



This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

# SPRING AI

## GENERATIVE ARTIFICIAL INTELLIGENCE CON JAVA

Simone Scannapieco

Corso base per Venis S.p.A, Venezia, Italia

Settembre 2025

### Note

[illegible]



## 🔧 IMPLICIT E ZERO-SHOT PROMPTING

### Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

- 1 Creazione enumeratori `ArtifactType.java` e `Sentiment.java`
- 2 Creazione modello di artefatto `Artifact.java`
- 3 Creazione modello richiesta per artefatto `ArtifactRequest.java`
- 4 Creazione *prompt template* per *sentiment analysis* di tipo *0-shot* per artefatti
- 5 Modifiche ad interfaccia ed implementazione del servizio Gemini
- 6 Modifica del controllore MVC per servizio Gemini
- 7 Test delle funzionalità con Postman/Insomnia

[illegible]

```
package it.venis.ai.spring.demo.data;

public enum ArtifactType {

    LIBRO("Libro"),
    FILM("Film"),
    SPETTACOLO("Spettacolo"),
    ARTICOLO("Articolo");

    private String artifactType;

    ArtifactType(String artifactType) {
        this.artifactType = artifactType;
    }

    public String getArtifactType() {
        return artifactType;
    }

}
```

## This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and extend across the width of the page. There are no margins, text, or other markings on the paper.

```
package it.venis.ai.spring.demo.data;

public enum Sentiment {

    MOLTO_POSITIVO("molto positivo"),
    POSITIVO("positivo"),
    NEUTRALE("neutrale"),
    NEGATIVO("negativo"),
    MOLTO_NEGATIVO("molto negativo");

    private String sentiment;

    Sentiment(String sentiment) {
        this.sentiment = sentiment;
    }

    public String getSentiment() {
        return sentiment;
    }

}
```

## This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

```
package it.venis.ai.spring.demo.model;

import java.util.Objects;

import it.venis.ai.spring.demo.data.ArtifactType;

public record Artifact(String title, String subtitle, ArtifactType type, String body) {

    @Override
    public String title() {
        return Objects.requireNonNullElse(this.title, "---");
    }

    @Override
    public String subtitle() {
        return Objects.requireNonNullElse(this.subtitle, "---");
    }
}
```

### Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

```
package it.venis.ai.spring.demo.model;

import java.util.UUID;

public record ArtifactRequest(UUID id, Artifact artifact) {

    public ArtifactRequest(String title, Artifact artifact) {
        this(UUID.randomUUID(), artifact);
    }

}
```

### Note

This image shows a single sheet of white paper with horizontal blue ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.



**File** get-artifact-sentiment-prompt.st

 Simone Scannapieco

### Note

[illegible]

```
package it.venis.ai.spring.demo.services;

import it.venis.ai.spring.demo.model.Answer;
import it.venis.ai.spring.demo.model.ArtifactRequest;
import it.venis.ai.spring.demo.model.DefinitionRequest;
import it.venis.ai.spring.demo.model.DefinitionResponse;
import it.venis.ai.spring.demo.model.Question;

public interface GeminiFromClientService {

    String getAnswerFromClient(String question);

    Answer getAnswerFromClient(Question question);

    Answer getDefinitionFromClient(DefinitionRequest definitionRequest);

    Answer getCustomFormatDefinitionFromClient(DefinitionRequest definitionRequest);

    Answer getJSONUserFormatDefinitionFromClient(DefinitionRequest definitionRequest);

    DefinitionResponse getJSONOutputConverterFormatDefinitionFromClient(DefinitionRequest definitionRequest);

    Answer getSentimentForArtifact(ArtifactRequest artifactRequest);

}
```

[illegible]

```
package it.venis.ai.spring.demo.services;

import java.util.Map;
import org.springframework.ai.chat.client.ChatClient;
import org.springframework.ai.template.st.StTemplateRenderer;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.core.io.Resource;
import org.springframework.stereotype.Service;
import it.venis.ai.spring.demo.model.Answer;
import it.venis.ai.spring.demo.model.DefinitionRequest;
import it.venis.ai.spring.demo.model.Question;

@Service
public class GeminiFromClientServiceImpl implements GeminiFromClientService {

    private final ChatClient chatClient;

    public GeminiFromClientServiceImpl(ChatClient.Builder chatClientBuilder) {
        this.chatClient = chatClientBuilder.build();
    }

    @Override
    public String getAnswerFromClient(String question) {
        return this.chatClient.prompt()
            .user(question)
            .call()
            .content();
    }
}
```

### Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

```

...
@Value("classpath:templates/get-artifact-sentiment-prompt.st")
private Resource artifactSentimentPrompt;

@Override
public Answer getSentimentForArtifact(ArtifactRequest artifactRequest) {
    List<String> stopSequences = Stream.of(" ", "\n").collect(Collectors.toList());
    Sentiment chatResponse = Sentiment.valueOf(this.chatClient.prompt()
        .options(ChatOptions.builder()
            .model("gemini-2.0-flash")
            .temperature(0.1)
            // .topP(1.0)
            // .topK(30)
            .maxTokens(10)
            // .frequencyPenalty(0.1)
            // .presencePenalty(0.1)
            .stopSequences(stopSequences)
            .build())
        .user(u -> u.text(this.artifactSentimentPrompt)
            .params(Map.of("recensione", artifactRequest.artifact().body(),
                "artefatto", artifactRequest.artifact().type())))
        .templateRenderer(StTemplateRenderer.builder().startDelimiterToken('{')
            .endDelimiterToken('}')
            .build())
        .call()
        .content()
        );
    return new Answer(chatResponse.getSentiment());
}
}

```

### Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

```
package it.venis.ai.spring.demo.controllers;

import org.springframework.web.bind.annotation.RestController;
...
import it.venis.ai.spring.demo.services.GeminiFromClientService;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;

@RestController
public class QuestionFromClientController {

    private final GeminiFromClientService geminiService;

    ...

    @PostMapping("/client/definition/json/user")
    public Answer getJSONUserFormatDefinition(@RequestBody DefinitionRequest definitionRequest) {

        return this.geminiService.getJSONUserFormatDefinitionFromClient(definitionRequest);

    }

    ...

    @PostMapping("/client/sentiment")
    public Answer getSentiment(@RequestBody ArtifactRequest artifactRequest) {

        return this.geminiService.getSentimentForArtifact(artifactRequest);

    }

}
```

### Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

13 / 13

[illegible]