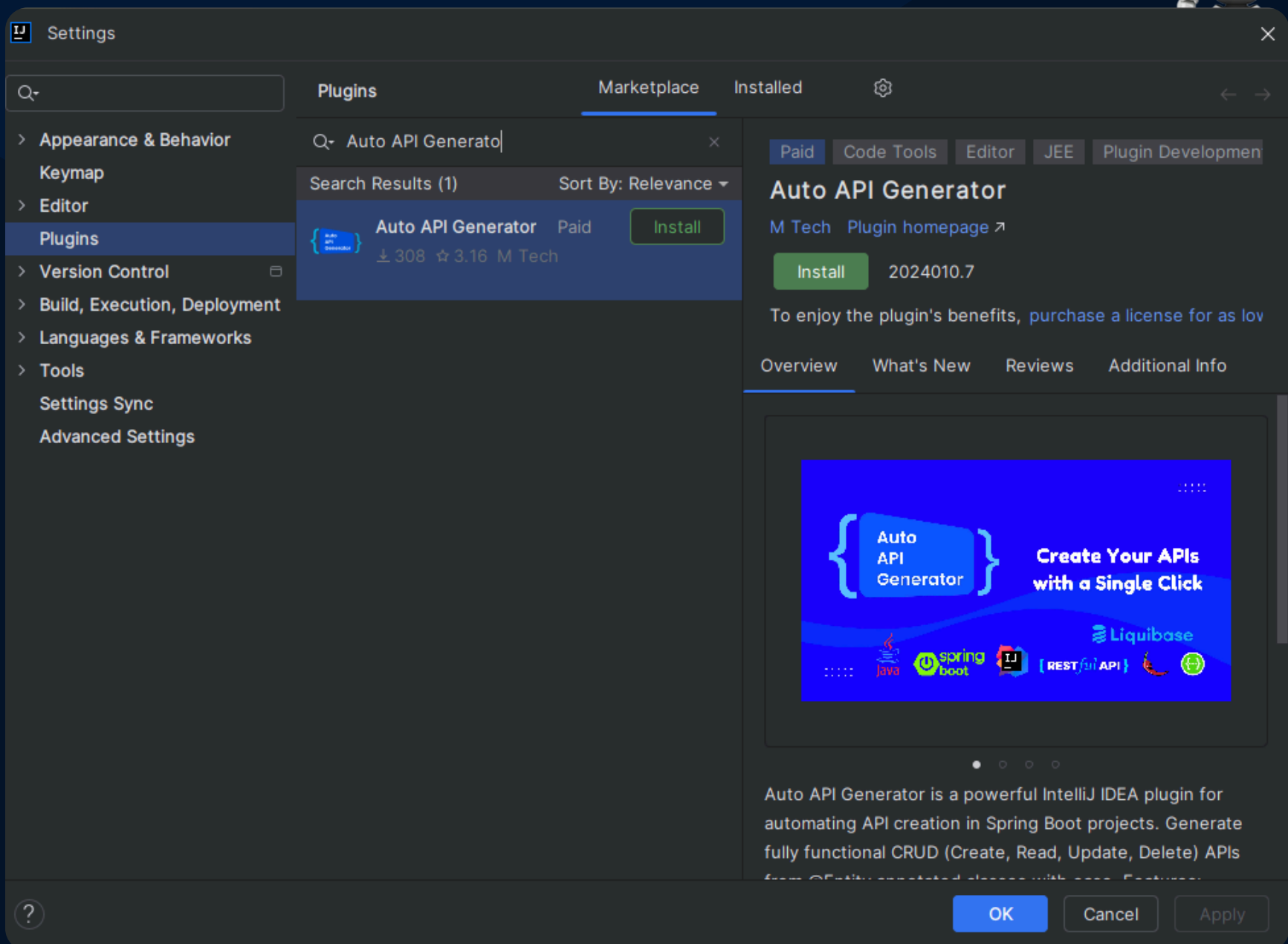




Fast API development with best practices!



