



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA



Programação Avançada em Java – AoR

11ª edição

Acertar o Rumo 2023/2024

Relatório Projeto

ForgeXperimental Projects

Alexandre Silva (2005122565)

Inês Cunha (2008006278)

Orientador: João Lourenço

Julho 2024

Índice

Introdução

Este projeto é desenvolvido no contexto da disciplina de Programação Avançada em Java, pertencente à 11ª edição do curso Acertar o Rumo, como o trabalho final da sua fase letiva.

A ideia para este projeto foi apresentada pela empresa Critical Software, que identifica a necessidade imperativa das empresas tecnológicas de permanecerem na vanguarda da inovação. A empresa reconhece que a verdadeira força motriz do progresso reside muitas vezes dentro da própria organização, nos seus colaboradores, cujas experiências e conhecimentos proporcionam novas perspetivas e *insights* valiosos.

Neste cenário, o objetivo é criar um "Laboratório de Inovação", um espaço dedicado à partilha e desenvolvimento criativo. Este laboratório será um ponto de encontro para colaboradores identificarem interesses comuns, desenvolverem novos projetos e acompanharem-nos desde a conceção inicial até à sua implementação. O ambiente deve fomentar a colaboração, a criatividade e a inovação contínua.

A aplicação a ser desenvolvida, baseada na web, deve atender aos requisitos especificados no enunciado do projeto, incorporando diversas funcionalidades essenciais. As principais secções da aplicação incluem:

1. **Secção Pessoal de Utilizador:** Esta secção deve fornecer informações gerais de perfil, registo de habilidades e interesses, bem como uma listagem dos projetos nos quais o utilizador está ou esteve envolvido. Além disso, deve permitir a troca de mensagens pessoais entre utilizadores.
2. **Gestão de Projetos:** Esta funcionalidade é crucial para a criação e edição de projetos, permitindo a associação de utilizadores, laboratórios e habilidades. Também deve suportar o registo de recursos necessários ou atribuídos a cada projeto e a gestão do estado do projeto, que pode variar entre Planeamento, Pronto, Aprovado, Em Progresso, Cancelado e Concluído.
3. **Gestão de Stocks de Componentes/Recursos:** Deve possibilitar a criação e edição de componentes, a pesquisa de componentes, a atribuição de recursos a projetos e a consulta de estatísticas de estoque para uma gestão eficiente dos recursos disponíveis.

No presente relatório serão descritos, detalhadamente, os requisitos, a arquitetura da aplicação, o modelo de dados adotado, os requisitos e pressupostos considerados para o desenvolvimento do sistema, bem como os testes realizados para garantir a sua funcionalidade e desempenho.

Requisitos

Ao longo do desenvolvimento do projeto, todos os requisitos definidos foram alcançados. O sistema implementado cumpre as seguintes especificações:

Requisitos Funcionais:

Gestão de Projetos:

- Criação, edição e exclusão de projetos.
- Associação de utilizadores aos projetos.
- Gestão dos estados dos projetos (Planning, Ready, Approved, In Progress, Cancelled, Finished).

Gestão de Utilizadores:

- Registo e autenticação de utilizadores.
- Atualização de perfis com informações de skills e interesses.
- Listagem dos projetos em que os utilizadores participam ou participaram.

Gestão de Recursos:

- Criação e edição de componentes.
- Pesquisa e atribuição de componentes aos projetos.
- Estatísticas detalhadas dos stocks de recursos.

Requisitos Não Funcionais:

Segurança:

- Implementação de medidas robustas para proteger os dados dos utilizadores.
- Assegurar que apenas utilizadores autorizados possam aceder e modificar informações sensíveis.

Usabilidade:

- Interface intuitiva e fácil de usar, garantindo uma experiência de utilizador positiva.

Desempenho:

- Otimização para respostas rápidas às solicitações dos utilizadores.

Compatibilidade:

- Funcionamento com os principais navegadores web, incluindo Chrome, Firefox, Safari e Edge.

Testes Realizados:

Testes Unitários:

- Os componentes do sistema foram testados utilizando JUnit para garantir o seu correto funcionamento.

Testes de Integração:

- Verificação da interação entre diferentes módulos do sistema utilizando Jest para assegurar que trabalham em conjunto de forma harmoniosa.

