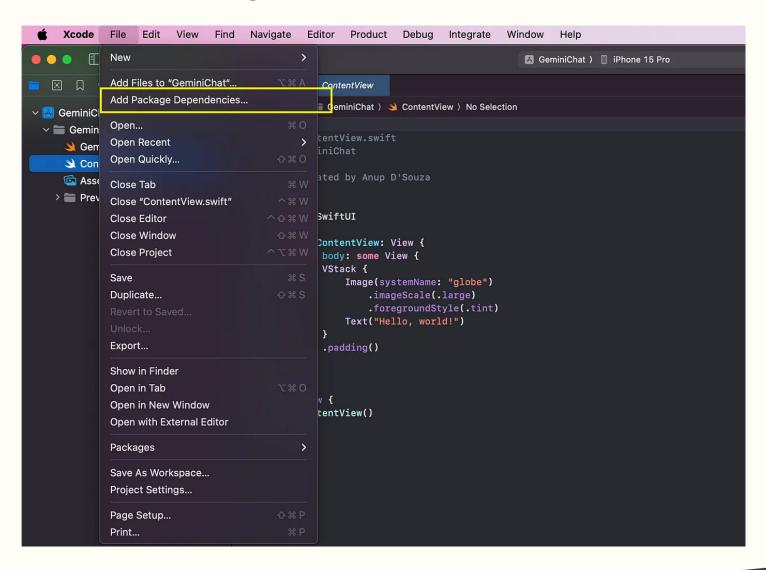
Gemini API Xcode

Tutorial

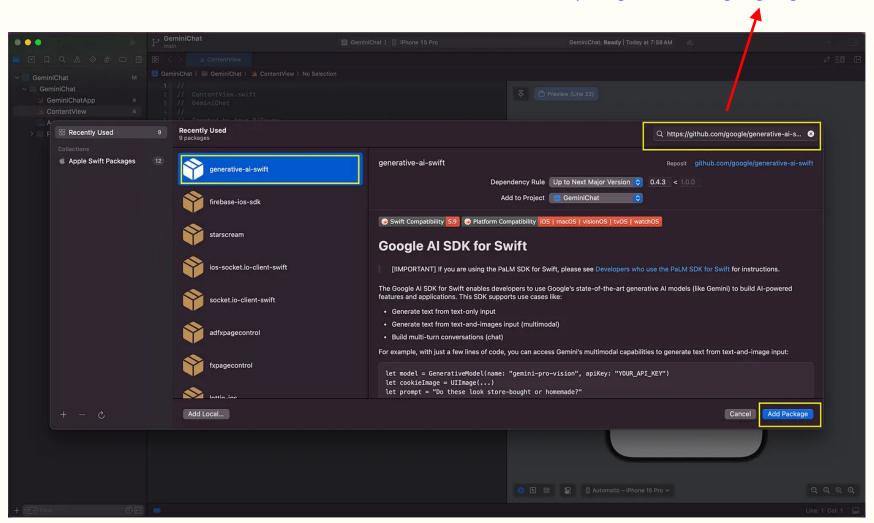


1) Adicionar dependência

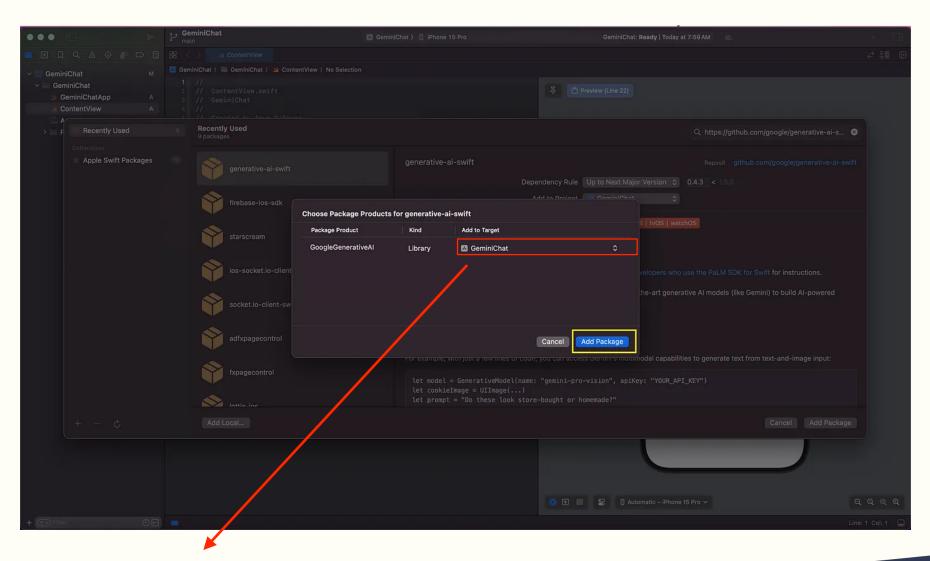


1) Adicionar dependência

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



1) Adicionar dependência



2) Criar uma API KEY

ii)

https://makersuite.google.com/app/apikey Se Google Al Studio makersuite.google.com/app/apikey Google Al Studio ⊙ Get API key API keys ① Create new Google Al Studio creates a new Google Cloud project for each new API key. You also can create an API key in an **一 My library** existing Google Cloud project. All projects are subject to the Google Cloud Platform Terms of Service 2. Allow Drive access Note: The Gemini API is currently in public preview. Production applications are not supported yet. ⊙ Create API key in new project or Create API key in existing project Getting started
