


## Desarrollo de un backend para la gestión del sistema penitenciario del Ecuador

*Development of a backend for the management of Ecuador's penitentiary system*

- <sup>1</sup> Byron Gustavo Loarte Cajamarca  <https://orcid.org/0000-0001-8954-8002>  
Universidad Escuela Politécnica Nacional, Escuela de Formación de Tecnólogos, Quito,  
Ecuador  
[byron.loarteb@epn.edu.ec](mailto:byron.loarteb@epn.edu.ec)



### Artículo de Investigación Científica y Tecnológica

Enviado: 07/07/2022

Revisado: 22/08/2022

Aceptado: 01/09/2022

Publicado: 16/09/2022

DOI: <https://doi.org/10.33262/concienciadigital.v5i3.2.2319>

Cítese:

Loarte Cajamarca, B. G. (2022). Desarrollo de un backend para la gestión del sistema penitenciario del Ecuador. *Conciencia Digital*, 5(3.2), 47-66.  
<https://doi.org/10.33262/concienciadigital.v5i3.2.2319>



*CONCIENCIA DIGITAL*, es una revista multidisciplinar, **trimestral**, que se publicará en soporte electrónico tiene como **misión** contribuir a la formación de profesionales competentes con visión humanística y crítica que sean capaces de exponer sus resultados investigativos y científicos en la misma medida que se promueva mediante su intervención cambios positivos en la sociedad. <https://concienciadigital.org>

La revista es editada por la Editorial Ciencia Digital (Editorial de prestigio registrada en la Cámara Ecuatoriana de Libro con No de Afiliación 663) [www.celibro.org.ec](http://www.celibro.org.ec)

Esta revista está protegida bajo una licencia *Creative Commons Attribution Non Commercial No Derivatives 4.0 International*. Copia de la licencia: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

**Palabras claves:**

Tiempo Real,  
Laravel, PHP,  
Sistema  
Penitenciario, API  
RESTFul

**Resumen**

**Introducción.** En el Ecuador, actualmente se han presentado un sin número de inconvenientes en lo que respecta al sistema penitenciario lo que ha provocado que no se disponga de medios tecnológicos para afrontar esta problemática de una vez por todas. Por otra parte, las últimas estadísticas muestran un déficit en la seguridad de los centros penitenciarios en el Ecuador lo que agrava aún más esta difícil situación ya que los controles para el ingreso de personas se realizan de forma manual. Es decir, un agente lleva el registro de todas las visitas que se realizan en hojas de papel y en raras ocasiones hacen uso de hojas de cálculo. Al igual que el registro de privados de libertad que dispone cada una de las cárceles se lo realiza de forma manual, lo que ocasiona una serie de inconvenientes como: no disponer el número exacto de privados de libertad que alberga cada una de las cárceles, no se dispone de una categorización de forma ordenada, información dispone únicamente en medios impresos, información desactualizada, no se dispone reportes estadísticos, etc. Por otra parte, el uso de medios tecnológicos al igual que la comunicación en tiempo real han tomado una mayor importancia en esta era digital. Es por esta razón, que el desarrollo y uso de aplicaciones web, aplicaciones móviles, aplicaciones de escritorio, aplicaciones web progresivas, entre otras han permitido automatizar tareas y llevar un mejor control de la información en pequeñas, medianas y grandes empresas. **Objetivo.** Desarrollar un backend para la gestión del sistema penitenciario del Ecuador. **Metodología.** El método aplicado fue deductivo con un enfoque cualitativo, con un diseño experimental y documentado. **Resultados.** Entre los resultados obtenidos en la investigación y la aplicación de las herramientas tecnológicas se ha logrado desarrollar un backend que les permita gestionar de forma adecuada el personal que labora dentro de las cárceles como son: directores, guardias y prisioneros. Además, de obtener información en tiempo real sobre los pabellones, cárceles y realizar reportes por parte de los guardias. **Conclusión.** Se concluye que la utilización de herramientas de desarrollo actuales y que sean escalables con el pasar del tiempo al igual que la implementación de una metodología de desarrollo de forma adecuada se ha logrado desarrollar una herramienta tecnológica que les permita a las máximas autoridades del sistema

---

penitenciario obtener información de manera detallada, eficiente, organizada y en tiempo real gracias al uso de la tecnología.

