

**DEFENSA HITO 3**

Nombre Completo: Univ. Ricardo David La Madrid Alarcón

Asignatura: PRIGRAMACION III

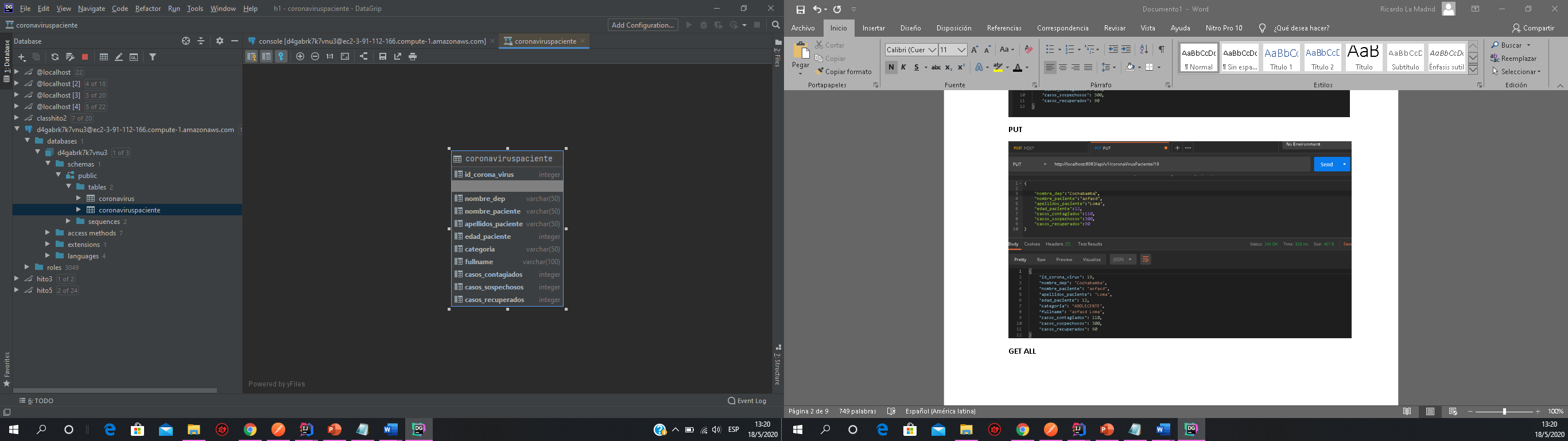
Carrera: INGENIERÍA DE SISTEMAS

Paralelo: BDA (1)

