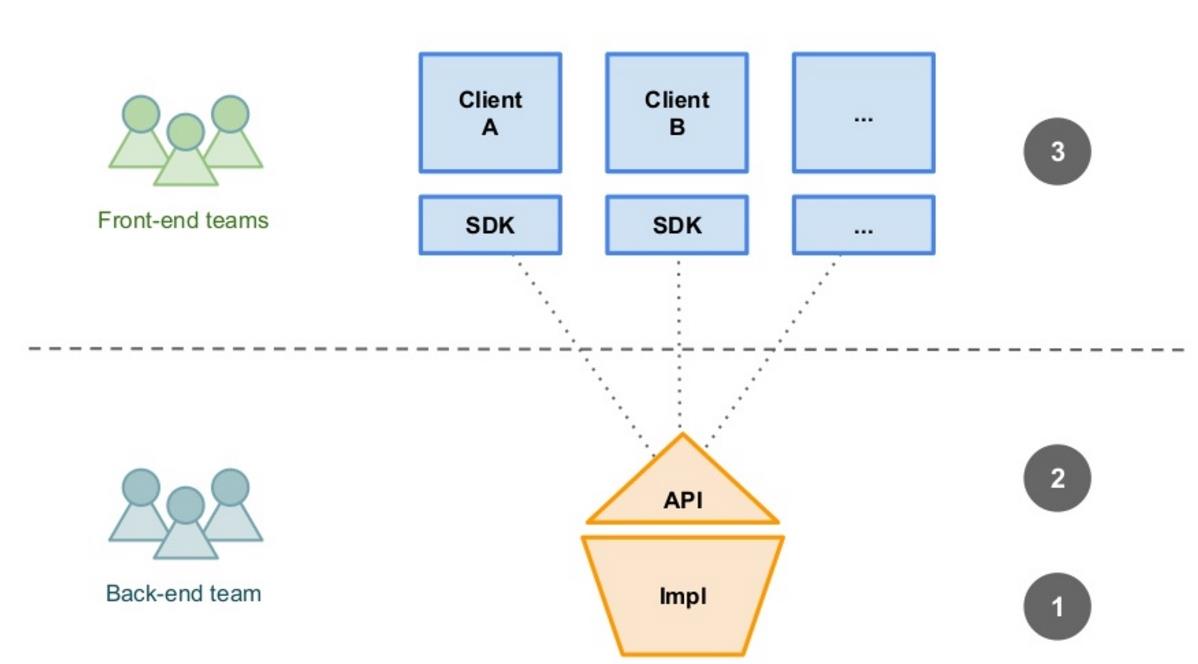
**API First**

Uma estratégia/modelo de desenvolvimento em que a primeira ordem de negócio é desenvolver uma API que coloque os interesses do desenvolvedor de destino em primeiro lugar para posteriormente construir o produto em seu topo, seja um site, um aplicativo móvel ou um software SaaS, nessa estratégia esta será a primeira e possivelmente única interface da aplicação. Ao se utilizar desta estratégia e construir em cima de APIs tendo os desenvolvedores em mente, muito esforço é poupado ao se estabelecer as bases onde outros desenvolvedores irão construir seus apps.

A API projetada vai ser para o próprio aplicativo tornando-se assim uma API REST similar ao mundo real, e ao mesmo tempo, compatível com o desenvolvimento de produto ou serviço.

