misiontic

**CICLO 4: MISION TIC 2022**

**DESARROLLO DE APLICACIONES WEB**

**“Sprint 2”**

Logotipo

Descripción generada automáticamente con confianza media**Equipo:**

**Integrantes**

* Angel Leonardo Ramirez Mahecha
* Juan Felipe Bonilla Padilla
* Juan David Garcia Mejia
* Ernesto Alejandro Morales Rojas
* Cristian Camilo Morales Rojas

**Presentado A:**

Tannia Lucia Hernández Rojas

**21 noviembre 2021**

**Seguimiento Sprint 2**

* **Actividad**

Diseño de base de datos en **MongoDB**

* **Objetivos**
* Entregar una base de datos funcional en MongoDB con todos los documentos y colecciones que satisfagan las necesidades del proyecto “Administración de proyectos de investigación”
* Entregar los comandos y consultas requeridas para administrar la información alojada en la base de datos MongoDB.
* **Protocolo Entrega**

El documento a entregar debe contener lo siguiente:

* Portada con los nombres de los integrantes del grupo
* Descripción del proceso que se llevó a cabo para cumplir con los requerimientos de este sprint
* Evidencia de la funcionalidad de aplicación (pantallazos)
* La ruta (URL) en la que se encuentran las bases de datos MongoDB Atlas
* Compartir enlace de Trello
* **Contenido**

[1. Creación de Base de datos 3](#_Toc88157359)

[ Diagrama UML 3](#_Toc88157360)

[ Creación de la BD 4](#_Toc88157361)

[2. Comandos y consultas 8](#_Toc88157362)

[ Buscar: 8](#_Toc88157363)

[ Actualizar 8](#_Toc88157364)

[ Borrar 9](#_Toc88157365)

[3. Repositorio Trello 10](#_Toc88157366)

[4. Repositorio GitHub 11](#_Toc88157367)

**Desarrollo**

# Creación de Base de datos

## Diagrama UML

A partir de las 23 historias de usuario registradas en la plataforma, en reunión, se decidieron los campos que llevarían en el diagrama UML, las tablas **“usuarios”** y **“proyectos”.**

Diagrama

Descripción generada automáticamente

Con el anterior diagrama como referencia, se modela la base de datos en Mongo DB

## Creación de la BD

Se creo el proyecto en Mongo Atlas y se dio acceso a todos los integrantes

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Se agregan dos colecciones

* + db.createCollection("usuario") y db.createCollection("proyecto")

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se agregan documentos en la colección “usuario”

* + db.usuario.insertOne({ identificacion: 1151, nombre: 'Leonardo', apellido: 'Ramirez', email: 'angelr521@hotmail.com', rol: 'Estudiante', estado: 'Pendiente', inscripciones: 'Pendiente' })
  + db.usuario.insertOne({ identificacion: 1252, nombre: 'Juan', apellido: 'Bonilla', email: 'jfbonillap@unal.edu.co', rol: 'Administrador', estado: 'Autorizado', inscripciones: 'Pendiente' })
  + db.usuario.insertOne({ identificacion: 1353, nombre: 'Camilo', apellido: 'Morales', email: 'c.zuma2828@gmail.com', rol: 'Lider', estado: 'Pendiente', inscripciones: 'Pendiente' })
  + db.usuario.insertOne({ identificacion: 1454, nombre: 'Ernesto', apellido: 'Rincon', email: 'earm57@gmail.com', rol: 'Administrador', estado: 'Autorizado', inscripciones: 'Pendiente' })
  + db.usuario.insertOne({ identificacion: 1555, nombre: 'Juan', apellido: 'Garcia', email: 'juan84\_10@hotmail.com', rol: 'Lider', estado: 'Pendiente', inscripciones: 'Pendiente' })

Desde Robo3t:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamenteCaptura de pantalla de computadora

Descripción generada automáticamente

Se agregan 5 documentos en la colección “proyectos”

