

Relatório Técnico - Especificações de Projeto

Campus Monitoring

Disciplina: IES - Introdução à Engenharia de Software

Data: Aveiro, 14 de Novembro de 2019

Alunos: 88969: Alexandre Lopes
89198: André Amarante
89323: Edgar Morais
51531: Gabriela Santos

Resumo do Projeto: Este projeto consiste num sistema de monitorização de espaços alimentares universitários, através da recolha de informação em tempo real de parâmetros como a temperatura, humidade, pressão atmosférica e níveis de CO₂, providenciando uma forma de assegurar um maior controlo por gestores e trabalhadores com vista a tornar o espaço adequado às suas funções, bem como uma forma de fornecer informações acerca de eventuais catástrofes com o objetivo de desencadear respostas rápidas e eficazes às mesmas.

Índice:

[1 Introdução](#)

[2 Conceptualização do Produto](#)

[Visão Global](#)

[Personas](#)

[Cenários Principais](#)

[3 Architecture notebook](#)

[Key requirements and constraints](#)

[Architectural view](#)

[Module interactions](#)

[4 Information perspective](#)

[5 References and resources](#)

1 Introdução

Este projeto prático tem como foco a simulação do ambiente de desenvolvimento de uma aplicação web em contexto empresarial. A integração com a disciplina de IES é assente na exposição aos alunos de princípios de gestão de projetos, paradigmas de arquitetura e ferramentas tecnológicas largamente usadas na indústria de software. Após esta etapa de exposição inicial, os alunos irão adquirir competências fundamentais para o dia-a-dia do trabalho de um engenheiro de software, tanto num âmbito organizacional, como de análise de soluções.

Relativamente ao âmbito organizacional, a equipa propõe-se a implementar uma seleção de princípios Agile para gestão do projeto. O desenvolvimento do mesmo será organizado em iterações de 2 semanas, cujo progresso será avaliado mediante a capacidade da equipa concluir as tarefas selecionadas do backlog, segundo o paradigma Scrum. A implementação de novas funcionalidades ocorrerá de modo incremental. Iremos utilizar também algumas práticas comuns de Extreme Programming, como a concepção de funcionalidades da aplicação na perspetiva de user stories geridas por um Product Owner. A decomposição das funcionalidades em tarefas mais simples e a hierarquização da prioridade das mesmas serão essenciais para atingir um bom fluxo de trabalho entre equipa.

Adicionalmente, pretendemos utilizar uma arquitetura *multilayered* como referência e analisar qual a melhor forma de obter uma interação ótima entre componentes da nossa aplicação. Nesse sentido, temos também como objetivo fazer escolhas informadas de quais as ferramentas tecnológicas (*frameworks*, *template engines*, bases de dados e *message brokers*) que nos permitirão tirar partido das suas características para a melhor implementação da nossa visão inicial para a aplicação.

Finalmente, para aumentar a produtividade da equipa e podermos concretizar entregas constantes das novas funcionalidades introduzidas no projeto, propomo-nos também a adotar mecanismos de DevOps ao longo do mesmo, bem como ferramentas de código partilhado e de gestão de tarefas.

2 Conceptualização do Produto

Visão Global

O produto Campus Monitoring encontraria o seu espaço no mercado como uma ferramenta essencial à execução do conceito de Edifícios Inteligentes. Este conceito mistura um domínio que tem sido tradicionalmente estático, como o edifício clássico, e confere-lhe características responsivas e dinâmicas. Com os avanços tecnológicos no campo de automação, atualmente é possível controlar à distância componentes de um edifício (como por exemplo, sistemas de ar condicionado, abertura de portas e janelas para ventilação, movimentar persianas), assim como utilizar sensores para medir em tempo real

o desempenho térmico e da qualidade do ar do mesmo.

A nossa solução centra-se na resolução do problema fulcral de como fazer a medição contínua e processamento das mudanças constantes de temperatura e níveis de CO₂ de um espaço público (neste caso, uma cantina universitária) em tempo real. A legislação atual de eficiência energética prevê um intervalo restrito de valores aceitáveis para estes parâmetros num edifício público, contudo, a fiscalização tem-se revelado demasiado difícil de concretizar, o que poderia ser solucionado pela nossa aplicação.

