



El futuro digital
es de todos

MinTIC



Formato de Informe de Seguimiento



Universidad de Caldas

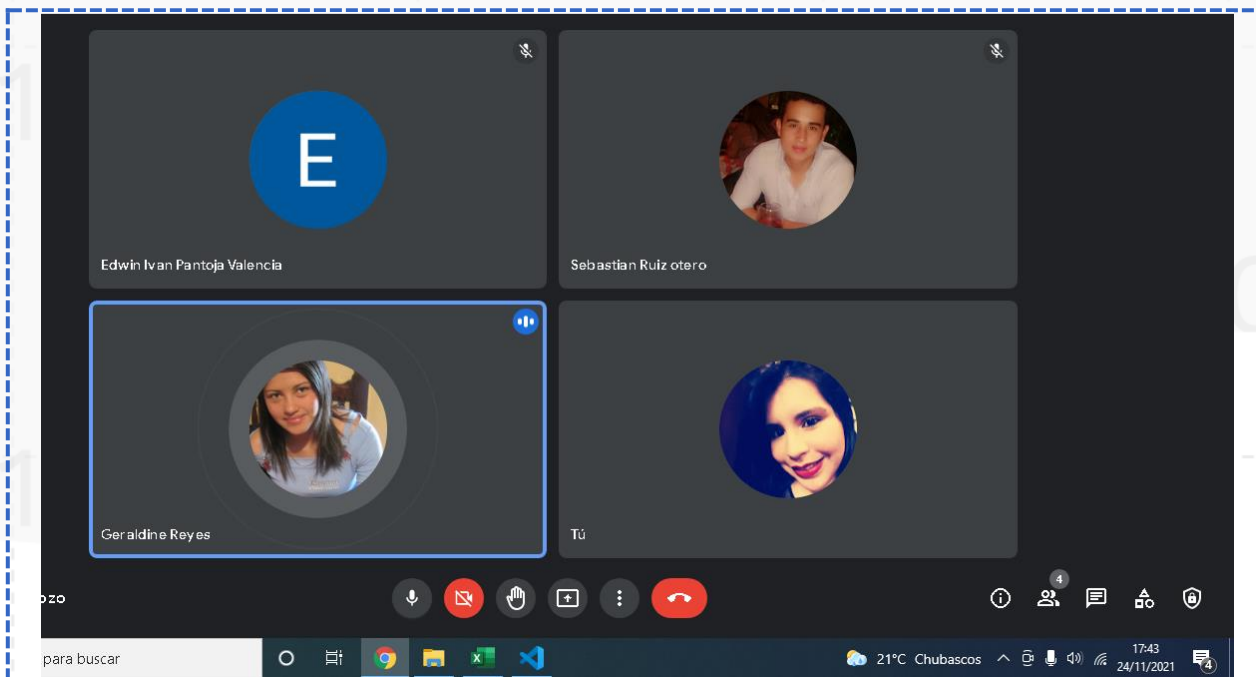


Formato de Informe de Seguimiento

Nombre del equipo	Team GSTIC
Sprint No:	3

1. Primera reunión (plan inicial del sprint) – Miércoles

- Pantallazo:





Observaciones:

Se realizó reunión de equipo el miércoles 24 de noviembre 2021; se revisó en qué consistía el sprint 3 el cual fue enviado por el profesor el día anterior en la noche y se definió el tablero Kanban (<https://github.com/tatty0318/Team-GSTIC.git>).

¿Qué debes hacer?

Teniendo la rest api solucionada en el framework loopback y la base de datos funcional

Tener presente

- Agregar las nuevas actividades en un tablero canva, marcando como terminadas las actividades realizadas del Sprint 1
 - Debemos agregar en el controlador de empleados, un método post (Login), el cual será utilizado para la autenticación del empleado y la autorización
 - Tener presente según videos de la semana implementación de encriptación de contraseña
 - Realizar pruebas funcionales utilizando postman y enviar los mensajes
-
- Entregar código funcional utilizando Loopback 4
 - Distribuir las actividades según los roles asignados en sus equipos de trabajo.