* + db.proyecto.insertOne({ nombre: **'Instalaciones Electricas'**, presupuesto: 120000, fechaInicio: new Date(), fechaFin: new Date('2022/11/17'), lider: '6195156ba02a705cdb6575c5', estado: 'Activo', fase: 'Iniciado', objetivos: [{ objetivoGeneral: 'Este es el Objetivo General' }, { objetivoEspecifico1: 'Este es el Objetivo Especifico 1' }, { objetivoEspecifico2: 'Este es el Objetivo Especifico 2' }], avance: [{descripcion1: 'Iniciando caso de estudio'}, {descripcion2: 'memorias de calculo'}]})
  + db.proyecto.insertMany([{ nombre: **'Automatizacion'**, presupuesto: 240000, fechaInicio: new Date(), fechaFin: new Date('2023/11/17'), lider: '6195156ba02a705cdb6575c5', estado: 'Activo', fase: 'Iniciado', objetivos: [{ objetivoGeneral: 'Este es el Objetivo General' }, { objetivoEspecifico1: 'Este es el Objetivo Especifico 1' }, { objetivoEspecifico2: 'Este es el Objetivo Especifico 2' }], avance: [{descripcion1: 'validacion se sensores y actuadores'}, {descripcion2: 'generacion grafcet'}]}, { nombre: **'Calidad Energia'**, presupuesto: 150000, fechaInicio: new Date(), fechaFin: new Date('2022/12/01'), lider: '61951609a02a705cdb6575c7', estado: 'Activo', fase: 'Desarrollo', objetivos: [{ objetivoGeneral: 'Este es el Objetivo General' }, { objetivoEspecifico1: 'Este es el Objetivo Especifico 1' }, { objetivoEspecifico2: 'Este es el Objetivo Especifico 2' }], avance: [{descripcion1: 'estudio factor de potencia'}, {descripcion2: 'analizador de redes electricas'}]}])
  + db.proyecto.insertMany([{ nombre: **'React'**, presupuesto: 500000, fechaInicio: new Date(), fechaFin: new Date('2022/05/17'), lider: '6195156ba02a705cdb6575c5', estado: 'Activo', fase: 'Iniciado', objetivos: [{ objetivoGeneral: 'Este es el Objetivo General' }, { objetivoEspecifico1: 'Este es el Objetivo Especifico 1' }, { objetivoEspecifico2: 'Este es el Objetivo Especifico 2' }], avance: [{descripcion1: 'Hooks'}, {descripcion2: 'react route dom'}]}, { nombre: **'FrontEnd'**, presupuesto: 150000, fechaInicio: new Date(), fechaFin: new Date('2022/02/01'), lider: '61951609a02a705cdb6575c7', estado: 'Activo', fase: 'Desarrollo', objetivos: [{ objetivoGeneral: 'Este es el Objetivo General' }, { objetivoEspecifico1: 'Este es el Objetivo Especifico 1' }, { objetivoEspecifico2: 'Este es el Objetivo Especifico 2' }], avance: [{descripcion1: 'estudio css'}, {descripcion2: 'estudio funciones'}]}])

Desde Robo3t:

Una captura de pantalla de una computadora

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Los objetivos se insertan como array [] \*\*

Captura de pantalla de computadora

Descripción generada automáticamente

# Comandos y consultas

## Buscar

* + db.proyecto.findOne()

Captura de pantalla de computadora

Descripción generada automáticamenteDesde Mongo Shell:

## Actualizar

* + db.usuario.updateOne({ \_id : ObjectId("61953a3ea02a705cdb6575d0")}, {$set: { estado : 'Autorizado' }})

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteAntes del comando:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteVerificar cambio de usuario