---

**Keywords:**

Real-Time,  
Laravel, PHP,  
Prison System,  
API RESTFul

**Abstract**

**Introduction.** In Ecuador, there are currently many problems in the penitentiary system, which has led to the lack of technological means to deal with this problem finally. On the other hand, the latest statistics show a deficit in the security of the penitentiary centers in Ecuador, which further aggravates this demanding situation, since the controls for the entry of people are conducted manually. An agent keeps a record of all visits on paper sheets and on rare occasions makes use of spreadsheets. Likewise, the register of inmates in each of the prisons is kept manually, which causes a series of inconveniences such as: not having the exact number of inmates in each of the prisons, not having an orderly categorization, and information is only available in printed media, outdated information, no statistical reports. On the other hand, the use of technological means, as well as real-time communication, have become more important in this digital era. For this reason, the development and use of web applications, mobile applications, desktop applications, and progressive web applications, among others, have allowed automating tasks and better control of information in small, medium, and large companies. **Objective.** To develop a backend for the management of Ecuador's penitentiary system. **Methodology.** The method applied was deductive with a qualitative approach, with an experimental and documented design. **Results.** Among the results obtained in the research and the application of the technological tools, it has been possible to develop a backend that allows them to manage the personnel working in the prisons, such as directors, guards, and prisoners. In addition, to obtain information in real-time about the wards, and prisons and to make reports by the guards. **Conclusion.** It is concluded that the use of current development tools that are scalable over time, as well as the implementation of an appropriate development methodology, has allowed the development of a technological tool that allows the maximum authorities of the penitentiary system to obtain detailed, efficient, organized information in real-time thanks to the use of technology.

---

## Introducción

En la actualidad el uso de un sistema de información basado en la web y que además sea en tiempo real supone una gran ventaja frente a los sistemas tradicionales ya que la información puede ser accedida desde cualquier parte del mundo las 24 horas del día los 7 días de la semana (Ruiz, 2021, p.2). Por otra parte, el surgimiento de nuevas tecnologías, lenguajes de programación, herramientas de desarrollo, Frameworks, librerías, patrones de desarrollo, etc. ha permitido que la industria del desarrollo de software se divida en dos estratos por un lado el desarrollo de aplicaciones del lado del cliente (frontend) y por otra el desarrollo de aplicaciones del lado del servidor (backend) y que gracias a la creación y el consumo de API's RESTful se logre una comunicación e intercambio de información sin importar el lenguaje de programación se esté utilizando (Cando, 2020, p.24). Es por esta razón, que en los últimos años diversas empresas de desarrollo de software han apostado por implementar este nuevo modelo de desarrollo para aprovecharse los beneficios que conlleva.

Por otra parte, existen 3 principales tecnologías que se deben dominar para la construcción de aplicaciones del lado del cliente que son: Html, Css, JS mientras que para la construcción de aplicaciones del lado del servidor existen varias tecnologías y lenguajes de programación que se deben dominar como: Java, JavaScript, PHP, Go, C++, C, Python, etc. (Guapi, 2022, p.3). Además, otro punto muy importante que se debe tener en cuenta al momento de comenzar el desarrollo de un nuevo software es el uso de algún Framework, ya que en la actualidad su uso es casi mandatorio (Guanoluisa, 2022, p.3).

Un Framework es un marco (entorno o ambiente) de trabajo que ofrece a los desarrolladores un conjunto de herramientas y directrices claramente establecidas en la construcción de un sistema software (Espinosa-Hurtado, 2021, p.134). Además, al ser una estructura previamente definida posibilita una serie de ventajas como: reutilización de código y componentes, facilitar el desarrollo de aplicaciones web, integración del patrón MVC, funcionalidades integradas, compatibilidad con librerías externas, etc. Es por esta razón, que en la actualidad existen diversos Frameworks para cada lenguaje de programación tales como: Spring Boot, Laravel, Django, Golang, Express, NestJS, etc.

Si bien un Framework permite agilizar el proceso de desarrollo del software otro punto importante es la forma en cómo se va a gestionar la información, para lo cual se dispone de dos tipos de Bases de Datos las de tipo relacionales (SQL) y no relaciones (NoSQL) (Bermeo, 2022, p.3). En ese sentido en una Base de datos relacional se prioriza el uso de tablas, columnas y sus respectivas relaciones para lograr mantener la integridad de los datos y una escalabilidad asegurada, mientras que una Base de datos NoSQL se establece que no requiere ningún tipo de relaciones y que a menudo se clasifican según su forma de almacenar los datos haciendo que las consultas para acceder a la información sean mucho más fáciles (Díaz, 2021, p.20). Además, es importante recalcar que cuando se

desarrolle una aplicación del lado del servidor y una Base de datos es necesario hacer uso de un ORM (Mapeo Objeto-Relacional) el cual es una técnica de programación para convertir las tablas de la Base de datos en una serie de clases que a menudo son llamadas entidades, evitando de esta manera hacer tediosas sentencias SQL ya que el ORM se encargará de realizarlas por nosotros sin importar el motor de Base de datos que se esté utilizando (Suárez, 2021, p.19).

Por otra parte, algunos de los aspectos que se debe tener en cuenta al momento de otorgar información hacia las aplicaciones del lado del cliente son las siguientes:

- Formato que se va a utilizar ya sea (XML o JSON)
- Niveles de acceso que pueden ser (endpoints públicos o protegidos)
- La disponibilidad de la información (pública o privada)

una vez identificado claramente lo que el equipo de desarrollo requiere y siempre enfocándose en las necesidades y objetivo del proyecto, comienza el proceso de la construcción de las API's utilizando para ello una arquitectura denominada REST la cual otorga una serie de reglas para la construcción de API's modernas y escalables en la actualidad, destacando las siguientes ventajas: un protocolo cliente/servidor, capa petición no debe mantener un estado, un conjunto de operaciones bien definidas, el uso de hipermedios, interfaz uniforme, etc. (Hernández, 2021, p.8).

