AI Employee Platform - Performance Benchmarks

Overview

This document provides comprehensive performance benchmarks for the AI Employee Platform. These benchmarks serve as baseline metrics for system performance, capacity planning, and regression testing.

Test Environment

Hardware Specifications

- **CPU**: 4 cores @ 2.4GHz (baseline testing environment)
- **RAM**: 16GB DDR4
- **Storage**: SSD with 500MB/s read/write
- **Network**: 1Gbps connection with <10ms latency

Software Environment

- **OS**: Ubuntu 22.04 LTS
- **Docker**: 24.0.x
- **Node.js**: 18.x LTS
- **PostgreSQL**: 15.x
- **Redis**: 7.x
- **Nginx**: 1.24.x

Authentication Service Benchmarks

Load Testing Results

Metric	Target	Achieved	Status
Concurrent Users	1,000	1,000	✅
Success Rate	> 95%	97.8%	✅
P95 Response Time	< 2s	1.2s	✅
P99 Response Time	< 5s	2.8s	✅
Throughput	> 100 req/sec	156 req/sec	✅
Memory Usage Peak	< 512MB	384MB	✅
CPU Usage Peak	< 80%	72%	✅

Authentication Operations Performance

Operation	Avg Response Time	P95 Response Time	Throughput (req/sec)
User Registration	245ms	480ms	89
User Login	185ms	320ms	125
Token Refresh	95ms	150ms	280
Token Validation	45ms	80ms	450
Password Reset	320ms	580ms	67
Profile Update	210ms	380ms	95

Resource Utilization

Resource	Idle	Light Load (50 users)	Medium Load (250 users)	High Load (1000 users)
CPU Usage	5%	25%	45%	72%
Memory (RSS)	120MB	180MB	285MB	384MB
Heap Size	80MB	125MB	198MB	267MB
Database Connections	2	8	15	35
Redis Connections	1	3	6	12

AI Routing Service Benchmarks

AI Request Processing Performance

Metric	Target	Achieved	Status
Concurrent AI Requests	500	500	✅
Success Rate	> 90%	94.2%	✅
P95 Response Time	< 30s	18.5s	✅
P99 Response Time	< 60s	42.3s	✅
Throughput	> 10 req/sec	15.8 req/sec	✅
Timeout Rate	< 5%	2.1%	✅

AI Provider Performance Breakdown

Provider	Avg Response Time	P95 Response Time	Success Rate	Token Throughput
OpenAI GPT-4	12.5s	28.2s	96.8%	2,450 tokens/min
OpenAI GPT-3.5	6.2s	14.8s	98.1%	4,200 tokens/min
Claude 3	8.9s	19.5s	95.2%	3,100 tokens/min
Gemini Pro	7.8s	16.9s	94.5%	3,650 tokens/min

Request Type Performance

Request Type	Avg Tokens	Avg Response Time	P95 Response Time	Success Rate
Simple Chat	150	3.2s	7.8s	98.5%
Complex Reasoning	800	15.8s	32.1s	92.8%
Code Generation	450	8.9s	18.7s	95.2%
Creative Writing	1200	22.4s	45.6s	89.3%
Data Analysis	650	12.1s	24.8s	94.1%

Resource Utilization - AI Routing

Resource	Idle	Light Load (50 req)	Medium Load (200 req)	High Load (500 req)
CPU Usage	8%	35%	58%	79%
Memory (RSS)	180MB	320MB	485MB	720MB
Active Connections	5	25	85	180
Queue Length	0	3	12	28

Database Performance Benchmarks

PostgreSQL Performance

Operation	Avg Response Time	P95 Response Time	Throughput (ops/sec)
User Query (Indexed)	2.1ms	5.8ms	1,250
User Insert	8.5ms	15.2ms	285
User Update	6.8ms	12.4ms	320
Transaction Query	3.2ms	7.9ms	980
Transaction Insert	12.1ms	22.8ms	195
AI Request Log	5.4ms	11.7ms	420
Complex Join Query	18.5ms	45.2ms	125

Redis Cache Performance

Operation	Avg Response Time	P95 Response Time	Throughput (ops/sec)
GET (Small Value)	0.8ms	1.2ms	4,500
SET (Small Value)	0.9ms	1.4ms	4,200
GET (Large Value)	2.1ms	3.8ms	1,800
SET (Large Value)	2.5ms	4.2ms	1,650
Hash Operations	1.2ms	2.1ms	3,200
List Operations	1.4ms	2.5ms	2,800
Cache Hit Ratio	-	-	87.2%

System Resource Benchmarks

Memory Usage by Service

Service	Idle	Light Load	Medium Load	High Load	Peak Observed
Auth Service	120MB	180MB	285MB	384MB	420MB
AI Routing	180MB	320MB	485MB	720MB	850MB
User Management	95MB	145MB	220MB	310MB	365MB
Billing Service	85MB	125MB	185MB	260MB	295MB
Plugin Manager	110MB	165MB	240MB	335MB	380MB
Notification	75MB	115MB	170MB	240MB	275MB
Admin Dashboard	450MB	520MB	680MB	920MB	1.1GB
Employee Portal	420MB	480MB	620MB	850MB	980MB
PostgreSQL	256MB	384MB	512MB	768MB	1.2GB
Redis	64MB	96MB	128MB	192MB	256MB

