



Fast API development with best practices!



MyBatis



File upload API



Gradle



Whether you're working with a single class or selecting hundreds at once, **Auto API Generator** has you covered! Simply choose your **entity** or **POJO** classes, and watch it generate clean, well structured APIs thousands in seconds perfectly tailored to your needs.

And much more ready with just one click.

```
@RestController no usages
@RequestMapping("/api/v1/files/pdf")
public class PdfUploadController {

    private final FilePdfService filePdfService; 4 usages

    public PdfUploadController(FilePdfService filePdfService) { this.filePdfService = f

    @GetMapping no usages
    public ResponseEntity<List<FileDto>> getAllPdfs(
        @RequestParam(required = false, defaultValue = "0") int page,
        @RequestParam(required = false, defaultValue = "10") int size
    ) {
        return ResponseEntity.ok(filePdfService.getAllPdfs(page, size));
    }
}
```

```
@GetMapping(value =("/{id}", produces = "application/pdf") no usages
public void downloadPdf(@PathVariable UUID id,
    HttpServletResponse response) {
    filePdfService.download(id, response);
    response.setContentType("application/pdf");
}

@PostMapping no usages
public ResponseEntity<UUID> upload(@RequestParam("file") MultipartFile file) {
    UUID pdf = filePdfService.upload(file);
    return ResponseEntity.ok(pdf);
}
```

```
@PostMapping no usages
public ResponseEntity<UUID> upload(@RequestParam
    UUID imageId = imageService.upload(file);
    return ResponseEntity.ok(imageId);
}

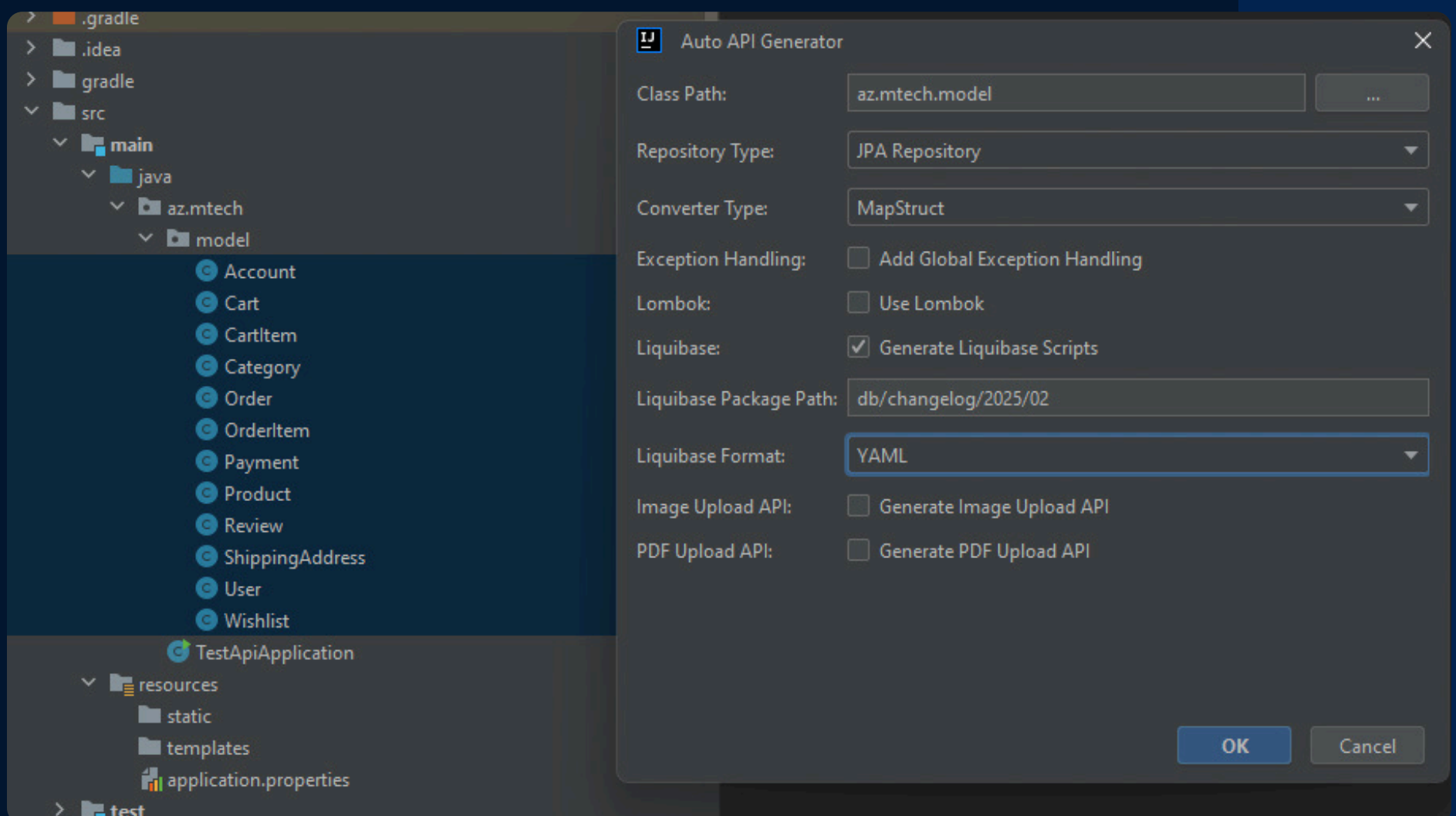
@PostMapping("/base64") no usages
public ResponseEntity<List<UUID>> uploadFiles(
    List<UUID> uuids = imageService.uploadBase
    return ResponseEntity.ok(uuids);
}

@GetMapping(value =("/{id}", produces = MediaType
public void downloadPhoto(@PathVariable UUID id
    HttpServletResponse
    imageService.downloadImage(id, response);
    response.setContentType(MediaType.IMAGE_PN
}
```