Finalmente, con el constante crecimiento de las Tecnologías de la Información y su gran aporte hacia la sociedad en el presente artículo se presenta el desarrollo de un backend para la gestión del sistema penitenciario del Ecuador abarcando todos los puntos citados anteriormente, es decir se parte de una problemática real en el que, aplicando una serie de herramientas tecnológicas, patrones de diseño y buenas prácticas de codificación se dará una solución para que pueda ser implementada en cualquier cárcel del Ecuador si se lo requiere.

### Metodología

Para el desarrollo del backend se ha utilizado la metodología de desarrollo ágil denominada Scrum, ya que otorga una serie de ventajas para el cumplimiento del objetivo que se ha propuesto. Además, otorga una serie de roles, artefactos y una serie de entregables funcionales que son primordiales para satisfacer los requisitos del cliente (Velasco, 2021, p.438).

Como parte de la metodología de desarrollo ágil se ha hecho uso de los roles de trabajo (*Product Owner, Scrum Máster, Development Team*), los cuales permitirán que el backend se desarrolle de forma organizada y en base a los requisitos que se han planteado al inicio (Córdova, 2020, p.5). Por otra parte, los artefactos que se han utilizado como: Recopilación de Requisitos, Historia de Usuario, Product Backlog y Sprint Backlog han

permitido que el desarrollo del proyecto sea de manera organizada cumpliendo con los tiempos establecidos y sobre todo que el producto software resultante sea de calidad (Cajamarca, 2019, p.201). En ese sentido, la Recopilación de Requisitos se ha utilizado para listar todos los requisitos a manera de lista ordenada para saber cuál es el alcance del backend, así como requisitos funcionales y no funcionales, las Historias de Usuario han permitido describir uno de los requisitos anteriormente mencionados de una manera mucho más detallada, el Product Backlog ha permitido agrupar cada uno de los requisitos en base a la complejidad del desarrollo y en la prioridad del negocio y el Sprint Backlog ha permitido por una parte establecer las iteraciones que va a tener el desarrollo del backend y por otra parte, el detalle de cada una de las tareas que debe tener cada iteración al igual que el tiempo de entrega.

### *Iteraciones*

En base a lo citado anteriormente, se ha obtenido 6 iteraciones que se describen a continuación, como se ilustra en la tabla 1. No obstante, cabe recalcar que el primer Sprint Backlog se ha utilizado para definir algunas cuestiones que están estrechamente relacionadas con los requisitos técnicos previo a la etapa de codificación del backend y sus módulos respectivos (Cola, 2022, p.7).

**Tabla 1**

*Iteraciones: agrupación de cada historia de usuario para cada iteración*

Sprint Backlog	Tareas
Configuración del ambiente de desarrollo	Estructura del proyecto del backend. Roles de usuarios. Definición de requerimientos y restricciones. Diseño de la Base de datos SQL.
Módulo de autenticación y recuperación de contraseña	Crear un endpoint para el inicio sesión. Crear un endpoint para el restablecimiento de contraseña. Crear un endpoint para el envío y verificación de email. Crear un endpoint para el cierre de sesión.
Módulo de dashboard y actualización del perfil	Crear un endpoint para la actualización de perfil. Crear un endpoint para la actualización del avatar.
Módulo para la gestión de directores, guardias y prisioneros	Crear varios endpoints para la gestión de directores. Crear varios endpoints para la gestión de Guardias. Crear varios endpoints para la gestión de Prisioneros.

**Tabla 1**

*Iteraciones: agrupación de cada historia de usuario para cada iteración (continuación)*

Sprint Backlog	Tareas
Módulo de pabellones, cárceles y asignaciones	<p>Crear varios endpoints para la gestión de Pabellones.</p> <p>Crear varios endpoints para la gestión de Cárceles.</p> <p>Crear varios endpoints para la asignación de guardias a pabellones.</p> <p>Crear varios endpoints para la asignación de prisioneros a cárceles.</p>
Módulo para la gestión de reportes	<p>Crear varios endpoints para la gestión de Reportes.</p> <p>Pruebas en el backend.</p> <p>Despliegue a producción.</p>
Pruebas y despliegue	<p>Pruebas de aceptación.</p> <p>Pruebas de rendimiento.</p> <p>Despliegue a Heroku.</p>

**Resultados**

Una vez que se ha culminado con éxito la etapa de planificación en base a la metodología ágil de desarrollo Scrum, en esta sección se presenta los resultados que se han obtenido por cada iteración.

