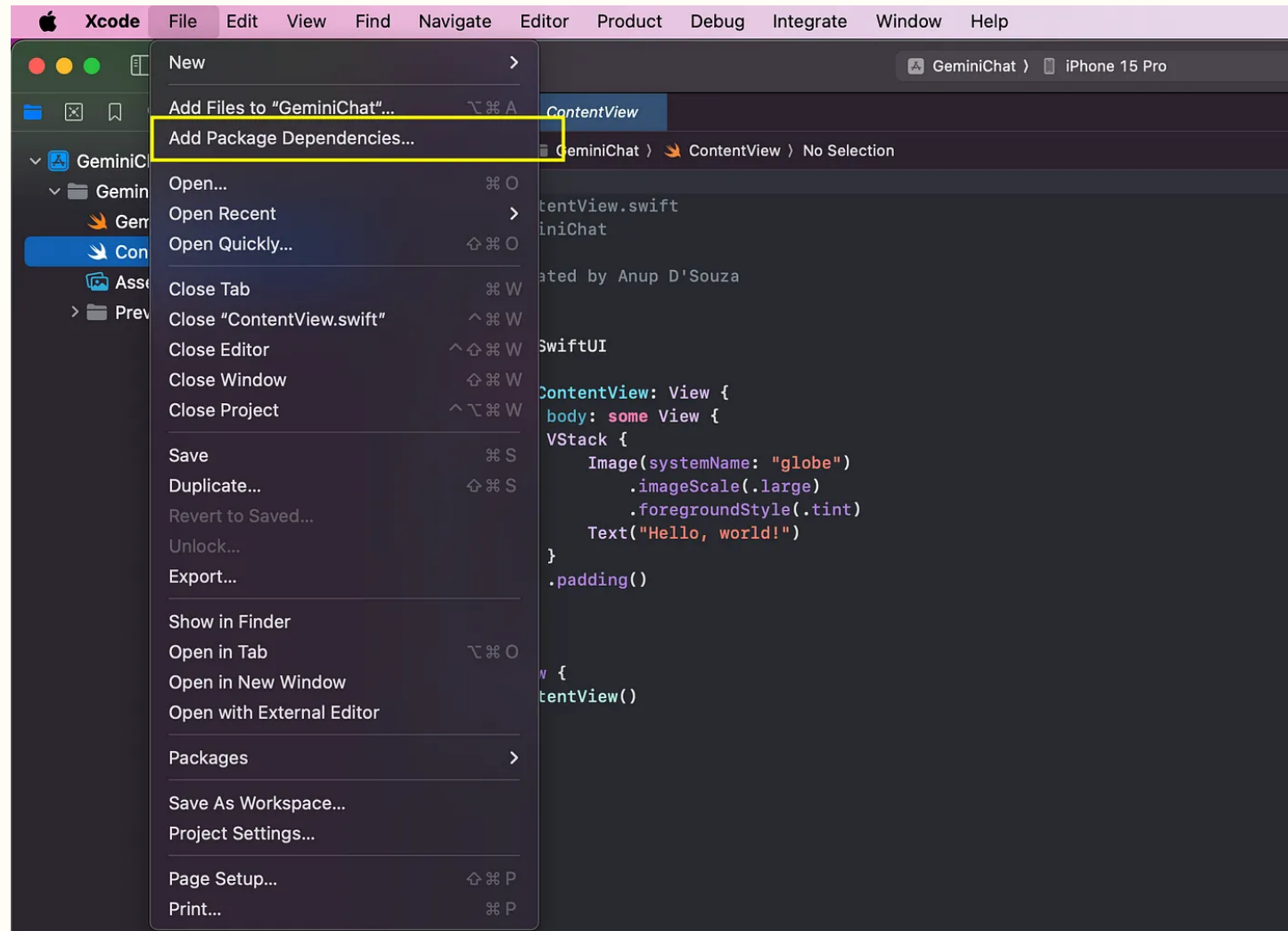


Gemini API Xcode

Tutorial

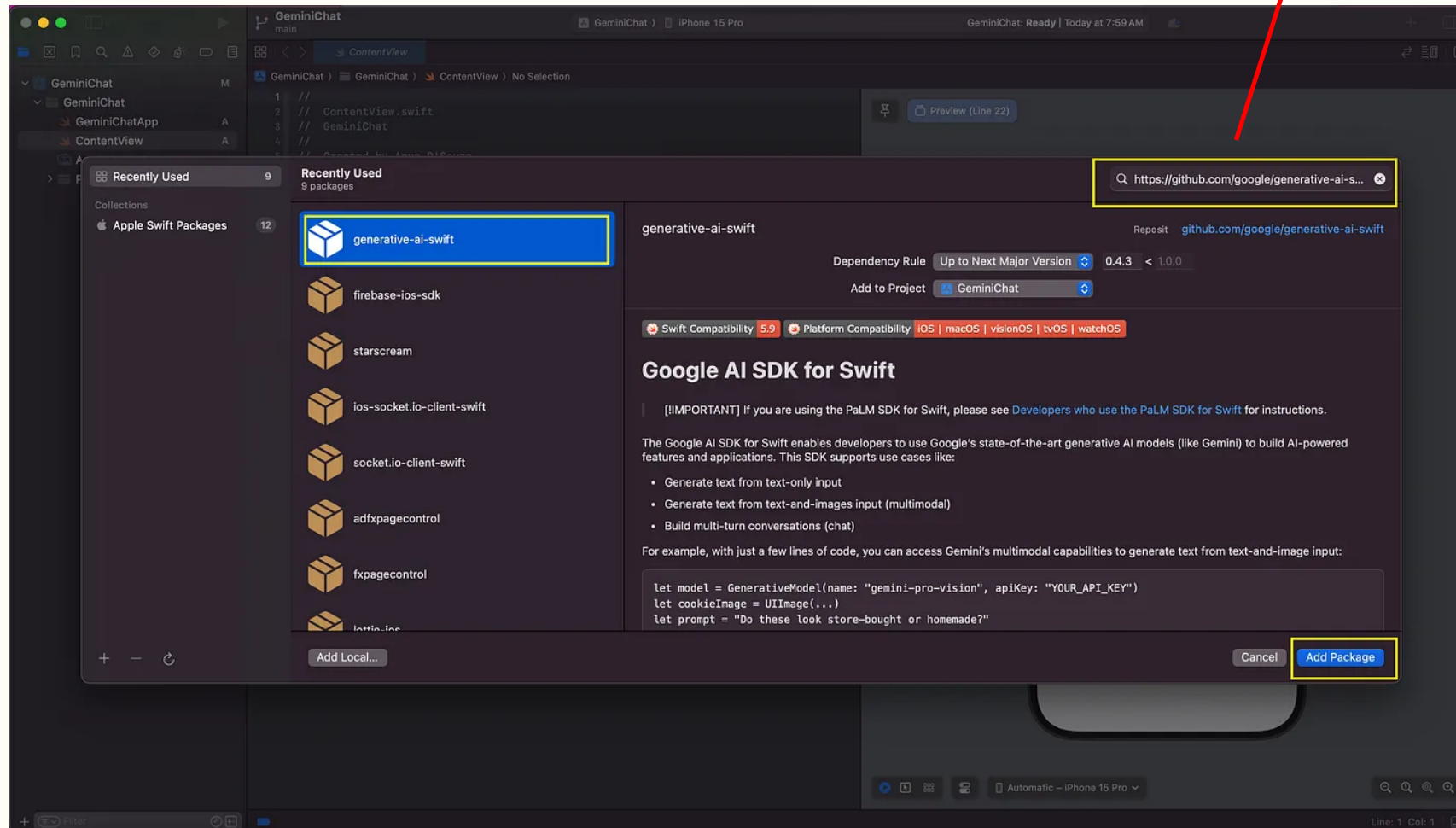