## Borrar

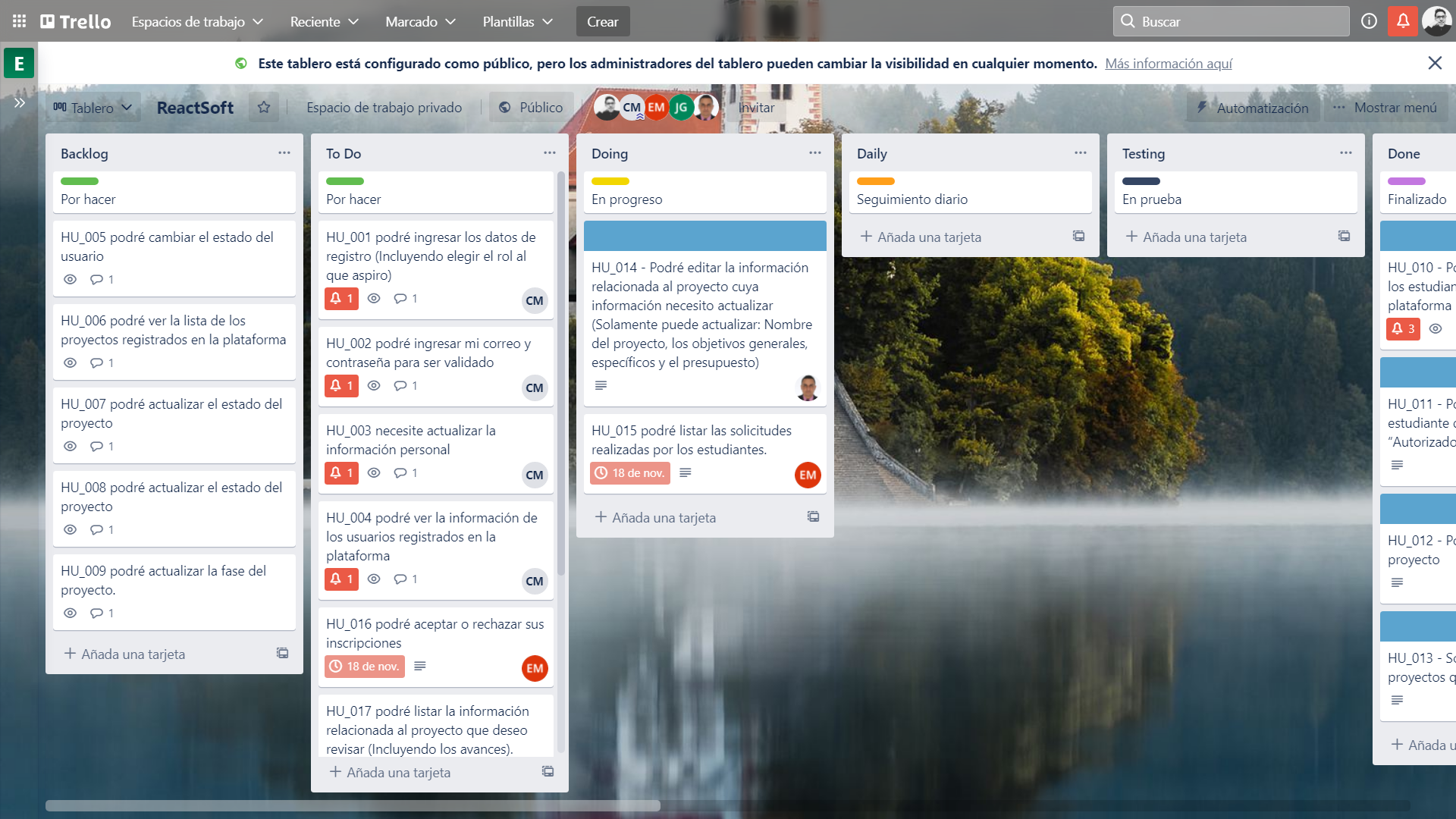
* + db.proyecto.findOneAndDelete()