*Sprint 0: Configuración del ambiente de desarrollo*

El backend dispone del siguiente diseño de la Base de Datos relacional, el cual está gestionado por el SGBD MySQL como se ilustra en la figura 1, adicional a ello el backend se ha desarrollado con la utilización del editor de código Visual Studio Code, el Framework Laravel en la versión 9 y bajo el patrón de arquitectura MVC (Iza, 2021, p.8), como se ilustra en la figura 2, además se dispone de los siguientes roles como Administrador, Director, Guardia, Prisionero, los cuales tienen acceso a los módulos una vez que inicien sesión respectivamente, por último se dispone de una serie de restricciones que se detallan a continuación:

- El perfil administrador puede realizar la gestión de directores, guardias y prisioneros.
- El perfil director puede realizar la gestión de pabellones y cárceles.
- El perfil director puede realizar la asignación de cárceles a pabellones.
- El perfil director puede realizar la asignación de guardias y prisioneros.
- El perfil guardia puede realizar la gestión de reportes.
- El perfil prisionero no puede acceder al sistema web.

Figura 1

Diseño de la base de datos relacional

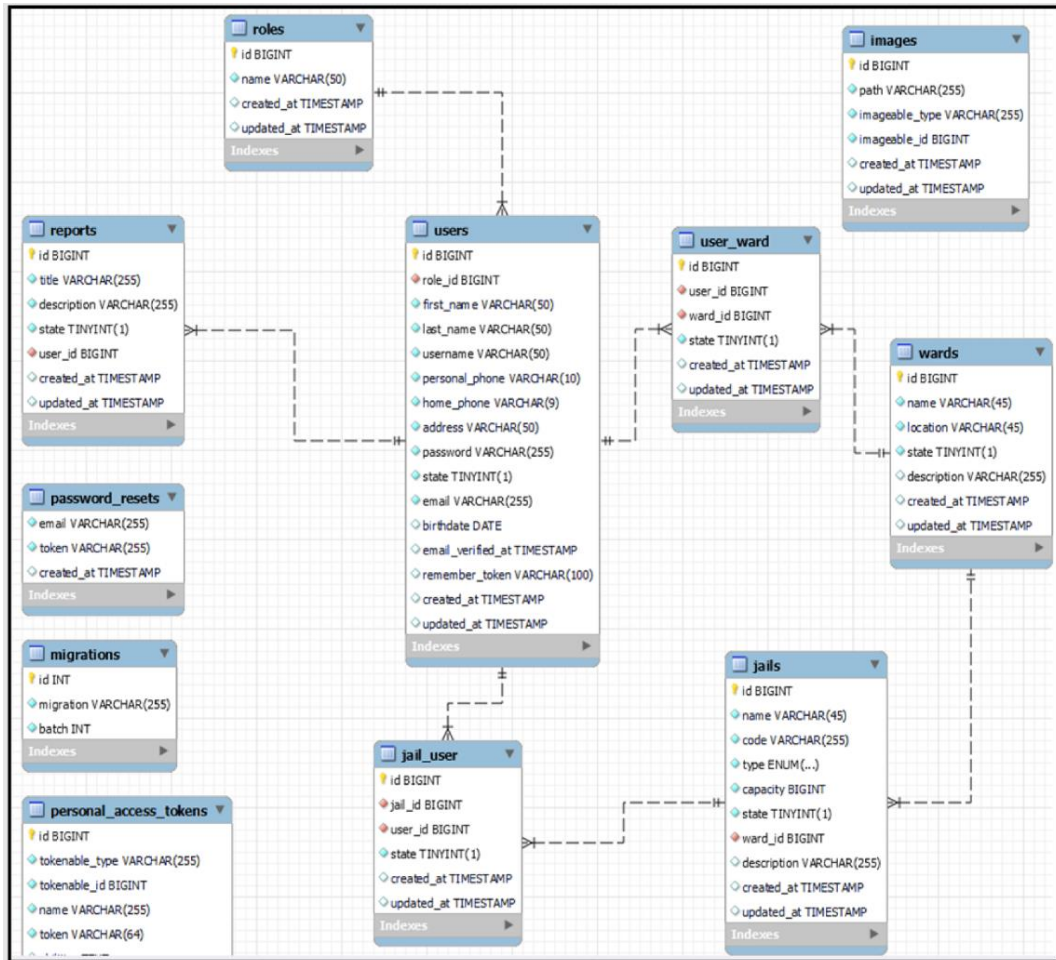
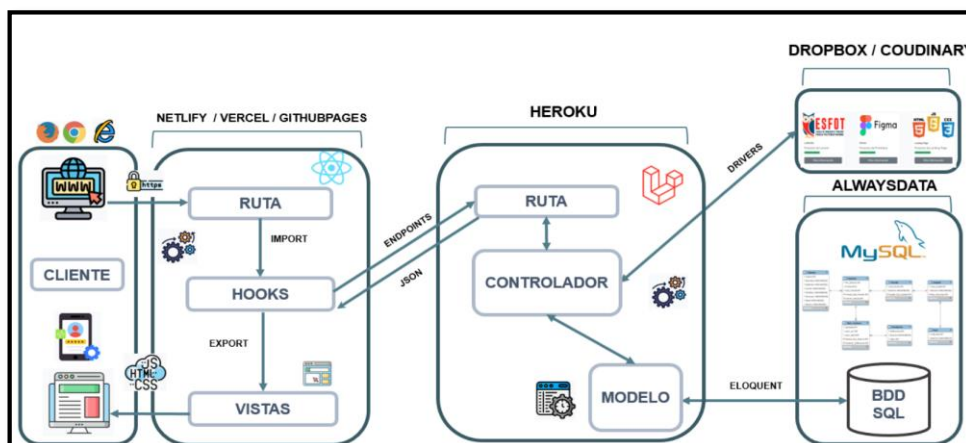