Testes de Sistema:

- Avaliação do sistema como um todo para confirmar que todos os requisitos funcionais e não funcionais foram atendidos.

Testes de Segurança:

- Análises realizadas com o *Postman* para identificar e corrigir vulnerabilidades, assegurando a proteção dos dados dos utilizadores.

Testes de Usabilidade:

- Sessões de teste para validar a interface e a experiência do utilizador.

Testes de Desempenho:

- Testes de carga e stress para garantir que o sistema mantém um desempenho adequado mesmo sob condições extremas.

Estas funcionalidades e especificações foram cuidadosamente implementadas e testadas, assegurando que o sistema atende às necessidades de organização e gestão dos projetos desenvolvidos no Laboratório de Inovação, proporcionando uma plataforma robusta, segura e eficiente para os utilizadores.

Arquitetura da Aplicação

A arquitetura da aplicação é projetada para suportar um sistema web baseado em Java no *backend* e *React* no *frontend*, utilizando MySQL como base de dados. Abaixo estão os principais componentes e suas interações:

Frontend: Desenvolvido em *React*, utilizando componentes específicos para *React* e integrando elementos de *frameworks* como *Bootstrap* para uma interface de utilizador modular e responsiva. A comunicação com o *backend* é realizada através de requisições HTTP para endpoints RESTful, seguindo os princípios de design REST para operações CRUD (Create, Read, Update, Delete).

Backend: Implementado em Java com o servidor de aplicação *WildFly*. Organizado em packages para melhor modularização e gerenciamento de código: **Entities:** representação das entidades do domínio que são mapeadas para tabelas no banco de dados. **DTOs (Data Transfer Objects):** utilizados para transferência eficiente de dados entre o *frontend* e o *backend*. **DAOs (Data Access Objects):** responsáveis por aceder e manipular os dados no banco de dados, seguindo o padrão DAO para abstrair as operações de persistência. **Serviços:** implementam a lógica de negócio da aplicação e fazem a interface entre os DAOs e os controladores REST. **WebSockets:** Implementação para comunicação bidirecional em tempo real entre o cliente (*frontend*) e o servidor (*backend*). **Beans:** componentes gerenciados pelo container EJB para serviços empresariais. **Enums:** definições de tipos enumerados para representar valores fixos. **Gestão de Envio de Emails:** funcionalidade para envio automatizado de emails, integrado à lógica de negócio.

Comunicação Frontend-Backend: Utilização de serviços RESTful para troca de dados, com *endpoints* bem definidos para cada recurso da aplicação. Implementação de DTOs específicos para cada operação, garantindo que apenas os dados necessários sejam transferidos entre as camadas.

Segurança: Validação de dados implementada em ambos os lados (*frontend* e *backend*) para evitar inconsistências e proteger contra ataques como injeção de SQL e XSS. Autenticação baseada em tokens (JWT - JSON Web Tokens), com gestão de sessões e os papéis dos utilizadores para controlar o acesso aos recursos da aplicação. Segurança de Dados: uso de HTTPS para criptografar a comunicação entre cliente e servidor, com medidas adicionais para prevenir vulnerabilidades como SQL injection e XSS.

Base de Dados: Utilização do MySQL para persistência de dados. Configuração adequada de índices e chaves estrangeiras para garantir integridade e desempenho das operações da base de dados. Senhas armazenadas de forma segura, utilizando o algoritmo MD5 para proteção contra acessos não autorizados.

Testes: Implementação de testes unitários com JUnit para testar componentes individuais do *backend*. Testes de integração com Jest para validar a integração correta entre o *frontend* e o *backend*. Utilização do Postman para testes de API, verificando se todos os *endpoints* respondem corretamente às requisições.

Ambiente de Desenvolvimento: Utilização do IntelliJ IDEA para desenvolvimento do *backend* em Java. Visual Studio Code como IDE para o desenvolvimento do *frontend* em React. MySQL Workbench para modelagem e administração da base de dados MySQL.

Modelo de Datos

Mockups

Os *mockups* desempenham um papel crucial no desenvolvimento de qualquer projeto de software, servindo como um mapa visual que orienta tanto o design quanto o desenvolvimento. Para o projeto em questão, elaboramos *mockups* detalhados utilizando a plataforma *Figma*, com o objetivo de capturar não apenas a estética visual, mas também a funcionalidade essencial da aplicação.

