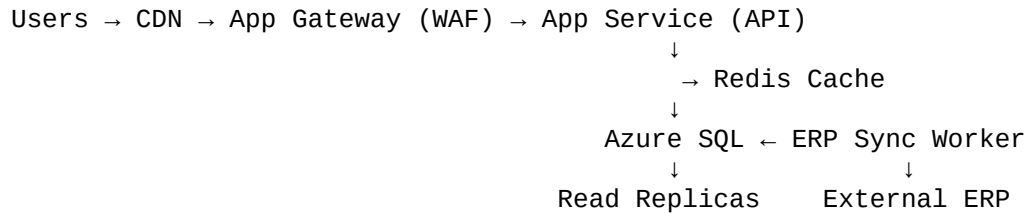


# Architecture Overview

## High-Level Design



## Key Components

### 1. API Layer (ASP.NET Core Web API)

- Auto-scaling: 2-20 instances based on CPU/memory/HTTP queue
- Endpoints: Product detail + paginated listing
- Redis caching: 85%+ hit ratio
- Rate limiting: 100 req/min per client

### 2. Data Layer (Azure SQL Business Critical)

- String-based IDs (e.g., "el-01", "cat-01")
- Proper indexes on CategoryId, Price
- Read replicas for reporting

### 3. Cache Layer (Redis)

- Product cache: 15-minute TTL
- Listing cache: 5-minute TTL
- Selective invalidation after ERP sync

### 4. Integration Layer (Background Worker)

- Scheduled sync every 15 minutes
- Only updates changed products (efficient)
- Full audit trail in SyncLogs

### 5. Monitoring (Application Insights)

- Request telemetry
- Performance metrics
- Custom events for sync operations

## Scalability Strategy

### Horizontal Scaling:

- CPU > 70% → scale out

- Memory > 80% → scale out
- HTTP queue > 100 → scale out

## ERP Integration Flow

1. **Fetch Categories** from ERP → Sync to database (create/update)
2. **Fetch Products** from ERP → Calculate hash for each
3. **Compare** hash with stored hash
4. **Update** only if hash changed (delta sync)
5. **Invalidate** relevant caches
6. **Log** operation to SyncLogs

## Security

- HTTPS only
- Rate limiting
- CORS policies
- SQL injection prevention (parameterized queries)
- Secrets in Azure Key Vault (production)

## Deployment

### Blue-Green Strategy:

1. Deploy to "Green" staging slot
2. Warm up
3. Run smoke tests
4. Swap slots (instant cutover)
5. Keep "Blue" as rollback target

---

**Design Patterns:** Repository, Cache-Aside, Background Worker, Clean Architecture