Como tal, iremos introduzir mecanismos de reação a eventos, sob a forma de alertas, que podem ser vistos pelos utilizadores da nossa aplicação como notificações no seu browser ou e-mail, o que garantiria uma monitorização 24/7.

Adicionalmente, pretendemos fornecer aos utilizadores do espaço um canal de comunicação para expressar feedback relativo às suas necessidades de conforto, visto que o desempenho dos mesmos no seu local de trabalho poderá ser significativamente afetado por condições adversas.

Uma aplicação com estas características poderia vir a ser integrada com facilidade num projeto tecnológico com maior escopo, que possibilitasse controlar equipamentos do edifício à distância para corrigir automaticamente desvios das condições ideais climatéricas da cantina após serem alertados; assim como estender o âmbito da aplicação para cobrir os restantes espaços públicos do campus.

Personas

Pedro Bastos: O Pedro tem 40 anos e é administrador dos serviços da cantina da Universidade de Aveiro há 6 meses. Devido à natureza do seu trabalho de escritório e horário exigente, não tem muita disponibilidade para falar diretamente com os seus funcionários ou dirigir-se múltiplas vezes durante o dia à cantina. Como tal, quer efetuar mudanças na gestão das condições do local de trabalho e ter um canal de comunicação direto para receber feedback. O seu objetivo principal é aumentar o desempenho dos seus trabalhadores, pelo que pretende a atividade profissional destes não seja perturbada por condições adversas e que estes se encontrem à vontade para expressar pontos negativos. Adicionalmente, a cantina poderá ser alvo de inspeções de fiscalização de eficiência energética, pelo que é do interesse do Pedro investigar potenciais problemas antecipadamente.

Motivação: O Pedro gostaria de aceder à visualização da evolução das condições de temperatura e de qualidade do ar ao longo do tempo e do histórico de alertas de violação de limites legislados para análise estatística dos dados. Para além disso, necessita um mecanismo de registo e quantificação dos principais problemas detectados no espaço de trabalho.

Maria Cardoso: A Maria tem 60 anos e trabalha como cozinheira na cantina da Universidade de Aveiro há cerca de 20 anos. Recentemente, foi diagnosticada com uma doença reumatológica cujos sintomas pioram com exposição ao frio. Nesse sentido, o seu

médico de família aconselhou-a a evitar temperaturas mais baixas. O chefe de equipa da cantina pretende acomodar a sua limitação, dado a vasta experiência da Maria como cozinheira e o seu contributo essencial para o bom funcionamento da cantina. Apesar da promessa de acomodação, muitas vezes a Maria passa horas no seu local de trabalho sob temperaturas demasiado frias, principalmente no Inverno, o que interfere com a sua capacidade de movimentação. Por vezes, ela comunica verbalmente o problema ao seu chefe, mas vê as suas preocupações serem ignoradas no meio do volume de trabalho.

Motivação: A Maria gostaria que as suas queixas fossem oficialmente registadas e de receber notificações personalizadas quando as condições de temperatura do seu local de trabalho se tornarem prejudiciais à sua saúde.

Cenários Principais

Pedro prepara-se para uma auditoria à cantina – O Pedro abre a aplicação e entra no menu de autenticação. Coloca os seus dados de funcionário da UA, o seu login é processado com sucesso e entra no menu principal. Nesta seção, escolhe a *opção correspondente à cantina*, entre diferentes espaços do campus. Na página da cantina, explora a opção de aceder a *registos dos históricos de evolução das condições máximas e mínimas* relativas a temperatura, humidade, pressão atmosférica e níveis de CO₂. Ele decide aceder à opção de *consulta de histórico de alarmes universais*. Estes dados proporcionam-lhe uma visão global da performance do espaço ao longo do tempo e ele irá usá-los na auditoria para mostrar o efeito positivo das mudanças que introduziu como novo administrador e que este espaço está a cumprir os regulamentos.