1) Adicionar dependência

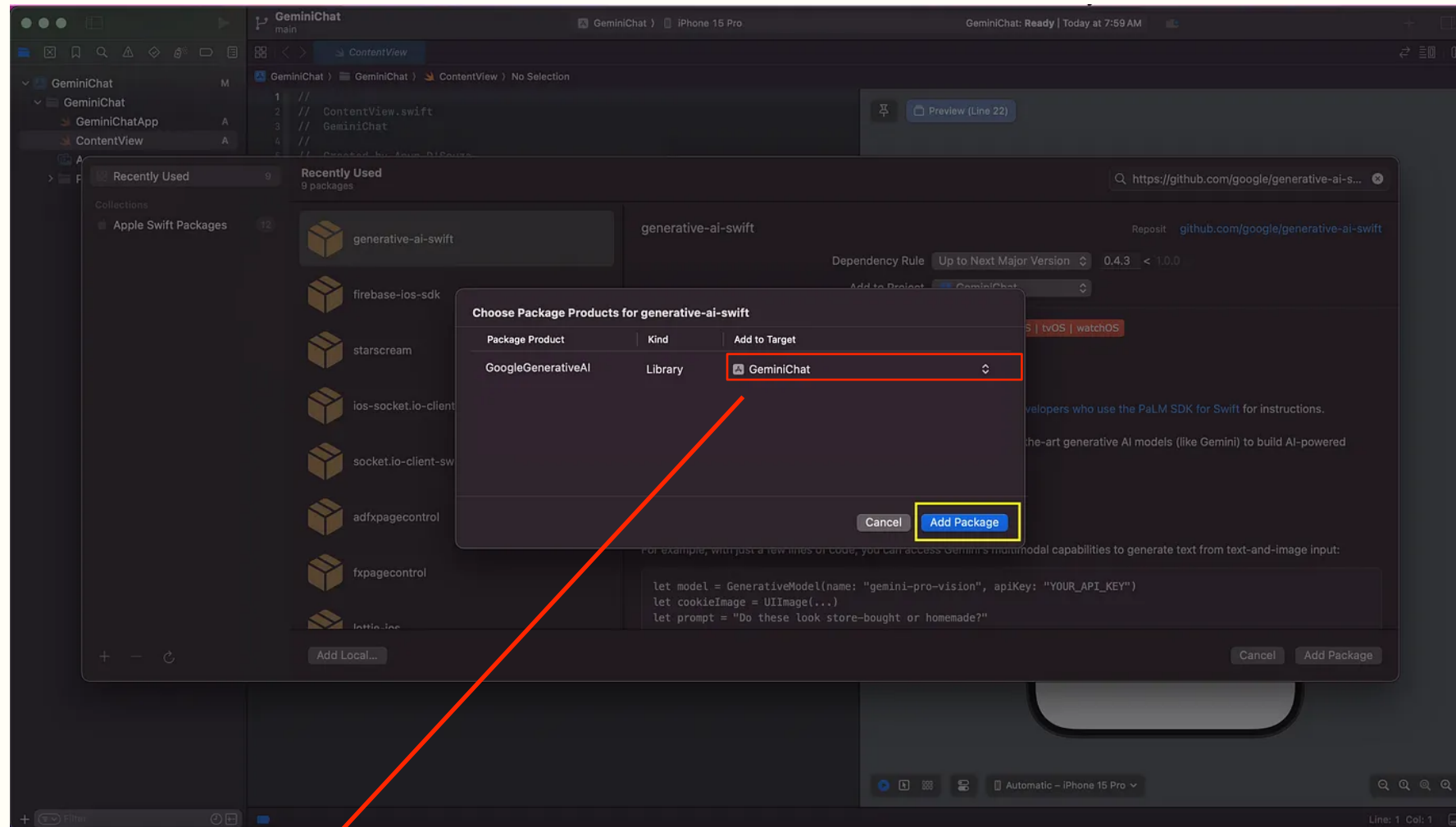


1) Adicionar dependência

<https://github.com/google/generative