Figura 2

Arquitectura del backend



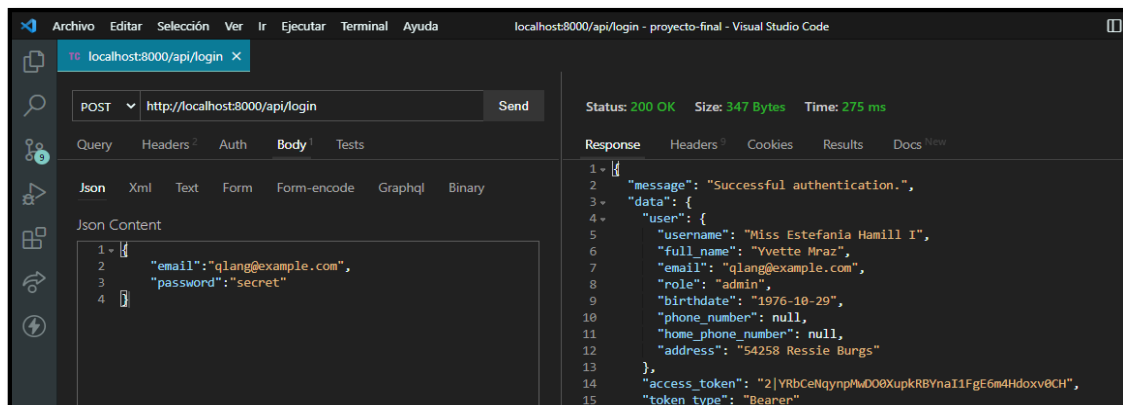


*Sprint 1: Módulo de autenticación y recuperación de contraseña*

Acorde a la planificación del Sprint Backlog el Sprint 1 corresponde al módulo de autenticación y recuperación de contraseña. En ese sentido, se ha creado un endpoint para el respectivo inicio de sesión como se ilustra en la figura 3, un endpoint para que los usuarios en el caso de que lo requieran puedan restablecer su contraseña por medio del envío de un email haciendo uso de la plataforma Mailtrap como se ilustra en la figura 4 y un endpoint para que los usuarios en el caso de que lo requieran puedan cerrar sesión como se ilustra en la figura 5.

**Figura 3**

*Endpoint para el inicio de sesión*



**Figura 4**

*Endpoint para restablecer contraseña*

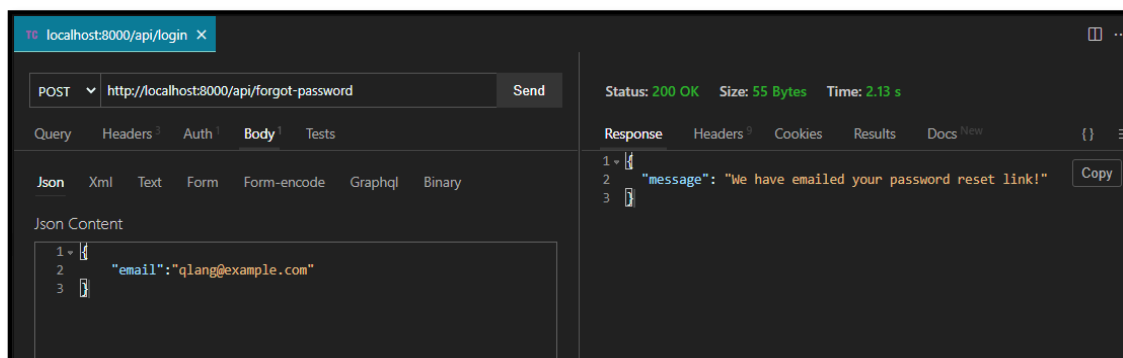
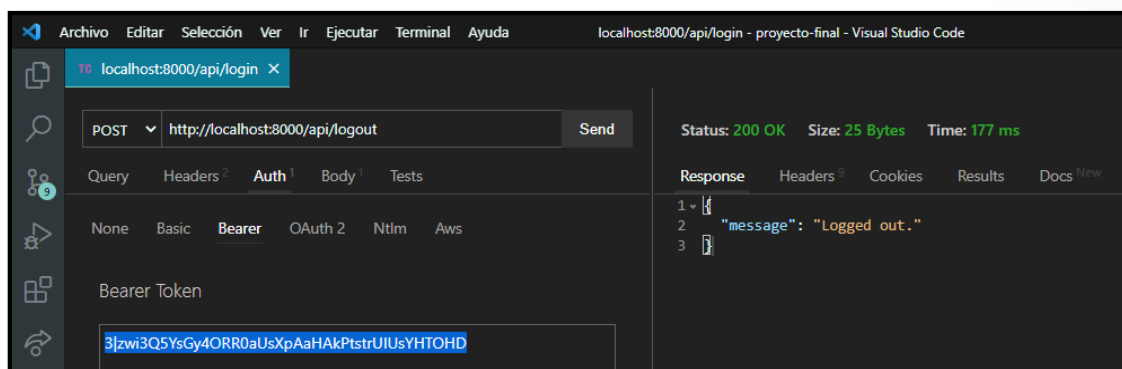


