

# Đề thi thực tập sinh Golang tại Techmaster

## Mục tiêu

Dự án mà bạn ứng tuyển đang phát triển ứng dụng di động học nhiều ngoại ngữ. Nhiệm vụ của lập trình viên Golang là xây dựng phần mềm backend sử dụng framework [Iris](#), cơ sở dữ liệu quan hệ [Postgresql](#) kết nối vào các dịch vụ AI như OpenAI, Claude, Gemini, DeepSeek, Grog, Azure, Quwen để tự động hóa việc:

- Tạo ra nội dung text, ảnh, audio xử lý
- Nhận kết quả trả về, xử lý cho phù hợp định dạng
- Lưu trữ dữ liệu vào cơ sở dữ liệu
- Tạo ra các API để phần mềm di động gọi

Công việc của bạn luôn phải hình dung ra chức năng càng chi tiết càng tốt, viết prompt bằng tiếng Việt hay Anh tùy thích, gửi lên cho AI sinh code. Bạn chạy thử code, nếu chưa đúng ý yêu cầu AI sửa lại, chạy thử, tự bạn chỉnh sửa code (bạn vốn là lập trình viên mà) cho đến khi đúng yêu cầu.

## Hướng dẫn làm đề thực hành

1. Bạn thoải mái hỏi AI để tìm giải pháp!
2. Thời hạn nộp bài là 6 ngày kể từ ngày nhận được đề thi. Cần gì cứ nhắn Zalo cho mình 0902209011 .
3. Code bạn cần đẩy lên github repository chia thành các thư mục con:
  - 01
  - 02
  - 03
  - ...
  - [ReadMe.md](#)

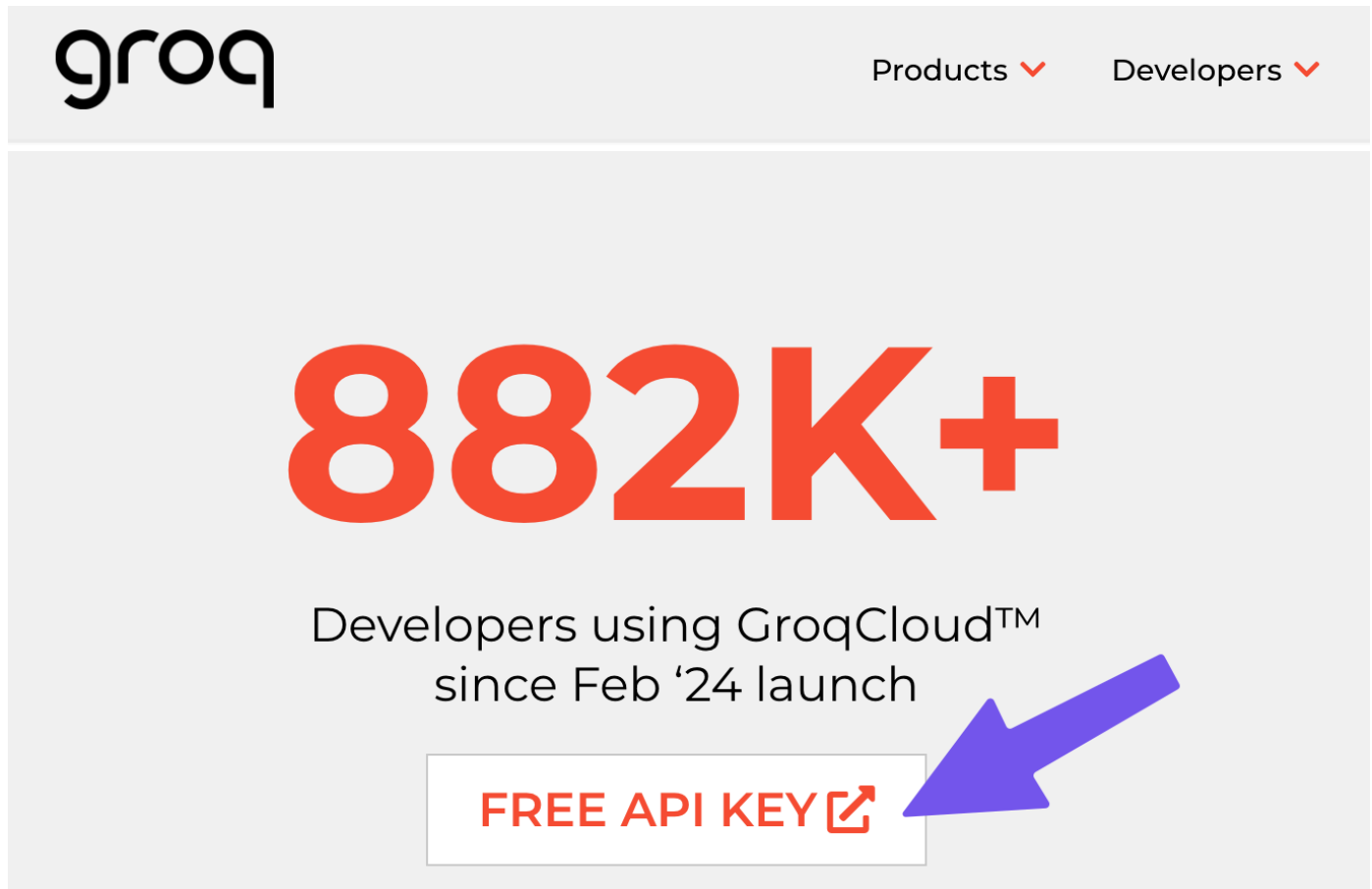
File [ReadMe.md](#) là file giới thiệu về dự án, cách chạy, các bước thực hiện. Hãy chụp ảnh, video minh họa để dễ dàng cho người khác đọc hiểu.

