



UNIVERSIDADE FEDERAL DO MARANHÃO  
COORDENAÇÃO DE ENGENHARIA DA COMPUTAÇÃO

EECP0021 - SISTEMAS DISTRIBUÍDOS

**ANTONIO LUCAS DA SILVA VALE**

**ANDRE MOURA LIMA**

**JEYSRAELLY ALMONE DA SILVA**

**VICTOR COELHO DA SILVA**

**PLANEJAMENTO DO PROJETO DE SISTEMAS DISTRIBUÍDOS - Calendário com  
base no TAP – Termo de abertura do projeto.**

SÃO LUÍS - MA

JUNHO/2025



<b>1 INTRODUÇÃO</b>	<b>3</b>
<b>2 ATUALIZAÇÃO DO PROJETO</b>	<b>4</b>
<b>3 ESTRUTURA ANALÍTICA DO PROJETO – FASES E PRINCIPAIS ENTREGAS</b>	<b>5</b>
3.1 ESTRUTURA ANALÍTICA ATUALIZADA	5
<b>4 Análise das Datas e Atrasos no Cronograma do Projeto</b>	<b>9</b>
<b>5 JUSTIFICATIVA PARA ATRASO NO CRONOGRAMA: SUBSTITUIÇÃO DO MIDDLEWARE INTERSCITY E REJEIÇÃO DA PLATAFORMA THINGSPEAK</b>	<b>10</b>
<b>CONCLUSÃO</b>	<b>11</b>



## 1 INTRODUÇÃO

Este trabalho apresenta um cronograma de planejamento voltado para o início e o desenvolvimento do projeto "**Alerta de Alagamentos para São Luís**". A estrutura do cronograma foi definida com base no Termo de Abertura do Projeto (TAP) previamente proposto, servindo como guia para a organização das atividades acadêmicas relacionadas ao tema. O objetivo é garantir uma execução ordenada das etapas planejadas, promovendo o acompanhamento eficiente do progresso do trabalho ao longo do período letivo.



## 2 ATUALIZAÇÃO DO PROJETO

Optou-se por **deixar em aberto a escolha do software de middleware** a ser utilizado na implementação do sistema, de modo a possibilitar a experimentação com diferentes plataformas de código aberto que atendem aos princípios de **sistemas distribuídos**. Entre as opções consideradas estão:

- **FIWARE**, plataforma voltada para cidades inteligentes, com componentes modulares e suporte à modelagem de contexto urbano.
- **Mainflux**, middleware escalável e seguro, com arquitetura baseada em microsserviços e suporte a múltiplos protocolos (como MQTT e HTTP).
- **ThingsBoard**, plataforma visual com suporte à comunicação com dispositivos IoT, criação de dashboards e regras de alarme.
- **InterSCity**, desenvolvido no Brasil com foco em cidades inteligentes, que será avaliado por meio do **playground disponibilizado pela UFMA**.

Por outro lado, a ferramenta **ThingSpeak**, inicialmente mencionada, **não foi adotada** como solução final por **não se caracterizar como um middleware de sistemas distribuídos**. Apesar de oferecer funcionalidades úteis para armazenamento e visualização de dados IoT, o ThingSpeak não contempla suporte nativo à comunicação entre múltiplos nós ou à distribuição de componentes, o que o torna limitado frente aos objetivos do projeto.

Assim, a abordagem aberta permite comparar as soluções de middleware disponíveis de forma crítica, considerando critérios como escalabilidade, facilidade de integração, suporte ao ESP32 e capacidade de adaptação ao contexto urbano de São Luís.

Com o avanço das análises, foi possível localizar o repositório do InterSCity no GitLab, contendo os arquivos necessários para sua implantação. Além disso, a equipe contará com a documentação técnica disponibilizada pelo professor responsável, bem como com o



acesso à rota hospedada sob seu domínio institucional, o que viabiliza a integração e testes do sistema com maior fidelidade. Diante disso, decidiu-se pela utilização efetiva do InterSCity como middleware principal do projeto, dado que sua arquitetura modular, voltada a cidades inteligentes, se alinha aos objetivos da solução proposta. A continuidade das próximas etapas dependerá, portanto, da efetiva disponibilização desse ambiente de testes e da infraestrutura correspondente, assegurando que os requisitos de distribuição, comunicação e escalabilidade possam ser atendidos conforme o planejado.