 ☑ Documentation Google Cloud. API key generated Discord community Use your API keys securely. Do not share them or embed them in code the public can view.

□ Сору

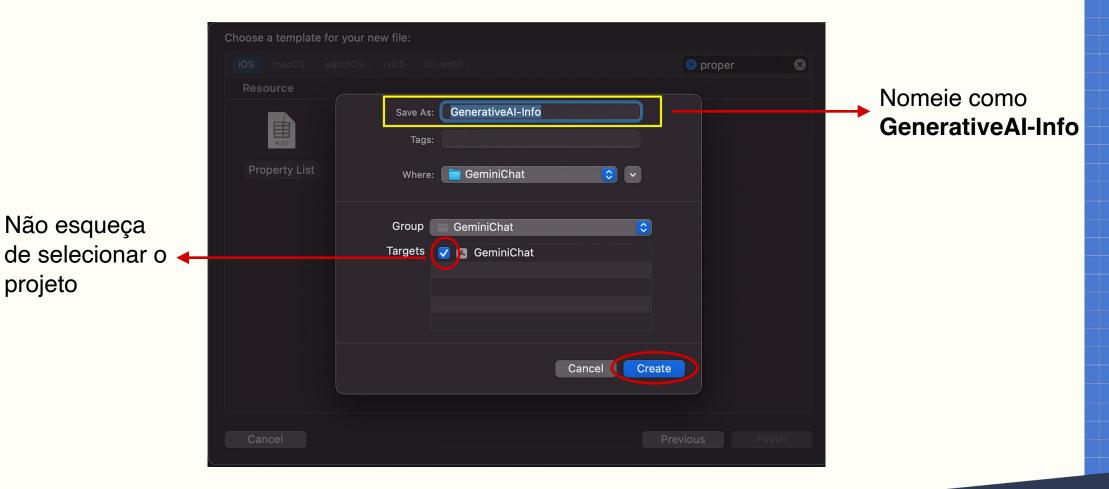
Quickly test the API by running a cURL command



3) Registrar a API KEY na Property List

projeto

File > New File > Property List



3) Registrar a API KEY na Property List

Nomeie a chave de "API_KEY" Cole sua chave Escolha o tipo como "String" ☐ GenerativeAl-Info Value Key Type ∨ Information Property List (1 item) API_KEY Bundle identifier Bundle name Bundle OS Type code **Bundle version** Bundle version string (short) Camera — Opt in for Portrait Effect Camera — Opt in for Reaction Eff... Camera — Opt in for Studio Light... Caps Lock Language Switch Cap... Clients allowed to add and remov... ClockKit Complication - Principal

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_makerspace



HACKATRUCK MAKERSPACE



www.hackatruck.com.br