Captura de pantalla de computadora

Descripción generada automáticamente

# Repositorio Trello

<https://trello.com/b/emYvOqXZ/reactsoft>

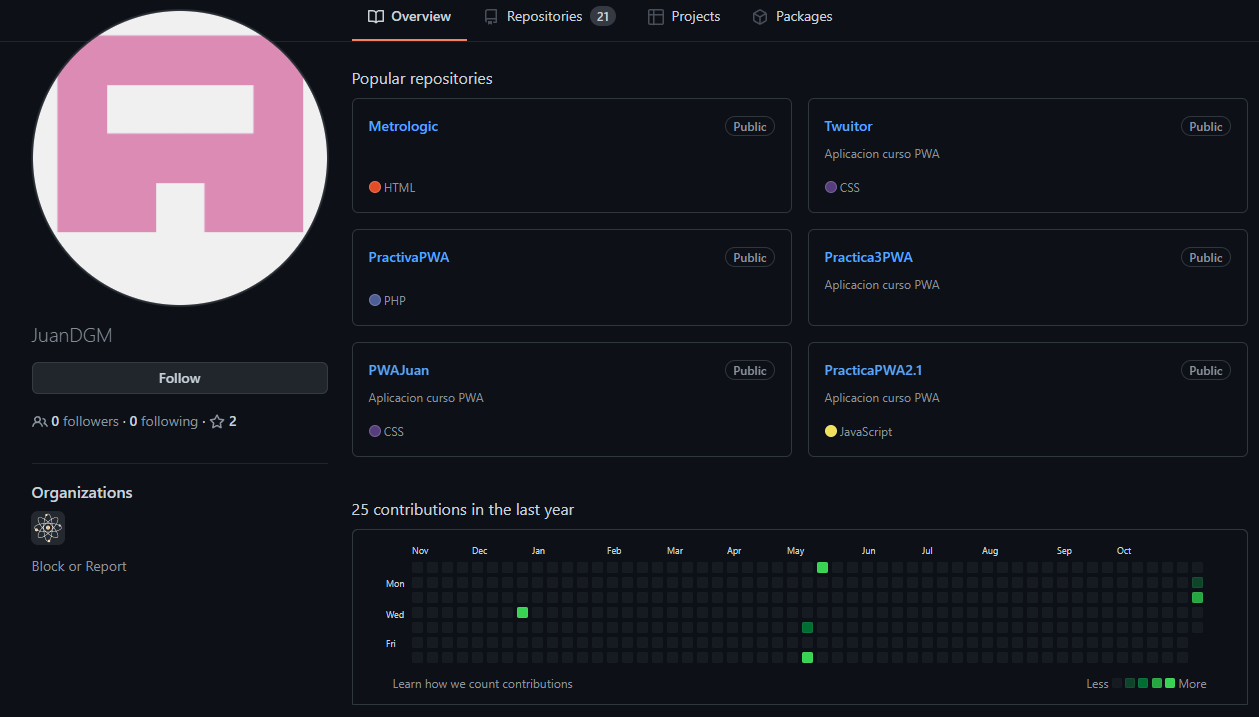
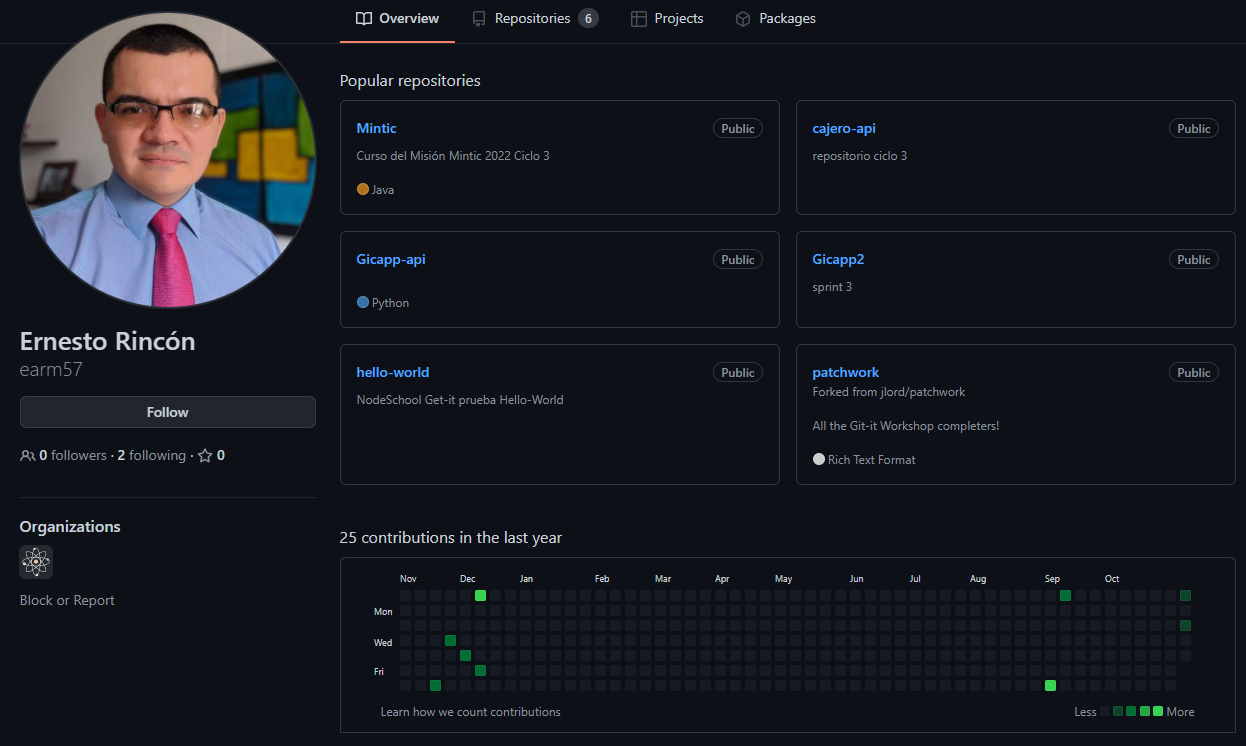