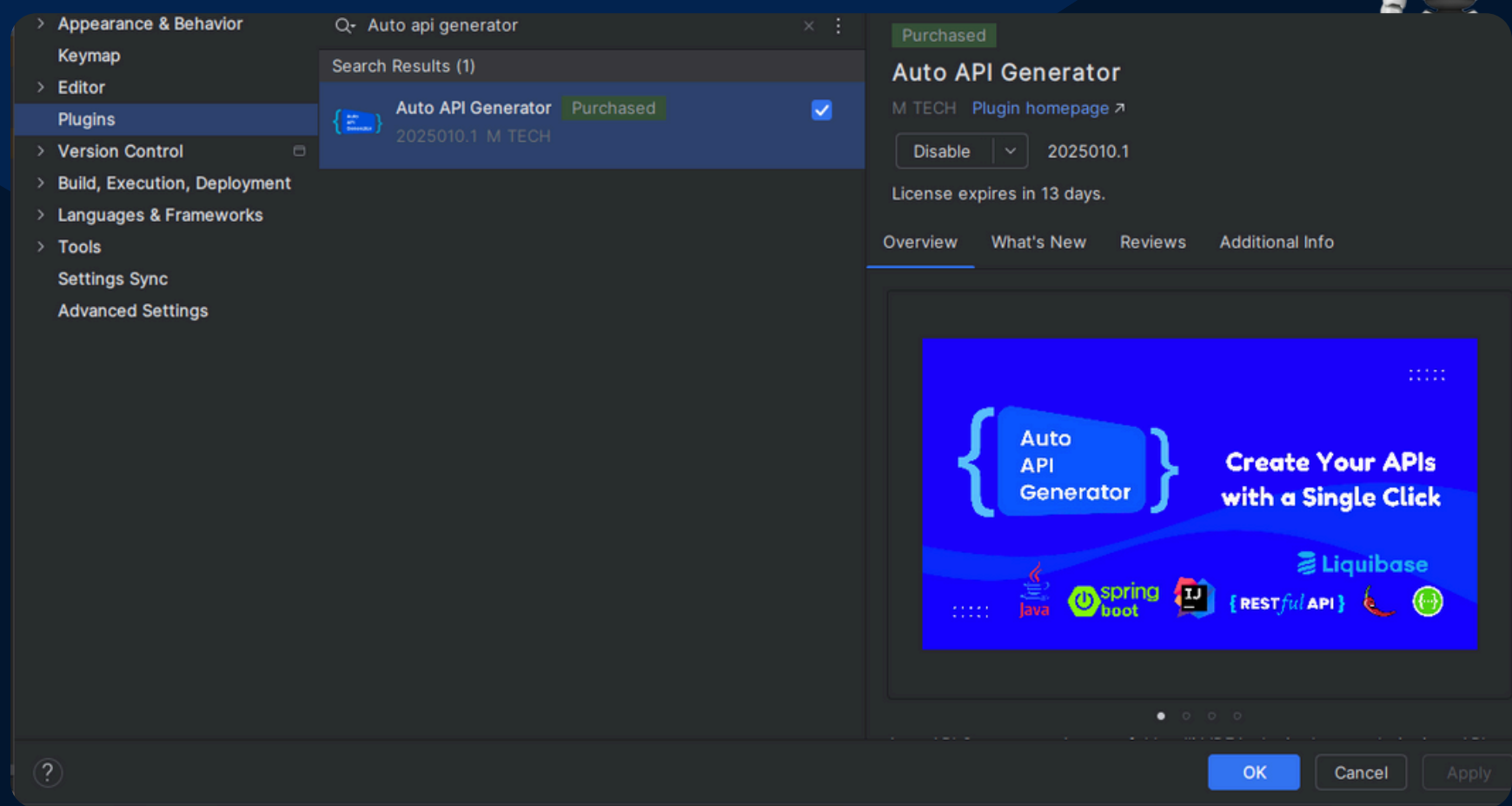
```
15 @RequiredArgsConstructor
16 @RequestMapping("/api/product")
17 public class ProductController {
18
19     private final ProductService service;
20
21     @GetMapping no usages
22     public List<ProductResponse> findAll() {
23         return service.findAll();
24     }
25
26     @GetMapping("/{id}") no usages
27     public Optional<ProductResponse> findById(@PathVariable Long id) {
28         return service.findById(id);
29     }
30 }
```