**Utilização**

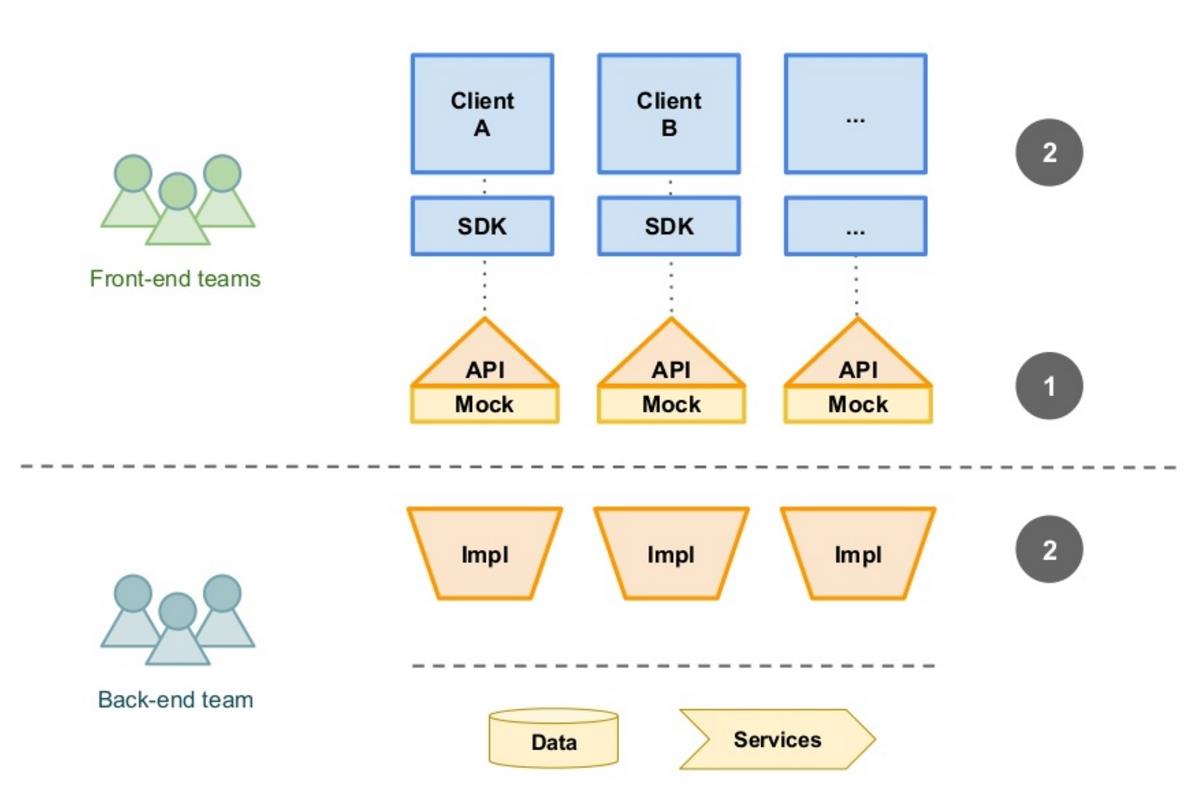
Normalmente o processo de desenvolvimento mais usual é síncrono, ou seja, após feito o projeto as equipes de back end começam a escrever o protótipo, enquanto as outras equipes como a de front-end ficam no aguardo e somente depois que o projeto tenha sido concluído é que pode ser feito um documento da API e compartilhado com as diferentes equipes.



Exemplo 1: Desenvolvimento síncrono

Quando uma mudança se torna necessária seja ela por motivos de bugs ou novas implementações o ciclo começa todo de novo, desperdiçando tempo e energia valiosos e consequentemente atrasando o lançamento do projeto para o mercado.

Por outro lado, com a utilização da API First o desenvolvimento é paralelo por todas as equipes sem a necessidade de esperar por alterações ou depender de uma equipe ou outra



Exemplo 2 API First

Neste modelo as APIs criadas são **mocks.** E as equipes de Back-end, Front-end e testes podem trabalhar simultaneamente com as APIs simuladas. Quando a API estiver totalmente pronta as equipes poderão alternar entre a API de produção ou de teste, economizando desta maneira muito tempo de desenvolvimento, adiantando o lançamento do produto e consequentemente poupando dinheiro, afinal tempo é dinheiro.