O processo de criação dos *mockups* no *Figma* foi guiado pelo desejo de apresentar uma representação tangível e intuitiva do produto final. Partindo de requisitos iniciais definidos, exploramos diversas abordagens visuais e funcionais, equilibrando criatividade com praticidade. A liberdade concedida para ajustar e expandir os *mockups* permitiu-nos explorar ideias adicionais que enriqueceram a experiência do utilizador e a funcionalidade do sistema.

Ao criar os *mockups*, focamos não apenas na estética visual, mas também na usabilidade e na experiência do utilizador. Cada elemento foi cuidadosamente projetado para refletir as necessidades identificadas durante a fase de planejamento, garantindo que a interface proporcionasse uma interação fluida e intuitiva. A escolha de componentes e layouts no *Figma* permitiu-nos simular com alguma precisão como os utilizadores interagiriam com a aplicação em diferentes cenários.

Além de replicar as funcionalidades inicialmente propostas, exploramos novas ideias e recursos que poderiam potencializar ainda mais o valor da aplicação. Isso incluiu melhorias na navegação, integração de novas ferramentas de gestão de dados e otimizações na interface de utilizador para melhorar a acessibilidade e a eficiência operacional.

Tecnologias Utilizadas

As tecnologias utilizadas neste projeto foram selecionadas com base nas ferramentas e *frameworks* ensinados durante o curso, visando aplicar os conhecimentos adquiridos de forma prática no desenvolvimento da aplicação. Cada tecnologia foi aplicada para atender aos requisitos funcionais e não funcionais estabelecidos, garantindo eficiência, segurança e escalabilidade no projeto.

Backend:

- **Java:** Utilizado como linguagem de programação principal para o desenvolvimento da lógica dos serviços da aplicação, alinhando-se com os conceitos aprendidos nas disciplinas Introdução à Programação em Java e Programação Avançada em Java.
- **WildFly:** Servidor de aplicação Java EE utilizado para hospedar e executar o backend, aplicando os princípios de servidores de aplicação estudados durante o curso.

Frontend:

- **React:** Biblioteca JavaScript ensinada durante o curso, utilizada para construir a interface do utilizador dinâmica e responsiva da aplicação web.
- **Bootstrap:** Framework CSS ensinado para criação de um design responsivo e componentes reutilizáveis, seguindo as melhores práticas de desenvolvimento frontend.

Base de Dados:

- **MySQL:** Sistema de gerenciamento de banco de dados relacional ensinado durante o curso, escolhido para armazenar e gerenciar os dados da aplicação de maneira segura e eficiente.
- **MySQL Workbench:** Ferramenta de modelagem e administração de banco de dados também aprendida no curso, utilizada para projetar e gerenciar o esquema do banco de dados MySQL.

Ferramentas de Desenvolvimento:

- **IntelliJ IDEA:** utilizado para desenvolvimento backend em Java, oferecendo suporte avançado para depuração e desenvolvimento.
- **Visual Studio Code:** Editor de código ensinado e utilizado para desenvolvimento frontend com React, conhecido por sua extensibilidade e integração com ferramentas externas.

Outras Tecnologias:

- **JSON Web Tokens (JWT):** Utilizado para autenticação e autorização entre o frontend e o backend da aplicação.
- **JUnit e Jest:** Frameworks de teste ensinados para realização de testes unitários e de integração no backend e frontend, garantindo a qualidade e estabilidade do código.

As tecnologias utilizadas neste projeto foram escolhidas com base nos conhecimentos adquiridos durante o curso, refletindo o alinhamento com as práticas e ferramentas recomendadas para o desenvolvimento de aplicações web modernas e escaláveis. A aplicação

dessas tecnologias permitiu alcançar os objetivos estabelecidos com sucesso, demonstrando a aplicação prática dos conceitos aprendidos e preparando para desafios futuros no campo da tecnologia da informação.