-ai-swift>



1) Adicionar dependência

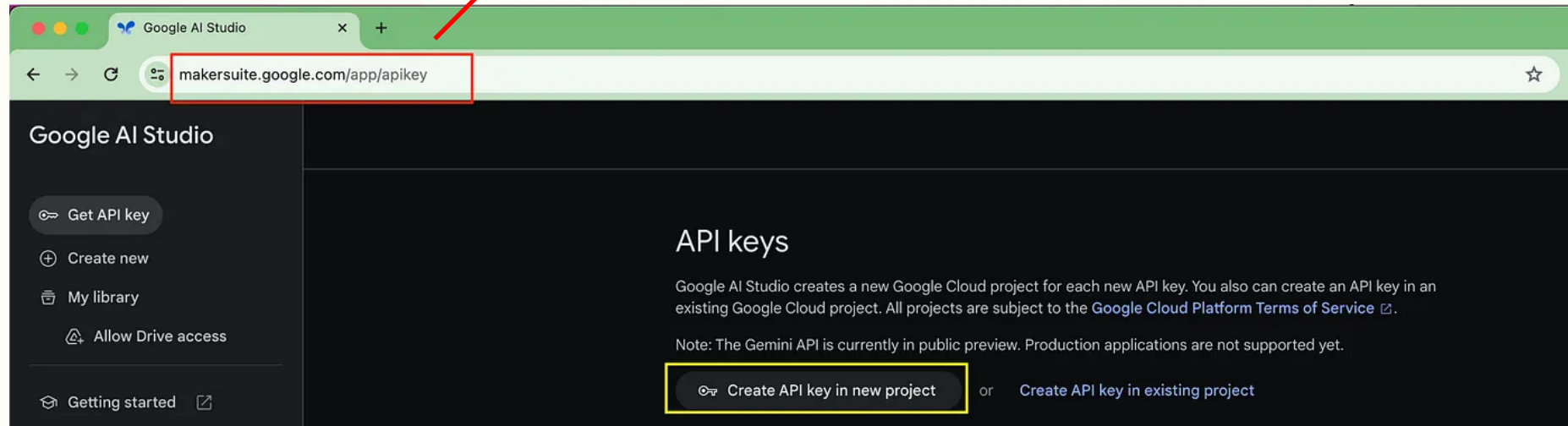


Verificar se corresponde ao seu projeto

