

# AzureAISearchService

September 4, 2026

## AzureAISearchService

**Kind:** Service

**Source:** [atloria-monorepo/apps/api/src/web-search/services/azure-ai-search.service.ts](#)

Azure AI Search Service  
Queries the indexed competitor documentation

`AzureAISearchService` is a NestJS backend service that queries the Azure AI Search index containing competitor documentation. It provides search, category-filtered search, document lookup by URL, and health-check capabilities for web-search consumers.

### Methods

| Method                        | Signature                                                       | Returns                                            | Description                            |
|-------------------------------|-----------------------------------------------------------------|----------------------------------------------------|----------------------------------------|
| <code>search</code>           | <code>search(query: SearchQuery)</code>                         | <code>Promise&lt;CompetitorSearchResult&gt;</code> | Search competitor documentation        |
| <code>searchByCategory</code> | <code>searchByCategory(category: string, limit: unknown)</code> | <code>Promise&lt;SearchResult[]&gt;</code>         | Search by category                     |
| <code>getDocumentByUrl</code> | <code>getDocumentByUrl(url: string)</code>                      | <code>Promise&lt;SearchResult&gt;</code>           | <code>null</code>                      |
| <code>healthCheck</code>      | <code>healthCheck()</code>                                      | <code>Promise&lt;boolean&gt;</code>                | Check if the search service is healthy |

### Where it refuses work

- `AzureAISearchService` stops the work with an early return when `!this.searchClient`, in 2 places.

### When something fails

- `AzureAISearchService` handles failure in 3 places: it turns it into a return value in all 3.

## Diagram

```
sequenceDiagram
    participant Client
    participant Consumer as Controller/Service Consumer
    participant Search as AzureAISearchService
    participant Azure as Azure AI Search Index

    Client->>Consumer: Request competitor documentation
    Consumer->>Search: search(query)
    Search->>Azure: Execute indexed document query
    Azure-->>Search: Matching search results
    Search-->>Consumer: CompetitorSearchResult
    Consumer-->>Client: Search response

    Note over Consumer,Search: Other supported operations:<br/>searchByCategory(), getDocumentByUrl
```

## Usage

```
import { Injectable } from '@nestjs/common';
import { AzureAISearchService } from '../azure-ai-search.service';

@Injectable()
export class CompetitorResearchService {
  constructor(
    private readonly azureAISearchService: AzureAISearchService,
  ) {}

  async findAuthenticationDocs() {
    const results = await this.azureAISearchService.search('authentication');

    return results;
  }

  async findDocument(url: string) {
    const document = await this.azureAISearchService.getDocumentByUrl(url);

    if (!document) {
      return null;
    }

    return document;
  }

  async isSearchAvailable(): Promise<boolean> {
    return this.azureAISearchService.healthCheck();
  }
}
```

## AI Coding Instructions

- Inject `AzureAISearchService` through NestJS dependency injection; do not instantiate it directly.
- Use `search()` for general competitor-documentation queries and `searchByCategory()` when the caller already has a category constraint.
- Handle `getDocumentByUrl()` returning `null` when the indexed URL cannot be found.
- Use `healthCheck()` for readiness or dependency monitoring rather than issuing a full search request.
- Treat returned search results as indexed data that may be stale or incomplete; validate URLs and metadata before using them in downstream workflows.