Requisitos e Tomadas de Decisão

O objetivo central deste projeto foi desenvolver uma aplicação web capaz de organizar e gerenciar eficientemente os projetos do Laboratório de Inovação. Os principais requisitos incluíam:

Secção Pessoal de Utilizador: Registo de um utilizador; perfil do utilizador com informações gerais; registo de habilidades e interesses; listagem de projetos atuais e anteriores do utilizador; mensagens pessoais enviadas e recebidas.

Gestão de Projetos: Criação e edição de projetos; associação de utilizadores aos projetos (criador, convidado e candidato); vinculação de projetos a laboratórios; registo de recursos necessários e atribuídos a cada projeto, criação e edição de tarefas relacionadas aos projetos; controle do estado do projeto.

Resumo estatístico: Apresentação de um resumo estatístico dos projetos em curso, apresentado o número total de projetos, o número médio de membros por projeto, o tempo médio de execução de um projeto, o número e percentagem de cada projeto no global e em cada laboratório.

Gestão de Stocks de Componentes/Recursos: criação e edição de componentes; pesquisa e atribuição de componentes a projetos; consulta de estatísticas dos recursos específicos.

Todos os requisitos estabelecidos foram completamente atendidos durante o desenvolvimento da aplicação. Cada funcionalidade foi implementada com foco nas necessidades específicas do Laboratório de Inovação, garantindo uma plataforma eficiente para colaboração e gerenciamento de projetos.

A dedicação em atender a todos os requisitos especificados, além daqueles que surgiram durante as sessões de perguntas e respostas e da reunião com o cliente, influenciou algumas escolhas estratégicas que são pertinentes mencionar aqui:

Criação de um super Admin. Este utilizador foi concebido para atender a necessidades avançadas de gestão e controle, garantindo maior flexibilidade e segurança na administração dos projetos. Foi criada uma página de configurações da aplicação que apenas o admin tem acesso, na qual pode definir o tempo de inatividade da aplicação, assegurando que as sessões dos utilizadores sejam gerenciadas de forma segura e eficiente, gerir o número máximo de membros nos projetos, garantindo que a estrutura dos projetos seja mantida conforme as diretrizes do laboratório e por fim, pode despromover gestores da aplicação, assegurando que a liderança da aplicação esteja sempre alinhada com os objetivos organizacionais e com o desempenho esperado. A inclusão de um super Admin proporciona um nível adicional de controle e administração sobre a aplicação, permitindo ajustes finos conforme as necessidades específicas do Laboratório de Inovação. Este papel garante que a aplicação não só atende às expectativas atuais, mas também é flexível o suficiente para se adaptar a futuros requisitos e mudanças operacionais.

Apresentação do Produto Final

Nesta seção, oferecemos uma visão abrangente do produto final da nossa aplicação. Utilizando capturas de tela e descrições detalhadas, ilustramos as principais funcionalidades e a interface do usuário. Isso permite uma compreensão clara de como a aplicação funciona e quais são suas características essenciais. Além disso, detalhamos o fluxo de uso, desde o login até a administração completa dos projetos, destacando a experiência do utilizador em cada etapa. Na página inicial

apresentamos de uma forma dinâmica e agradável a aplicação e as suas funcionalidades (figura 1). Composta por varias imagens interativas e um *header* com os botões de Sign Up, caso o utilizador não tenha conta e pretenda criar uma, Login, para entrar na sua conta e os botões PT e EN que se mantêm presentes no header em todas as páginas da aplicação, permitindo assim ao utilizador

mudar a linguagem da aplicação sempre que pretenda. Carregando no botão de registo, abre na página o modal de registo no qual o utilizador tem de preencher com o seu email e password (figura 2), e quando submete é enviado um email para o seu email com um link que carregando, é direcionado para a página de registo do utilizador (figura 3). Nesta página são

