



QUEM SOMOS?

A **Atmosfera Tech** é uma empresa dedicada a desenvolver soluções tecnológicas inovadoras, com o objetivo de melhorar o cotidiano das pessoas. Nosso principal produto, Atmos, é um aplicativo de previsão do tempo, projetado para oferecer informações precisas e em tempo real, com uma interface simples, funcional e de fácil navegação. A plataforma foi desenvolvida para garantir que qualquer pessoa, em qualquer lugar, possa acessar rapidamente os dados que precisa para tomar decisões mais informadas sobre o clima.

Na Atmosfera Tech, buscamos excelência em tudo o que fazemos, com o compromisso de inovar continuamente para criar produtos que impactam positivamente a vida das pessoas e o mundo ao nosso redor.



QUEM SOMOS NÓS?



**VICTOR
MELO**

Diagramas



**GABRIEL
VÍNICIUS**

Integração
com a API



**THAYLIZE
SANTOS**

Design e
repositórios



**MAXWELL
MACIEL**

Interface Gráfica

SKILLS DESENVOLVIDAS



**VICTOR
MELO**

Antes: Criar GitHub, e-mail, diagrama.

Após: GitHub configurado, e-mail criado e diagrama elaborado.



**GABRIEL
VÍNICIUS**

Antes: Java básico (variáveis, for), sem experiência com APIs ou JSON.

Após: Manipulação de APIs e JSON, uso de try e catch, execução de apps Java em IDE.



**THAYLIZE
SANTOS**

Antes: Design, Figma, Canva, GitHub básico.

Após: Gerenciamento completo de repositórios, identidade visual e interfaces refinadas.



