Here's a summary of the deployment architecture for a task management system designed to support 10,000 daily users:

**System Requirements**

* 1,000-1,500 concurrent users during peak hours
* 150-300 requests per second at peak load
* Initially less than 100GB storage needed

**Core Architecture Components**

1. **Kubernetes Cluster**: Managed service (AKS, EKS, or GKE) with 3-5 nodes
2. **API Layer**: Starting with 3 replicas, scaling up to 10 with Horizontal Pod Autoscaler
3. **Database**: Managed SQL database or self-hosted SQL Server with 4 vCPUs and 16GB RAM
4. **Optional Redis Cache**: For frequently accessed data with appropriate TTL values
5. **Authentication**: AWS Cognito for user management
6. **API Gateway**: Kubernetes Ingress with SSL termination and rate limiting
7. **Monitoring & Logging**: Application Insights or Prometheus + Grafana

**Scalability Strategy**

* Initial setup comfortably handles 10,000 daily users
* For growth beyond this: consider microservices architecture, database scaling with read replicas, and multi-region deployment

**Deployment Approach**

* CI/CD pipeline with staged environments
* Blue/Green deployment for zero-downtime updates

**Estimated Cost**

* $550-1,000 per month for the complete cloud-based setup

The architecture is designed to be robust yet cost-effective, with clear recommendations for starting conservatively and scaling based on actual usage patterns and growth.