

ZAX-Code Assistant

your 24/7 coding companion

Presented by:

Tarni

Akanchha

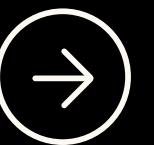
Shweta



PROBLEM STATEMENT:

- **Challenges:**
 - Difficulty in quickly finding accurate coding solutions.
 - Time-consuming and frustrating debugging processes.
 - Challenges in understanding different problem-solving approaches.
 - Limited support for multiple programming languages in a single tool.
- **Impact:**
 - Reduced productivity.
 - Slower learning curve for students.
 - Increased time spent on repetitive tasks.

SOLUTION:



Overview:

Zax Code Assistant is designed to assist with code generation, debugging, and providing explanations for coding problems.



Core Capabilities:

Accepts code-related queries as prompts.

Capable of generating code, debugging, and explaining intuitions.

Supports multiple programming languages.

Functionality:

Input: User inputs a coding query.

AI Processing: Utilizes Mistral's AI API to process the query.

Output: Generates accurate code, debugs existing code, and provides explanatory comments.



TECH-STACK

Backend: Python, Flask, Mistral AI API

Frontend: HTML, CSS, JavaScript.

PROJECT BENEFICIARIES



Students:

- Gain a better understanding of coding problems and solutions.
- Learn different approaches and coding practices.



Educational Institutions:

- Provide a powerful tool for teaching coding.
- Enhance the learning experience for students.



Software Development Teams:

- Increase productivity and reduce time spent on debugging.
- Access to a versatile tool supporting multiple languages.

EXPECTED RESULTS:

Efficiency:

Significant reduction in time spent on coding and debugging tasks.

Learning:

Improved understanding of coding concepts for students and developers.

Productivity:

Enhanced productivity for software development teams.

Quality:

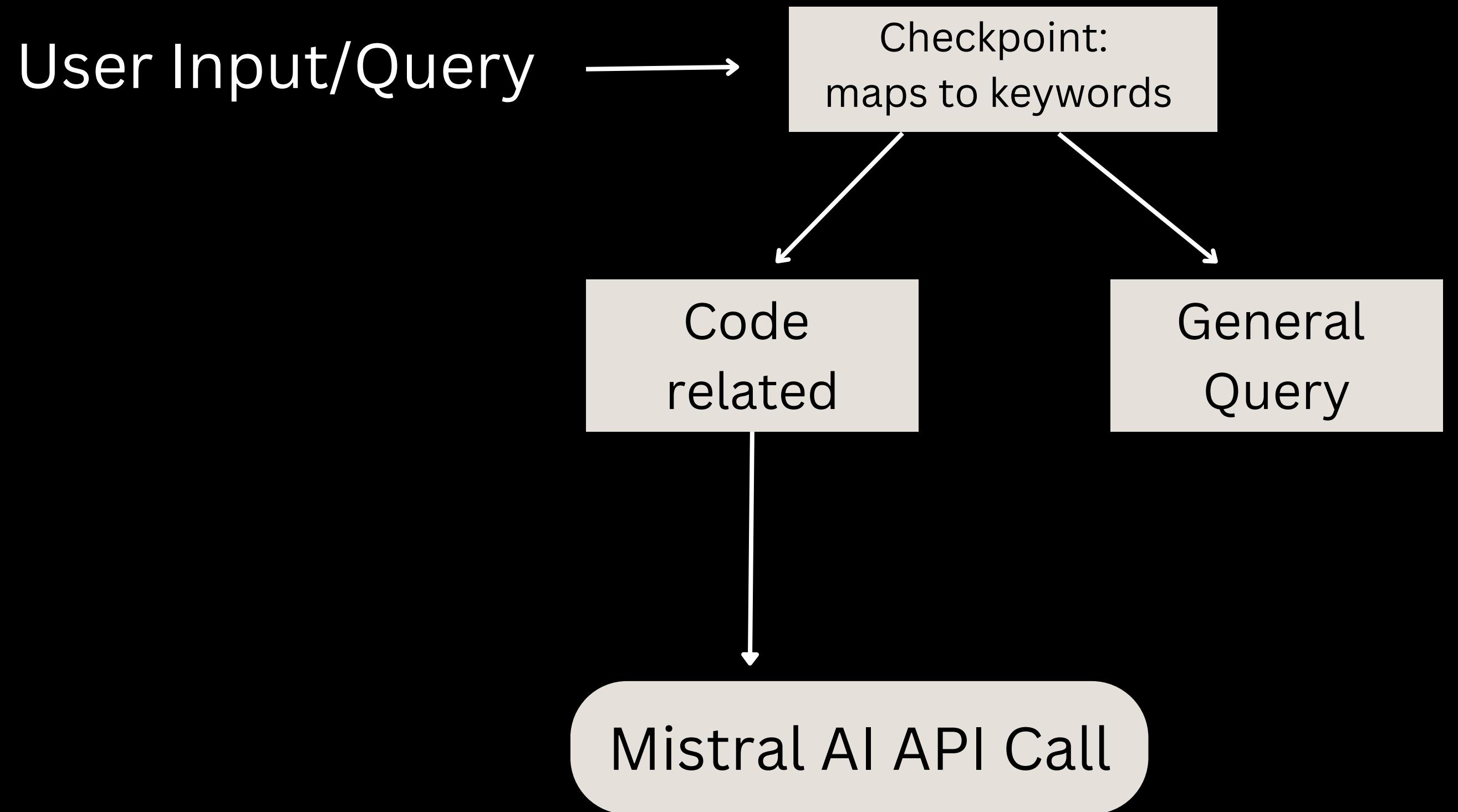
Higher quality code with fewer errors.

