB
* Authentication of HTTP requests

Database

* Hosted with AWS RDS, providing high availability
* Automated backups and replication

**Quality requirements supported**

Scalability: Vercel provides automatic concurrency scaling, load balancer will be used to scale the backend, AWS read replicas achieve horizontal scaling

Reliability: AWS RDS deployment provides multi-AZ deployments for high availability as well as automated backups

Maintainability: Achieved with modular containerized services and stateless frontend

Performance: Frontend has a global CDN, caching done on backend  
  
