


RIMBERIO.CO 



ПРОЕКТ + ДОПОВІДЬ

ПІДГОТУВАВ
СТУДЕНТ 2-ГО КУРСУ
ГРУПИ МІТ-11
ПУГАЧ НАЗАР



ВАРІАНТ 18

ЗБРОЯ ТА БОЄПРИПАСИ: КУЛЕМЕТИ



*ЦЕЙ ПРОЕКТ Є ВЕБ ЗАСТОСУНКОМ ДЛЯ
ОБЛІКУ КІЛЬКОСТІ КУЛЕМЕТІВ РІЗНИХ
ВИДІВ НА УМОВНОМУ СКЛАДІ.
ПЕРЕДБАЧЕНІ МОЖЛИВОСТІ
ДОДАВАННЯ, РЕДАГУВАННЯ ТА
ВИДАЛЕННЯ, ТОБТО CRUD ОПЕРАЦІЇ*


БЕК-ЕНД ФРЕЙМОВОРК

*МИ ВОКОРИСТАЛИ SPRING FRAMEWORK 3
REST API*



БЕК-ЕНД БАЗА ДАНИХ

МИ ВОКОРИСТАЛИ POSTGRESGL ТА
HIBERNATE ДЛЯ ЗБЕРІГАННЯ ДАНИХ В
БАЗІ ДАНИХ

```
1  spring.datasource.url=jdbc:postgresql://localhost:  
2 spring.datasource.username=sample  
3 spring.datasource.password=1111  
4  
5 spring.jpa.show-sql=true  
6  
7 spring.jpa.hibernate.ddl-auto=update
```

БЕК-ЕНД ENTITY

НАШ КЛАС MACHINEGUN МАЄ
ПОЛЯ ID, NAME, PHOTOURL,
DESCRIPTON TA AMOUNT

```
@Entity 9 usages Pugach-Nazar
@Table(name = "lab5Table")
public class MachineGun {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;
    private String name; 3 usages
    private String photoUrl; 3 usages
    private String description; 3 usages
    private int amount; 3 usages

    public MachineGun() { Pugach-Nazar
}
}
```

БЕК-ЕНД КОНТРОЛЕР

МИ ВОКОРИСТАЛИ АНОТАЦІЮ
`@RestController` В КОНТРОЛЕРІ ТА
ДОДАЛИ ВСІ CRUD ОПЕРАЦІЇ

```
▼ @RestController  🌐 Pugach-Nazar
@CrossOrigin(origins = "*")
public class MachineGunController {

    private IMachineGunRepository repo; 7 usages

    MachineGunController(IMachineGunRepository repo)

    @GetMapping(🌐"/machineGuns")  🌐 Pugach-Nazar
    List<MachineGun> getMachineGuns() { return (List<MachineGun>) repo.findAll(); }

    @PostMapping(🌐"/machineGuns")  🌐 Pugach-Nazar
    MachineGun createMachineGuns(@RequestBody MachineGun machineGun) {
        return repo.save(machineGun);
    }

    @PutMapping(🌐"/machineGuns/{id}")  🌐 Pugach-Nazar
    MachineGun updateMachineGun(@RequestBody MachineGun machineGun, @PathVariable int id) {
        return repo.findById(id).map(machineGun::update).orElseThrow();
    }

    @DeleteMapping(🌐"/machineGuns/{id}")  🌐 Pugach-Nazar
    void deleteMachineGun(@PathVariable int id){
        repo.deleteById(id);
    }
}
```

