

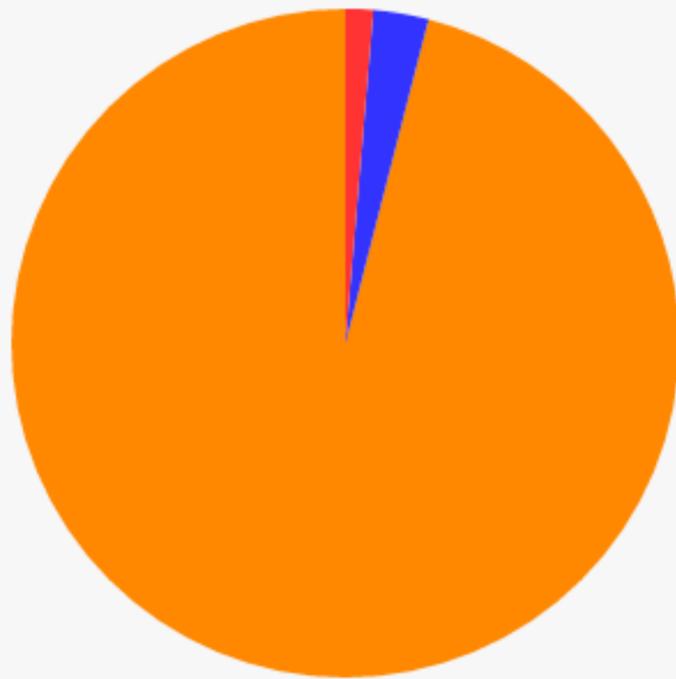
# Mobile vs AI

20.03.2025

# What used to happen

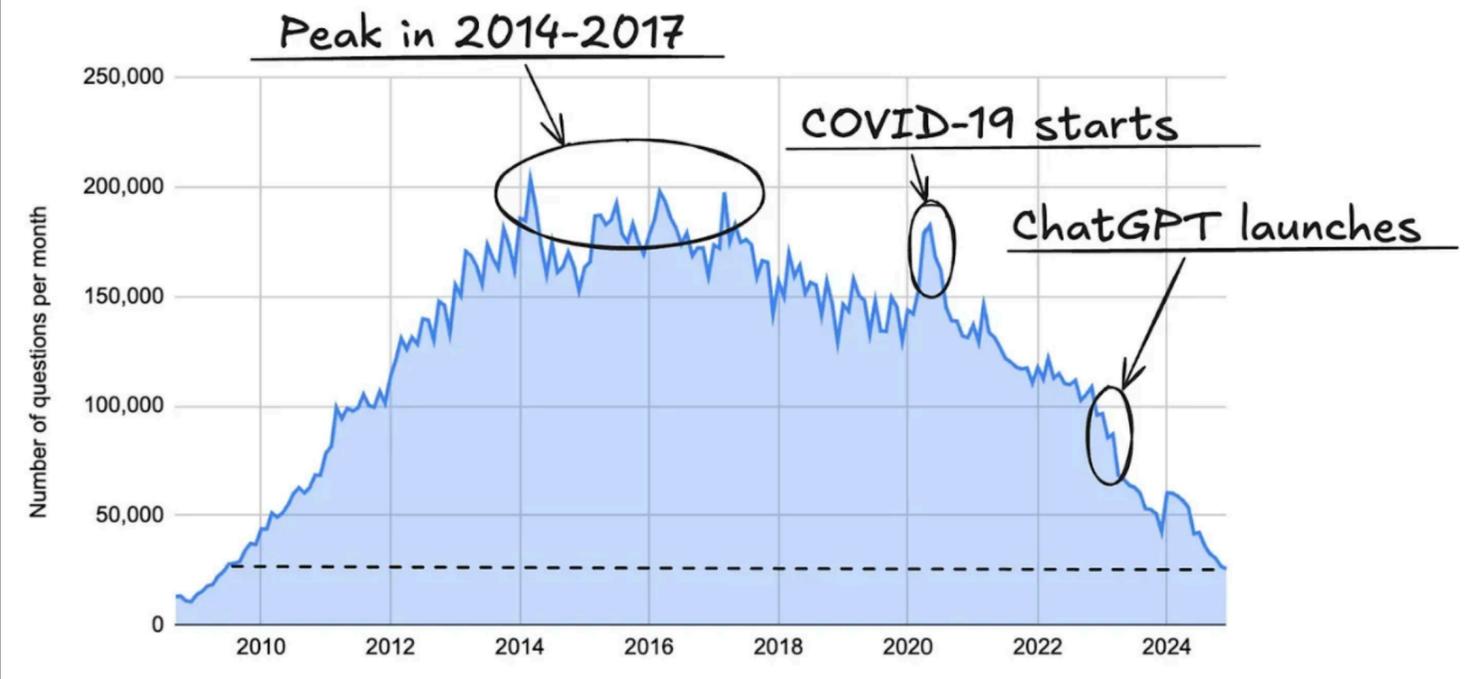
## Programming help

- Ask other people
- Textbook
- Stack overflow



imgflip.com

## Monthly questions asked on StackOverflow



# Young generation of devs



Doesn't know what stackoverflow is!