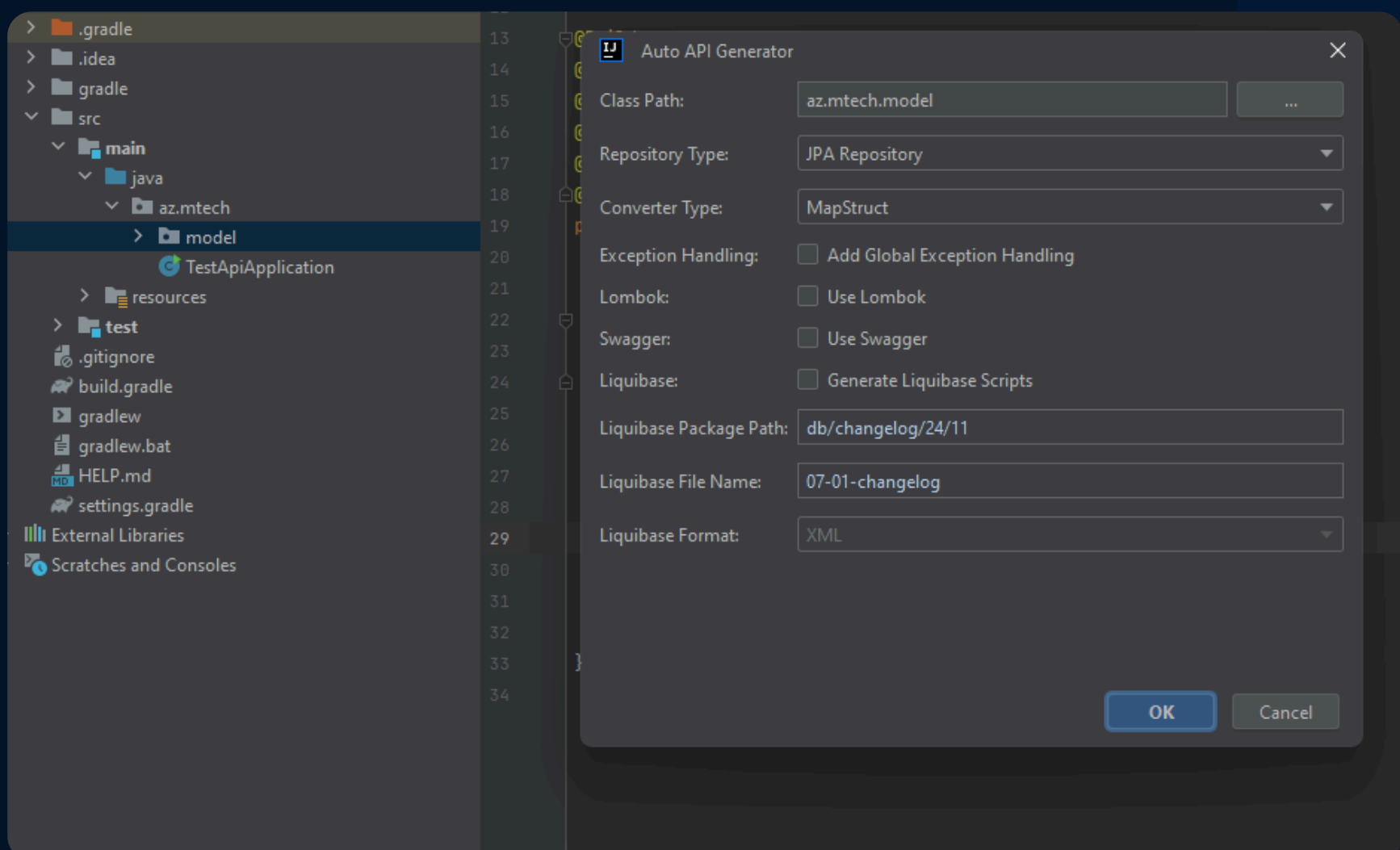
Imagine you have 100 **entity** classes and only 30 minutes to write APIs for each. It might seem impossible at first, but with the Auto API plugin, you can make it happen effortlessly!



