 summary of all the improvements made to the auth service components:

1. Redis Service Improvements ✅

* Connection management with retry logic
* Error handling with custom RedisError
* Session management
* MFA secret storage
* Comprehensive logging
* Graceful shutdown
* Connection health checks

1. Message Queue Improvements ✅

* Robust connection handling
* Retry mechanism for failed messages
* Dead letter queue support
* Message correlation IDs
* Comprehensive logging
* Error handling with custom MessageQueueError
* Queue management (auth\_events, user\_events, notification\_events)
* Graceful shutdown

1. Error Handling Improvements ✅ Added specialized error types:

Code

CopyInsert

- BaseError

- AuthError

- ValidationError

- ForbiddenError

- NotFoundError

- ConflictError

- RateLimitError

- RedisError

- MessageQueueError

- DatabaseError

- ServiceUnavailableError

1. Logging Improvements ✅ Added specialized log files:

Code

CopyInsert

- error.log (errors only)

- combined.log (all logs)

- security.log (security events)

- exceptions.log (uncaught exceptions)

- rejections.log (unhandled rejections)

Features:

* Request logging middleware
* Error logging middleware
* Custom log formatting
* Environment-based logging
* Log rotation
* Stack trace preservation
* Request context tracking

1. Security Features ✅

* Token validation
* Rate limiting
* MFA support
* Session management
* Security event logging
* Input validation
* Error sanitization

1. Integration Points ✅ Services properly integrated with:

* Redis for session/token management
* RabbitMQ for event messaging
* Logging system
* Error handling system

1. Best Practices Implemented ✅

* Singleton pattern for services
* Graceful shutdown handlers
* Connection pooling
* Error standardization
* Comprehensive logging
* Security hardening
* Performance optimization

1. Development Features ✅

* Debug logging in development
* Pretty console output
* Stack traces in development
* Request timing
* Correlation IDs