# Câu 1: Gọi vào Groq API

Groq là dịch vụ AI cung cấp API cho phép gọi đến các model như:

- Llama 3.1
- GPT-4o
- Claude 3.5 Sonnet
- Gemini 1.5 Pro

Hiện nay Groq cung cấp dịch vụ gọi vào API miễn phí.



The screenshot shows the Groq website header with the logo and navigation links for 'Products' and 'Developers'. The main content area features a large red '882K+' statistic, followed by the text 'Developers using GroqCloud™ since Feb '24 launch'. Below this is a button labeled 'FREE API KEY' with an external link icon. A large blue arrow points to the button.

Hãy viết ứng dụng web Golang dùng Iris framework có một ô text area cho người dùng viết prompt, nút gửi đi, ứng dụng gọi vào Groq API và trả về kết quả cho người dùng xem. Chú ý kết quả trả về dạng Markdown do đó cần hiển thị sao cho đẹp (có nghĩa bạn phải parse Markdown response).

7.9 or 7.11 which number is bigger?

Gửi đi

To determine which number is bigger between **7.9** and **7.11**, let's compare them step by step.

**1. Align the Decimal Places:**

7.90    and    7.11

(Note: We write **7.9** as **7.90** to have the same number of decimal places for easier comparison.)

**2. Compare the Whole Number Part:**

- Both numbers have the same whole number part, which is **7**.

**3. Compare the Tenths Place:**

- In **7.90**, the tenths place is **9**.
- In **7.11**, the tenths place is **1**.

Since **9** (from **7.90**) is greater than **1** (from **7.11**), we can conclude that **7.90** is larger than **7.11**.

## Câu 2: Sinh file SSML từ hội thoại

Bài này bạn chỉ cần dùng HTML và JavaScript thuần để code một trang web. Bạn có một danh sách các voice có sẵn:

- en-US-AndrewMultilingualNeural
- en-US-ChristopherNeural
- en-US-EricNeural
- vi-VN-HoaiMyNeural
- vi-VN-NamMinhNeural

bạn chọn 2 voice để sinh file SSML từ hội thoại gốc

A: Chào Lan! Mình là James, đến từ Hoa Kỳ. Rất vui được gặp bạn.  
B: Chào James! Mình là Lan, đến từ Việt Nam. Rất vui được làm quen với bạn.  
A: Bạn làm nghề gì vậy, Lan?  
B: Mình là cô giáo dạy ngoại ngữ. Còn bạn?  
A: Mình là kỹ sư hàng không.  
B: Nghe thú vị quá! Bạn đến Việt Nam lâu chưa?  
A: Mình mới đến đây được vài ngày.  
B: Hy vọng bạn sẽ thích Việt Nam!  
A: Cảm ơn Lan!

