# Web2py - Introdução ao desenvolvimento ágil para web utilizando Python.

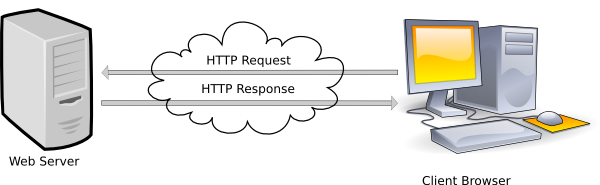
Neste livro, vamos aprender como desenvolver uma aplicação rápida para web, utilizando um framework construído em Python e persistência em banco de dados.

## Web Framework

Antes de falar sobre o framework propriamente dito, vamos entender como funciona uma aplicação básica web: Uma aplicação web é composta de diversas funções que são executadas em determinadas URL’s, o resultado é renderizado para o navegador do usuário. Dessa forma, os web frameworks servem como um facilitador de desenvolvimento, com ferramentas construídas para facilitar o seu desenvolvimento.

## Protocolo HTTP

Ao navegar na web, você deve ter percebi no seu navegador, mais precisamente na barra de endereços, o protocolo HTTP (Hypertext Transfer Protocol), que é base da web atualmente, sua função é realizar a comunicação entre cliente e servidor e para isso utiliza os serviços request (requisitar) e response (responder).



## Sobre o web2py

Web2py é um framework para aplicações web de código aberto escrito na linguagem de programação Python. O web2py permite que os desenvolvedores web programem conteúdo web dinâmico usando Python. Ele foi projetado para ajudar a reduzir tarefas tediosas de desenvolvimento web, como o desenvolvimento de formulários web do zero, apesar de um desenvolvedor web poder construir um formulário do zero se necessário.

O web2py foi originalmente projetado como uma ferramenta de ensino com ênfase na facilidade de uso e implementação. Entretanto, ele não possui quaisquer arquivos de configuração de nível de projeto. O projeto do web2py foi inspirado pelos frameworks Ruby on Rails e Django. Assim como esses frameworks, o web2py foca no desenvolvimento rápido, favorece a abordagem convenção sobre configuração e segue o padrão arquitetural modelo-visão-controlador (MVC).

## Modelo MVC

Model-view-controller (MVC) é um padrão de arquitetura de software que separa a representação da informação da interação do usuário com ele. O modelo (model) consiste nos dados da aplicação, regras de negócios, lógica e funções. Uma visão (view) pode ser qualquer saída de representação dos dados, como uma tabela ou um diagrama. É possível ter várias visões do mesmo dado, como um gráfico de barras para gerenciamento e uma visão tabular para contadores. O controlador (controller) faz a mediação da entrada, convertendo-a em comandos para o modelo ou visão. As ideias centrais por trás do MVC são a reusabilidade de código e separação de conceitos.

## MODELS

Nos models são guardadas informações de bancos de dados, conexão, tabelas e colunas. A partir disso, poderemos gravar e armazenar dados da aplicação e interagir com eles: visualizar, editar, deletar, etc.

Os models também são sempre os primeiros arquivos a serem lidos, em ordem alfabética, pelo web2py. Ou seja, se eu tenho um arquivo db.py e um model.py, as informações do arquivo db.py serão lidas e interpretadas antes de model.py. Portanto, se alguma informação presente em db.py depender de algo em model.py, ocorrerá um erro!

Além disso, tudo o que for definido em um model poderá ser acessado em todas as partes da aplicação.

## CONTROLLERS

Nos controllers são definidas as funções da aplicação que farão com que ela funcione da forma como designada. São nos **controllers** que poderemos fazer com que a aplicação interaja com o banco de dados e envie informações para o usuário. No web2py, cada função sem parâmetros no controller é mapeada para uma URL, ou seja, cada função define uma diferente página da sua aplicação.

Caso você passe parâmetros para uma função, ela deixa de ser mapeada como uma página e passa a ser uma função Python comum, que poderá ser chamada no código. Isso é especialmente feito em **models**, pois assim a função será acessível em todas as camadas de código.

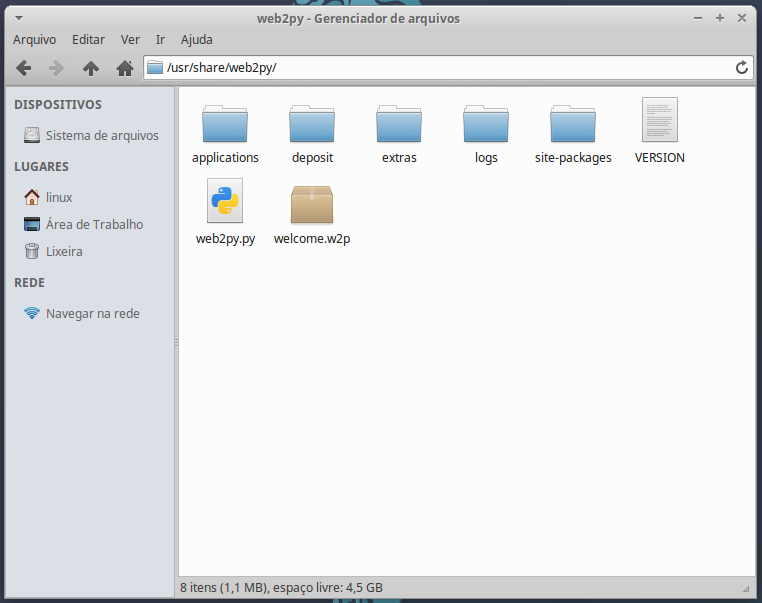
## VIEWS

As views definem o que a aplicação vai exibir para o usuário e a forma como isso se dará. No web2py, views são arquivos HTML que podem receber código Python. O código Python em um view deve estar dentro dos delimitadores {{ }} para ser executado.

Views podem herdar ou incluir outras views. Isto é, você pode fazer com que uma view contenha outra, exibindo ambas em conjunto.

## DESENVOLVENDO SUA PRIMEIRA APLICAÇÃO COM WEB2PY

O primeiro passo é organizar o seu ambiente de desenvolvimento. Ao criar uma nova aplicação na interface administrativa do web2py, teremos a respectiva pasta da aplicação dentro do nosso diretório do web2py. Para começar a desenvolver, precisamos entender como funciona as estruturas de pastas de nosso Framework



A maior parte do conteúdo desta pasta não fará grande diferença para você enquanto desenvolve, mas existem alguns que merecem um certo destaque:

O diretório applications contém todas as aplicações que você criou no web2py. Cada aplicação é uma pasta que contém diversos outros conteúdos que analisaremos também adiante.

O diretório examples contém arquivos de exemplo que você pode usar como modelo para várias funcionalidades do web2py. Por exemplo: configurações de rota, configurações de logging, configurações do appengine, entre outros.

O diretório gluon contém os módulos e arquivos que literalmente constroem o framework.

O arquivo web2py.py, que é o arquivo que inicia a execução do framework.