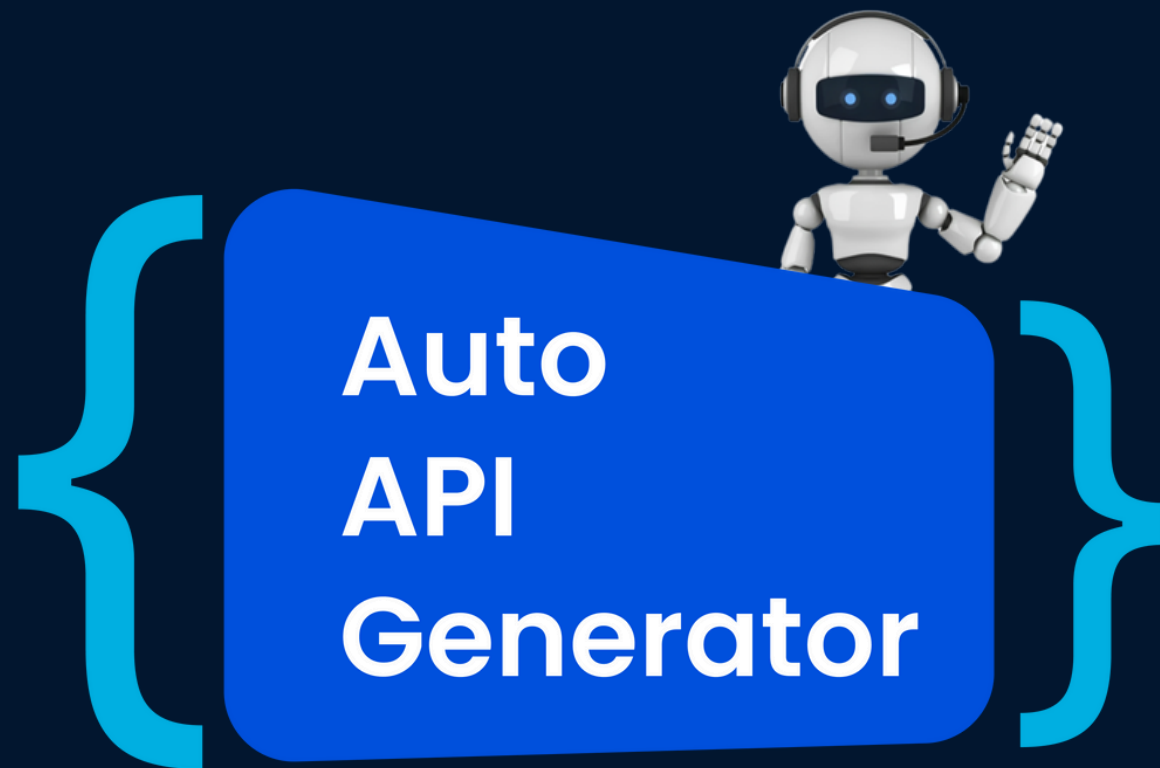
```
databaseChangeLog:
- changeSet:
    id: User-01
    author: autoapigen
    changes:
    - createTable:
        tableName: user
        columns:
        - column:
            name: id
            type: VARCHAR(255)
            constraints:
            nullable: true
        - column:
            name: name
            type: VARCHAR(255)
            constraints:
            nullable: true
        - column:
            name: surname
            type: VARCHAR(255)
            constraints:
            nullable: true
```

The plugin automatically generates **controllers**, **services**, **repositories**, **DTOs**, **mappers**, **exceptions**, and **Liquibase** configurations—all structured with the best practices in mind. You can choose between **Hibernate** or **MyBatis** for your persistence layer. It creates a complete project package structure along with well-organized code. Additionally, the plugin includes an image and PDF **upload API** feature for seamless file handling. You can also enable **Lombok** for greater efficiency and cleaner code!



Imagine handling 100 **entity** or **POJO** classes and generating thousands of APIs within minutes. With Auto API Generator, it's that easy! Streamline your workflow, deliver clean, production-ready code, and stay focused on innovation and growth—all without the repetitive boilerplate.





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