Dado que todavía se tenían inconvenientes con la tabla mensajes empleados sigue en progreso en el tablero Kanban.

tatty0318 / Team-GSTIC Public

Unwatch 1 ☆

<> Code Issues 17 Pull requests Discussions Actions Projects 1 Wiki Security Insights Settings

Team GSTIC
Updated 1 minute ago

Filter cards + Add c

To do	In progress	Done
<ul style="list-style-type: none">Crear post(login) #13 opened by tatty0318Encriptación de contraseña #14 opened by tatty0318Crear métodos de validación #15 opened by tatty0318Generar autenticación #16 opened by tatty0318	<ul style="list-style-type: none">Crear CRUD mensajesEmpleados #10 opened by tatty0318Crear tabla mensajesEmpleados #9 opened by tatty0318Realizar pruebas del CRUD mensajesEmpleados #11 opened by tatty0318	<ul style="list-style-type: none">Agregar servicio twilio al proyecto #8 opened by tatty0318realizar pruebas en celular mensajesEmpleados #12 opened by tatty0318Planeación 2 sprint #5 opened by tatty0318crear cuenta en twilio #7 opened by tatty0318

Automated as To do Manage

Automated as In progress Manage

Automated as Done Manage



2. Reunión diaria de seguimiento - Jueves



The image shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like 'src' and 'controllers', and files like 'personas.controllers.ts'. The code editor shows the implementation of the 'personas.controllers.ts' file, including imports, class decorators, and the 'identificarPersona' method.

```

src > controllers > TS personas.controllers.ts > ...
27 @service(AutenticacionService)
28 public servicioAutenticacion: AutenticacionService,
29 {}
30
31 @post("/IdentificarPersona", {
32   responses: {
33     "200": {
34       description: "Identificacion de usuarios"
35     }
36   }
37 })
38 async identificarPersona(
39   @requestBody() credenciales: Credenciales
40 ) {
41   let p = await this.servicioAutenticacion.IdentificarPersona(credenciales.usuario, c
42   if (p) {
43     let token = this.servicioAutenticacion.GenerarTokenJWT(p);
44     return {
45       datos: {
46         nombre: p.nombres + " " + p.apellidos,
47         correo: p.email,
48         id: p.id
49       },
50       tk: token
51     }
52   } else {
53     throw new HttpErrors[401]("Datos invalidos");
54   }
55 }

```

Se creo el post para identificar la persona a través del modelo credenciales y el servicio de autenticación

