

```

package com.dai.webServer.Resources;

import java.net.URI; import java.util.List; import java.util.Optional;

import org.springframework.beans.factory.annotation.Autowired; import
org.springframework.http.ResponseEntity; import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.DeleteMapping; import
import org.springframework.web.bind.annotation.GetMapping; import
org.springframework.web.bind.annotation.PathVariable; import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping; import
org.springframework.web.bind.annotation.RequestBody; import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController; import
org.springframework.web.servlet.support.ServletUriComponentsBuilder;

import com.dai.webServer.Exceptions.HouseNotFoundException; import
com.dai.webServer.Exceptions.UtilizadorNotFoundException; import
com.dai.webServer.Objects.House; import com.dai.webServer.Objects.User;
import com.dai.webServer.Repos.HouseRepository;

@RequestMapping("/") @RestController public class HouseResources {

    @Autowired
    private HouseRepository houseRepository;

    //listar todas as casas
    @GetMapping("/house")
    public List<House> retrieveAllHouse() {
        return houseRepository.findAll();
    }

    //Criar Casa
    @PostMapping("/house")
    public ResponseEntity<Object> createHouse(@RequestBody House house) {

        House savedHouse = houseRepository.save(house);

        URI location = ServletUriComponentsBuilder.fromCurrentRequest().path("/{id}")
            .buildAndExpand(savedHouse.getId_house()).toUri();

        return ResponseEntity.created(location).build();
    }

    //listar casa por Id
    @GetMapping("/house/{id}")
    public House retrieveHouse(@PathVariable long id) {
        Optional<House> house = houseRepository.findById(id);

```

```

        if (!house.isPresent())
            throw new HouseNotFoundException("id-" + id);

        return house.get();
    }

    //Alterar dados da casa
    @PostMapping("/house/{id}")
    public ResponseEntity<Object> updateHouse(@RequestBody House house, @PathVariable long id) {

        Optional<House> houseOptional = houseRepository.findById(id);

        if (!houseOptional.isPresent())
            return ResponseEntity.notFound().build();

        house.setId_house(id);

        houseRepository.save(house);

        return ResponseEntity.noContent().build();
    }

    //apagar casa
    @PostMapping("/house/delete/{id}")
    public String deleteHouse(@PathVariable long id) {
        System.out.print("funciona");
        houseRepository.deleteById(id);
        System.out.print("funciona");
        return "done";
    }
}

```