Interfaz de usuario gráfica, Aplicación

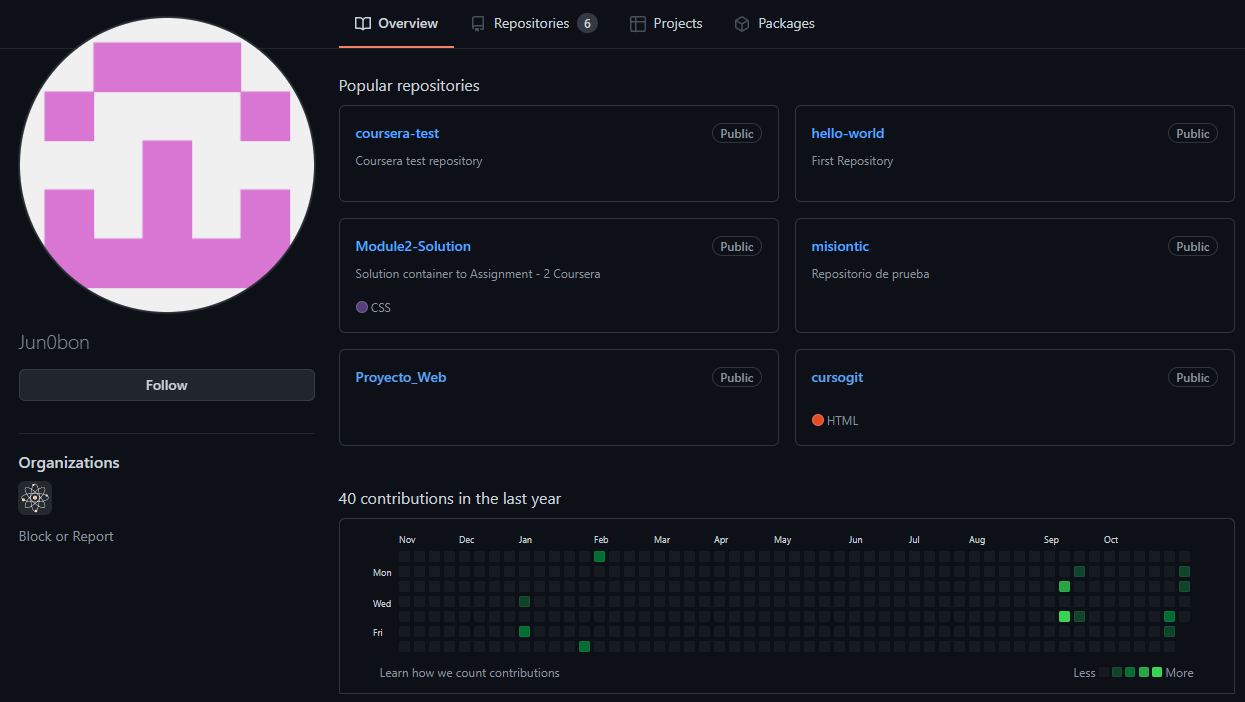
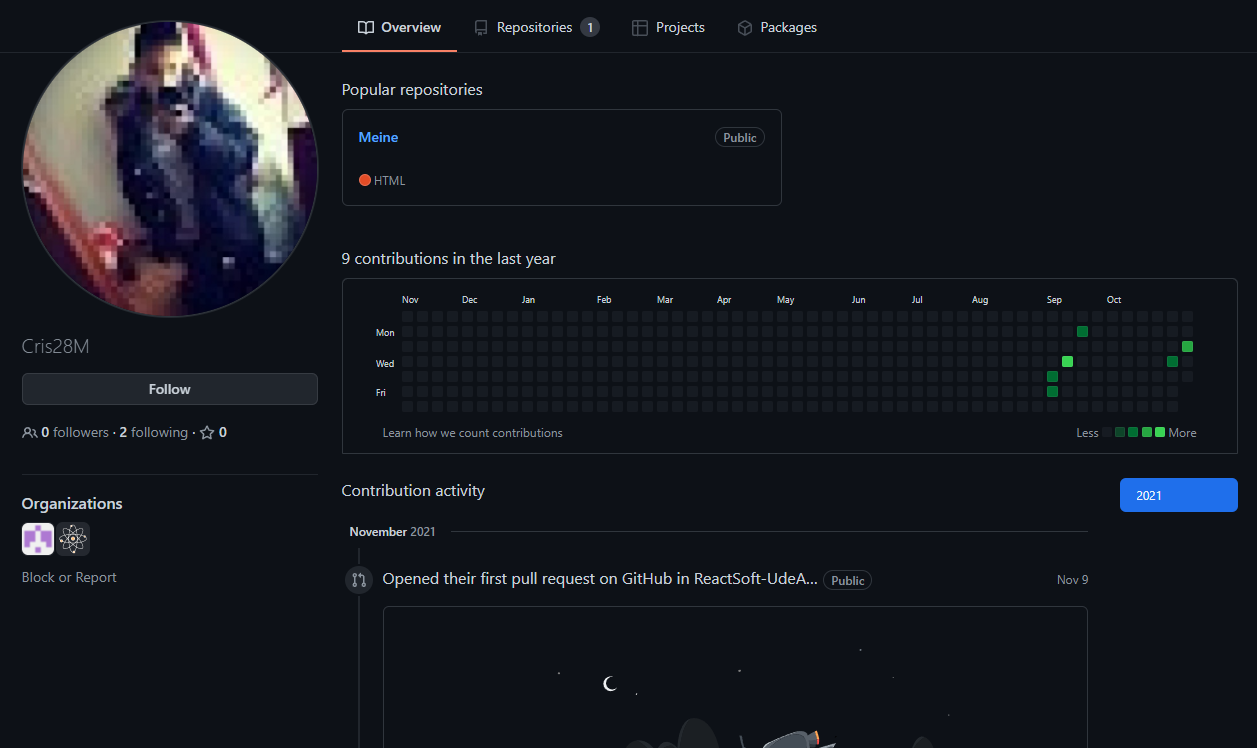
Descripción generada automáticamente