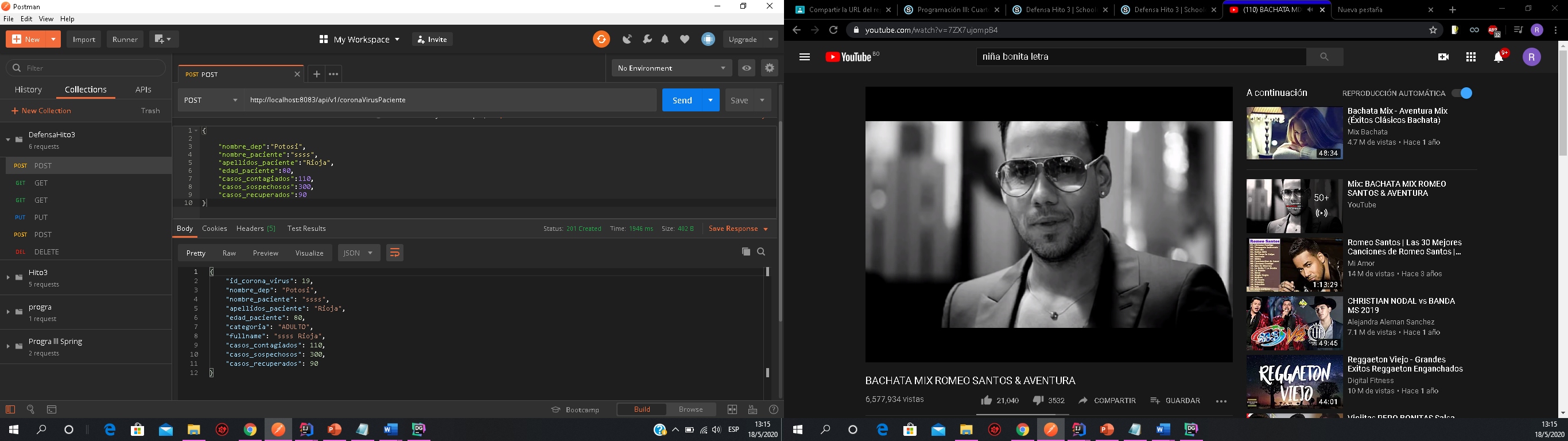
Docente: Lic. William R. Barra Paredes

fecha: 30/03/2020

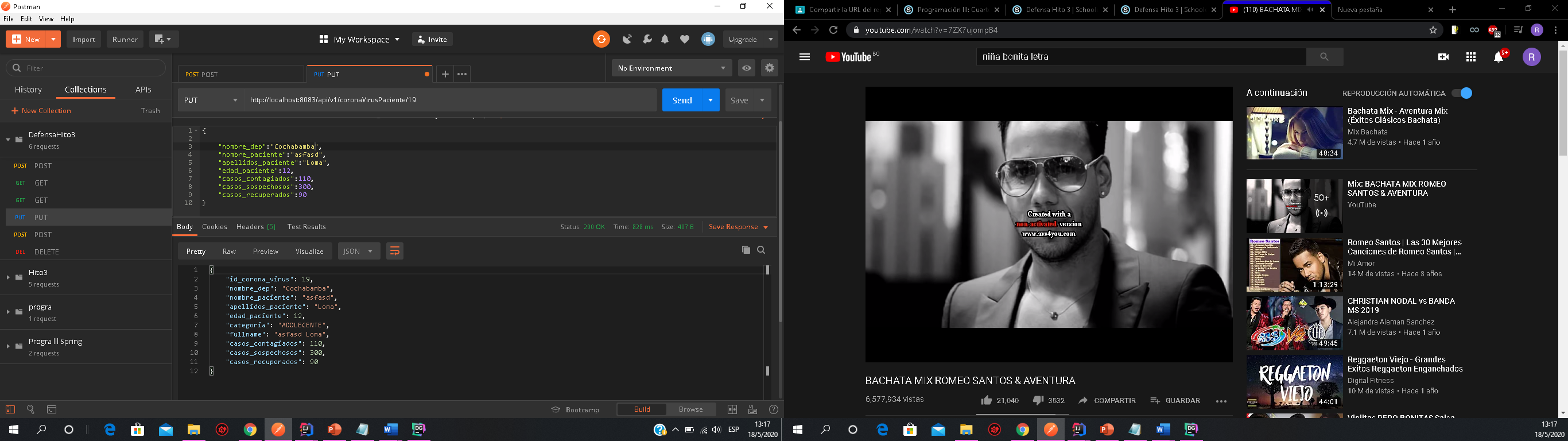
**TABLA**



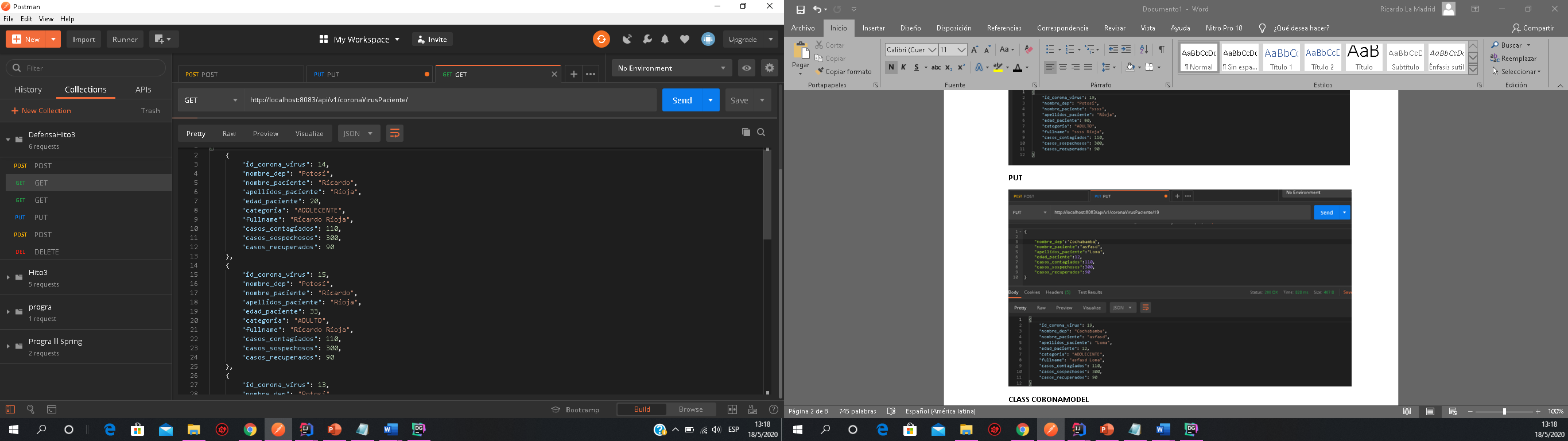