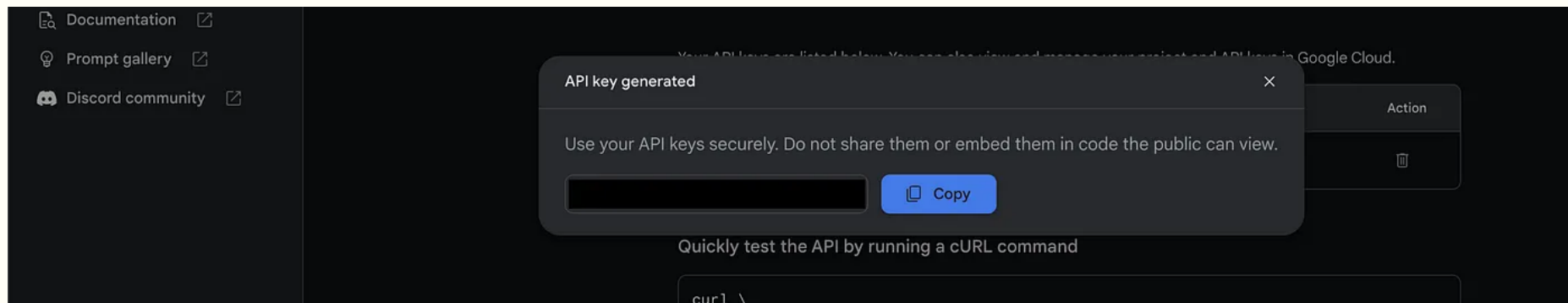
2) Criar uma API KEY

<https://makersuite.google.com/app/apikey>

i)