### 3 ESTRUTURA ANALÍTICA DO PROJETO – FASES E PRINCIPAIS ENTREGAS

#### 3.1 ESTRUTURA ANALÍTICA ATUALIZADA

ETAPA	1. Pesquisa e Planejamento
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Estudo sobre sensores de nível, comparação entre Wi-Fi e GSM (SIM800L), levantamento dos requisitos para integração com a plataforma e uso do protocolo <b>MQTT</b> para comunicação.
OBSERVAÇÕES	Durante essa etapa foi feito um levantamento de possíveis temas de trabalho e como poderíamos realizar um projeto que fosse alinhado à realidade da nossa cidade.
DATA DE INÍCIO E TÉRMINO	18/04/2025 até 25/04/2025

ETAPA	2. Aquisição de Componentes
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Visualização de softwares de simulação on-line, como o tinkercad para utilização inicial durante experimentos, podendo então verificar o funcionamento dos componentes e realizar testes básicos de funcionamento desses.
OBSERVAÇÕES	Durante essa etapa fizemos uma aferição dos possíveis softwares que poderíamos utilizar para auxiliar em testes básicos e também estabelecemos como meta a utilização da placa esp32, para posterior prototipação física.
DATA DE INÍCIO E TÉRMINO	12/06/2025 até 30/06/2025



--	--

ETAPA	4. Montagem do Protótipo
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Montagem do circuito no protoboard com sensor de nível e módulo de comunicação. Estrutura básica para envio de dados via MQTT.
OBSERVAÇÕES	Etapa não concluída e por isso sem observações das atividades realizadas.
DATA DE INÍCIO E TÉRMINO	30/06/2025 até 10/07/2025

ETAPA	3. Pesquisa e Comparação de Middlewares de Sistemas Distribuídos
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Realização de pesquisa técnica sobre diferentes middlewares de código aberto com suporte a sistemas distribuídos, a fim de avaliar quais se adequam melhor ao projeto "Alerta de Alagamentos para São Luís". As opções analisadas incluem FIWARE, Mainflux, ThingsBoard e InterSCity (via playground da UFMA), levando em conta critérios como compatibilidade com o ESP32, protocolos suportados, escalabilidade, facilidade de integração e suporte a visualização de dados.
OBSERVAÇÕES	Durante esta etapa, será feita uma análise comparativa entre os middlewares mencionados, com foco na identificação de vantagens e limitações de cada um para o cenário proposto. A ferramenta ThingSpeak



	foi desconsiderada por não ser um middleware de sistemas distribuídos. A etapa ainda se encontra <b>em andamento</b> , aguardando testes práticos e definição final da ferramenta a ser adotada.
DATA DE INÍCIO E TÉRMINO	10/07/2025 até 12/07/2025

ETAPA	5. Desenvolvimento do Código
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Programação do Arduino para: leitura do sensor, envio de dados via <b>MQTT</b> , e conexão com a (registro de recurso, envio de dados de sensores).
OBSERVAÇÕES	Etapa não concluída e por isso sem observações das atividades realizadas.
DATA DE INÍCIO E TÉRMINO	12/07/2025 até 12/07/2025

ETAPA	6. Testes de Campo
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Testes em ambiente de simulação: medição de níveis de água, envio dos dados por MQTT, verificação do recebimento e registro na plataforma escolhida. Ajustes no código.
OBSERVAÇÕES	Etapa não concluída e por isso sem observações das atividades realizadas.
DATA DE INÍCIO E TÉRMINO	12/07/2025 até 13/07/2025





ETAPA	7. Montagem Final e Acabamento
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Montagem definitiva em caixa, revisão da comunicação MQTT e testes de estabilidade com a plataforma selecionada e otimização da alimentação.
OBSERVAÇÕES	Etapa não concluída e por isso sem observações das atividades realizadas.
DATA DE INÍCIO E TÉRMINO	13/07/2025 até 13/07/2025

ETAPA	8. Documentação e Apresentação
RESPONSÁVEL	Equipe de projetos
DESCRIÇÃO	Produção do relatório final incluindo: arquitetura do sistema com MQTT, fluxograma de comunicação, uso da plataforma com vídeo de demonstração (opcional) e apresentação final.
OBSERVAÇÕES	Etapa não concluída e por isso sem observações das atividades realizadas.
DATA DE INÍCIO E TÉRMINO	14/07/2025 até 14/07/2025



#### **4 Análise das Datas e Atrasos no Cronograma do Projeto**

A análise do cronograma mostra que algumas tarefas do projeto estão significativamente atrasadas, especialmente a partir da Etapa 3, que trata do desenvolvimento do sistema de controle e processamento. A tarefa 3.2, que envolve a implementação do software de controle para coleta e processamento dos dados dos sensores, deveria ter sido concluída entre os dias 5 e 10 de maio, mas até o momento apresenta apenas 20% de progresso. As demais tarefas dessa etapa, bem como todas as atividades das Etapas 4, 5 (exceto a 5.1), 6 e 7, ainda não foram iniciadas, mesmo estando com prazos vencidos ou próximos do vencimento.