**MAXWELL
MACIEL**

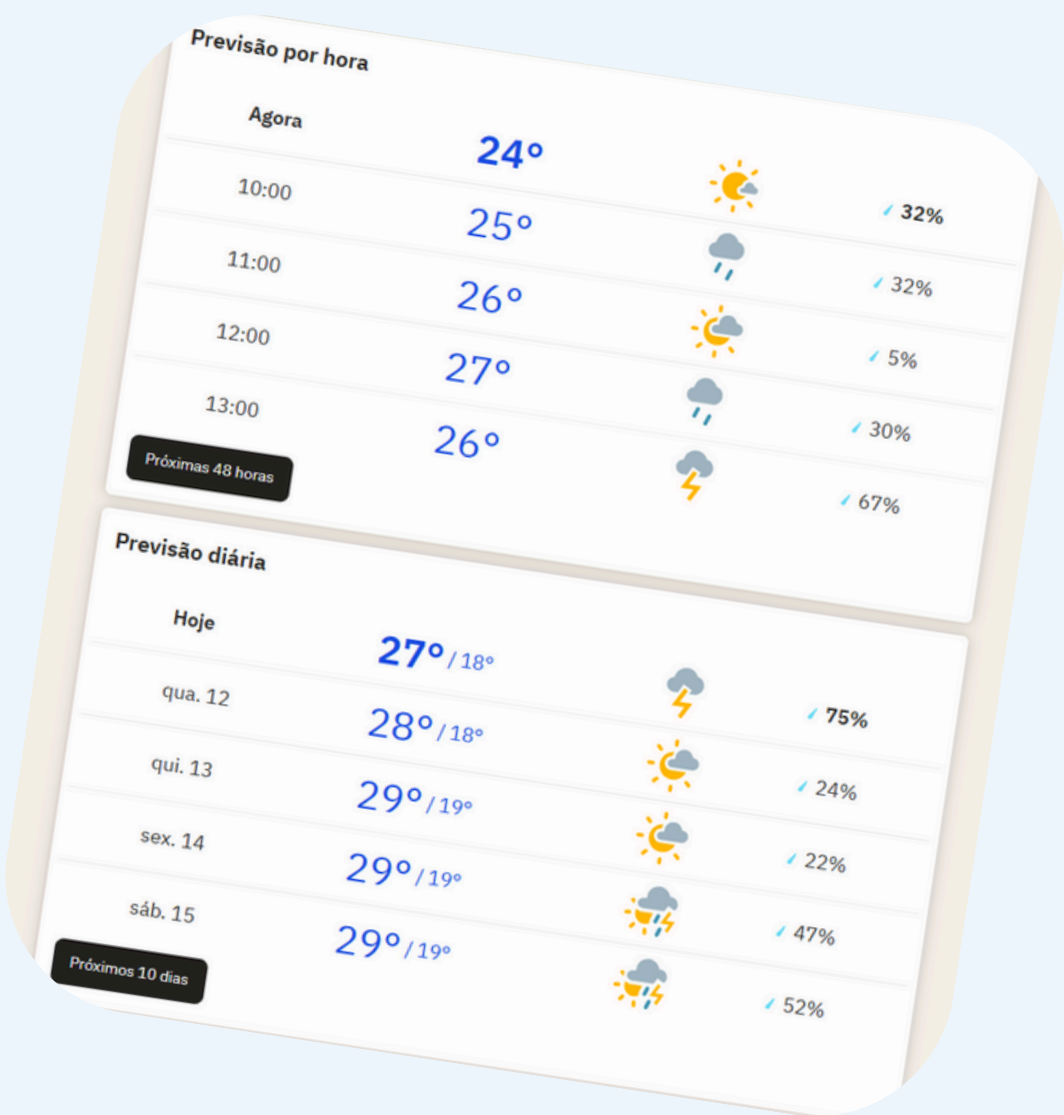
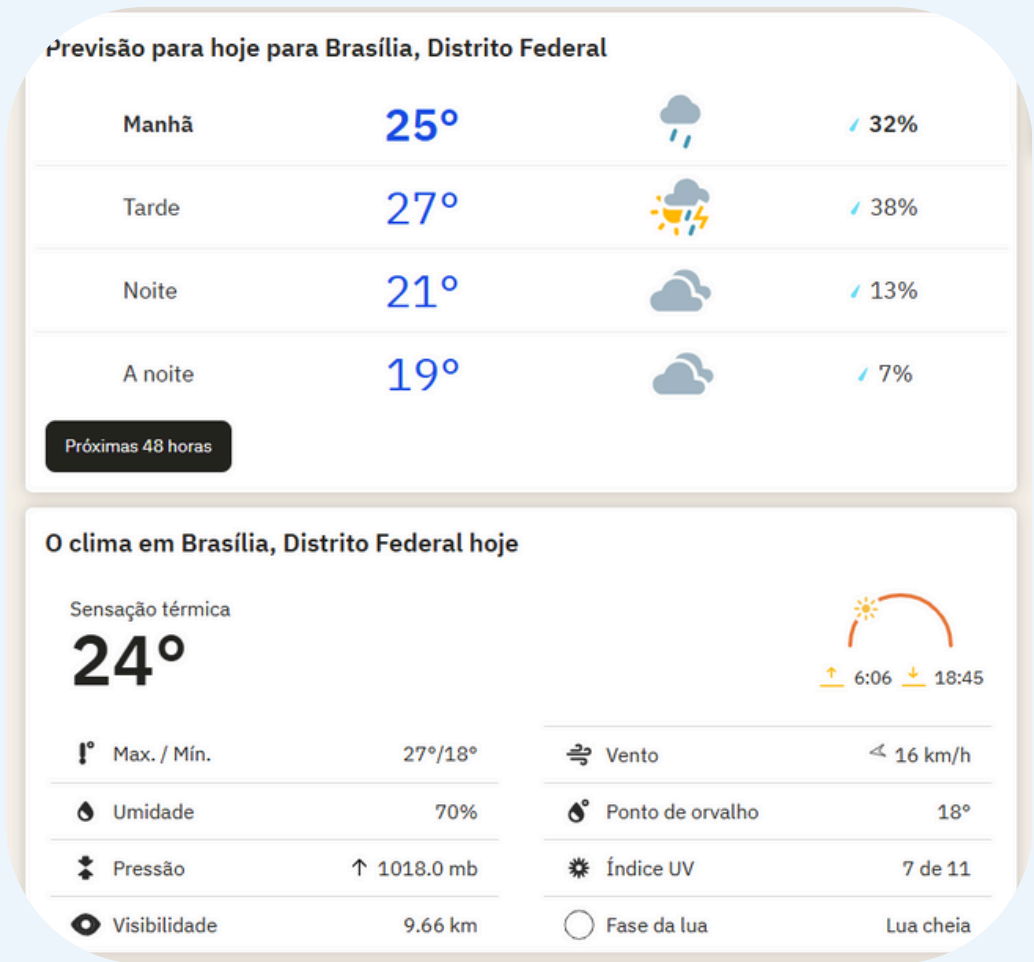
Antes: Conhecimento básico de POO e interfaces gráficas.