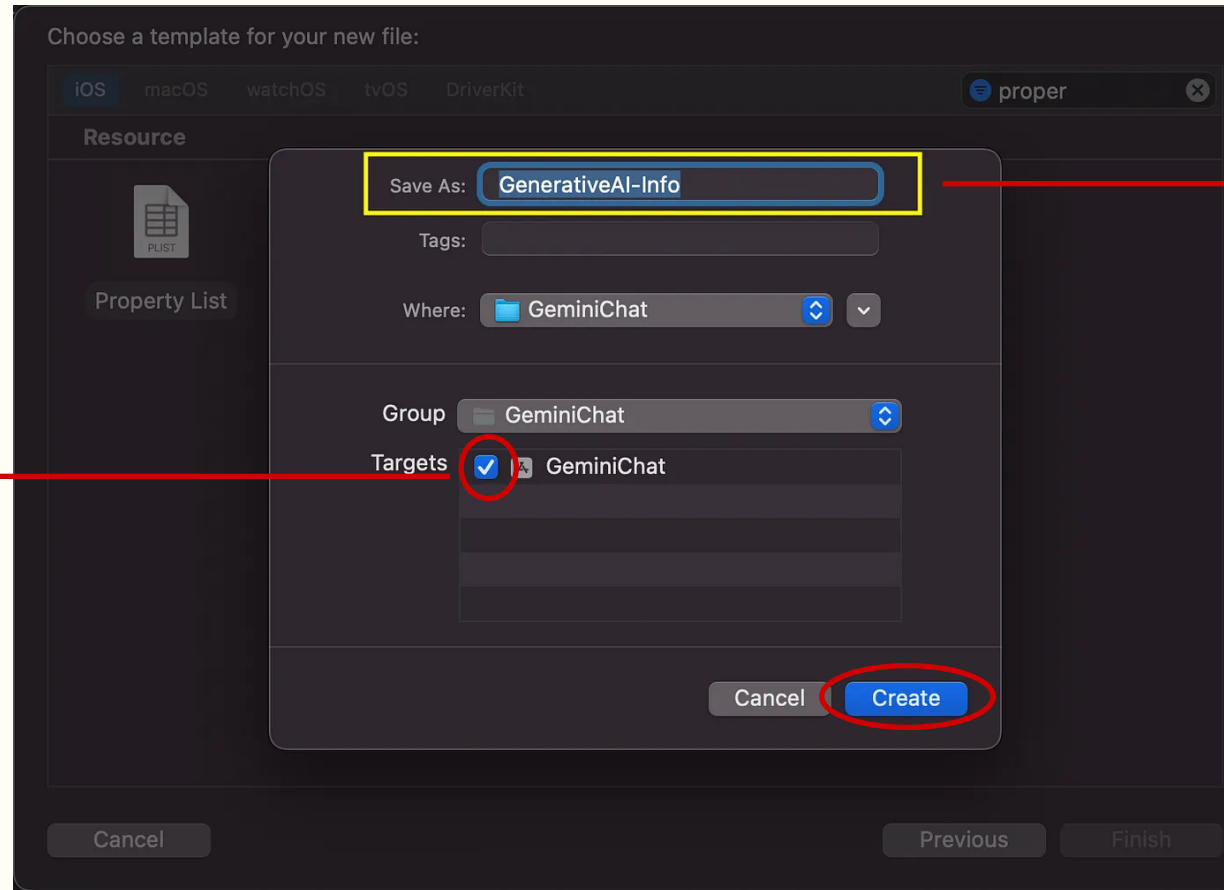
ii)



