



Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”  
Факультет інформатики та обчислювальної техніки  
Кафедра інформаційних систем та технологій

Лабораторна робота №7  
ТРПЗ  
**Тема:** Патерни проектування. HTTP-сервер

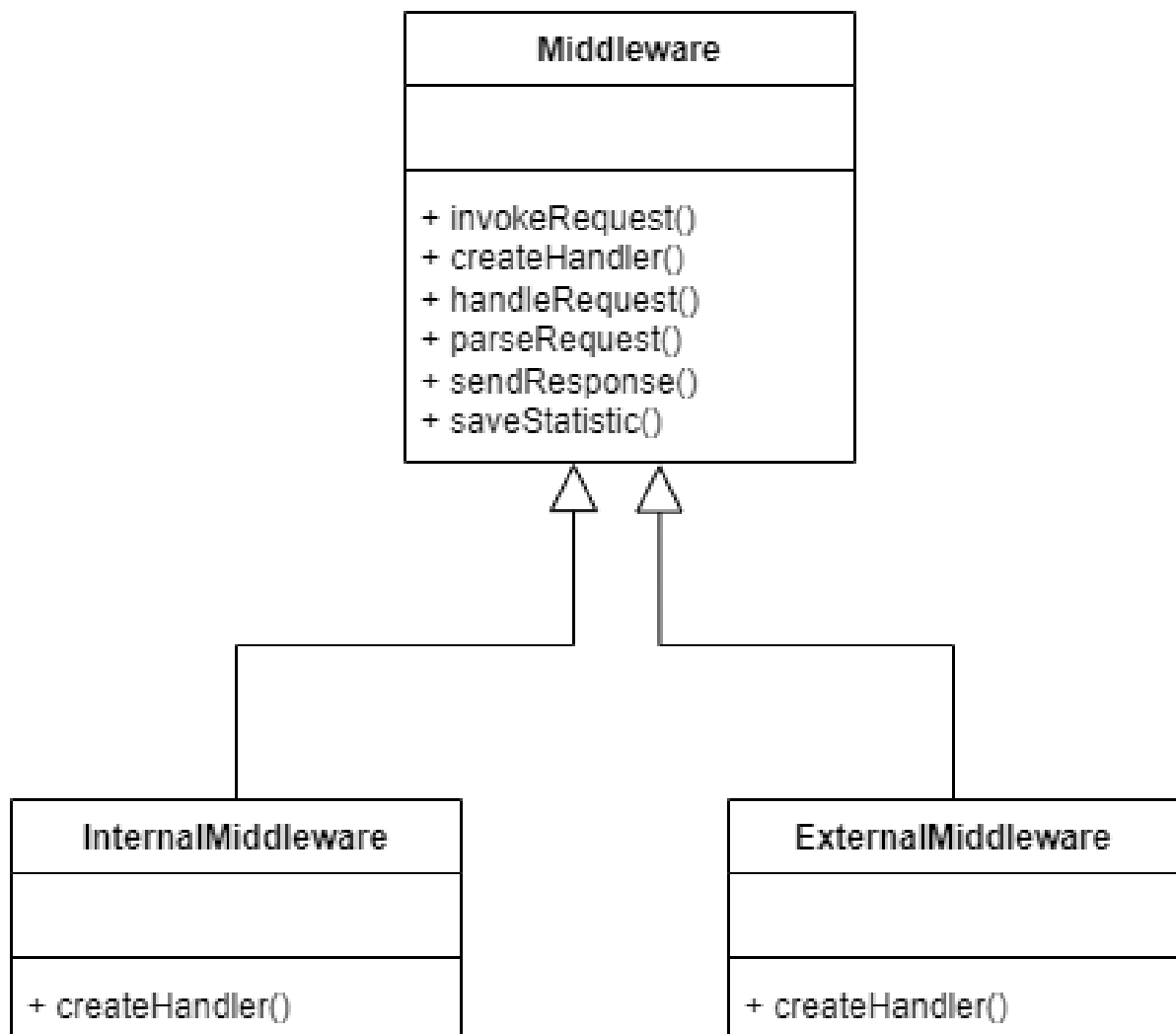
Виконав:  
студент групи ІА-31  
Машин Д. І.

Київ 2025

**Мета:** Вивчити структуру шаблонів «Mediator», «Facade», «Bridge», «Template method» та навчитися застосовувати їх в реалізації програмної системи

### Завдання

- Ознайомитись з короткими теоретичними відомостями.
- Реалізувати частину функціоналу робочої програми у вигляді класів та їхньої взаємодії для досягнення конкретних функціональних можливостей.
- Реалізувати один з розглянутих шаблонів за обраною темою.
- Реалізувати не менше 3-х класів відповідно до обраної теми.
- Підготувати звіт щодо виконання лабораторної роботи. Поданий звіт повинен містити: діаграму класів, яка представляє використання шаблону в реалізації системи, навести фрагменти коду по реалізації цього шаблону.



```

14 public abstract class Middleware {
    1 usage 2 implementations
15 public abstract Handler createHandler();
16
    2 usages
17 public void invokeRequest(Socket socket) throws IOException {
18     long requestTime = System.currentTimeMillis();
19     HttpRequest request = parseRequest(socket.getInputStream());
20     Handler handler = createHandler();
21     HttpResponse response = handleRequest(handler, request);
22     long responseTime = System.currentTimeMillis();
23     sendResponse(response, socket.getOutputStream());
24     long time = responseTime - requestTime;
25     //saveStatistic(request, response, time);
26 }
    1 usage
27 public HttpResponse handleRequest(Handler handler, HttpRequest request) { return handler.handle(request); }
    1 usage
30 public HttpRequest parseRequest(InputStream inputStream) throws IOException {
31     HttpRequestParser httpRequestParser = new HttpRequestParser();
32     return httpRequestParser.parse(inputStream);
33 }
34
    1 usage
35 public void sendResponse(HttpResponse response, OutputStream outputStream) throws IOException {
36     response.send(outputStream);
37 }
    no usages
38 public void saveStatistic(){
39 }
..

```

```

    1 usage
7 public class InternalMiddleware extends Middleware{
    2 usages
8     Map<String, HttpRequestCallback> routes;
    1 usage
9     public InternalMiddleware(Map<String, HttpRequestCallback> routes){
10         this.routes = routes;
11     }
    1 usage
12     @Override
13     public Handler createHandler() { return new InternalHandler(routes); }
16 }

```

```

3 public class ExternalMiddleware extends Middleware{
4
    1 usage
5     @Override
6     public Handler createHandler() {
7         return new ExternalHandler();
8     }
9 }

```