Após: Aprendizado de novas linguagens, aprofundamento em POO e uso avançado de interfaces gráficas.

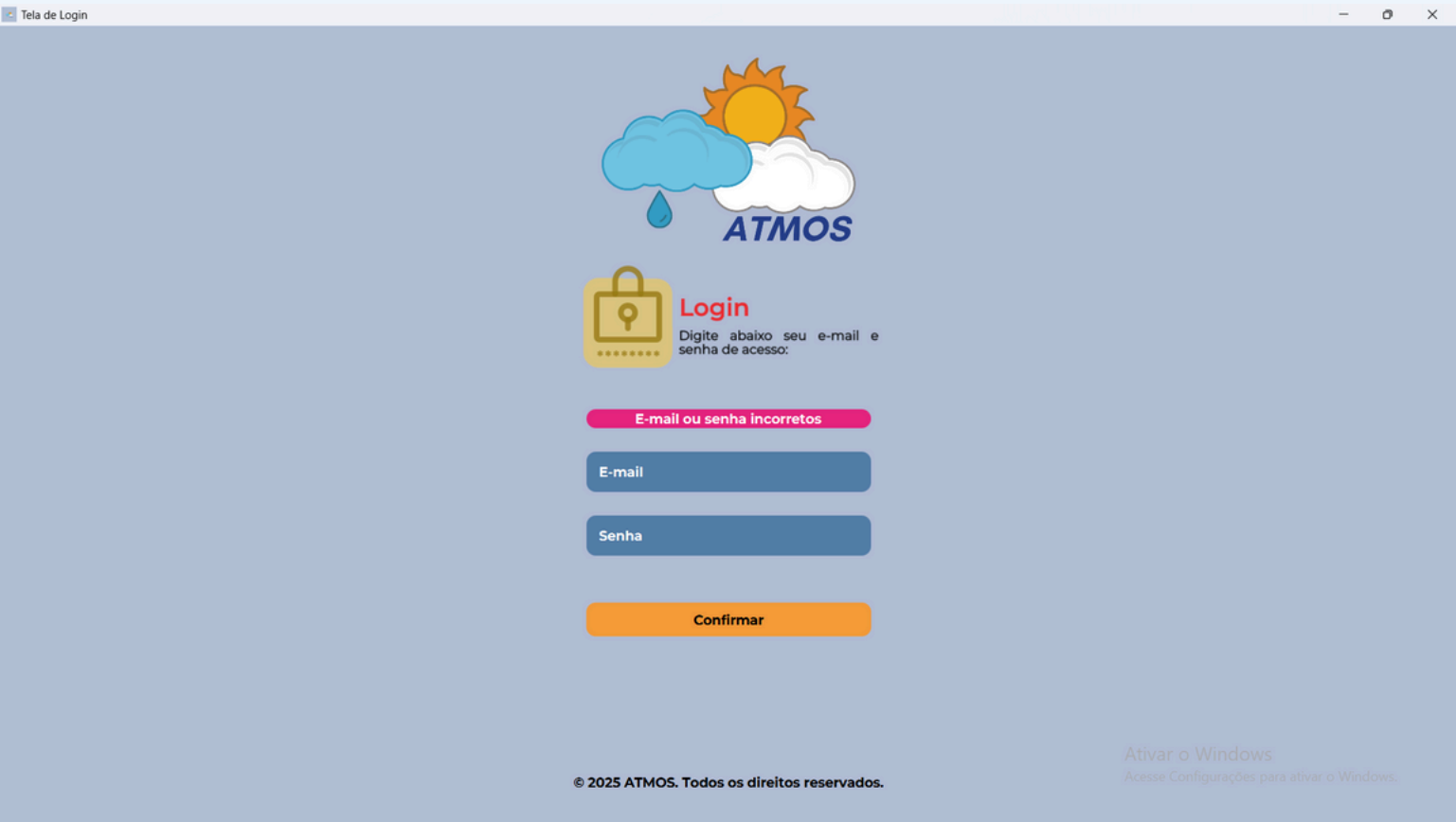
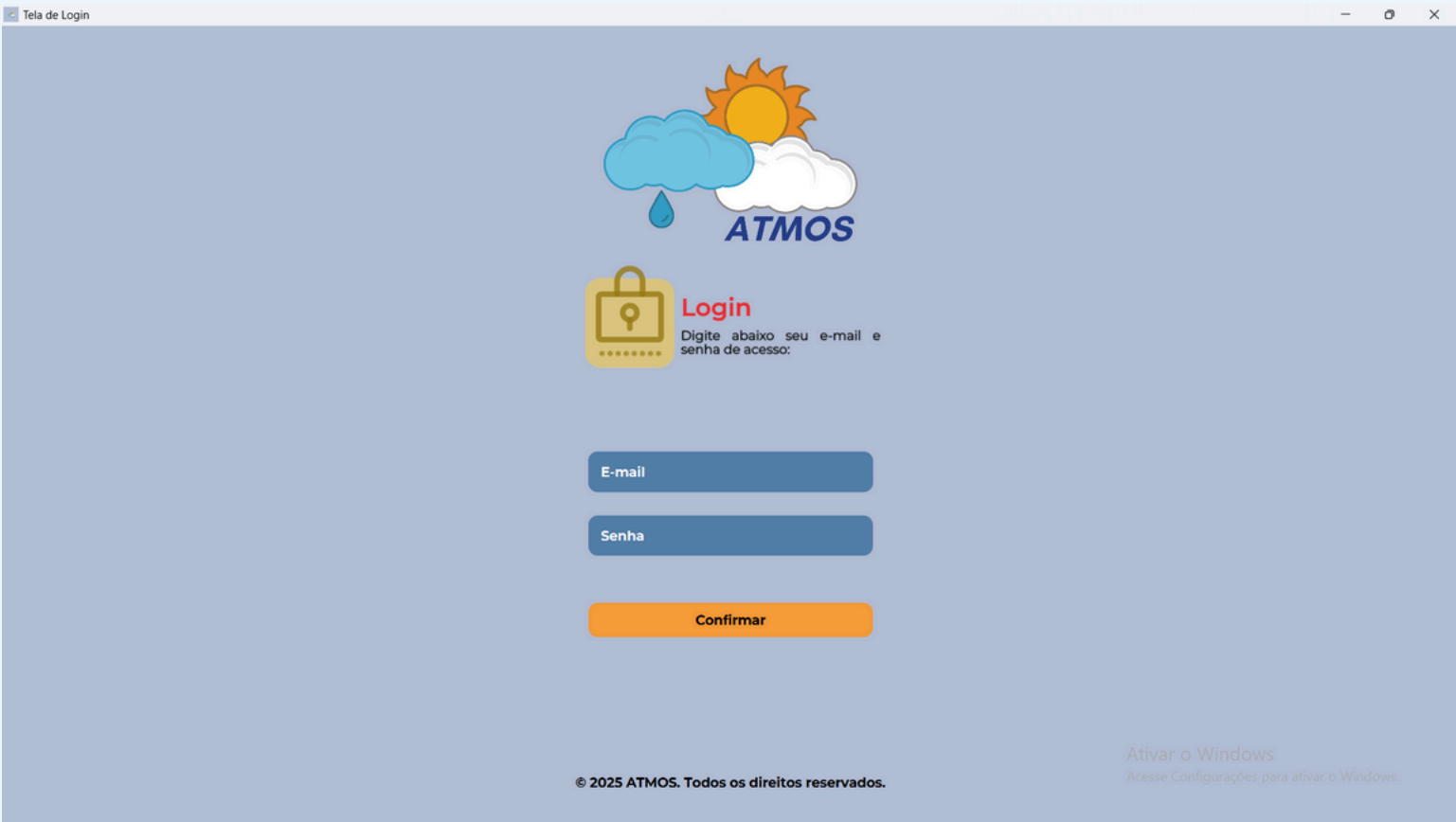
OBJETIVOS INICIAIS:

O objetivo inicial era criar um aplicativo simples e responsivo, inspirado no site weather.com, que fornecesse informações meteorológicas precisas e de fácil acesso.