# Repositorio GitHub

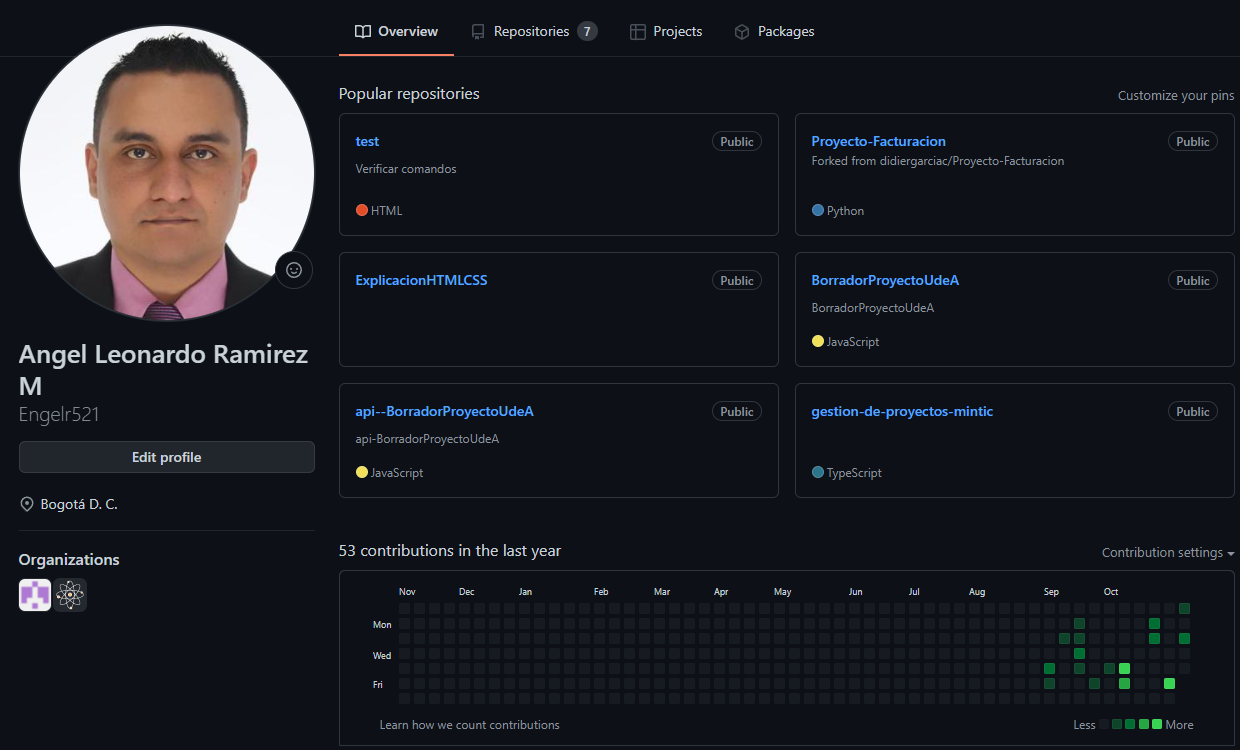
Juan David García Mejía Ernesto Alejandro Rincón Mora

Juan Felipe Bonilla Padilla Cristian Camilo Morales Rojas

Angel Leonardo Ramirez Mahecha



Se creó la organización de ReactSoft, para poder generar los diferentes repositorios para el proyecto, este se puede visualizar en el siguiente link:

<https://github.com/ReactSoft-UdeA>

1. CONFIGURACIÒN API.
2. SERVICIOS DE PROYECTO

**Historia de usuario: HU\_017**

**Como**líder

**Dado**que estoy viendo la lista de los proyectos que he registrado en la plataforma

**Cuando**necesite realizar una revisión a uno de mis proyectos

**Entonces**podré listar la información relacionada al proyecto que deseo revisar (Incluyendo los avances).

**Objetivo: Adicionar al servicio que obtiene el proyecto de un usuario, el populate de avances. (se genera desde proyectos)**

1. SERVICIOS: USUARIOS
2. SERVICIO: INSCRIPCIONES
3. **SERVICIO AVANCES (JUAN DAVID GARCÍA)**

**Historia de usuario: HU\_018**

**Como**líder

**Dado**que estoy viendo la lista de los avances registrados en uno de los proyectos que lidero

**Cuando**necesite agregar observaciones a un avance en uno de mis proyectos

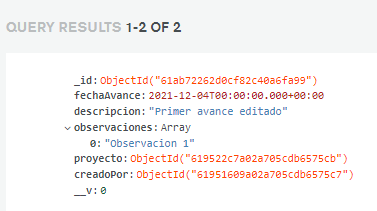
**Entonces**podré actualizar el campo de observaciones del avance seleccionado.