Versatility:

A versatile tool capable of handling multiple programming languages.



DATABASE ARCHITECTURE:





ZAX - Code Assistant

Enter your question or code:

How to calculate the fibonacci number? Code in javascript and explain the code.

Do you want to document the code?

Submit

ZAX - Code Assistant

Enter your question or code:

How to calculate the fibonacci number? Code in javascript and explain the code.

Do you want to document the code?

Submit

Response:

Sure, I'd be happy to help with that. Here's a simple JavaScript function that calculates the nth Fibonacci number:

```
javascript
function fibonacci(n) {
  if (n <= 1) {
    return n;
  } else {
    let a = 0;
    let b = 1;
```

```
let b = 1;
let temp;

for (let i = 2; i <= n; i++) {
    temp = a + b;
    a = b;
    b = temp;
}

return b;
}
```

This function works by starting with the first two Fibonacci numbers (0 and 1) and then iteratively calculating the next Fibonacci number as the sum of the previous two. It does this in a loop that runs `n` times, where `n` is the position of the Fibonacci number you want to find.

Here's a breakdown of the code:

1. The function `fibonacci` takes one argument, `n`, which is the position of the Fibonacci number you want to find.
2. If `n` is less than or equal to 1, the function returns `n` directly, because the first two Fibonacci numbers are 0 and 1.
3. If `n` is greater than 1, the function initializes two variables, `a` and `b`, to the first two Fibonacci numbers.
4. The function then enters a loop that runs `n - 1` times. In each iteration of the loop, it calculates the next Fibonacci number as the sum of `a` and `b`, and then updates `a` and `b` to be the last two Fibonacci numbers calculated so far.
5. After the loop finishes, the function returns `b`, which is the `n`th Fibonacci number.

ZAX - Code Assistant

Enter your question or code:

How to calculate the fibonacci number? Code in javascript and explain the code.

Do you want to document the code?

Submit

Response:

Title: Fibonacci Number Calculator

Description:

This JavaScript function calculates the Fibonacci number for a given index. The Fibonacci sequence is a series of numbers in which each number is the sum of the two preceding ones, usually starting with 0 and 1.

Function/Method:

- `fibonacci(n)`

each number is the sum of the two preceding ones, usually starting with 0 and 1.

Function/Method:

- `fibonacci(n)`

Parameters:

- `n`: A non-negative integer representing the index of the Fibonacci sequence.

Return Values:

- The function returns the Fibonacci number at the given index.

Example Usage:

```
javascript
console.log(fibonacci(0)); // Output: 0
console.log(fibonacci(1)); // Output: 1
console.log(fibonacci(2)); // Output: 1
console.log(fibonacci(3)); // Output: 2
console.log(fibonacci(10)); // Output: 55
```

ADVANTAGES



INCREASED ACCESSIBILITY

- Multi-language support , beginner accessibility

FOSTER INNOVATION

- Fosters innovation and collaboration among newbies as well as advanced coders

BOOSTS INNOVATION

- Shifts the focus to solve more technical and creative problem-solving

LEARNING TOOL

- For newcomers and advanced coders alike



TEAM

TARNI

SHWETA

AKANCHHA



THANK YOU



Github : <https://github.com/Akanchha12/ZAXAssistant>

Deployed Link : <https://zax-assistant-53ty.vercel.app/>