Pedro analisa feedback dos seus funcionários na reunião semanal – Na página principal da cantina, o Pedro seleciona a *opção de exibir críticas*. Ele ordena as críticas por *ordem cronológica* e por *nível de insatisfação* (apresentado de 1 a 5). O Pedro utiliza estes dados na sua reunião semanal com a equipa de gestão da cantina para aferir o grau de contentamento dos seus funcionários e estratégias de melhoria das condições para o mesmos.

Maria prepara-se para sair de casa para o trabalho – Ao iniciar o seu dia, a Maria tem agora por hábito abrir a aplicação, autenticar-se, dirigir-se à página da cantina e selecionar a opção para *visualizar as condições atuais*. Assim, poderá escolher o seu vestuário de forma a estar mais preparada para se sentir confortável no seu local de trabalho.

Maria realiza as suas tarefas diárias no local de trabalho – Durante o dia, a Maria é *notificada por e-mail* que a temperatura da cantina excede o mínimo que é seguro para alguém com os seus problemas de saúde. Como tal, ela interrompe as suas tarefas e comunica verbalmente ao chefe de equipa que há um problema para resolver. Após a introdução da aplicação, ela notou melhorias no seu espaço de trabalho e tem agora mais facilidade em desempenhar as suas tarefas, com a sua condição de saúde a ser acomodada. Ela acede à página da cantina, *seleciona a opção de deixar uma crítica* e deixa uma crítica positiva e uma pontuação mais elevada ao seu espaço de trabalho.

3 Architecture notebook

Requisitos chave e limitações

- O sistema deve fornecer um mecanismo de autenticação de funcionários e administradores através da inserção do email da universidade.
- O sistema deve fornecer uma forma de persistência, segura, estável e funcional, de modo a preservar os dados relativos a funcionários e administradores, bem como dos registos dos sensores.
- O sistema deve ter integração com o DBMS, de forma a tornar possível a gestão, organização e acesso a dados armazenados na base de dados.
- O sistema deve apresentar um desempenho fluído e otimizado.
- O sistema deve fornecer uma interface adequada, que tenha um bom equilíbrio entre a apresentação de toda a informação relevante e a simplicidade de utilização.
- O sistema deve ter implementado um mecanismo de notificações que permita aos utilizadores configurarem eventos para serem notificados aquando da sua ocorrência.
- O sistema deve impedir que um utilizador tenha acesso a dados confidenciais e privados de outros utilizadores.
- O sistema deve disponibilizar em tempo real as condições do espaço em que está instalado, medido através de sensores físicos.
- O sistema deve ter uma ligação contínua à Internet.
- O sistema deve ser capaz de resistir e continuar em funcionamento em condições anormais de temperatura e níveis de humidade, pressão atmosférica e CO2.
- O sistema deve ser robusto e ser instalado de forma estratégica, de forma a interferir o menos possível com o fluxo de trabalho do corpo trabalhador existente, bem como resistir a eventuais ocorrências

