

# Joao Otavio Manieri - 12021BSI263

25/06/2025

## 1. Introdução

No contexto da disciplina de Sistemas Distribuídos, o presente relatório tem como objetivo aplicar os conceitos estudados em sala de aula por meio da reengenharia de um sistema originalmente construído de forma não distribuída, ou seja, **stand-alone**. A proposta consiste em selecionar um software de uso real, empregado em ambiente comunitário, e projetar uma nova versão distribuída que explore os benefícios proporcionados por essa arquitetura.

Para isso, foi escolhido o **Sistema de Controle de Acesso da Academia Império Fit**, localizada em São Carlos – SP (minha cidade natal). Embora não tenha participado do desenvolvimento do software, possuo contato próximo com a gerência e com alguns instrutores (frequento o local regularmente quando estou lá), o que garante conhecimento prático sobre a operação cotidiana da ferramenta.

O sistema atual é simples: executa-se em um único computador instalado na recepção da academia e serve exclusivamente para **registrar a entrada e a saída de alunos** por meio de liberação da recepcionista. Ele gera relatórios impressos no fim do dia para controle interno, mas não possui integração em rede, tampouco disponibiliza acompanhamento em tempo real para alunos ou treinadores. Trata-se, portanto, de uma aplicação **stand-alone**, cuja funcionalidade restringe-se ao ponto de atendimento local.

A escolha do sistema se justifica por sua simplicidade arquitetural e pelo potencial de expansão para uma solução distribuída que trará benefícios diretos à equipe administrativa, aos profissionais de educação física e, principalmente, aos alunos. Nas seções seguintes, apresentaremos: (i) uma descrição detalhada do sistema atual, suas funcionalidades e limitações; e (ii) uma proposta de reengenharia baseada nos princípios estudados em sala, com foco em modularidade, escalabilidade, transparência de distribuição e colaboração.

---

## 2. Descrição do Sistema Atual (Stand-alone)

O **Sistema de Controle de Acesso da Academia Império Fit** foi desenvolvido para registrar frequência diária dos alunos. Executa-se como uma aplicação desktop de execução local (Windows), operando em apenas um computador.

## 2.1 Finalidade e Contexto de Uso

A principal finalidade do sistema é:

- Registrar automaticamente horários de **entrada** de cada aluno.
- Permitir à recepção verificar rapidamente se a mensalidade está ativa (campo “status” no cadastro).
- Gerar **relatório impresso** no término do expediente com contagem de acessos.

O público atendido são alunos de diferentes planos (diária, mensal e anual) e os próprios funcionários (instrutores, recepcionistas e pessoal de limpeza). O software foi adotado pela academia para substituir o antigo livro-ponto manual, conferindo maior confiabilidade aos registros.

## 2.2 Funcionalidades Principais

1. **Cadastro de Alunos**
  - Nome, CPF, plano, data de vencimento da mensalidade, status (ativo/inadimplente).
2. **Registro de Acesso**
  - Anuncio previa a recepcionista.
  - Gravação de data/hora de entrada.
3. **Relatório Diário**
  - Impressão de lista com nome do aluno, horário de entrada.
4. **Alerta de Inadimplência**
  - Pop-up na tela quando o aluno se encontra com mensalidade vencida.
5. **Interface Gráfica Local**
  - Painel simples com botões: “Cadastrar”, “Consultar”, “Imprimir Relatório”, “Configurações”.

**Limitações notáveis:** inexistência de backup automático, ausência de acesso remoto, impossibilidade de vários terminais simultâneos e falta de integração com aplicativos de treino ou plataformas de gestão financeira.

## 2.3 Organização Interna (Módulos Conceituais)

Módulo	Descrição
Interface Gráfica (UI)	Formulários para cadastro/consulta, botões de operação
Módulo de Leitura	Driver de comunicação com leitor USB

<b>Registro de Sessão</b>	Mantém dados de entrada/saída na memória e grava em arquivo local
<b>Gerador de Relatório</b>	Compila dados do dia em arquivo PDF ou impressora
<b>Configurações Locais</b>	Define parâmetros como horário de fechamento, impressora padrão, etc.

## 2.4 Forma de Operação

- O recepcionista liga o computador, abre o sistema, faz login local.
- Cada aluno aproxima e solicita liberacao dando seu nome ou cpf.
- O sistema grava a hora de entrada em um arquivo local.
- Ao final do expediente, gera-se o relatório diário em PDF e imprime-se uma via.
- Não há armazenamento em nuvem, tampouco replicação; uma falha no HD resulta em perda de histórico.

### Principais limitações:

- **Nenhuma visibilidade em tempo real** para alunos ou professores sobre frequência.
- **Falta de integração** com aplicativos de treino, financeiro (boleto/mensalidade) ou CRM.
- **Dependência de um único ponto de falha** (computador da recepção).
- **Ausência de histórico longo** (archiving manual em pen-drive).
- **Incapacidade de gerar relatórios analíticos** (ex.: horários de pico, taxa de churn).