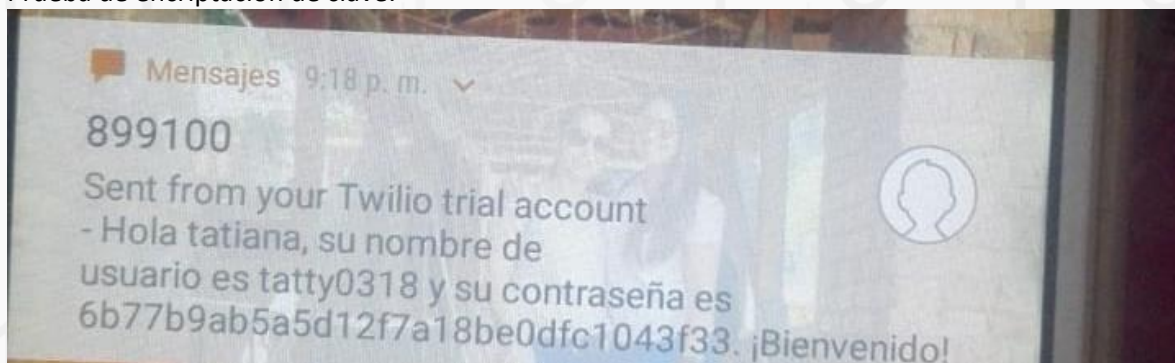
```
SPRINT1
  TS sprint1.datasource.ts M
  > estrategias M
  > models M
    TS credenciales.model.ts M
    TS empleado.model.ts M
    TS empresa.model.ts M
    TS index.ts M
    TS notificaciones-persona.model.ts M
    TS persona.model.ts M
    ① README.md M
  > repositories M
  > services M
    TS autenticacion.service.ts M
    TS index.ts M
    TS notificacion.service.ts M
    TS application.ts M
    TS index.ts M
    TS migrate.ts M
    TS openapi-specs.ts M
    TS sequence.ts M
    .dockerignore M
    .eslintrc M
    .eslintrc.js M
    .gitignore M

src > controllers > TS persona.controllers.ts >
  47
  28
  29
  30
  31
  32
  33
  34
  35
  36
  37
  38
  39
  40
  41
  42
  43
  44
  45
  46
  47
  48
  49
  50
  51
  52
  53
  54
  55

  @service(AutenticacionService)
  public servicioAutenticacion: AutenticacionService,
) { }

  @post("/identificarPersona", {
    responses: {
      '200': {
        description: "Identificacion de usuarios"
      }
    }
  })
  async identificarPersona(
    @requestBody() credenciales: Credenciales
  ) {
    let p = await this.servicioAutenticacion.IdentificarPersona(credenciales.usuario, cr
    if (p) {
      let token = this.servicioAutenticacion.GenerarTokenJWT(p);
      return {
        datos: {
          nombre: p.nombres + " " + p.apellidos,
          correo: p.email,
          id: p.id
        },
        tk: token
      }
    } else {
      throw new HttpErrors[401]("Datos invalidos");
    }
  }
}
```

Prueba de encriptación de clave:



Se crearon los métodos de identificación y validación

```
src > services > TS autenticacion.service.ts > ...
38 }
39 return false;
40 } catch {
41 return false;
42 }
43 }
44 GenerarTokenJWT(Persona: Persona) {
45 let token = JWT.sign({
46 data: {
47 id: Persona.id,
48 correo: Persona.email,
49 nombre: Persona.nombres + " " + Persona.apellidos,
50 }
51 },
52 Llaves.claveJWT);
53 return token;
54 }
55
56 ValidarTokenJWT(token: string) {
57 try {
58 let datos = JWT.verify(token, Llaves.claveJWT);
59 return datos;
60 } catch {
61 return false;
62 }
63 }
64 }
65 }
```

Y se solicita autenticación en el método post de empresa y en todos los métodos de empleados menos en el get

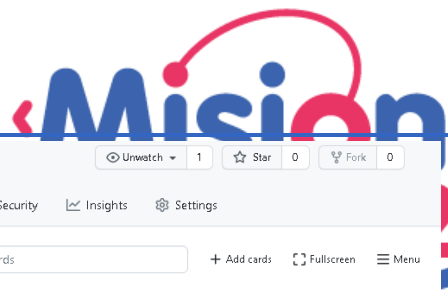