File SSML (Speech Synthesis Markup Language) cần được được sinh ra từ hội thoại trên:

```
<speak xml:lang="vi-VN"><voice name="en-US-AndrewMultilingualNeural">Chào Lan! Mình là  
<voice name="vi-VN-HoaiMyNeural">Chào James! Mình là Lan, đến từ Việt Nam. Rất vui được  
<voice name="en-US-AndrewMultilingualNeural">Bạn làm nghề gì vậy, Lan?</voice>  
<voice name="vi-VN-HoaiMyNeural">Mình là cô giáo dạy ngoại ngữ. Còn bạn?</voice>  
<voice name="en-US-AndrewMultilingualNeural">Mình là kỹ sư hàng không.</voice>  
<voice name="vi-VN-HoaiMyNeural">Nghe thú vị quá! Bạn đến Việt Nam lâu chưa?</voice>  
<voice name="en-US-AndrewMultilingualNeural">Mình mới đến đây được vài ngày.</voice>  
<voice name="vi-VN-HoaiMyNeural">Hy vọng bạn sẽ thích Việt Nam!</voice>  
<voice name="en-US-AndrewMultilingualNeural">Cảm ơn Lan!</voice></speak>
```

Giao diện như hình dưới đây:

Voice A ▼

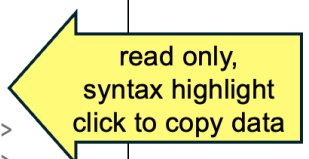
Voice B ▼

Xuất SSML

A: Chào Lan! Mình là James, đến từ Hoa Kỳ. Rất vui được gặp bạn.  
B: Chào James! Mình là Lan, đến từ Việt Nam. Rất vui được làm quen với bạn.  
A: Bạn làm nghề gì vậy, Lan?  
B: Mình là cô giáo dạy ngoại ngữ. Còn bạn?  
A: Mình là kỹ sư hàng không.  
B: Nghe thú vị quá! Bạn đến Việt Nam lâu chưa?  
A: Mình mới đến đây được vài ngày.  
B: Hy vọng bạn sẽ thích Việt Nam!  
A: Cảm ơn Lan!



```
<speak xml:lang="vi-VN"><voice name="en-US-AndrewMultilingualNeural">Chào Lan! Mình là  
James, đến từ Hoa Kỳ. Rất vui được gặp bạn.</voice>  
<voice name="vi-VN-HoaiMyNeural">Chào James! Mình là Lan, đến từ Việt Nam. Rất vui được làm  
quen với bạn.</voice>  
<voice name="en-US-AndrewMultilingualNeural">Bạn làm nghề gì vậy, Lan?</voice>  
<voice name="vi-VN-HoaiMyNeural">Mình là cô giáo dạy ngoại ngữ. Còn bạn?</voice>  
<voice name="en-US-AndrewMultilingualNeural">Mình là kỹ sư hàng không.</voice>  
<voice name="vi-VN-HoaiMyNeural">Nghe thú vị quá! Bạn đến Việt Nam lâu chưa?</voice>  
<voice name="en-US-AndrewMultilingualNeural">Mình mới đến đây được vài ngày.</voice>  
<voice name="vi-VN-HoaiMyNeural">Hy vọng bạn sẽ thích Việt Nam!</voice>  
<voice name="en-US-AndrewMultilingualNeural">Cảm ơn Lan!</voice></speak>
```



# Câu 3: Tạo hội thoại từ prompt và trích xuất từ mới trong hội thoại

*Học từ bài 1, chúng ta đến với một thách thức thú vị hơn một chút. Điểm bài này rất cao do đó hãy cố gắng nhé*

## 3.1 Thực hành thủ công để cảm nhận trước

**Bước 1:** Trong DeepSeek, Quwen, Groq hay Chat-GPT bạn có thể tạo ra hội thoại bằng cách gõ prompt:

Tạo một hội thoại bằng tiếng Việt, gồm 6 câu, ngắn gọn, đơn giản, hỏi đường đi đến hồ Hoàn Kiếm ở Hà nội giữa một Mỹ tên James và người Việt nam tên Lan. Chỉ xuất ra hội thoại không cần giải thích.