3) Registrar a API KEY na Property List

File > New File > Property List

Não esqueça
de selecionar o
projeto



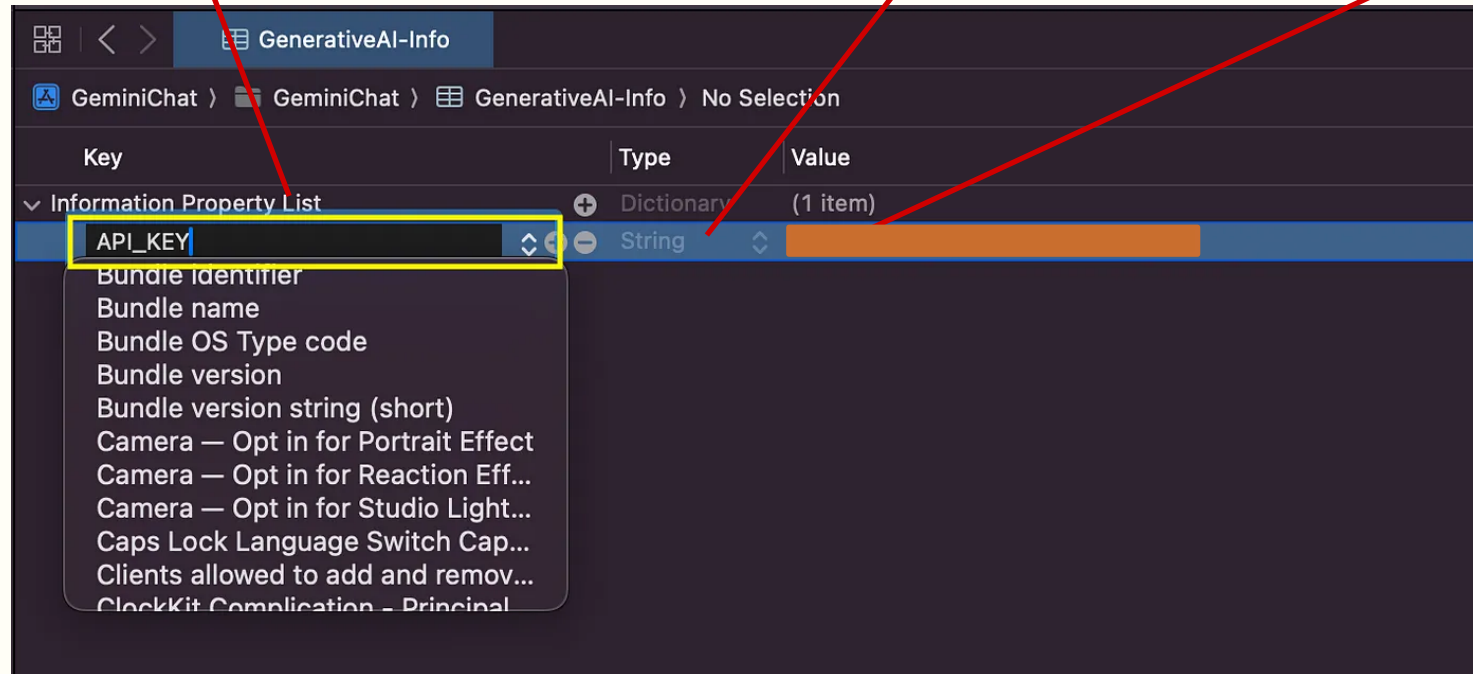
Nomeie como
GenerativeAI-Info

3) Registrar a API KEY na Property List

Nomeie a chave de “API_KEY”

Escolha o tipo como “String”

Cole sua chave



3) Validando a API KEY no projeto

Crie um arquivo *Swift File* chamado **APIKey** e escreva o seguinte código:


```
import Foundation

enum APIKey {

    static var `default`: String {
        guard let filePath = Bundle.main.path(forResource: "GenerativeAI-Info", ofType: "plist")
        else {
            fatalError("Couldn't find file 'GenerativeAI-Info.plist'.")
        }
        let plist = NSDictionary(contentsOfFile: filePath)
        guard let value = plist?.object(forKey: "API_KEY") as? String else {
            fatalError("Couldn't find key 'API_KEY' in 'GenerativeAI-Info.plist'.")
        }
        if value.starts(with: "_") {
            fatalError("Follow the instructions at https://ai.google.dev/tutorials/setup to get an API key")
        }
        return value
    }
}
```


4) Chamar a API no arquivo de View

Verificar se está nessa versão



```
import SwiftUI
import GoogleGenerativeAI

struct ContentView: View {

    let model = GenerativeModel(name: "gemini-2.0-flash", apiKey: APIKey.default)
    @State var textInput = ""
    @State var aiResponse = "Hello! How can I help you today?"
```

4) Chamar a API no arquivo de View

```
func sendMessage() {
    aiResponse = ""

    Task {
        do {
            let response = try await model.generateContent(textInput)

            guard let text = response.text else {
                textInput = "Sorry, I could not process that.\nPlease try again."
                return
            }

            textInput = ""
            aiResponse = text

        } catch {
            aiResponse = "Something went wrong!\n\(error.localizedDescription)"
        }
    }
}
```



/hackatruck



@hackatruck_makerspace



HACKATRUCK MAKERSPACE



www.hackatruck.com.br