```
EXPLORADOR TS empresa.controller.ts M X
src > controllers > TS empresa.controller.ts > ...
19 constructor() {
20 @getRepository(EmpresaRepository)
21 public empresaRepository: EmpresaRepository,
22 } {}
23
24 @authenticate("admin")
25 @post('/empresas')
26 @response(200, {
27 description: 'Empresa model instance',
28 content: {'application/json': {schema: getModelSchemaRef(Empresa)}},
29 })
30 async create(
31 @requestBody({
32 content: {
33 'application/json': {
34 schema: getModelSchemaRef(Empresa, {
35 title: 'NewEmpresa',
36 }
37 }
38 }
39 }
40 })
41 empresa: Empresa,
42 ): Promise<Empresa> {
43 return this.empresaRepository.create(empresa);
44 }
45
46 @get('/empresas/count')
47 @response(200, {
```



```
EXPLORADOR
src > controllers > TS empleado.controller.ts > ...
9 } from '@loopback/repository';
10 import {
11   del, get,
12   getModelSchemaRef, param, patch, post, put, requestBody,
13   response
14 } from '@loopback/rest';
15 import {Empleado} from '../models';
16 import {EmpleadoRepository} from '../repositories';
17
18 @authenticate("admin") //Me protege todas las acciones del controlador excepto aqu
19 export class EmpleadoController {
20   constructor(
21     @repository(EmpleadoRepository)
22     public empleadoRepository: EmpleadoRepository,
23   ) { }
24
25
26 @post('/empleados')
27 @response(200, {
28   description: 'Empleado model instance',
29   content: {'application/json': {schema: getModelSchemaRef(Empleado)}}),
30 })
31 async create(
32   @requestBody({
33     content: {
34       'application/json': {
35         schema: getModelSchemaRef(Empleado, {
36           title: 'NewEmpleado',
37
```

```
38   })
39   return this.empleadoRepository.count(where);
40 }
41
42 @authenticate.skip()
43 @get('/empleados')
44 @response(200, {
45   description: 'Array of Empleado model instances',
46   content: {
47     'application/json': {
48       schema: {
49         type: 'array',
50         items: getModelSchemaRef(Empleado, {includeRelations: true}),
51       },
52     },
53   },
54 })
55 async find(
56   @param.filter(Empleado) filter?: Filter<Empleado>,
57 ): Promise<Empleado[]> {
58   return this.empleadoRepository.find(filter);
59 }
60
61 @patch('/empleados')
62 @response(200, {
63
```