Figura 5

*Endpoint para el cierre de sesión*

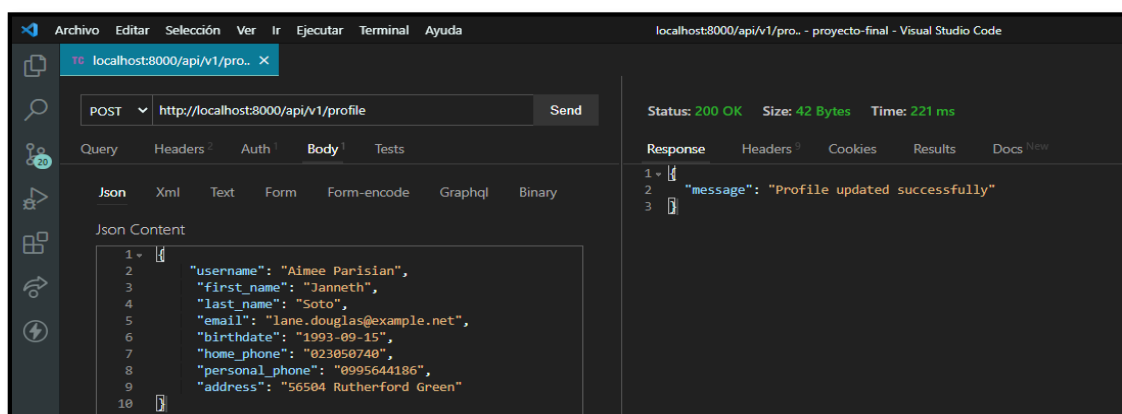


*Sprint 2: Módulo de dashboard y actualización del perfil*

Acorde a la planificación del Sprint Backlog el Sprint 2 corresponde al módulo de dashboard y actualización del perfil. En ese sentido, se ha creado un endpoint para que los usuarios en el momento de iniciar sesión puedan actualizar su información personal si lo requieren como se ilustra en la figura 6 y un endpoint para que los usuarios puedan actualizar su avatar personal si lo requieren como se ilustra en la figura 7.

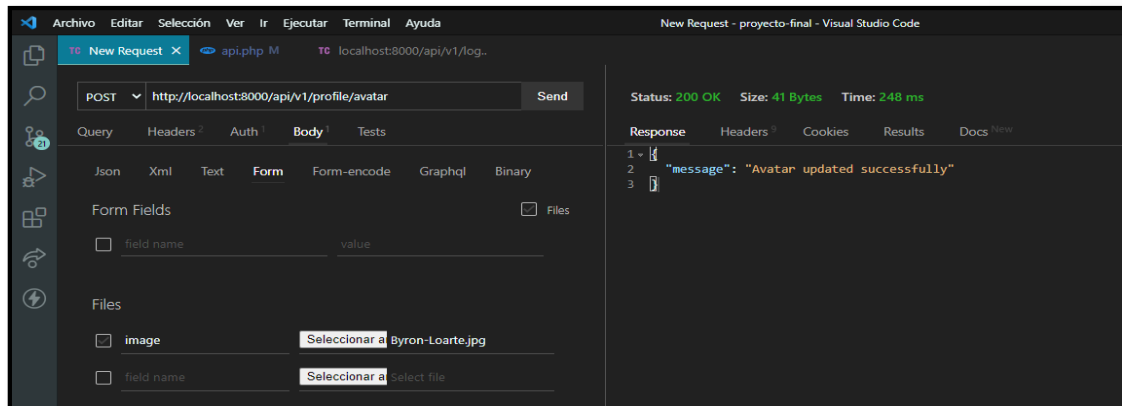
Figura 6

*Endpoint para actualiza información personal*



**Figura 7**

*Endpoint para actualiza avatar personal*



*Sprint 3: Módulo para la gestión de directores, guardias y prisioneros*

Acorde a la planificación del Sprint Backlog el Sprint 3 corresponde al módulo para la gestión de directores, guardias y prisioneros. En ese sentido, se han creado varios endpoints para realizar la gestión de los usuarios con perfil director, guardia y prisionero como se ilustran en las figuras 8, 9 y 10.