# AI assistants

- Github Copilot (both)
- Cursor (both)
- Gemini (Android)
- Android Studio - loads of AI plugins
- XCode - only a handful so far

*“Mobile is a little bit behind compared to others”*

# Picking an agent

Good to try new ones, few are actually comfortable to use.

- Integration with IDE
- Ability to detect technology

```
import Foundation
// write a compilable bubblesort method
// MARK: - CodeAI Output
public class BubbleSort {

    public static void bubbleSort(int[] arr) {
        int n = arr.length;
        boolean swapped;

        for (int i = 0; i < n - 1; i++) {
            swapped = false;
            for (int j = 0; j < n - i - 1; j++) {
                if (arr[j] > arr[j + 1]) {
```

# Legal side

When using for enterprise solution, read through T&C

of the American Arbitration Association. Changes to the Terms GreekApp LLC reserves the right to change these Terms at any time, with or without notice. Users are responsible for regularly reviewing these Terms. Continued use of the App Governing Law and Dispute

# Reviews from engineers

- Most are using Copilot. Those who don't have paid version, use their private accounts (re: privacy concerns)
- Most common use-cases:
  - Code completion - 🤔 - takes too many attempts to get it right.
  - Code generation for models - ✅ - needs API definition
  - Unit tests - 🤔 - too much guessing, hardly ever compilable
  - Conversion from other language ?
    - ObjC -> Swift - no insight
    - Java -> Kotlin - partially usable
  - Q&A - 🤔 - good for creating summary of large sources, proposed solutions are questionable
  - Personal assistant - ✅ - help with emails, composing responses, large chat conversations



Neo: *"I suppose the most obvious question is, how can I trust you?"*

The Oracle: *"Bingo! It is a pickle, no doubt about it. The bad news is there's no way if you can really know whether I'm here to help you or not. So it's really up to you. You just have to make up your own damn mind to either accept what I'm going to tell you or reject it. Candy?"*

*-THE MATRIX RELOADED (2003)*

# So..

## What does it take to generate an app?

- Set the context with the documentation of a choice
- Come up with clear requirements
- Specify the stack
- Best to have modern hardware
- Manage your expectations

Demo

# Prompt

@Codebase @Swift @MVVM iOS @Apple UI guidelines @WeatherAPI Create a Weather app using SwiftUI and Combine and NavigationStack for navigation. I'd like it to do the following

1. Show the list of cities: New York, London, Dublin, Rome, Paris, Minsk, Istanbul, Tokyo, Pretoria and make it searchable using .searchable
2. when a city is selected redirect to the city details view. Fetch city forecast from weather API using the API Key: <your API key> and base URL @https://api.weatherapi.com/v1 and present them on the screen in the same manner as the default ios weather app.
3. add unit tests as much as possible.

# Conclusions

- Outcome: POC level solutions, potential for code generation, localization, migration.
- Mission: Share publicly the insights and solutions. AI learns from public knowledge.
- Focus: Shared responsibility - context is the key, quality context -> great outcomes. Waiting for maturity or be part of the growth.
- Fact: it's already widely used, focus on HOW rather than IF.

# Bonus

- OpenAI integration with JIRA confluence:
- Full code: <https://github.com/jardon-u/chat>

## Stack:

- Chainlit for frontend
- Claude Sonnet 3.7 deployed on AWS bedrock
- Atlassian python SDK

The image shows a chat interface on the left and its corresponding Python code on the right. The chat interface displays a user query: "Who is the CEO of Arlo?". The AI response is: "I'll look that up for you in Arlo's Confluence." Below this, it indicates the tool used: "Used confluence\_search". The final response is: "Based on the information found in Arlo's Confluence, the CEO of Arlo is Matthew McRae (also appears as Matt McRae in some documents). This information appears in multiple documents including corporate communications policies and department listings where he's identified as 'Chief Executive Officer.'".

The Python code on the right is a script for a chat application. It imports necessary libraries like chainlit, boto3, os, json, and the Atlassian Confluence SDK. It sets up a Bedrock client and a Confluence client. The main logic is in the `converse` function, which handles the chat flow, including tool calls for searching Confluence. The `confluence_search` tool is defined as an asynchronous function that uses the Confluence SDK to search for documents. The `process_tools` function processes the results of tool calls and updates the chat messages. The `on_message` function is decorated with `@chainlit.on_message` and handles the overall chat logic, including sending responses and processing tool calls.

**Thank you!**

*Any questions?*