



Tarea 7: Api

10 de Abril

En esta tarea haremos **endpoints** de un **api rest** para usarlos en tareas posteriores:

```

GET    /api/usuario/correo           // para comprobar si existe o no el usuario

GET    /api/obra/cuantas             // el número total de obras

GET    /api/obra/?desde=0&hasta=10  // para un paginador

GET    /api/obra/12                  // datos de la obra con id 12

DELETE /api/obra/12                  // borrar la obra con id 12
  
```

Seguimos la [documentación de prisma](#), pero teniendo en cuenta posibles errores al acceder a la BD

```

// router/api.mjs
router.get('/usuario/:correo', async (req, res) => {
  try {
    const correo = req.params.correo // Capturamos el ID desde los parametros
    logger.debug(`----- GET USER ${correo}`)

    const usuario = await prisma.usuario.findUnique({
      where: {correo}
    })
  }
}
  
```

```
    if (usuario) {  
        res.status(200).json({ok:true, data:correo})  
    } else {  
        res.status(404).json({ok:false, msg:`${correo} not found`})  
    }  
  
    } catch (error) {  
        res.status(500).json({ok:false, msg:error})  
    }  
})
```

Testing

Utlizaremos la extensión de VSCode [REST Client](#) que permite la comprobación y depuración del API muy sencilla. Los endpoints se ponen en un archivo con extensión **.http**:

```
// test-api.http  
  
GET http://localhost:8000/api/usuario/pp@pp.com  
  
###  
  
GET http://localhost:8000/api/obra/12  
  
...
```

Referencias:

- [REST API components & How to read them](#)