

Simulador do Servidor recebendo dados de um Dispositivo

Documentação da Aplicação de Python na Global Solutions

Resumo de nossa ideia para Global Solutions:

A nossa ideia se resume a um dispositivo simples tipo Arduino ou Esp32, que faz medições das condições ambientais ao seu redor e envia esses dados para um servidor. Assim coletando mais dados para prever melhor as condições ambientais futuras como enchentes.

Resumo da utilização do Python para nosso projeto:

Para o nosso projeto decidimos fazer um simulador do funcionamento dos Servidores ao receber dados transmitidos pelos nossos dispositivos. Neste caso os Dados são inseridos pelo usuário e a partir desses dados a simulação do Servidor faz estimativas de qual será o nível do Rio. Após todos os dados serem inseridos pelo usuário, o programa exibe todos os dados para o usuário. E caso a previsão do nível do Rio é que ele irá ultrapassar a altura segura, ele fara alterações a mensagem exibida para o Usuário com um alerta.

Objetivo com este projeto:

Demonstrar como os servidores processariam as informações coletadas pelos dispositivos e como seriam capazes de alertar usuários caso os dados indiquem uma situação de risco.

Aviso:

Os cálculos matemáticos para fazer previsões são arbitrários e não possuem base em estáticas reais, isso foi feito para simplificar a utilização do programa pelo usuário, já que para fazer cálculos verídicos, seria necessária uma quantidade enorme de dados, específico de cada local, de cada trecho de um rio.

Documentação do código:

Import:

Nosso código utiliza a biblioteca Random para alterar a temperatura por uma quantidade aleatória dentro de um limite. (Isso será explicado mais afundo na parte das funções).

Definições de Variáveis e Listas:

Antes de entrarmos no código, definimos alguns valores que seria definido junto do dispositivo, como as coordenadas em que o dispositivo se localiza. O dispositivo se localizará utilizando seu modulo de GPS e enviará essa informação uma vez, assim definindo suas coordenadas como uma variável constante. Sabendo sua coordena, o dispositivo ou o servido, pesquisaria pela internet o rio mais próximo, para definir as constantes como nome do rio local, escoamento do rio local e a partir de qual altura acima do seu nível padrão, as populações locais estariam em risco. Tudo isso seria definido antes do processamento de dados, como o objetivo desse projeto é apenas simular o processamento de dados, esses valores serão constantes já definidas neste projeto, para simplificar.

Nesta seção também definimos as listas, para que os dados “coletado pelo dispositivo” possam ser armazenados.

Funções:

Por questão de modularidade, o projeto inteiro é feito por funções, uma função para cada etapa do processamento dos dados. Para facilitar a compreensão da programação do projeto, as funções serão explicadas em ordem em que são chamadas, ou seja, na ordem em que são executadas.

main():

A nossa primeira função é bem simples, uma função que engloba todas as outras funções, para que o código inteira seja executado chamando apenas está função. Dentro dela temos um loop **for** que repete a quantidade de vezes definida pela próxima função **(quantidade_de_dados())**.

Este loop é essencial para registrar a quantidade de dados pedido pelo usuário e adicionar, manipular e inserir os dados em suas respectivas listas, nas posições certas.

quantidade_de_dados():

Esta é uma função que pede para o usuário inserir um valor inteiro referente a quantidade de dados que ele deseja fornecer ao servidor. Em outras palavras, a quantidade de dias, que é o mesmo que a quantidade de dados. Esta função também é uma função recursiva, pois caso o usuário insira um valor que não seja um número inteiro, um **int**, a função retorna um erro que chama está mesma função, pedindo ao usuário repetidamente para inserir um valor inteiro, até ele inserir um valor inteiro.

receber_dados(i):

Nesta função, pedimos para o usuário inserir um valor que representa o valor que o servidor receberia do dispositivo em certo dia. Por ser um valor baseado na porcentagem da umidade do solo, o usuário só consegue inserir um valor entre 0 e 100. Assim como a última função, está é uma função recursiva, em que caso o valor inserido seja inválido, ele chama esta função repetidamente até o usuário inserir um valor válido.

previsao_elementos(i):

Esta função define os valores baseados nos valores inseridos pelo usuário para a umidade do solo. Ele faz estimativas da temperatura e quantidade de chuva baseado na diferença entre o valor inserido neste loop e no último loop. Inserindo os na lista. Caso este seja o primeiro valor inserido, ele insere valores padrões pré-definidos. Se o valor inserido agora for maior que o último valor inserido, uma previsão de chuva maior é inserida no dado anterior, já que umidade maior do solo agora, indica que havia previsão de chuva para o dado anterior, e o valor é proporcional para a diferença dos dados entre um loop e o outro, para mostrar uma relação entre umidade do solo e chuva.

Já a temperatura, partimos de um valor padrão e assumimos que a temperatura abaixa quando chove e sobe se não chove. Já que para reduzir a umidade no solo, a uma necessidade de uma temperatura maior. O valor da temperatura subir ou descer é limitado a um valor entre 0 e 3 graus.

previsao_rio(i):

Esta função dá uma previsão do nível do rio, baseado na previsão de chuva. Ele soma o valor anterior do nível do rio a um aumento esperado devido a quantidade de chuva. Então dependendo de quantos milímetros de chuva é esperado, ele aumenta o nível do rio em certa quantidade. Caso a previsão seja que não terá chuva, o nível do rio abaixa por uma quantidade fixa, baseada no escoamento do rio (variável definida antes do processamento de dados), parando em um limite (assumimos que não haverá uma seca e que a fonte do rio continuará fornecendo água para manter um nível mínimo do rio).

exibir_dados():

Nesta função, todos os dados são exibidos para o usuário através da função print. Primeiro ele exibe para o usuário de qual dispositivo esses dados vieram (exibindo as coordenadas únicas do dispositivo), em seguida ele exibe uma lista dos valores de cada dia através de outro **for loop**. Indicando primeiro o Dia, seguido da umidade do solo, temperatura, previsão de chuva e por último a previsão do nível do rio. Caso o nível do rio tenha uma previsão que ultrapasse a altura limite de segurança para

a população local, esta função destaca a parte da lista com um alerta, indicando a previsão do nível do rio e ressaltando que uma enchente é provável.