**POST**



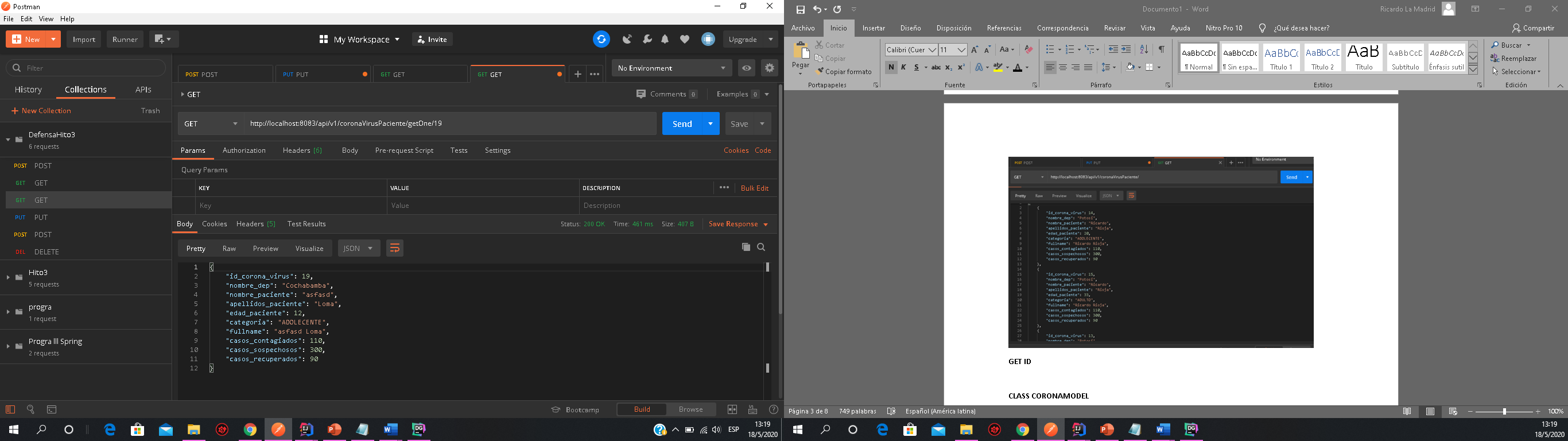
**PUT**



**GET ALL**



**GET ID**



**CLASS CORONAMODEL**

import javax.persistence.\*;  
  