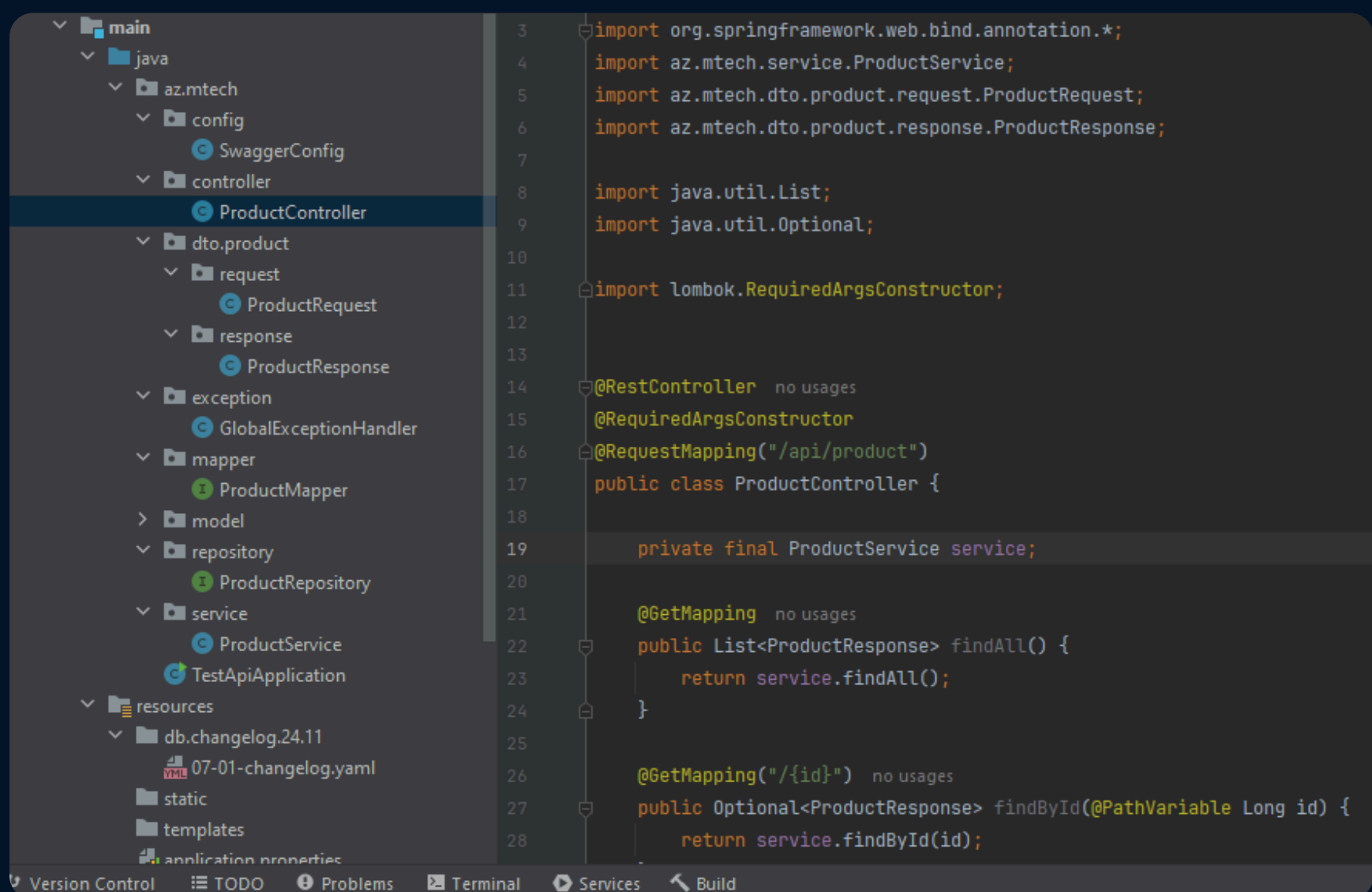
The plugin automatically generates **controllers**, **services**, **repositories**, **DTOs**, **mappers**, **exceptions**, and **Liquibase** configurations, all structured with the best practices in mind. It creates a complete project package structure along with well-organized code. Plus, you can easily enable **Lombok** and **Swagger** for even greater efficiency and cleaner code!