Kết quả trả về có dạng:

James: Chào bạn! Bạn có thể chỉ tôi đường đến hồ Hoàn Kiếm không?  
Lan: Chào bạn! Bạn đang ở đâu?  
James: Tôi đang ở phố Tràng Tiền.  
Lan: Từ đây, bạn đi thẳng, rẽ phải vào đường Đinh Tiên Hoàng, sẽ thấy hồ.  
James: Cảm ơn bạn nhiều!  
Lan: Không có gì, chúc bạn đi vui!

**Bước 2:** Hãy lưu hội thoại trên vào một bảng có tên là `dialog` trong cơ sở dữ liệu Postgresql.

**Bước 3:** Sau đó bạn lại gõ tiếp prompt:

Từ hội thoại trên hãy lọc ra danh sách các từ quan trọng, bỏ qua danh từ tên riêng cần học. Không cần giải thích xuất kết quả ra dạng JSON trong thẻ ``words``.

Kết quả trả về có dạng:

```
{  
  "words": [  
    "chào",  
    "bạn",  
    "chỉ",  
    "đường",  
    "hồ",  
    "đang",  
    "phố",  
    "đi thẳng",  
    "rẽ phải",  
    "thấy",  
    "cảm ơn",  
    "không có gì",  
    "chúc",  
    "vui"  
  ]  
}
```

**Bước 4:** Viết tiếp prompt để dịch các từ trong json sang tiếng Anh:

Dịch từng từ trong danh sách dưới sang tiếng Anh rồi trả JSON  
gồm mảng trong đó mỗi phần tử sẽ gồm từ tiếng Việt và từ  
tiếng Anh tương đương. Không cần giải thích.

Kết quả trả về có dạng:

```

{
  "translated_words": [
    { "vi": "chào", "en": "hello" },
    { "vi": "bạn", "en": "you" },
    { "vi": "chỉ", "en": "show" },
    { "vi": "đường", "en": "road" },
    { "vi": "hồ", "en": "lake" },
    { "vi": "đang", "en": "currently" },
    { "vi": "phố", "en": "street" },
    { "vi": "đi thẳng", "en": "go straight" },
    { "vi": "rẽ phải", "en": "turn right" },
    { "vi": "thấy", "en": "see" },
    { "vi": "cảm ơn", "en": "thank you" },
    { "vi": "không có gì", "en": "you're welcome" },
    { "vi": "chúc", "en": "wish" },
    { "vi": "vui", "en": "happy" }
  ]
}

```

Hãy lưu danh sách từ vào bảng có tên là `word` trong cơ sở dữ liệu PostgreSQL.

## 3.2 Tự động hóa toàn bộ các bước thủ công trên

Hãy lập trình một ứng dụng web dùng Iris framework gọi vào Groq API, model `deepseek-r1-distill-llama-70b` để tự động hóa toàn bộ các bước thủ công trên.

Bạn nên hiển thị kết quả mỗi bước ra giao diện web sau đó lưu vào các bảng trong cơ sở dữ liệu PostgreSQL.

### 3.3 Cấu trúc bảng cho sẵn để bạn tạo

```
-- Create dialog table
CREATE TABLE dialog (
    id BIGSERIAL PRIMARY KEY,
    lang VARCHAR(2) NOT NULL, //vi: Vietnamese, en: English
    content TEXT NOT NULL //Lưu toàn bộ nội dung hội thoại
);

-- Create word table
CREATE TABLE word (
    id BIGSERIAL PRIMARY KEY,
    lang VARCHAR(2) NOT NULL, //vi: Vietnamese, en: English
    content TEXT NOT NULL, //Lưu gốc
    translate TEXT NOT NULL //Lưu dịch ra tiếng Anh
);

-- Bảng trung gian quan hệ nhiều nhiều giữa word và dialog
CREATE TABLE word_dialog (
    dialog_id BIGINT REFERENCES dialog(id) ON DELETE CASCADE,
    word_id BIGINT REFERENCES word(id) ON DELETE CASCADE,
    PRIMARY KEY (dialog_id, word_id)
);
```