



# Universidad de Costa Rica

FACULTAD DE INGENIERÍA  
ESCUELA DE CIENCIAS DE LA COMPUTACIÓN E INFORMÁTICA

CI0128 – PROYECTO INTEGRADOR: INGENIERÍA DE SOFTWARE Y  
BASES DE DATOS  
PROFESORAS: REBECA OBANDO Y ALEXANDRA MARTÍNEZ

## REPORTE: SPRINT 3

Nombre del Equipo: Ta' Bueno

Miembros:

Emmanuel D. Solís - B97670 (Scrum Master)

[emmanuel.solispomares@ucr.ac.cr](mailto:emmanuel.solispomares@ucr.ac.cr)

Gabriel Zúñiga - B98755

[gabriel.zunigaorozco@ucr.ac.cr](mailto:gabriel.zunigaorozco@ucr.ac.cr)

Jan Murillo - B95447

[jan.murillo@ucr.ac.cr](mailto:jan.murillo@ucr.ac.cr)

Kevin Arguedas - B80626

[kevin.arguedasmuriel@ucr.ac.cr](mailto:kevin.arguedasmuriel@ucr.ac.cr)

Luis D. Chinchilla - B82227

[luis.chinchillaotarola@ucr.ac.cr](mailto:luis.chinchillaotarola@ucr.ac.cr)

## 1. Código Fuente del Sistema

Se encuentra tanto adjunto en esta carpeta comprometida como en el enlace del GitHub previamente compartido.

## 2. Code Review

A realizarse según nuestra cita para el día viernes 22 de Julio a la 1:30pm.

## 3. Unit Testing

Proyecto de pruebas se encuentra en la carpeta */Planilla/planilla\_backend\_testing*.

## 4. UI Testing

Videos de prueba de ejecución de las pruebas, junto con el respectivo reporte de los **casos de prueba**, además del proyecto que crea *Selenium* se encuentran en la carpeta de este directorio: *ui\_testing*.

Cabe mencionar de forma relevante que se hicieron las pruebas **grabándolas en el navegador**, no por código, según el enunciado permitía la libertad de ejecutar cualquiera de las dos formas.

## 5. Deployment de la aplicación en Azure

Escogimos la plataforma **Azure** para el despliegue de la aplicación dado que es la que mayormente conocemos y donde también teníamos el uso de créditos gratis para poder hacerlo.

Algo importante de estipular es que nuestro repositorio de GitHub originalmente tiene en él los tres proyectos: archivos de la base de datos, backend y frontend. Por lo que fue necesario crear dos repositorios copias de estos por aparte, privados, para tener en cada uno de forma separada el backend y el frontend, y entonces poder hacer el respectivo *deployment* de cada uno.

### 5.1. Base de Datos

Para hacer el *deployment* de la base de datos fue crear una **nueva base de datos** en Azure, y usando los scripts que teníamos guardado de la creación de tablas, funciones y procedimientos, además de los índices y transacciones; pudimos crear la nueva base de datos sin problema alguno.

Primero tuvimos que crear un servidor de base de datos, y luego entonces crear la base de datos para el proyecto. En la Figura 1 podemos ver el estado actual de dicho servidor.

Una vez que ya teníamos el servidor creamos la base de datos para el proyecto en sí, esta la podemos ver en la Figura 2.

Y al tener estas dos cosas ya teníamos la base de datos en Azure para luego poder ser usada por las otras dos fases de la aplicación: *Backend* y *Frontend*.

### 5.2. Backend

Para hacer el *deployment* del backend fue más sencillo pues solo fue necesario tomar el repositorio separado del backend y en Azure crear un nuevo *App Service* el cual se le configura que es un .NET API y como esta conectado por medio de GitHub, Azure la conexión y *deployment* de forma automática por debajo. Con eso ya tenemos tanto la base de datos como el backend (API) listos y en línea, faltando solo el frontend. Podemos ver la vista general del backend en la Figura 3.