{ **RESTfulAPI** }

 **Liquibase**

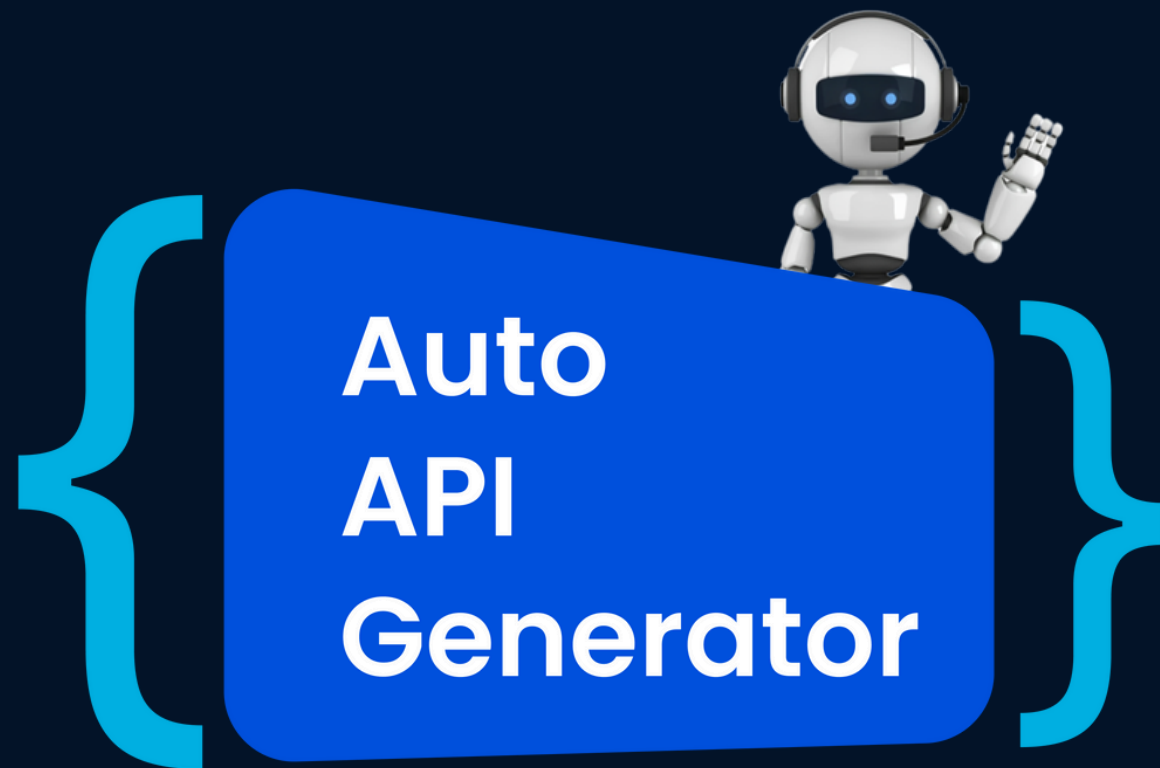


Simply create your **entity** class, and the **Auto API Generator** plugin will handle everything based on it, delivering clean, well-structured code following best practices.



The screenshot shows an IDE with a project structure on the left and a Java code file on the right. The project structure includes a 'main' directory with 'java' and 'resources' subdirectories. The 'java' directory contains 'az.mtech' (with 'config', 'controller', 'dto.product', 'exception', 'mapper', 'model', 'repository', and 'service' subdirectories) and 'resources' (with 'db.changelog.24.11', 'static', and 'templates' subdirectories). The 'controller' directory contains 'ProductController'. The 'dto.product' directory contains 'request' (with 'ProductRequest') and 'response' (with 'ProductResponse'). The 'service' directory contains 'ProductService'. The 'resources' directory contains 'db.changelog.24.11' (with '07-01-changelog.yaml'), 'static', and 'templates'. The 'ProductController' file is selected, and its code is displayed on the right. The code includes imports for Spring Framework, Az.Mtech service and DTOs, Java utility classes, and Lombok. It defines a REST controller with two endpoints: 'findAll' and 'findById'.

```
3 import org.springframework.web.bind.annotation.*;
4 import az.mtech.service.ProductService;
5 import az.mtech.dto.product.request.ProductRequest;
6 import az.mtech.dto.product.response.ProductResponse;
7
8 import java.util.List;
9 import java.util.Optional;
10
11 import lombok.RequiredArgsConstructor;
12
13
14 @RestController no usages
15 @RequiredArgsConstructor
16 @RequestMapping("/api/product")
17 public class ProductController {
18
19     private final ProductService service;
20
21     @GetMapping no usages
22     public List<ProductResponse> findAll() {
23         return service.findAll();
24     }
25
26     @GetMapping("/{id}") no usages
27     public Optional<ProductResponse> findById(@PathVariable Long id) {
28         return service.findById(id);
29     }
30 }
```



**More functionality will be added soon.
Save time, speed up your work, and make
your life easier with the Auto API
Generator plugin!**