  
@Entity  
@Table(name = "coronaviruspaciente")  
public class CoronaModel {  
  
 @Id  
 @GeneratedValue(strategy = *GenerationType*.*AUTO*)  
 private int id\_corona\_virus;  
 @Column(name = "nombre\_dep")  
 private String nombre\_dep;  
 @Column(name = "nombre\_paciente")  
 private String nombre\_paciente;  
 @Column(name = "apellidos\_paciente")  
 private String apellidos\_paciente;  
 @Column(name = "edad\_paciente")  
 private int edad\_paciente;  
 @Column(name = "categoria")  
 private String categoria;  
 @Column(name = "fullname")  
 private String fullname;  
 @Column(name = "casos\_contagiados")  
 private int casos\_contagiados;  
 @Column(name = "casos\_sospechosos")  
 private int casos\_sospechosos;  
 @Column(name = "casos\_recuperados")  
 private int casos\_recuperados;  
  
 public int getId\_corona\_virus() {  
 return id\_corona\_virus;  
 }  
  
 public void setId\_corona\_virus(Integer *id\_corona\_virus*) {  
 *this*.id\_corona\_virus = *id\_corona\_virus*;  
 }  
  
 public String getNombre\_dep() {  
 return nombre\_dep;  
 }  
  
 public void setNombre\_dep(String *nombre\_dep*) {  
 *this*.nombre\_dep = *nombre\_dep*;  
 }  
  
 public String getNombre\_paciente() {  
 return nombre\_paciente;  
 }  
  
 public void setNombre\_paciente(String *nombre\_paciente*) {  
 *this*.nombre\_paciente = *nombre\_paciente*;  
 }  
  
 public String getApellidos\_paciente() {  
 return apellidos\_paciente;  
 }  
  
 public void setApellidos\_paciente(String *apellidos\_paciente*) {  
 *this*.apellidos\_paciente = *apellidos\_paciente*;  
 }  
  
 public int getEdad\_paciente() {  
 return edad\_paciente;  
 }  
  
 public void setEdad\_paciente(Integer *edad\_paciente*) {  
 *this*.edad\_paciente = *edad\_paciente*;  
 }  
  
 public int getCasos\_contagiados() {  
 return casos\_contagiados;  
 }  
  
 public void setCasos\_contagiados(Integer *casos\_contagiados*) {  
 *this*.casos\_contagiados = *casos\_contagiados*;  
 }  
  
 public int getCasos\_sospechosos() {  
 return casos\_sospechosos;  
 }  
  
 public void setCasos\_sospechosos(Integer *casos\_sospechosos*) {  
 *this*.casos\_sospechosos = *casos\_sospechosos*;  
 }  
  
 public int getCasos\_recuperados() {  
 return casos\_recuperados;  
 }  
  
 public void setCasos\_recuperados(Integer *casos\_recuperados*) {  
 *this*.casos\_recuperados = *casos\_recuperados*;  
 }  
  
 public String getCategoria() {  
 return categoria;  
 }  
  
 public void setCategoria(String *categoria*) {  
 *this*.categoria = *categoria*;  
 }  
  
 public String getFullname() {  
 return fullname;  
 }  
  
 public void setFullname(String *fullname*) {  
 *this*.fullname = *fullname*;  
 }  
}

**CLASS CONTROLLERCORONA**

import defensahito3.defensahito3.Model.CoronaModel;  
import defensahito3.defensahito3.services.CoronaService;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.*HttpStatus*;  
import org.springframework.http.ResponseEntity;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.*List*;  
  
@RestController  
@RequestMapping(value = "/api/v1/")  
public class UserControllerCorona {  
  
 @Autowired  
 private CoronaService coronaService;  
  
  
 @PostMapping("/coronaVirusPaciente")  
 public ResponseEntity save(@RequestBody CoronaModel *corona*) {  
 try {  
 return new ResponseEntity<>(coronaService.save(*corona*), *HttpStatus*.*CREATED*);  
 } catch (Exception *e*) {  
 return new ResponseEntity<>(null, *HttpStatus*.*EXPECTATION\_FAILED*);  
 }  
 }  
  
  
 @DeleteMapping("/coronaVirusPaciente/delete/")  
 public ResponseEntity<String> delete(CoronaModel *cModel*) {  
 try {  
 coronaService.delete(*cModel*);  
 return new ResponseEntity<>("person successfully deleted", *HttpStatus*.*OK*);  
 } catch (Exception *e*) {  
 return new ResponseEntity<>(null, *HttpStatus*.*EXPECTATION\_FAILED*);  
 }  
 }  
  