---

## 3. Proposta de Reengenharia para Sistema Distribuído

### 3.1 Benefícios da Arquitetura Distribuída

1. **Acompanhamento de frequência em múltiplos dispositivos**
  - Aluno poderá checar frequência própria via aplicativo ou portal.
  - Instrutor visualiza lista em tempo real para chamada e controle de treinos.
2. **Integração com financeiro**

- Atualização automática de status de pagamento (integração com ERP ou gateway).
- 3. **Escalabilidade horizontal**
  - Adição de totens de autoatendimento em outros pontos da academia (área de peso, estúdio de ginástica).
- 4. **Alta disponibilidade**
  - Replicação síncrona em servidor na nuvem; falha local não compromete operação.
- 5. **Relatórios analíticos**
  - Dashboard de horários de pico, retenção de alunos, uso de planos.
- 6. **Notificações pró-ativas**
  - Push para aluno com lembrete de frequência mínima ou expiração de plano.
- 7. **Transparência de distribuição**
  - Para o usuário final, a experiência de “bater carteirinha” permanece a mesma, porém o backend distribui carga e dados.

## 3.2 Arquitetura Distribuída Proposta

### 3.2.1 Componentes

Componente	Função
API Gateway	Roteia requisições dos leitores, totens e apps
Serviço de Acesso	Lógica de entrada/saída, verificação de status de aluno
Serviço Financeiro	Consulta/atualiza mensalidade via integração com ERP
Serviço de Relatórios	Geração de dashboards e KPIs
Banco de Dados (PostgreSQL)	Persistência de alunos, acessos, pagamentos
Fila de Mensagens (RabbitMQ)	Evento “acesso registrado” → envio de notificações / atualização de relatórios

<b>Aplicativo Móvel / PWA</b>	Portal do aluno (frequência, histórico, avisos)
<b>Totens de Autoatendimento</b>	Leitores adicionais distribuídos pela academia
<b>Painel de Instrutor</b>	Interface web em tablets para controle de turmas

### 3.2.2 Fluxo Típico

1. Aluno aproxima carteirinha no leitor (dispositivo IoT ou leitor USB).
2. Leitor envia requisição ao **API Gateway** → **Serviço de Acesso**.
3. Serviço verifica status na base e na **Fila** despacha evento "Acesso OK".
4. **Serviço de Relatórios** consome evento, atualiza dashboards em tempo real.
5. **Aplicativo do aluno** e **Painel do instrutor** recebem update via WebSocket.

### 3.2.3 Tecnologias

- **Node.js + NestJS** (Serviços)
- **React / React Native** (Apps e painéis)
- **PostgreSQL** (dados relacionais) + **Redis** (caches)
- **RabbitMQ** (mensageria)
- **Docker Compose / Kubernetes** (orquestração)
- **Prometheus + Grafana** (monitoramento)
- **JWT + OAuth2** (autenticação)

---

## 3.4 Interação do Usuário com o Sistema

- **Aluno**
  - Continua passando carteirinha no leitor.
  - Visualiza frequência, vencimento do plano, mensagens motivacionais no app.
- **Instrutor**
  - Abre o painel em tablet; vê lista de alunos presentes, histórico de faltas.
  - Marca presença em aulas coletivas; envia alerta para faltosos recorrentes.
- **Recepção**
  - Usa dashboard administrativo para cadastrar novos alunos, imprimir etiquetas de código de barras.
  - Consulta em tempo real inadimplentes e bloqueia acesso automaticamente.

---

## 3.5 Detalhes Técnicos de Implementação

Área	Detalhe
<b>Desenvolvimento</b>	Monorepo Git + CI/CD GitHub Actions
<b>Deploy</b>	Cluster Kubernetes (K3s on-prem + backup na AWS)
<b>Segurança</b>	TLS em toda a malha de serviços, token de curta duração, refresh tokens
<b>Observabilidade</b>	Logs estruturados (Elastic Search), traces (Jaeger), métricas (Prometheus)
<b>Alta disponibilidade</b>	2 réplicas por serviço crítico; Postgres em modo streaming replication
<b>Backup</b>	Dump diário + snapshot S3 + retenção de 30 dias

## 4. Considerações Finais

Este relatório apresentou a reengenharia do **Sistema de Controle de Acesso da Academia Império Fit**, originalmente executado de forma local e isolada, transformando-o em uma solução distribuída capaz de:

- Acompanhar frequência em tempo real em múltiplos dispositivos.
- Integrar dados de pagamento e inadimplência automaticamente.
- Prover dashboards gerenciais para tomada de decisão.
- Escalar facilmente a partir de novos pontos de leitura sem refatorar toda a aplicação.

Ao alinhar-se aos princípios de sistemas distribuídos — escalabilidade, transparência, abertura e tolerância a falhas — a nova arquitetura demonstra como pequenas academias podem modernizar seus processos, oferecendo valor agregado a alunos e funcionários sem perder a simplicidade da experiência diária.