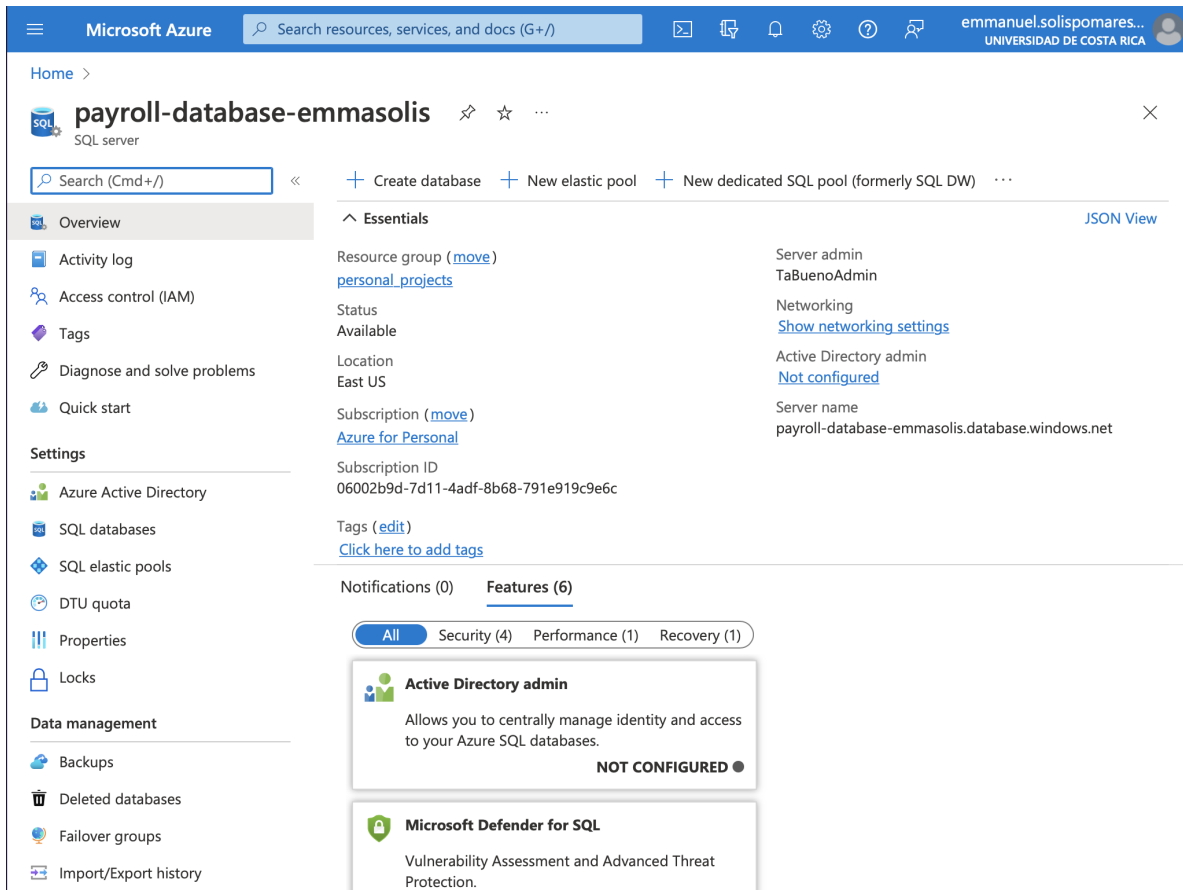


Figura 1: Vista general del servidor de base de datos en Azure.

## 6. Sprint Backlog

Se encuentra en el siguiente enlace de la profesora Rebeca Obando: <https://ingesoftg001.atlassian.net/jira/software/projects/TB/boards/5/backlog>. Con todos los requerimientos solicitados según el enunciado.

## 7. Entregable de las ceremonias

A ser entregado el lunes 27 de junio por medio del correo institucional según indicación de la profesora; véase Figura ??.

## 8. Mock Ups

Se encuentran en el archivo *sprint2/mock\_ups.pdf*.

## 9. Implementación de la Base de Datos

Nuestra base de datos *Ta Bueno* se encuentra implementada con las tablas que teníamos en los diagramas de Modelo EntidadRelación y Lógico que fueron previamente presentados y aprobados por la profesora

