
Tourist agency consultant

Release '1.0.0'

edikis, Cherry-pashka, DmitryKRX, 1makeyourdreams1

May 02, 2025

CONTENTS

1	Tourist_agency_consultant	1
1.1	database module	1
1.2	faiss_service module	1
1.3	main module	2
1.4	prompts module	2
1.5	tour_agent module	2
1.6	utils module	2
	Python Module Index	3
	Index	5

TOURIST_AGENCY_CONSULTANT

1.1 database module

```
class database.Database
    Bases: object
    async create_pool()
    async register_user_if_not_exists(user)
    async save_feedback(user, feedback_text)
```

1.2 faiss_service module

```
class faiss_service.CreateIndexRequest(*, key: str, dimension: int)
    Bases: BaseModel
    dimension: int
    key: str
    model_config: ClassVar[ConfigDict] = {}
    Configuration for the model, should be a dictionary conforming to [ConfigDict][pydantic.config.ConfigDict].

class faiss_service.SearchQuery(*, key: str, query: List[float], k: int = 5)
    Bases: BaseModel
    k: int
    key: str
    model_config: ClassVar[ConfigDict] = {}
    Configuration for the model, should be a dictionary conforming to [ConfigDict][pydantic.config.ConfigDict].
    query: List[float]

async faiss_service.add_vectors(key: str, vectors: List[List[float]])
    Add vectors to index

async faiss_service.create_index(request: CreateIndexRequest)
    Create new index
```

```
async faiss_service.search(query: SearchQuery)
```

Find nearest neighbors in the specified index

1.3 main module

```
class main.FeedbackState
```

Bases: StatesGroup

```
waiting_feedback = <State 'FeedbackState:waiting_feedback'>
```

```
async main.feedback_request(message: Message, state: FSMContext)
```

```
async main.handle_message(message: Message)
```

```
async main.on_startup(_)
```

```
async main.save_feedback(message: Message, state: FSMContext)
```

```
async main.start(message: Message)
```

1.4 prompts module

1.5 tour_agent module

```
class tour_agent.AgentAPI(agent: LangGraphAgent)
```

Bases: object

```
class tour_agent.LangGraphAgent
```

Bases: object

```
assistant(state)
```

1.6 utils module

```
utils.print_messages(messages)
```

PYTHON MODULE INDEX

d

database, [1](#)

f

faiss_service, [1](#)

m

main, [2](#)

p

prompts, [2](#)

t

tour_agent, [2](#)

u

utils, [2](#)

INDEX

A

`add_vectors()` (in module `faiss_service`), 1
`AgentAPI` (class in `tour_agent`), 2
`assistant()` (`tour_agent.LangGraphAgent` method), 2

C

`create_index()` (in module `faiss_service`), 1
`create_pool()` (`database.Database` method), 1
`CreateIndexRequest` (class in `faiss_service`), 1

D

`database`
 module, 1
`Database` (class in `database`), 1
`dimension` (`faiss_service.CreateIndexRequest` attribute), 1

F

`faiss_service`
 module, 1
`feedback_request()` (in module `main`), 2
`FeedbackState` (class in `main`), 2

H

`handle_message()` (in module `main`), 2

K

`k` (`faiss_service.SearchQuery` attribute), 1
`key` (`faiss_service.CreateIndexRequest` attribute), 1
`key` (`faiss_service.SearchQuery` attribute), 1

L

`LangGraphAgent` (class in `tour_agent`), 2

M

`main`
 module, 2
`model_config` (`faiss_service.CreateIndexRequest` attribute), 1
`model_config` (`faiss_service.SearchQuery` attribute), 1
module

`database`, 1
 `faiss_service`, 1
 `main`, 2
 `prompts`, 2
 `tour_agent`, 2
 `utils`, 2

O

`on_startup()` (in module `main`), 2

P

`print_messages()` (in module `utils`), 2
`prompts`
 module, 2

Q

`query` (`faiss_service.SearchQuery` attribute), 1

R

`register_user_if_not_exists()`
 (`database.Database` method), 1

S

`save_feedback()` (`database.Database` method), 1
`save_feedback()` (in module `main`), 2
`search()` (in module `faiss_service`), 1
`SearchQuery` (class in `faiss_service`), 1
`start()` (in module `main`), 2

T

`tour_agent`
 module, 2

U

`utils`
 module, 2

W

`waiting_feedback` (`main.FeedbackState` attribute), 2