 @PutMapping("/coronaVirusPaciente/{id\_corona\_virus}")  
 public ResponseEntity<CoronaModel> updateMaterias(@PathVariable("id\_corona\_virus") Integer *id\_corona\_virus*, @RequestBody CoronaModel *cModel*) {  
 try {  
 CoronaModel pUpdate = coronaService.update(*cModel*, *id\_corona\_virus*);  
 if (pUpdate != null) {  
 return new ResponseEntity<>(pUpdate, *HttpStatus*.*OK*);  
 } else {  
 return new ResponseEntity<>(*HttpStatus*.*NOT\_FOUND*);  
 }  
 } catch (Exception *e*) {  
 return new ResponseEntity<>(null, *HttpStatus*.*INTERNAL\_SERVER\_ERROR*);  
 }  
 }  
  
  
 @GetMapping("/coronaVirusPaciente/")  
 public ResponseEntity<*List*<CoronaModel>> getAllPersons() {  
 try {  
 *List*<CoronaModel> corona = coronaService.getAllPersons();  
  
 if (corona.isEmpty()) {  
 return new ResponseEntity<>(*HttpStatus*.*NO\_CONTENT*);  
 } else {  
 return new ResponseEntity<>(corona, *HttpStatus*.*OK*);  
 }  
 } catch (Exception *e*) {  
 return new ResponseEntity<>(null, *HttpStatus*.*INTERNAL\_SERVER\_ERROR*);  
 }  
 }  
  
 @GetMapping("/coronaVirusPaciente/getOne/{id\_corona\_virus}")  
 public ResponseEntity<CoronaModel> getPersonByIdPer(@PathVariable("id\_corona\_virus") Integer *id\_corona\_virus*) {  
 try {  
 CoronaModel pModel = coronaService.getPersonByIdPer(*id\_corona\_virus*);  
  
 if (pModel != null) {  
 return new ResponseEntity<>(pModel, *HttpStatus*.*OK*);  
 } else {  
 return new ResponseEntity<>(*HttpStatus*.*NOT\_FOUND*);  
 }  
 } catch (Exception *e*) {  
 return new ResponseEntity<>(null, *HttpStatus*.*INTERNAL\_SERVER\_ERROR*);  
 }  
 }  
}

**CLASS CORONASERVICE**

import defensahito3.defensahito3.Model.CoronaModel;  
import defensahito3.defensahito3.repo.*CoronaRepo*;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Service;  
  
import java.util.ArrayList;  
import java.util.*List*;  
import java.util.Optional;  
  
@Service  
public class CoronaService implements *CoronaServiceInterface* {  
 @Autowired  
 private *CoronaRepo* coronaRepo;  
  
 @Override  
 public CoronaModel save(CoronaModel *cModel*) {  
 if (*cModel*.getEdad\_paciente() >= 21 || *cModel*.getEdad\_paciente() <=69 ) {  
 *cModel*.setCategoria("ADULTO");  
 String fullname = (*cModel*.getNombre\_paciente() + " " + *cModel*.getApellidos\_paciente());  
 *cModel*.setFullname(fullname);  
 } else if (*cModel*.getEdad\_paciente() >= 11 && *cModel*.getEdad\_paciente() <= 20 ) {  
 *cModel*.setCategoria("ADOLECENTE");  
 String fullname = (*cModel*.getNombre\_paciente() + " " + *cModel*.getApellidos\_paciente());  
 *cModel*.setFullname(fullname);  
 } else if (*cModel*.getEdad\_paciente() <= 10) {  
 *cModel*.setCategoria("NINO");  
 String fullname = (*cModel*.getNombre\_paciente() + " " + *cModel*.getApellidos\_paciente());  
 *cModel*.setFullname(fullname);  
  
 }else if (*cModel*.getEdad\_paciente() >= 70){  
 return null;  
 }  
 else {  
 *cModel*.setCategoria("ERROR");  
 }  
 return coronaRepo.save(*cModel*);  
 }  
  
