1. Crear un nuevo paquete dentro del paquete **.resources**  con el nombre **.vo**
2. Dentro de este se crearan los mismos objetos del paquete **.models** pero sin las notaciones, para utilizarlo como parámetro en los controladores.

Ejemplo

package me.camilocastillo.reservation.resources.vo;

import lombok.Data;

*@Data*

public class ClienteVO {

    private *String* idCli;

    private *String* nombreCli;

    private *String* apellidoCli;

    private *String* identificacionCli;

    private *String* direccionCli;

    private *String* telefonoCli;

    private *String* emailCli;

}

VO - Virtual Object

Implementacion de controladores

package me.camilocastillo.reservation.resources;

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

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 me.camilocastillo.reservation.model.Cliente;

import me.camilocastillo.reservation.negocio.services.ClienteService;

import me.camilocastillo.reservation.resources.vo.ClienteVO;

/\*\*

\* Clase que representa el servicio web de Cliente

\*/

*@RestController*

*@RequestMapping*("/api/cliente")

public class ClienteResource {

    private final *ClienteService* clienteService;

    public ClienteResource(*ClienteService* *clienteService*) {

*this*.clienteService = clienteService;

    }

*@PostMapping*

    public *ResponseEntity*<Cliente> createCliente(*@RequestBody* *ClienteVO* *clienteVO*){

*Cliente* cliente = **new** Cliente();

        cliente.setNombreCli(clienteVO.getNombreCli());

        cliente.setApellidoCli(clienteVO.getApellidoCli());

        cliente.setDireccionCli(clienteVO.getDireccionCli());

        cliente.setTelefonoCli(clienteVO.getTelefonoCli());

        cliente.setEmailCli(clienteVO.getEmailCli());

        return **new** *ResponseEntity*<>(*this*.clienteService.create(cliente),HttpStatus.CREATED);

    }

*@PutMapping*("/{identificacion}")

    public *ResponseEntity*<Cliente> updateCliente(*@PathVariable*("identificaion") *String* *identificacion*, *@RequestBody* *ClienteVO* *clienteVO*){

*Cliente* cliente = *this*.clienteService.findByIdentificacion(identificacion);

        if(cliente == null) {

            return **new** *ResponseEntity*<Cliente>(HttpStatus.NOT\_FOUND);

        }else {

            cliente.setNombreCli(clienteVO.getNombreCli());

            cliente.setApellidoCli(clienteVO.getApellidoCli());

            cliente.setDireccionCli(clienteVO.getDireccionCli());

            cliente.setTelefonoCli(clienteVO.getTelefonoCli());

            cliente.setEmailCli(clienteVO.getEmailCli());

            return **new** *ResponseEntity*<>(*this*.clienteService.update(cliente),HttpStatus.OK);

        }

    }

}