A única exceção é a tarefa 5.1, que trata da busca por outro software para substituir o middleware Interscity. Essa atividade foi realizada dentro do prazo, entre os dias 1º e 10 de junho, e resultou na escolha de um novo middleware. No entanto, as atividades de simulação e teste da infraestrutura de energia (5.2 e 5.3), que dependem dessa substituição, seguem paradas.

Na Etapa 6, responsável pelos testes e validação do sistema, apenas a tarefa inicial (6.1) foi concluída. Todas as demais, que envolvem testes práticos e ajustes com base nos resultados, ainda não foram executadas. Isso é preocupante, pois essas tarefas são fundamentais para garantir que o sistema funcione corretamente antes da fase final de implementação.

A Etapa 7, referente à entrega final e ao monitoramento do sistema, está prevista para ocorrer entre os dias 10 e 31 de julho. No entanto, com o volume de tarefas atrasadas nas etapas anteriores, há um risco real de que o projeto não seja finalizado dentro do prazo previsto.

Em resumo, os principais atrasos foram causados pela necessidade de substituir o middleware originalmente escolhido, o que exigiu tempo adicional para pesquisa, análise e adaptação. Agora, é essencial revisar o cronograma, redefinir prioridades e acelerar a execução das atividades pendentes para evitar impactos maiores na conclusão do projeto.



Cabe ressaltar que nenhuma atividade prevista no cronograma original foi cancelada. Todas as tarefas que sofreram interrupções ou atrasos foram apenas postergadas, em função da necessidade de reavaliação técnica e da definição do middleware a ser utilizado. Com a recente decisão de prosseguir com a utilização do InterSCity, as atividades anteriormente suspensas serão retomadas, respeitando as dependências técnicas e a nova organização dos prazos. A equipe mantém o compromisso de concluir todas as etapas previstas, assegurando a entrega completa e funcional do sistema proposto.

## **5 JUSTIFICATIVA PARA ATRASO NO CRONOGRAMA: SUBSTITUIÇÃO DO MIDDLEWARE INTERSCITY E REJEIÇÃO DA PLATAFORMA THINGSPEAK**

O presente relatório tem como objetivo justificar o atraso ocorrido no cronograma do projeto, previamente estabelecido na fase de planejamento inicial. Tal atraso decorreu da necessidade de substituição do middleware Interscity, cuja adoção mostrou-se inviável diante das exigências técnicas do projeto.

Diante dessa situação, iniciou-se um processo sistemático de busca e avaliação de alternativas, com o intuito de identificar um middleware que atendesse de forma mais adequada aos requisitos definidos. A análise considerou critérios como escalabilidade, segurança, qualidade da documentação técnica, bem como a compatibilidade com a arquitetura previamente delineada.

Durante esse processo, a plataforma ThingSpeak foi avaliada como uma possível candidata. Entretanto, após uma análise minuciosa de suas funcionalidades e da sua aderência aos padrões e exigências do projeto, concluiu-se que a referida plataforma não possuía a flexibilidade e a capacidade de integração necessárias, principalmente no que diz respeito às customizações e adaptações exigidas para o contexto específico da implementação.

A rejeição do ThingSpeak, embora tenha contribuído para o desvio no cronograma, mostrou-se essencial para assegurar a qualidade e a robustez da solução a ser entregue. O tempo adicional investido na busca por um middleware mais compatível reforça



o comprometimento da equipe com a entrega de um produto alinhado às expectativas, requisitos técnicos e operacionais do projeto.

Atualmente, a equipe permanece empenhada na avaliação de outras alternativas tecnológicas que possam atender, de forma mais eficaz, aos padrões definidos.

## CONCLUSÃO

Diante dos objetivos propostos e das adaptações realizadas ao longo do desenvolvimento, este trabalho apresenta um cronograma estruturado para orientar o andamento do projeto "**Alerta de Alagamentos para São Luís**". Optou-se por **deixar em aberto a escolha do middleware**, considerando alternativas de código aberto como **FIWARE**, **Mainflux**, **ThingsBoard** e **InterSCity** (por meio do playground da UFMA), que oferecem suporte a sistemas distribuídos e são compatíveis com o ESP32.

A ferramenta **ThingSpeak**, inicialmente mencionada, foi descartada por **não se caracterizar como um middleware de sistemas distribuídos**, o que limita sua aplicação no contexto deste projeto.

Dessa forma, o cronograma elaborado serve como apoio para a organização das atividades e para o acompanhamento do desenvolvimento e atualização do Gráfico de Gantt, permitindo que o grupo avance de forma mais estruturada na implementação da solução proposta.



**Anexo I** - Link do repositório.

[https://github.com/AndreMouraL/Sistemas-distribuido\\_Sensor-de-alagamento.git](https://github.com/AndreMouraL/Sistemas-distribuido_Sensor-de-alagamento.git)