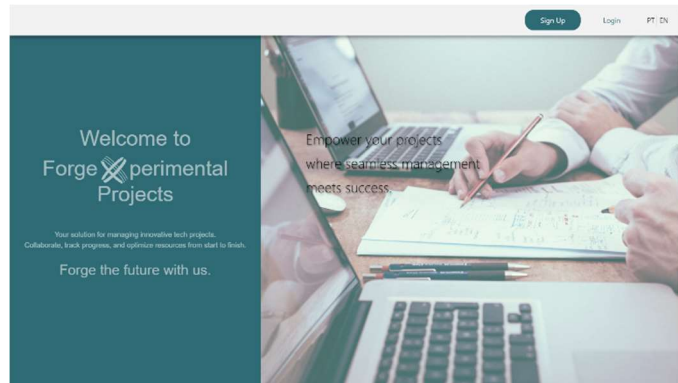


Figura 1

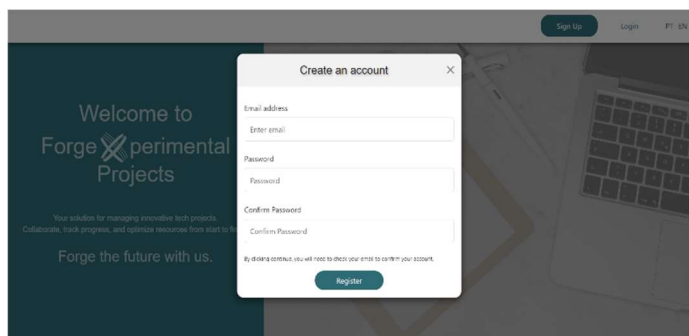


Figura 2

exibidos todos os campos de preenchimento dos dados do utilizador necessários para criar a sua conta: primeiro nome, apelido, laboratório, nickname, imagem e biografia, sendo que o primeiro nome, o apelido e laboratório são campos de preenchimento obrigatórios, dando erro com essa indicação e não permitindo o utilizador registar-se sem os preencher.

Após o registo efetuado com sucesso o utilizador é encaminhado (neste caso já com a sua conta) para a página principal da aplicação.

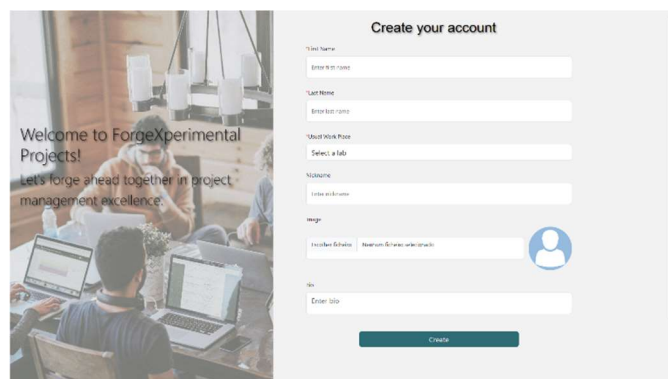


Figura 3

Conclusão

Consideramos que concluímos este projeto com sucesso, alcançando todos os objetivos estabelecidos e superando as dificuldades que foram surgindo. Estamos satisfeitos com o resultado, onde todas as funcionalidades propostas no enunciado foram implementadas com eficiência e robustez. A aplicação desenvolvida atende de forma abrangente às necessidades de organização e gestão dos projetos no Laboratório de Inovação, proporcionando uma interface intuitiva e agradável para os utilizadores.

Durante todo o processo, enfrentamos desafios substanciais que nos incentivaram a explorar e aplicar os conhecimentos adquiridos ao longo do curso. O apoio e orientação dos nossos professores ao longo do ano letivo foram fundamentais, fornecendo-nos não apenas conhecimento teórico, mas também orientações práticas que ajudaram no desenvolvimento do projeto. Em particular, o nosso orientador desempenhou um papel crucial ao fornecer *insights* valiosos e orientação experiente, garantindo que mantivéssemos o foco nos objetivos e superássemos obstáculos de forma eficaz.

A colaboração estreita entre os membros do grupo de trabalho foi essencial para o sucesso do projeto. Distribuímos responsabilidades de maneira equitativa, aproveitando as nossas competências individuais para fortalecer cada aspeto da aplicação. Esta abordagem não só facilitou a superação de desafios técnicos, mas também promoveu um ambiente de aprendizagem mútua e crescimento académico.

Em suma, este projeto não apenas nos desafiou como programadores, mas também reforçou a importância do trabalho em grupo e da aprendizagem contínua. Agradecemos sinceramente aos nossos professores pelo apoio inestimável ao longo deste percurso, que nos preparou não apenas para os desafios deste projeto, mas também para as exigências futuras da nossa carreira profissional.