

Proposta de Evolução do Projeto

Escritório EBAC

Esta proposta visa apresentar melhorias evolutivas para o software atual, garantindo maior segurança, conectividade, confiabilidade e capacidade de análise dos dados do sistema. As evoluções propostas incluem estratégias de backup, conectividade aprimorada, tratativa de erros e monitoria avançada de usuários.

1. Implementação de Backup

Proposta:

- Automatizar o processo de backup dos dados de usuários.
- Realizar backups diários, semanais e mensais para diferentes níveis de restauração.
- Manter versões históricas dos arquivos para recuperação em casos de corrupção ou exclusão acidental.
- Armazenar os backups em locais externos, como serviços em nuvem (ex.: Google Drive, OneDrive) ou servidores de backup dedicados.

Importância:

- Garante a **segurança dos dados**, protegendo contra perdas acidentais ou falhas no sistema.
- Facilita a **recuperação rápida** em cenários de desastre, minimizando o tempo de inatividade do sistema.

2. Melhoria da Conectividade

Proposta:

- Migrar o armazenamento de dados para um **banco de dados relacional**, como MySQL ou PostgreSQL, que permita o acesso remoto e o armazenamento centralizado.

- Integrar o sistema a uma **API RESTful**, possibilitando a comunicação com o banco de dados via rede.
- Implementar um **servidor web** para acessar o sistema de qualquer dispositivo conectado à rede local ou à internet.

Importância:

- Permite uma **maior flexibilidade de acesso**, inclusive em dispositivos móveis ou em diferentes locais físicos.
- Facilita o **processo de backup**, pois os dados estarão centralizados em um único servidor ou serviço em nuvem.

3. Tratativa de Erros e Garantia de Qualidade

Proposta:

- Estabelecer uma **equipe de testes** dedicada, que realizará testes unitários, de integração e de usabilidade.
- Introduzir o uso de ferramentas automatizadas de testes (ex.: Selenium, JUnit) para identificar falhas no código.
- Desenvolver um **mecanismo de logs e alertas**, onde possíveis erros sejam registrados e notificados à equipe de desenvolvimento.

Importância:

- Reduz a quantidade de **erros em produção**, proporcionando uma experiência mais fluida e segura para o usuário final.
- Permite uma resposta rápida e assertiva da equipe de desenvolvimento, minimizando os impactos negativos ao sistema.

4. Monitoria de Usuários e Dashboards

Proposta:

- Criar uma interface de **dashboard interativa** para gerentes e administradores do sistema, utilizando bibliotecas como **Recharts** ou **Chart.js**.
- Exibir métricas importantes, como:
 - Número total de usuários cadastrados.

- Número de usuários excluídos por período (mensal, semanal, diário).
- Frequência de consultas realizadas ao sistema, com possibilidade de **filtro por datas**.
- Relatórios exportáveis em **PDF** ou **Excel** para análises externas.
- Implementar gráficos que forneçam uma **visão clara e objetiva** do uso do sistema ao longo do tempo.

Importância:

- Permite que a equipe administrativa tome **decisões estratégicas**, baseadas em dados reais.
- Facilita a identificação de padrões, como picos de consultas ou momentos de maior exclusão de usuários.
- Garante um maior **controle operacional**, prevenindo abusos ou usos inadequados do sistema.

Essas evoluções proporcionam não apenas uma maior **segurança e estabilidade** ao sistema, mas também ampliam suas funcionalidades e sua capacidade de análise e monitoramento. Ao adotar estas melhorias, o escritório **EBAC** poderá gerenciar seus dados de maneira mais eficiente, segura e estratégica, possibilitando um crescimento sustentável e adaptável às futuras necessidades do negócio.