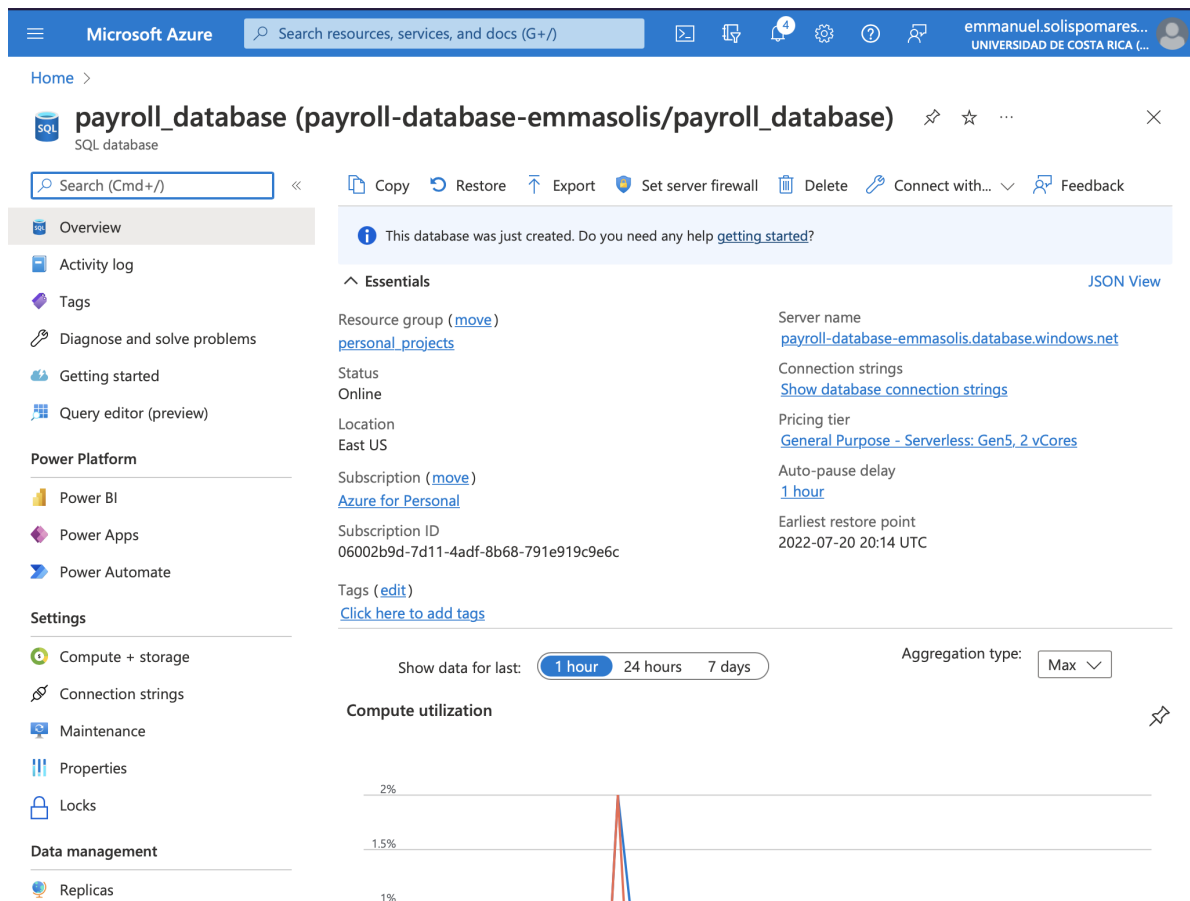


Figura 2: Vista general del de la base de datos *payroll* en Azure.

Alexandra, los diagramas se encuentran adjuntos en el archivo *sprint2/database\_diagrams.pdf*.

También se adjuntan los scripts que fueron usados para la creación de tablas, funciones, procedimientos y demás de la base de datos; estos se encuentran dentro de la carpeta *sprint2/database\_scripts/*.

## 10. Presentación durante Sprint Review

Se realizará la exposición el día 27 de Julio del 2022 debido a problemas técnicos con la base de datos fuera de nuestro alcance.

## 11. Autoevaluación y Coevaluación

Cada miembro del equipo se encarga individualmente de enviar su documento de auto y coevaluación.

Microsoft Azure Search resources, services, and docs (G+/)

Home > payroll-api-tabueno App Service

Search (Cmd+/) Browse Stop Swap Restart Delete Refresh Get publish profile

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Microsoft Defender for Cloud
- Events (preview)

Deployment

- Quickstart
- Deployment slots
- Deployment Center

Settings

- Configuration
- Authentication
- Application Insights (preview)
- Identity
- Backups
- Custom domains

Click here to access Application Insights for monitoring and profiling for your ASP.NET Core app. →

Essentials

Resource group (move) [personal\\_projects](#)

Status Running

Location Central US

Subscription (move) [Azure for Personal](#)

Subscription ID 06002b9d-7d11-4adf-8b68-791e919c9e6c

Tags (edit) [Click here to add tags](#)

URL <https://payroll-api-tabueno.azurewebsites.net>

Health Check [Not Configured](#)

App Service Plan [ASP-personalprojects-93cf \(\\$1: 1\)](#)

GitHub Project [https://github.com/emasp2001/pi\\_ingeBases-backend](https://github.com/emasp2001/pi_ingeBases-backend)

JSON View

Diagnose and solve problems

Our self-service diagnostic and troubleshooting experience helps you identify and resolve issues with your web app.

Application Insights

Application Insights helps you detect and diagnose quality issues in your apps, and helps you understand what your users actually do with it.

Http 5xx 100

Data In

Data Out

Figura 3: Vista general Backend API en Azure.