**Figura 8**

*Endpoint para listar directores*

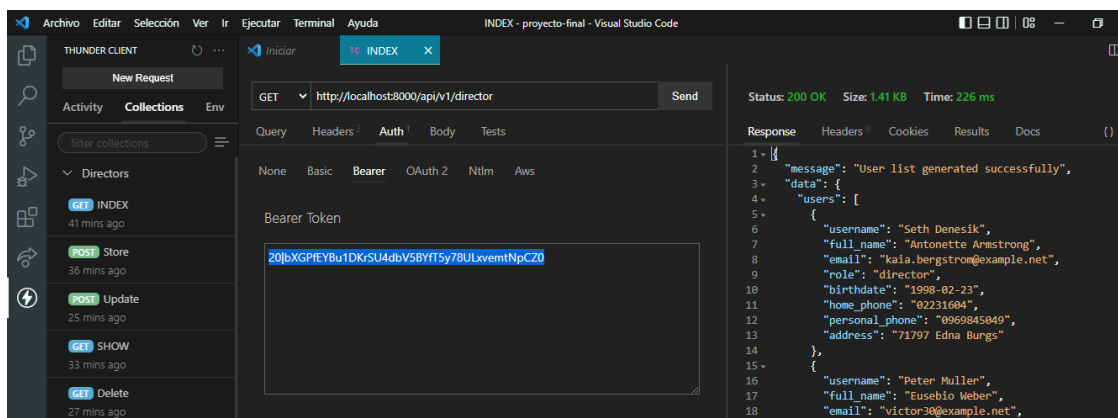


Figura 9

Endpoint para listar directores

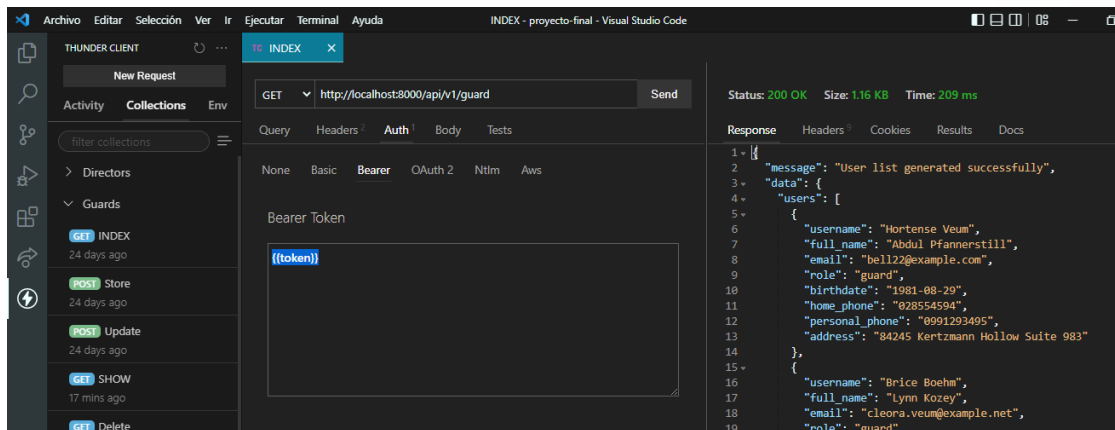
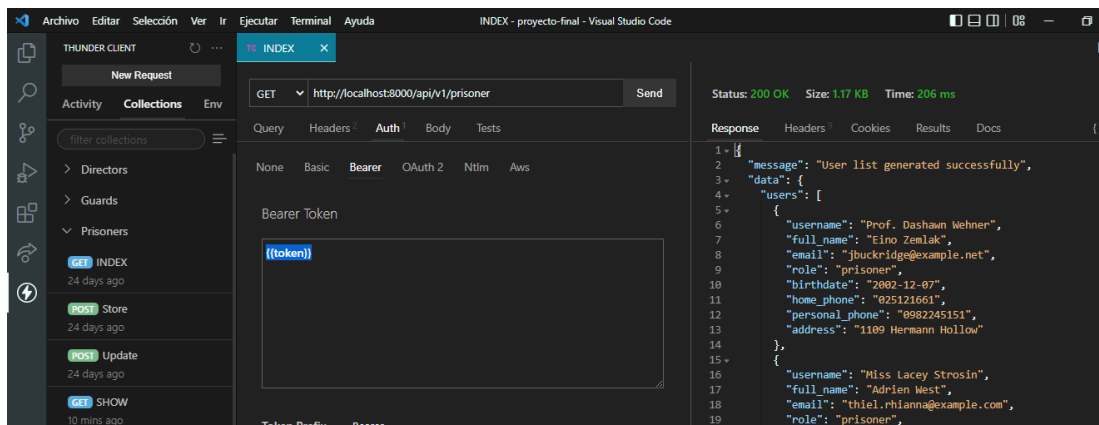


Figura 10

Endpoint para listar prisioneros



#### Sprint 4: Módulo de pabellones, cárceles y asignaciones

Acorde a la planificación del Sprint Backlog el Sprint 3 corresponde al módulo de pabellones, cárceles y asignaciones. En ese sentido, se han creado varios endpoints para realizar la gestión de pabellones y cárceles como se ilustran en las figuras 11 y 12, mientras que se han creado varios endpoints para realizar la asignación de guardias a pabellones como se ilustra en la figura 13 y varios endpoints para realizar la asignación de prisioneros a cárceles como se ilustra en la figura 14.

