📁 **Project Structure to Create**

```
smart-task-planner/
 ├── src/
 │   ├── main/
 │   │   ├── java/com/smartplanner/
 │   │   │   ├── controller/
 │   │   │   │   ├── PlanController.java
 │   │   │   ├── service/
 │   │   │   │   ├── PlanService.java
 │   │   │   │   ├── impl/PlanServiceImpl.java
 │   │   │   ├── mock/
 │   │   │   │   ├── MockLlmClient.java
 │   │   │   ├── dto/
 │   │   │   │   ├── GoalRequest.java
 │   │   │   │   ├── PlanResponse.java
 │   │   │   │   ├── TaskDto.java
 │   │   │   └── SmartTaskPlannerApplication.java
 │   │   ├── resources/
 │   │   │   ├── application.properties
 ├── pom.xml
 └── README.md
```

🧠 **1. pom.xml (dependencies)**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```xml
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <groupId>com.smartplanner</groupId>

  <artifactId>smart-task-planner</artifactId>

  <version>0.0.1-SNAPSHOT</version>

  <name>smart-task-planner</name>

  <description>Smart Task Planner Assignment</description>

  <properties>

    <java.version>17</java.version>

  </properties>

  <dependencies>


    <dependency>

      <groupId>org.springframework.boot</groupId>

      <artifactId>spring-boot-starter-web</artifactId>

    </dependency>


    <dependency>

      <groupId>org.springframework.boot</groupId>

      <artifactId>spring-boot-starter-validation</artifactId>

    </dependency>


    <dependency>

      <groupId>org.springframework.boot</groupId>

      <artifactartifactId>spring-boot-starter-test</artifactId>

      <scope>test</scope>

    </dependency>
```

```xml
        <dependency>

            <groupId>org.projectlombok</groupId>

            <artifactId>lombok</artifactId>

            <optional>true</optional>

        </dependency>


    </dependencies>


    <build>

        <plugins>

            <plugin>

                <groupId>org.springframework.boot</groupId>

                <artifactId>spring-boot-maven-plugin</artifactId>

            </plugin>

        </plugins>

    </build>

</project>
```

🚀 2. Main Application

```java
package com.smartplanner;


import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;


@SpringBootApplication

public class SmartTaskPlannerApplication {

    public static void main(String[] args) {
```

```java
        SpringApplication.run(SmartTaskPlannerApplication.class, args);

    }

}
```

📥 3. DTOs

GoalRequest.java

```java
package com.smartplanner.dto;


import jakarta.validation.constraints.NotBlank;

import lombok.Data;


@Data

public class GoalRequest {

    @NotBlank(message = "goal must not be blank")

    private String goal;


    @NotBlank(message = "targetDuration must not be blank")

    private String targetDuration;

}
```

TaskDto.java

```java
package com.smartplanner.dto;


import lombok.Data;


@Data

public class TaskDto {

    private int id;

    private String task;
```

```java
    private String description;

    private String duration;

    private Integer dependsOn;

}
```

PlanResponse.java

```java
package com.smartplanner.dto;


import lombok.Data;

import java.time.Instant;

import java.util.List;


@Data

public class PlanResponse {

    private String planId;

    private String goal;

    private String targetDuration;

    private List<TaskDto> tasks;

    private Instant createdAt;

}
```

🧠 4. Mock AI Client

MockLlmClient.java

```java
package com.smartplanner.mock;


import com.smartplanner.dto.TaskDto;

import org.springframework.stereotype.Component;


import java.util.ArrayList;
```

```java
import java.util.List;

@Component
public class MockLlmClient {

    public List<TaskDto> generateTasks(String goal, String targetDuration) {

        List<TaskDto> tasks = new ArrayList<>();

        TaskDto t1 = new TaskDto();
        t1.setId(1);
        t1.setTask("Research competitors");
        t1.setDescription("Analyze top competitor products and audience insights");
        t1.setDuration("1 day");
        t1.setDependsOn(null);

        TaskDto t2 = new TaskDto();
        t2.setId(2);
        t2.setTask("Design wireframes");
        t2.setDescription("Draft layout for homepage and major sections");
        t2.setDuration("2 days");
        t2.setDependsOn(1);

        TaskDto t3 = new TaskDto();
        t3.setId(3);
        t3.setTask("Develop frontend");
        t3.setDescription("Implement UI screens and responsiveness");
        t3.setDuration("4 days");
```

```java
        t3.setDependsOn(2);


        tasks.add(t1);

        tasks.add(t2);

        tasks.add(t3);


        return tasks;

    }

}
```

🛠 5. Service Layer

PlanService.java

```java
package com.smartplanner.service;


import com.smartplanner.dto.GoalRequest;

import com.smartplanner.dto.PlanResponse;


public interface PlanService {

    PlanResponse generatePlan(GoalRequest request);

}
```

PlanServiceImpl.java

```java
package com.smartplanner.service.impl;


import com.smartplanner.dto.GoalRequest;

import com.smartplanner.dto.PlanResponse;

import com.smartplanner.dto.TaskDto;

import com.smartplanner.mock.MockLlmClient;

import com.smartplanner.service.PlanService;
```

```java
import org.springframework.stereotype.Service;

import java.time.Instant;

import java.util.List;

import java.util.UUID;

@Service
public class PlanServiceImpl implements PlanService {

    private final MockLlmClient mockLlmClient;

    public PlanServiceImpl(MockLlmClient mockLlmClient) {
        this.mockLlmClient = mockLlmClient;
    }

    @Override
    public PlanResponse generatePlan(GoalRequest request) {

        List<TaskDto> tasks = mockLlmClient.generateTasks(request.getGoal(),
request.getTargetDuration());

        PlanResponse response = new PlanResponse();
        response.setPlanId(UUID.randomUUID().toString());
        response.setGoal(request.getGoal());
        response.setTargetDuration(request.getTargetDuration());
        response.setTasks(tasks);
        response.setCreatedAt(Instant.now());
```

```java
        return response;

    }

}
```

🌍 6. Controller

PlanController.java

```java
package com.smartplanner.controller;


import com.smartplanner.dto.GoalRequest;

import com.smartplanner.dto.PlanResponse;

import com.smartplanner.service.PlanService;

import jakarta.validation.Valid;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.*;


@RestController

@RequestMapping("/api")

public class PlanController {


    private final PlanService planService;


    public PlanController(PlanService planService) {

        this.planService = planService;

    }


    @PostMapping("/generate-plan")

    public ResponseEntity<PlanResponse> generatePlan(@Valid @RequestBody
GoalRequest request) {
```

```java
        return ResponseEntity.ok(planService.generatePlan(request));

    }


    @GetMapping("/health")

    public String health() {

        return "UP";

    }
}
```