CPU Usage by Service

Service	Idle	Light Load	Medium Load	High Load	Peak Observed
Auth Service	2%	15%	35%	65%	78%
AI Routing	5%	25%	45%	75%	89%
User Management	1%	8%	20%	40%	52%
Billing Service	1%	5%	15%	30%	42%
Plugin Manager	2%	12%	25%	50%	68%
Notification	1%	6%	18%	35%	48%
Nginx Gateway	1%	5%	12%	25%	32%

Network Performance

API Gateway Benchmarks

Metric	Value	Target	Status
Max Requests/sec	2,500	> 1,000	✅
Avg Response Time	45ms	< 100ms	✅
P95 Response Time	120ms	< 200ms	✅
P99 Response Time	280ms	< 500ms	✅
SSL Handshake Time	85ms	< 150ms	✅
Connection Pool Size	1,000	> 500	✅

WebSocket Performance (Notifications)

Metric	Value	Target	Status
Concurrent Connections	5,000	> 1,000	✅
Message Latency	15ms	< 50ms	✅
Messages/sec	10,000	> 5,000	✅
Connection Setup Time	120ms	< 200ms	✅
Memory per Connection	8KB	< 16KB	✅

End-to-End Scenarios

Complete User Journey Performance

Scenario	Steps	Total Time	P95 Time	Success Rate
New User Registration	4	2.8s	5.2s	98.5%
Login & First AI Request	3	15.2s	32.8s	94.8%
Dashboard Load & Navigation	5	1.8s	3.4s	99.2%
Plugin Installation	6	4.5s	8.9s	96.7%
Billing & Payment	7	3.2s	6.8s	97.9%
Settings Update	4	1.9s	3.7s	99.1%

Performance Scaling Characteristics

Horizontal Scaling Results

Instances	Max Users	Throughput (req/sec)	Avg Response Time	Resource Efficiency
1	1,000	156	650ms	100% (baseline)
2	1,800	285	580ms	91%
3	2,600	420	520ms	85%
4	3,200	540	480ms	81%

Database Scaling

Connection Pool	Max Concurrent	Query Response Time	CPU Usage	Memory Usage
20	200	3.2ms	25%	384MB
50	500	4.1ms	45%	512MB
100	1,000	5.8ms	68%	768MB
150	1,500	8.2ms	85%	1.1GB

Performance Regression Thresholds

Alert Thresholds

Metric	Warning	Critical	Action
Response Time (P95)	+20% from baseline	+50% from baseline	Scale up
Error Rate	> 2%	> 5%	Investigate immediately
Memory Usage	> 80% of limit	> 95% of limit	Add resources
CPU Usage	> 75% sustained	> 90% sustained	Scale horizontally
Database Connections	> 80% of pool	> 95% of pool	Increase pool size
Cache Hit Ratio	< 80%	< 70%	Optimize caching

Performance Targets by Service Level

Service Level	Response Time	Availability	Throughput	Error Rate
Premium	P95 < 1s	99.9%	500+ req/sec	< 0.1%
Standard	P95 < 2s	99.5%	200+ req/sec	< 0.5%
Basic	P95 < 5s	99.0%	100+ req/sec	< 1.0%

Optimization Recommendations

High Priority Optimizations

1. **Database Query Optimization**
   - Implement additional indexes for frequently queried columns
   - Use connection pooling with optimal pool sizes
   - Consider read replicas for read-heavy operations

2. **Caching Strategy Enhancement**
   - Increase cache TTL for stable data
   - Implement cache warming for critical data
   - Use Redis clustering for higher throughput

3. **AI Routing Optimization**
   - Implement request queuing for better throughput
   - Add circuit breakers for provider failures
   - Use connection pooling for AI provider APIs

Medium Priority Optimizations

1. **Memory Management**
   - Implement garbage collection tuning for Node.js services
   - Use streaming for large data processing
   - Optimize memory allocation patterns

2. **Network Optimization**
   - Enable HTTP/2 for better multiplexing
   - Implement request compression
   - Use CDN for static assets

3. **Code Optimization**
   - Profile and optimize CPU-intensive operations
   - Implement async processing for heavy operations
   - Use efficient data structures

Testing Methodology

Load Testing Process

1. **Environment Preparation**
   - Clean database with seed data
   - Warm up all services

   - Clear all caches
   - Reset all metrics

2. **Test Execution**
   - Gradual ramp-up over 2-5 minutes
   - Sustained load for 10-15 minutes
   - Gradual ramp-down over 2-5 minutes
   - Cool-down period for metric collection

3. **Data Collection**
   - Response time percentiles
   - Error rates by endpoint
   - Resource utilization metrics
   - Database performance metrics
   - Cache hit ratios

4. **Analysis and Reporting**
   - Compare against baseline metrics
   - Identify performance regressions
   - Generate optimization recommendations
   - Update capacity planning models

Continuous Performance Monitoring

- **Daily**: Automated performance smoke tests
- **Weekly**: Comprehensive load testing suite
- **Monthly**: Capacity planning review
- **Quarterly**: Full performance benchmark update

Conclusion

These benchmarks provide a comprehensive baseline for the AI Employee Platform's performance characteristics. Regular monitoring against these benchmarks ensures system reliability, identifies optimization opportunities, and supports capacity planning decisions.

For questions or additional performance testing requirements, contact the Platform Engineering team.

Document Version: 1.0
Last Updated: August 8, 2025
Next Review: November 8, 2025
Owner: Platform Engineering Team