Quedando el tablero Kanban de la siguiente manera:



tatty0318 / Team-GSTIC Public

Unwatch 1 Star 0 Fork 0

<> Code Issues 17 Pull requests Discussions Actions Projects 1 Wiki Security Insights Settings

Team GSTIC
Updated 4 days ago

Filter cards + Add cards Fullscreen Menu

1 To do + ...

- ✓ Pruebas funcionales validación y autenticación
#17 opened by tatty0318

Automated as To do Manage

3 In progress + ...

- ✓ Crear CRUD mensajesEmpleados
#10 opened by tatty0318
- ✓ Crear tabla mensajesEmpleados
#9 opened by tatty0318
- ✓ Realizar pruebas del CRUD mensajesEmpleados
#11 opened by tatty0318

Automated as In progress Manage

12 Done + ...

- ✓ Crear post(login)
#13 opened by tatty0318
- ✓ Generar autenticación
#16 opened by tatty0318
- ✓ Crear métodos de validación
#15 opened by tatty0318
- ✓ Encriptación de contraseña
#14 opened by tatty0318

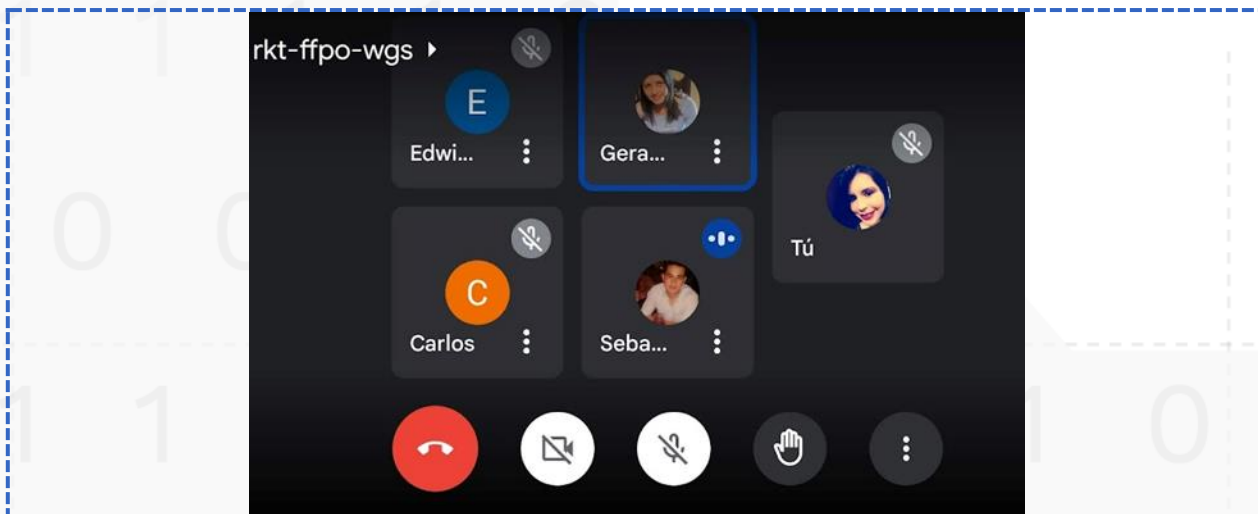
Automated as Done Manage

+ Add column



3. Reunión diaria de seguimiento – viernes

- Pantallazo:



Observaciones:

El jueves se había solicitado apoyo al profesor con el fin de aclarar como realizar lo de la integración de la tabla mensajesEmpleados.

De acuerdo con la explicación se debe realizar de forma que con un `find id()` se encuentre el id del empleado para el caso de nosotros persona y se envíe el mensaje a ese empleado encontrado. Sin embargo, no se comprende bien todavía como se debe realizar el proceso por lo cual como equipo se propone realizar revisión a la clase una vez esta sea subida y volver a revisar en próxima reunión.



4. Reunión diaria de seguimiento – Sábado

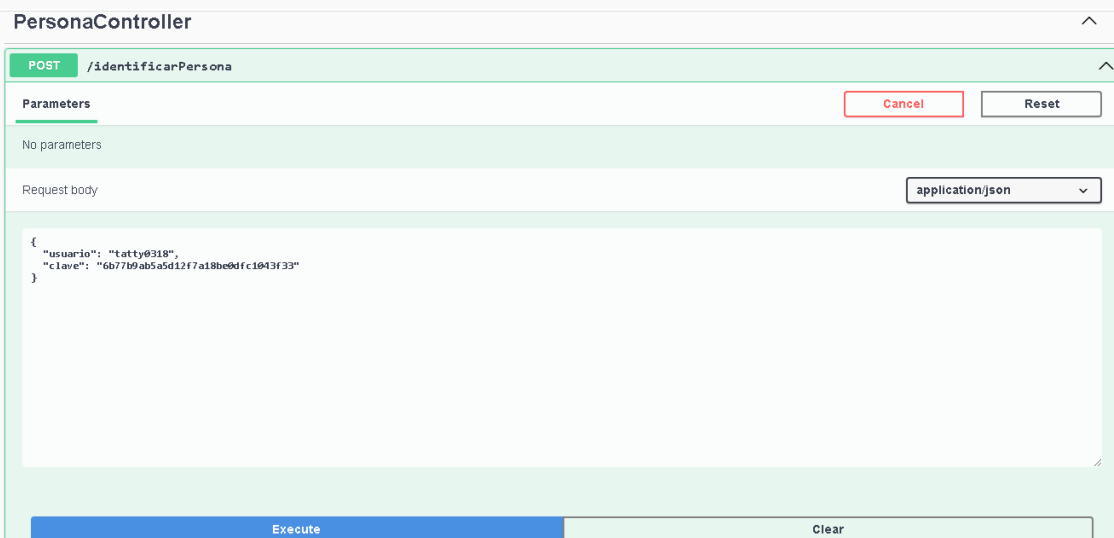
- Pantallazo:

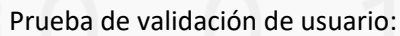



Observaciones:

Se realizaron las pruebas de la validación y autenticación de la contraseña dando como finalizado las actividades del tercer sprint.

