

TECHNOLOGIE UND ARCHITEKTUR

Sirion Feingold

Recap

eonum
EVOLVING SOFTWARE



Überblick Technologien



Ruby



Ruby on
Rails



ChatGPT

Inhalt



RUBY-GEMS

+

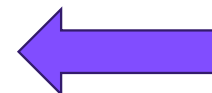
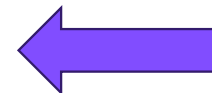


Was ist ein Ruby-Gem?

- Wiederverwendbarer Ruby-Code
- Innerhalb eines Gems:
 - Code, Dokumentation, Gemspec

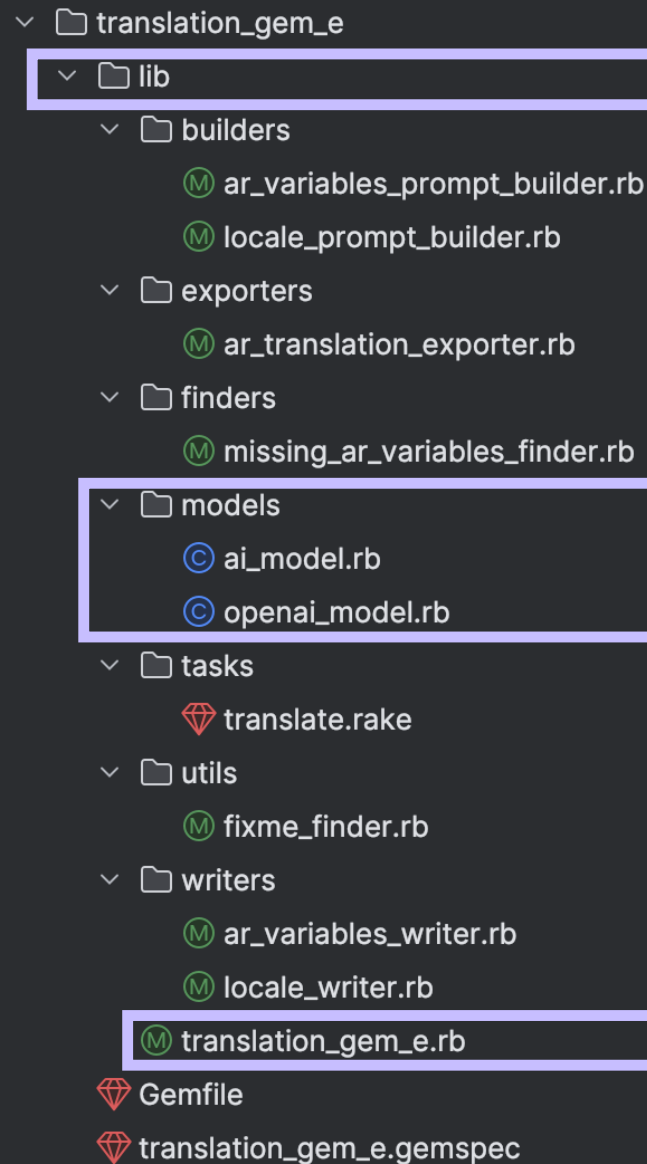
```
$ tree
```

```
.  
├── hola.gemspec  
└── lib  
    └── hola.rb
```



Architektur unseres Gems

- Single Responsibility Principle
- Generisches AI-Modell als Superklasse



The screenshot displays the file explorer of a Ruby gem project named 'translation_gem_e'. The 'lib' directory is expanded, showing subdirectories 'builders', 'exporters', 'finders', 'models', 'tasks', 'utils', and 'writers'. The 'models' directory is highlighted with a red box, containing 'ai_model.rb' and 'openai_model.rb'. The 'translation_gem_e.rb' file is also highlighted with a red box. The file explorer shows the following structure:

- translation_gem_e
 - lib
 - builders
 - ar_variables_prompt_builder.rb
 - locale_prompt_builder.rb
 - exporters
 - ar_translation_exporter.rb
 - finders
 - missing_ar_variables_finder.rb
 - models
 - ai_model.rb
 - openai_model.rb
 - tasks
 - translate.rake
 - utils
 - fixme_finder.rb
 - writers
 - ar_variables_writer.rb
 - locale_writer.rb
 - translation_gem_e.rb
 - Gemfile
 - translation_gem_e.gemspec

```
translation_gem_e
├── lib
│   ├── builders
│   │   ├── ar_variables_prompt_builder.rb
│   │   └── locale_prompt_builder.rb
│   ├── exporters
│   │   └── ar_translation_exporter.rb
│   ├── finders
│   │   └── missing_ar_variables_finder.rb
│   ├── models
│   │   ├── ai_model.rb
│   │   └── openai_model.rb
│   ├── tasks
│   │   └── translate.rake
│   ├── utils
│   │   └── fixme_finder.rb
│   └── writers
│       ├── ar_variables_writer.rb
│       ├── locale_writer.rb
│       └── translation_gem_e.rb
├── Gemfile
└── translation_gem_e.gemspec
```

Feature unseres Gems

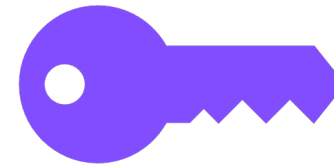


ANFRAGE AN CHATGPT

Idee



Ruby-OpenAI Gem



API access Key in .env File

OpenAI_Model: Configuration

```
# Configuration mit Azure Openai
OpenAI.configure do |config|
  config.access_token = ENV['GITHUB_TOKEN'] # Fetch API key
  config.uri_base = 'https://models.inference.ai.azure.com'
  config.api_type = :azure
  config.api_version = '2023-03-15-preview'
end
```

OpenAI_Model: System Prompt

```
# building prompt
```

```
system_prompt = <<~PROMPT
```

```
"You are a translator. Your task is to provide translations in #{target_language}.#{' '}
```

```
Your response must be formatted as JSON. Use an index starting from 0 as the key and the translation as the value.
```

```
Please use professional medical terminology as used by doctors.
```

```
Ensure the translation is clear, accurate, and formal. Do not include any other text outside of the  
Hash structure.
```

```
PROMPT
```

```
# optional context
```

```
system_prompt += "\nAdditional context to consider: #{context}" if context
```

Aufruf OpenAI-API

```
response = @client.chat(  
  parameters: {  
    model: 'gpt-4o-mini', # ChatGPT Model --> gpt-4o is VERY slow; may require request_timeout > 5min  
    messages: [  
      {  
        role: 'system',  
        content: system_prompt # with additional context  
      },  
      {  
        role: 'user',  
        content: prompt  
      }  
    ],  
    temperature: 0 # decides how "creative" ChatGPT is allowed to be.  
  }  
)
```

NOCH FRAGEN

**DANKE FÜR EURE
AUFMERKSAMKEIT!**