**Conclusão**

A capacidade de se adaptar cada vez mais rápido a novas funcionalidades e implementações na área de desenvolvimento de software é uma das se não a mais importante, por isso a tendência é que cada vez mais o modelo API Fisrt seja implementado e utilizado por programadores, pois com ele o sistema estará livre para crescer, adaptar-se e acomodar novos consumidores de serviços existentes sem precisar parar tudo para rearquitetar um sistema fechado.

**Vocabulário**

**Software SaaS** = No modelo SaaS, o fornecedor do software se responsabiliza por toda a estrutura necessária à disponibilização do sistema, e o cliente por sua vez utiliza o software via internet, pagando um determinado valor pelo serviço

**REST** = É definido como um conjunto de restrições de arquitetura.

**Mocks** =Objetos mock, objetos simulados ou simplesmente mock são objetos que simulam o comportamento de objetos reais de forma controlada.

**SDK** = conjunto de ferramentas de desenvolvimento de software que permite a criação de aplicações para um certo pacote de software.

**API** = APIs são mecanismos que permitem que dois componentes de software se comuniquem usando um conjunto de definições e protocolos

**Bibliografia**

[O que é uma API? – Guia de APIs para iniciantes – AWS (amazon.com)](https://aws.amazon.com/pt/what-is/api/)

[Kit de desenvolvimento de software – Wikipédia, a enciclopédia livre (wikipedia.org)](https://pt.wikipedia.org/wiki/Kit_de_desenvolvimento_de_software)

[Objeto mock – Wikipédia, a enciclopédia livre (wikipedia.org)](https://pt.wikipedia.org/wiki/Objeto_mock)

[API-First: uma abordagem de desenvolvimento | MANDIC](https://blog.mandic.com.br/artigos/uma-abordagem-de-desenvolvimento-api-first/)

[O que é API First? Entenda essa abordagem de desenvolvimento (callix.com.br)](https://blog.callix.com.br/o-que-api-first/#:~:text=O%20API%20First%2C%20como%20o,design%20das%20APIs%20%C3%A9%20priorizado.)