Prueba de Identificación de personas





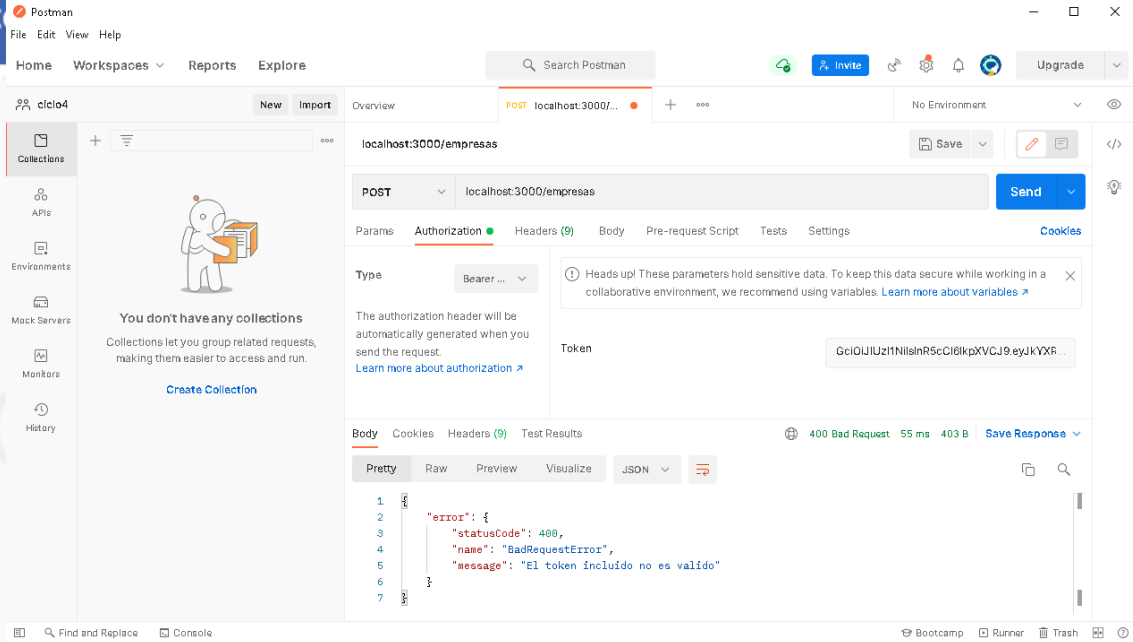
-
- The screenshot shows the Postman application interface. On the left sidebar, there are sections for 'Collections' (with a message 'You don't have any collections'), 'Environments', 'Mock Servers', 'Monitors', and 'History'. The main area displays a POST request to 'localhost:3000/empresas'. The request has a status of '400 Bad Request' and a response time of '51 ms'. The response body is a JSON object indicating an error: 'No se ha incluido un token en la solicitud.' (No token was included in the request).
- Postman Interface Details:**
- Top Bar:** File, Edit, View, Help. Search Postman. Upgrade button.
 - Left Sidebar:**
 - Collections:** You don't have any collections. Collections let you group related requests, making them easier to access and run. [Create Collection](#)
 - Environments:** No Environment selected.
 - Mock Servers:**
 - Monitors:**
 - History:**
 - Main Area:**
 - Request:** POST localhost:3000/empresas
 - Params:** none, form-data, x-www-form-urlencoded, raw, binary, GraphQL
 - Body:** This request does not have a body
 - Response:** 400 Bad Request, 51 ms. [Save Response](#)
 - Response Body:**

```

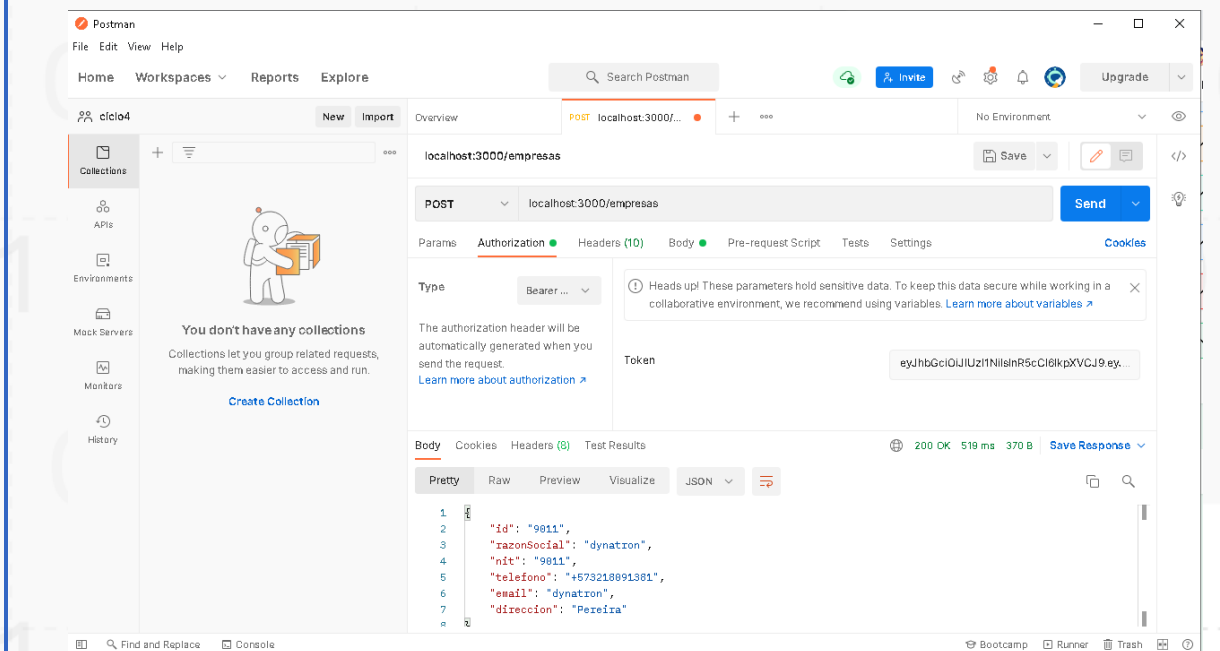
{
  "error": {
    "statusCode": 400,
    "name": "BadRequestError",
    "message": "No se ha incluido un token en la solicitud."
  }
}

