

# Fase de Descoberta, Mapa de Expectativas e Priorização MoSCoW

**Título do Projeto:** Sistema para Monitoramento de Temperatura em Estufas do grupo – Fatec Jacareí

**Grupo:** SafeTemp

**Data:** 18 /09

## Criando o Mapa de Expectativas

**Projeto:** Sistema para Monitoramento de Temperatura em Estufas – Fatec Jacareí

**Objetivo:** Desenvolver uma solução IoT integrada com backend em nuvem, visualização mobile/web e alertas automáticos, promovendo segurança e confiabilidade no controle de temperatura de estufas acadêmicas.

## Usuários-Alvo

- Pesquisadores/Professores responsáveis pelos experimentos.
- Alunos que acompanham os cultivos.
- Equipe de manutenção/apoio técnico da Fatec.

Sentimentos desejados	Funcionalidades essenciais	Frustrações a evitar
<b>Segurança:</b> Garantia de que os dados são confiáveis e o sistema reage a anomalias.	<b>1. Integração IoT:</b> Sensor(es) de temperatura conectados via ESP32 (ou similar).	Perda de dados por falhas na nuvem ou sensores.
<b>Tranquilidade:</b> Alertas automáticos oferecem supervisão contínua.	Comunicação em tempo real por Wi-Fi.	Interface difícil ou mal adaptada para mobile.
<b>Confiança:</b> Monitoramento em tempo real e dados precisos.	Possibilidade de expansão para outros sensores (umidade, luz, etc.).	Alertas com falhas ou excessivamente frequentes.

<b>Controle:</b> Acesso remoto e capacidade de intervenção.	<b>2. Backend em Nuvem:</b> Recepção dos dados via APIs REST.	Sistema que exige manutenção frequente.
<b>Praticidade:</b> Sistema de fácil navegação e configuração.	Processamento e armazenamento contínuo dos dados.	Falta de compatibilidade entre dispositivos.
<b>Engajamento:</b> Incentivo ao uso em projetos e aulas.	Implementação em container Docker	Baixa confiabilidade nas medições.
<b>Eficiência:</b> Redução de erros humanos e tempo gasto.	Disponível em serviços cloud (Ex: Azure, AWS ou Heroku).	Alto consumo de bateria ou dados móveis.
<b>Satisfação:</b> Cumprimento das necessidades reais do usuário.	<b>3. Aplicação Mobile e Web:</b> Acesso multiplataforma (mobile/web).	Dificuldade para exportar ou interpretar os dados.
	Visualização em tempo real dos dados (gráficos, status de sensores).	Falta de suporte técnico ou instruções claras.
	Interface responsiva e centrada no usuário.	
	Definição e visualização de limites de temperatura.	
	Funcionalidade de exportar dados (PDF/planilha).	
	<b>4. Notificações e Alertas:</b> Envio automático de SMS em casos de temperatura fora dos limites.	
	Sistema de cadastro e gerenciamento de usuários e alertas.	
	<b>5. Processo e Engenharia de Software:</b> Containerização com Docker (backend e serviços).	
	Implementação de CI/CD para automatizar build e deploy.	
	Documentação completa da arquitetura e do sistema.	

Indicadores de sucesso	Requisitos de UX
Uptime do sistema acima de 99%.	SUS score $\geq 80$ (satisfação com usabilidade).
Precisão de medição dentro de $\pm 0,5$ °C.	Aumento da frequência de uso ao longo do tempo.

Entrega de SMS com taxa acima de 95%	Tempo médio de resposta aos alertas.
Latência de exibição de dados < 5 segundos.	Satisfação (NPS – Net Promoter Score).
Consumo otimizado (bateria e dados).	NPS ≥ 8 (recomendação por parte dos usuários).

## Priorização MoSCoW

Obrigatórios	Importantes	Desejáveis	Não serão feitos agora
Interface intuitiva e menus simples.	Acessibilidade: bom contraste, texto legível, compatível com leitores de tela.	Modo escuro.	Gamificação para usuários.
Gráficos claros e códigos de cor para temperatura.	Personalização básica: limites de alerta ajustáveis.	Exportação de relatórios em PDF.	Dashboards totalmente customizáveis.
Feedback visual imediato para ações (ex: envio de alerta).	Cadastro de usuários simplificado.	Visualização offline dos últimos dados armazenados localmente.	
Design responsivo (mobile/tablet/PC).	Ajuda integrada (dicas/tutoriais).		
Carregamento rápido dos dados e gráficos.			