**Objetivo: Crear servicio de edición de avance, que permita registrar observaciones**

1. Registrar una observación a un avance

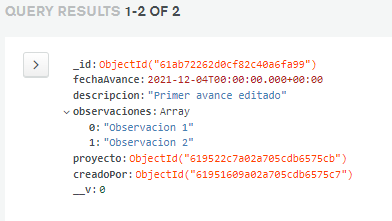
|  |  |
| --- | --- |
| 1. A continuación, se muestra el query **“EditarAvance”** enviando los parámetros de edición requeridos, como la nueva observación y definición de los atributos de retorno requeridos por el cliente-front. | 1. La siguiente imagen, muestra la respuesta de la edición, donde se evidencia el almacenamiento de la nueva observación. |

**Imagen nueva observación registrada al avance en la base de datos en Atlas mongodb**



|  |  |
| --- | --- |
| Para insertar una nueva observación, en el objeto de envio de datos se adiciona la nueva observación, como se muestra a continuación. | La siguiente imagen muestra la respuesta del servicio, donde se evidencia el almacenamiento de la nueva observación, en una estructura de lista de observaciones. |

**Imagen de las observaciones en la base de datos, donde se evidencia quedan listadas dentro de un array de observaciones:**



**Historia de usuario: HU\_021**

**Como**estudiante

**Dado**que mi inscripción a un proyecto fue aceptada

**Cuando**requiera listar los avances de un proyecto en el que estoy inscrito

**Entonces**podré ver la lista de los avances del proyecto registrados

**Objetivo: crear servicio que obtenga los avances de proyecto filtrado por usuario**

|  |  |
| --- | --- |
| 1. **Con el query Avances(Id):[Avance], se envía el \_id del usuario que reportó el avance, y la definición de la información de retorno requerida por el front** | 1. **El servicio retorna una lista de avances, con el detalle precargado del proyecto e información del usuario que realizo el registro de avance** |

**Historia de usuario: HU\_022**

**Como**estudiante

**Dado**que estoy viendo la lista de proyectos

**Cuando** requiera registrar avances a un proyecto en el que estoy inscrito (Y este se encuentra en estado activo)

**Entonces**podré ingresar la descripción de mi avance en el proyecto

**Objetivo: crear servicio que permita el registro de un avance. (la restricción que permita o no el registro de un avance según el estado del proyecto, se proporciona en dicha interfaz que inhabilita la opción de registrar avance)**

1. **Desde la mutación “CrearAvance” en la consola de Apollo, se diligencia la información a crear del nuevo avance asociado al proyecto y usuario.**

|  |  |
| --- | --- |
| 1. Datos del primer avance registrado en variables de Apollo server | 1. Mutación que pasa los parámetros registrados en las variables, y la definición del objeto a retornar en la respuesta. |

1. A continuación se presenta los retornos de almacenamiento del avance tanto en la consola de Apollo server como del documento registrado en la base de datos de mongodb, específicamente en la colección de avances.

|  |  |
| --- | --- |
|  |  |

**Historia de usuario: HU\_023**

**Como**estudiante

**Dado**estoy viendo la lista de avances de un proyecto

**Cuando**requiera actualizar la información de un avance

**Entonces**podré modificar la descripción del avance

**Objetivo: crear servicio que permita editar un avance**

|  |  |
| --- | --- |
| 1. En el servicio anterior, se obtuvo el siguiente a array de avances que tiene un usuario registrado, como se muestra a continuación. | 2. El usuario autorizado, podrá editar el avance requerido. A continuación, se muestra el servicio en Apollo, con los respectivos parámetros para su edición y retornos esperados. |

1. La respuesta al servicio presentada a continuación, muestra la edición correcta del avance registrado.

|  |  |
| --- | --- |
|  |  |