```

- Con el token incluido pero invalido



- Token valido





The screenshot shows a Trello board for 'tatty0318 / Team-GSTIC'. The board has three columns: 'To do' (0 cards), 'In progress' (3 cards), and 'Done' (13 cards). The 'In progress' column contains three cards: 'Crear CRUD mensajesEmpleados' (#10), 'Crear tabla mensajesEmpleados' (#9), and 'Realizar pruebas del CRUD mensajesEmpleados' (#11). The 'Done' column contains ten cards, including 'Pruebas funcionales validación y autenticación' (#17), 'Crear post(login)' (#13), 'Generar autenticacion' (#16), and 'Crear métodos de validación' (#15). The board is updated 4 days ago and has a search bar for filtering cards.

5. Reunión diaria de seguimiento – domingo

- Pantallazo:



Observaciones:

No se hizo reunión síncrona si no que se estuvo trabajando de manera asíncrona sin embargo se sigue con el mismo inconveniente. De la tabla mensajeEmpleado



tatty0318 / Team-GSTIC Public

Unwatch 1 Star 0 Fork 0

< Code Issues 17 Pull requests Discussions Actions Projects 1 Wiki Security Insights Settings

Team GSTIC
Updated 4 days ago

Filter cards + Add cards Fullscreen Menu

0 To do + ...

3 In progress + ...

13 Done + ...

+ Add column

Automated as To do Manage

Automated as In progress Manage

Automated as Done Manage

- Crear CRUD mensajesEmpleados #10 opened by tatty0318
- Crear tabla mensajesEmpleados #9 opened by tatty0318
- Realizar pruebas del CRUD mensajesEmpleados #11 opened by tatty0318
- Pruebas funcionales validación y autenticación #17 opened by tatty0318
- Crear post(login) #13 opened by tatty0318
- Generar autenticacion #16 opened by tatty0318
- Crear métodos de validación #15 opened by tatty0318