Utilizando a API OpenWeather, o app deveria consumir e exibir dados como temperatura, umidade, previsão do tempo, entre outros, de maneira clara e intuitiva. Além disso, a interface deveria ser agradável aos olhos, com um design moderno e fluido, garantindo uma experiência de usuário agradável em diferentes dispositivos e tamanhos de tela.



COMO FICARAM / A ENTREGA



TOOLBOX



ECLIPSE IDE

Foi usado para o código principal, usando o JAVAFX para a criação das interfaces



CANVA

Foi usado para a confecção dos slides e obtenção de alguns ícones.



FIGMA

Foi usado para fazer os designs iniciais das telas.



INTELIJ IDE

Foi usado para a primeira versão de consumo da API.



YOUTUBE

Foi usado para ver tutoriais a respeito da criação de telas, tiração de dúvidas, etc..



GPT

Usado para melhorar a legibilidade do código, tirar dúvidas, e adicionar comentários no código final.



ECLIPSE IDE - JAVA FX

O JavaFX é uma plataforma para a criação de interfaces gráficas (GUIs) rica e moderna em aplicações Java. Ele fornece um conjunto de bibliotecas e ferramentas para desenvolver interfaces de usuário com funcionalidades avançadas, como animações, efeitos gráficos, mídia, entre outros. JavaFX substitui o antigo Swing, oferecendo uma abordagem mais moderna e mais fácil de usar para construir aplicações desktop. Além disso, JavaFX é altamente personalizável, permitindo o uso de CSS para estilizar os componentes.

Para criar uma interface básica com JavaFX, você estende a classe `Application`, sobrescreve o método `start()`, cria os componentes (como botões) e os adiciona a um layout (como `StackPane`). Em seguida, cria uma `Scene` e a associa ao `Stage` para exibir a janela. Para aplicar estilos, você cria um arquivo CSS, vincula-o à cena com `getStylesheets().add()` e define os estilos para os componentes, como cor de fundo e tamanho da fonte, permitindo personalizar a aparência da aplicação sem alterar o código Java.

TOOLBOX



ECLIPSE IDE - JAVA FX

Para instalar o JavaFX no Eclipse, primeiro é necessário garantir que o JDK 8 ou superior esteja instalado, pois o JavaFX vem integrado a partir dessa versão. Em seguida, no Eclipse, baixe o pacote do JavaFX diretamente do site oficial da Oracle e extrair o conteúdo.

No Eclipse, criei um novo projeto Java e, nas configurações do projeto, acessei as propriedades, vá até a aba "Java Build Path" e adicione as bibliotecas do JavaFX, indicando a pasta onde o JavaFX foi extraído. Também foi necessário configurar a variável de ambiente PATH e adicionar o caminho para a biblioteca do JavaFX nas opções de execução do projeto.

Finalmente, no código, você pode importar as classes do JavaFX e começar a desenvolver interfaces gráficas. Para integrar com a API, foi necessário baixar os componentes json em sua versão .jar, e por um processo similar, os integrar na pasta.

PRINCIPAIS BARREIRAS E DESAFIOS

Ao longo do projeto, foram enfrentados diversos desafios para que o mesmo fosse possível de ser realizado. No meio destes desafios, encontramos também barreiras intransponíveis.

DESAFIOS:

- Integração da API com a interface;
- Download e utilização do JAVAFX;
- Processo criativo complexo;
- Diagramas

BARREIRAS:

- Tempo: Mesmo com mais de 2 meses, não conseguimos concluir tudo o que planejamos.
- API: A versão gratuita da OpenWeatherMap não forneceu dados como qualidade do ar e fases da lua.
- Design: O design desejado foi simplificado devido às limitações de estilização no Java.





POSSÍVEIS MELHORIAS

1.PROTEÇÃO DA CHAVE API

Não expor a chave no código
usar criptografia

2.USAR BANCO DE DADOS

Histórico de pesquisa
Cache de dados
Usuários e preferências

3.INTERNACIONALIZAÇÃO (i18n)

Retorna mensagens de erro em diferentes idiomas



**OBRIGADO
PELA ATENÇÃO!!**