 @Override  
 public CoronaModel update(CoronaModel *cModel*, Integer *id\_corona\_virus*) {  
 Optional<CoronaModel> corona = coronaRepo.findById(*id\_corona\_virus*);  
 CoronaModel coronaUpdate = null;  
 if (corona.isPresent()) {  
 coronaUpdate = corona.get();  
 coronaUpdate.setNombre\_dep(*cModel*.getNombre\_dep());  
 coronaUpdate.setNombre\_paciente(*cModel*.getNombre\_paciente());  
 coronaUpdate.setApellidos\_paciente(*cModel*.getApellidos\_paciente());  
 coronaUpdate.setEdad\_paciente(*cModel*.getEdad\_paciente());  
 if (*cModel*.getEdad\_paciente() >= 21) {  
 coronaUpdate.setCategoria("ADULTO");  
 } else if (*cModel*.getEdad\_paciente() >= 11 && *cModel*.getEdad\_paciente() <= 20 ) {  
 coronaUpdate.setCategoria("ADOLECENTE");  
 } else if (*cModel*.getEdad\_paciente() <= 10) {  
 coronaUpdate.setCategoria("NINO");  
 }  
 coronaUpdate.setFullname(*cModel*.getNombre\_paciente() + " " + *cModel*.getApellidos\_paciente());  
 coronaUpdate.setCasos\_contagiados(*cModel*.getCasos\_contagiados());  
 coronaUpdate.setCasos\_sospechosos(*cModel*.getCasos\_sospechosos());  
 coronaUpdate.setCasos\_recuperados(*cModel*.getCasos\_recuperados());  
 coronaRepo.save(coronaUpdate);  
  
 }  
  
 return coronaUpdate;  
 }  
 @Override  
 public Integer delete(CoronaModel *cModel*) {  
 return 0;  
 }  
  
 @Override  
 public *List*<CoronaModel> getAllPersons() {  
 *List*<CoronaModel> corona = new ArrayList<CoronaModel>();  
 coronaRepo.findAll().forEach(corona::add);  
  
 return corona;  
 }  
  
 @Override  
 public CoronaModel getPersonByIdPer(Integer *id\_corona\_virus*) {  
 Optional<CoronaModel> corona = coronaRepo.findById(*id\_corona\_virus*);  
 CoronaModel cModel = null;  
  
 if (corona.isPresent()) {  
 cModel = corona.get();  
 }  
 return cModel;  
 }  
}

**INTERFACE CORONAREPO**

import defensahito3.defensahito3.Model.CoronaModel;  
import org.springframework.data.jpa.repository.*JpaRepository*;  
  
public interface *CoronaRepo* extends *JpaRepository*<CoronaModel, Integer> {  
  
  
}

**INTERFACE CORONASERVICEINTERFACE**

import defensahito3.defensahito3.Model.CoronaModel;  
  
import java.util.*List*;  
  
public interface *CoronaServiceInterface* {  
  
 public CoronaModel save(CoronaModel *cModel*);  
  
 public CoronaModel update(CoronaModel *cModel*, Integer *id\_corona\_virus*);  
 public Integer delete(CoronaModel *cModel*);  
 public *List*<CoronaModel> getAllPersons();  
 public CoronaModel getPersonByIdPer(Integer *id\_corona\_virus*);  
  
}

**CONEXIÓN**

spring.jpa.database=*POSTGRESQL*spring.jpa.show-sql=true  
spring.jpa.hibernate.ddl-auto=update  
spring.datasource.driver-class-name=org.postgresql.Driver  
spring.datasource.url=jdbc:postgresql://ec2-3-91-112-166.compute-1.amazonaws.com:5432/d4gabrk7k7vnu3  
spring.datasource.username=ytutllpgunpvgx  
spring.datasource.password=f8c7db36c571d71d03efaec82abdd381624737f5a51b5f648c002503fab23b47  
spring.jpa.properties.hibernate.jdbc.lob.non\_contextual\_creation=true  
  
server.port=8083