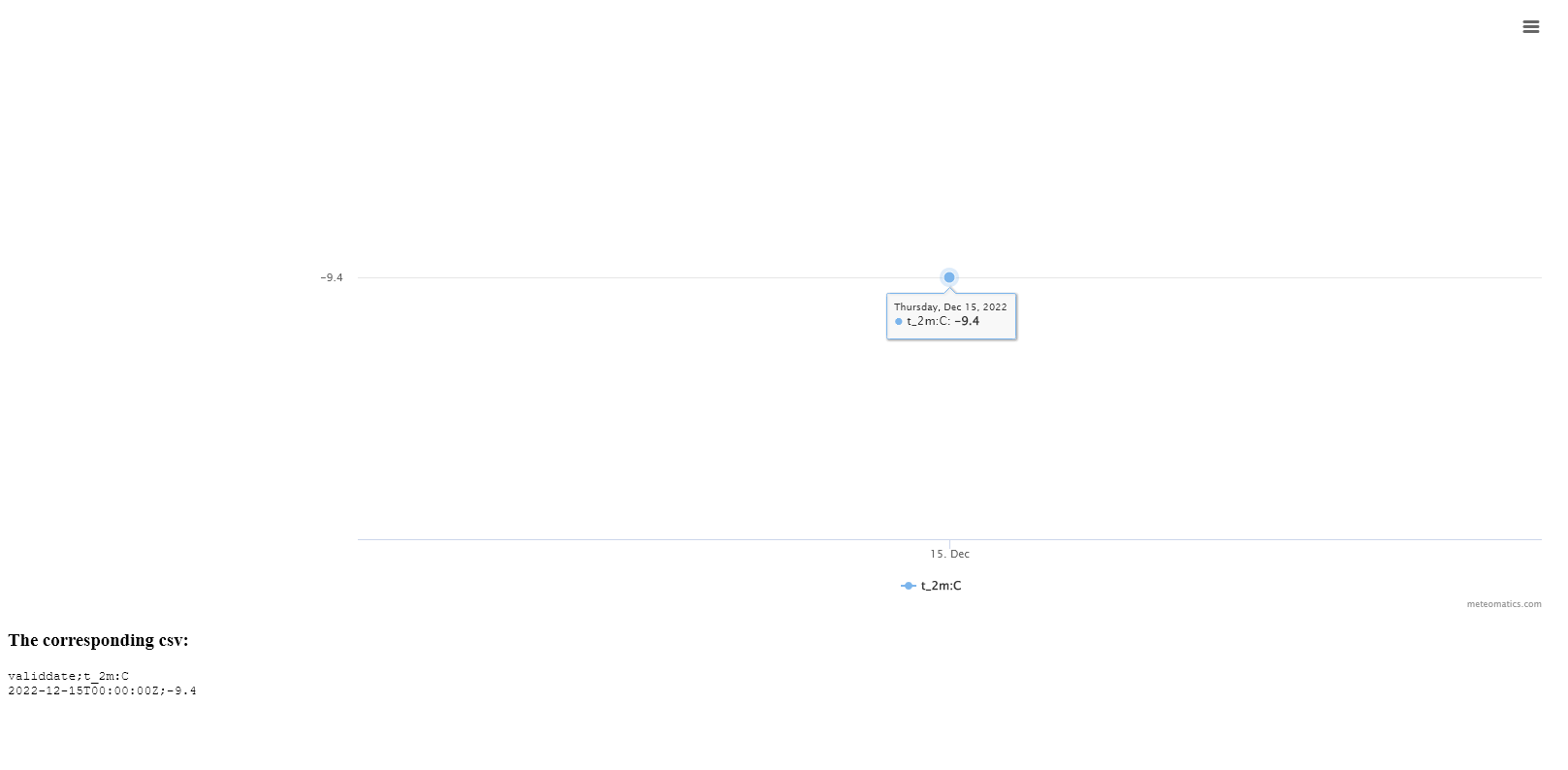
Todos acessados em 02/12/2022.

**API’S**

Meteomatics Weather APIEndpoint**:**<https://api.meteomatics.com/2022-12-15T00:00:00Z/t_2m:C/52.520551,13.461804/html>

Meio de autenticação: Usuário e senha.

Parâmetros de entrada: **t\_c2m: C** (temperatura a dois metros acima do solo em graus Celsius), latitude e longitude (em graus decimais).

Resposta: Página Html 

**Códigos de retorno HTTP**

**Código: 102 – Processing**

**Mensagem**: indica que o servidor recebeu e está processando a requisição, mas nenhuma resposta está disponível ainda

**Código: 205 – No Content**

**Mensagem**: Não há conteúdo para enviar para esta solicitação, mas os cabeçalhos podem ser úteis.

**Código: 300 – Multiple Choice**

**Mensagem**: A requisição tem mais de uma resposta possível. O usuário deve escolher uma delas (não há maneira padrão para efetuar a escolha).

**Código: 302 – Found**

**Mensagem**: A URL do recurso requerido foi mudada temporariamente

**Código: 400– Bad request**

**Mensagem**: O servidor não compreendeu a requisição devido a sintaxe inválida.

**Código: 401 – Unauthorized**

**Mensagem**: O cliente deve s autenticar para obter a resposta solicitada

**Código: 404– Forbidden**

**Mensagem**: O cliente não tem direito de acesso ao conteúdo portanto o servidor não lhe concedera a resposta.

**Código: 423 – Locked**

**Mensagem**: O recurso sendo acessado está travado

**Código: 429 – Too many Requests**

**Mensagem**: O usuário enviou muitas requisições em um dado espaço de tempo (“limitação de frequência”).

**Código: 451 – Unavailable For Legal Reasons**

**Mensagem**: O usuário requisitou um recurso ilegal, como uma página censurada por um governo por exemplo.

**Ferramentas para API’S**