ФРОНТ-ЕНД ANGULAR

*ДЛЯ ФРОНТУ МИ ВИКОРИСТАЛИ
ANGULAR, ВІН ДАЄ ВСІ НЕОБХІДНІ
МОЖЛИВСОТІ ДЛЯ СТВОРЕННЯ
ПОТРІБНОГО НАМ ВІДОБРАЖЕННЯ*



ANGULAR

Інтерфейси

МИ СТВОРИЛИ ДВА ІНТЕРФЕЙСИ
MACHINE-GUN TA REST-RESPONSE

```
export interface MachineGun {  
  id?:number,  
  name: string,  
  photoUrl: string,  
  description: string,  
  amount: number,  
  _links?:{  
    self: {  
      href: string  
    },  
    machineGun: {  
      href: string  
    }  
  }  
}
```

```
import { MachineGun } from "../machine-gun"  
  
export interface RestResponse {  
  _embedded:{  
    machineGuns:MachineGun[]  
  },  
  _links: {  
    self: {  
      href: string  
    },  
    profile: {  
      href: string  
    }  
  },  
  page: {  
    size: number,  
    totalElements: number,  
    totalPages: number,  
    number: number  
  }  
}
```

Сервіс

СЕРВІС ЗА
ДОПОМОГОЮ
ЯКОГО МИ
ВІДПРВЛЯЄМО
ЗАПИТИ ДО
СЕРВЕРА. МАЄ
GET, POST, PUT
ТА DELETE
МЕТОДИ

```
export class Service1Service {  
  
  url:string = "http://localhost:8080/machineGuns"  
  
  constructor(private http:HttpClient) { }  
  
  getRest():Observable<MachineGun[]>{  
    return this.http.get<MachineGun[]>([this.url])  
  }  
  
  postMachineGun(machineGun:MachineGun):Observable<MachineGun>{  
    return this.http.post<MachineGun>(this.url, machineGun)  
  }  
  
  putMachineGun(machineGun:MachineGun):Observable<MachineGun>{  
    return this.http.put<MachineGun>(this.url+"/"+machineGun.id, machineGun)  
  }  
  
  deleteMachineGun(machineGun:MachineGun):Observable<MachineGun>{  
    return this.http.delete<MachineGun>(this.url+"/"+machineGun.id,  
  }  
}
```


ANGULAR

В APP.COMPONENT
ЗНАХОДЯТЬСЯ МЕТОДИ ЗА
ДОПОМОГОЮ ЯКИХ
КОРИСТУВАЧ ВЗЯЄМОДІЄ З
СЕРВЕРНОЮ ЧАСТИНОЮ ЧЕРЕЗ
ВІДОБРАЖЕННЯ





```
title = 'lab5f';  
machineGuns:MachineGun[]=[];  
@Input() machineGun?:MachineGun;  
selectedMachineGun!:MachineGun;  
  
table:boolean=false;  
showcreateForm:boolean = false;  
showeditform:boolean = false;  
  
constructor(private service:Service1Service){}  
  
> getRest():void{ ...  
}  
  
> createMachineGun(machineGun:MachineGun){ ...  
}  
  
> onSelect(machineGun:MachineGun):void{ ...  
}  
  
> updateMachineGun():void{ ...  
}  
  
> deleteMachineGun(machineGun:MachineGun):void{ ...  
}
```

САЙТ

Machine guns

Get machine guns

Add new machine gun

Назва	Фото	Опис	Кількість	Дії
рпк		кулемет рпк	23	<button>Delete</button> <button>Edit</button>
m249		m249 machine gun	112	<button>Delete</button> <button>Edit</button>
m240		кулемет m240	58	<button>Delete</button> <button>Edit</button>
рпк6		кулемет рпк	5	<button>Delete</button> <button>Edit</button>

САЙТ

Machine guns

Get machine guns

Add new machine gun

Name

Photo Url

Description

Amount

Add machine gun

САЙТ

Name

рпкб

Photo Url

https://upload.wikimedia.org/wikipedia/commons/thumb/a/aa/Soviet_RPK-74.JPEG/225px-Soviet_RPK-74.JPEG

Description

кулемет рпк

Amount

5

Update

ДЯКУЮ ЗА УВАГУ



НЕ СКАЖУ)



notMyMail.com



notMyMail.com



notMyMail.com