Visão da Arquitetura

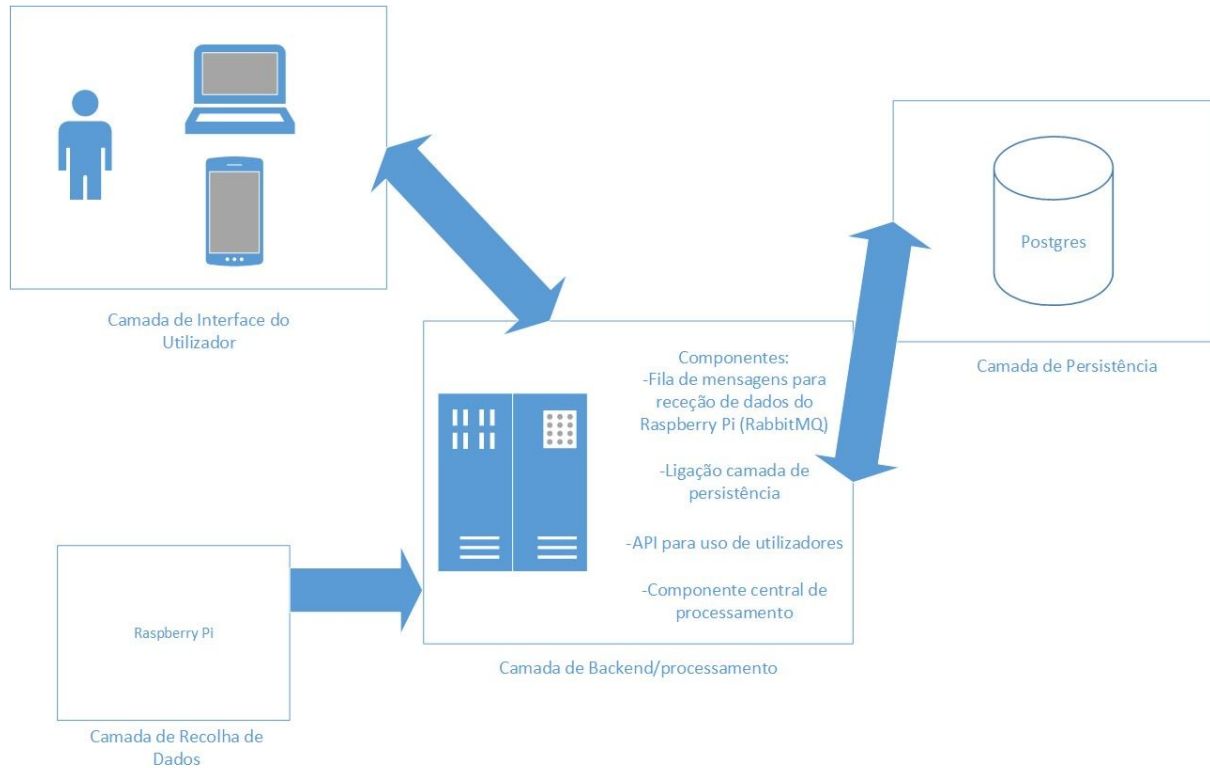
A arquitetura da solução, tal como proposto, será multilayered, podendo ser identificadas 4 camadas/módulos gerais: camada de persistência, camada de backend, camada de user interface and interaction e camada de geração de dados sensoriais.

A camada de dados sensoriais é uma camada de hardware constituída pelo Raspberry pi 3b+ com sensor BME680. A camada de persistência contém a base de dados de armazenamento dos dados recolhidos pelo Raspberry Pi. A decisão de uso de base de dados relacional foi feita pela interdependência dos dados recolhidos e necessidade das vantagens proporcionadas por este tipo de base de dados. Foi escolhido PostgreSQL em vez de MySQL pelas seguintes razões: Postgres tem Pgadmin como gui intuitiva e fácil de usar; mais compatível com SQL do que MySQL, logo mais facilmente usada por quem tem background em SQL server; bastante usada em sistemas em larga escala onde operações de escrita e leitura rápidas são cruciais e os dados têm de ser validados; múltiplos reads sem lock possíveis; grande comunidade para suporte e fóruns de ajuda; suporte a extensões (como Timescale) que serão necessárias no projeto.

A camada de backend trata do heavy work de processamento dos dados, para gestão de dados de interesse (valores elevados de CO2, valor mais baixo de temperatura

num certo dia/mês, etc) e passagem dos dados à camada de user interface, que é responsável pela interação com o utilizador. Nesta última camada, a interação com o utilizador é feita através de web app.

Primeira versão do diagrama de arquitetura:



Interações entre módulos

A comunicação entre os módulos tem a camada de backend como elemento central. Os dados recolhidos do Raspberry Pi (na camada de recolha de dados) são enviados para esta camada que trata da inserção na camada de persistência (base de dados) e do processamento em runtime dos dados que vai controlar o comportamento da interface da aplicação. Este comportamento inclui os possíveis alertas quando são detectadas condições adversas ou outros sinais que consideremos relevantes. Assim, a camada de backend interage de forma unidirecional com a camada de recolha de dados (os dados dessa camada são passados sempre diretamente para a camada backend) e de forma bidirecional com as restantes camadas (isto porque há troca de dados tanto originária, como destinada para a camada backend) .

4 Information perspective

5 Referências e recursos