Figura 11

*Endpoint para listar pabellones*

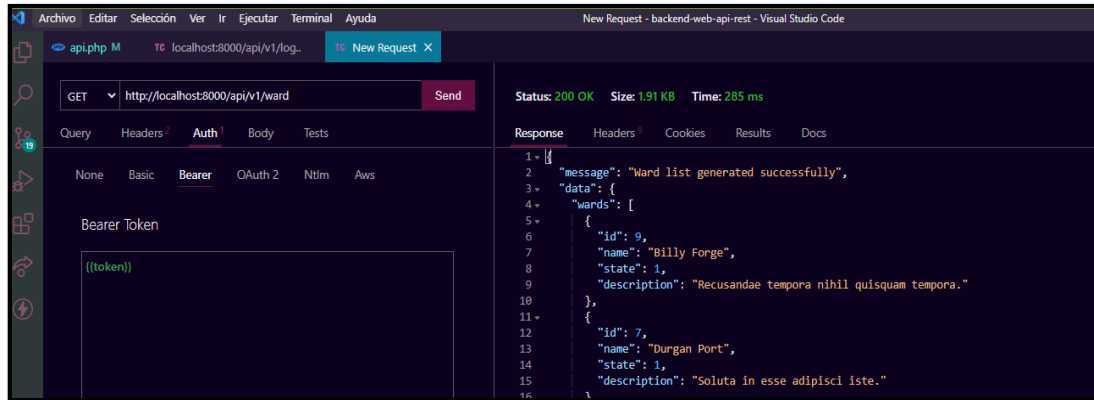


Figura 12

*Endpoint para listar cárceles*

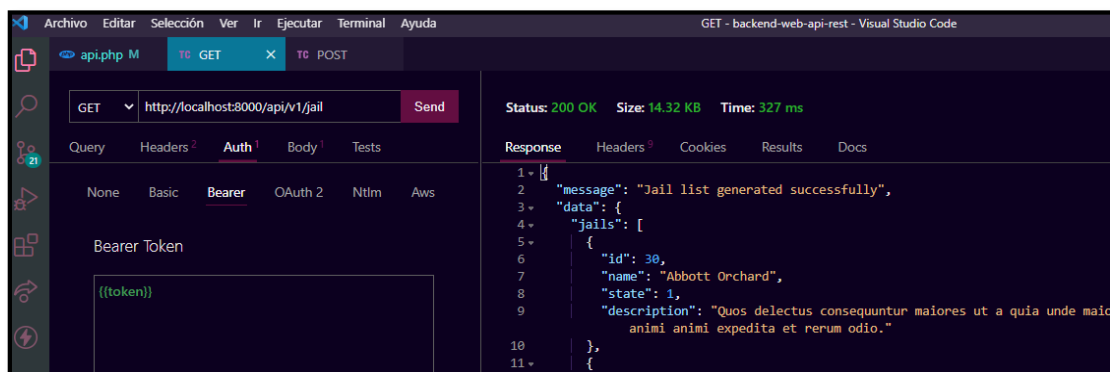


Figura 13

*Endpoint para asignar guardias a pabellones*

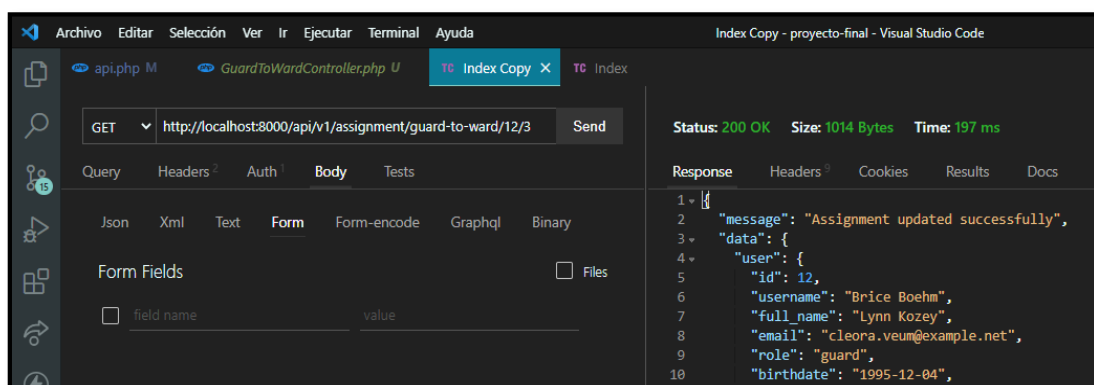
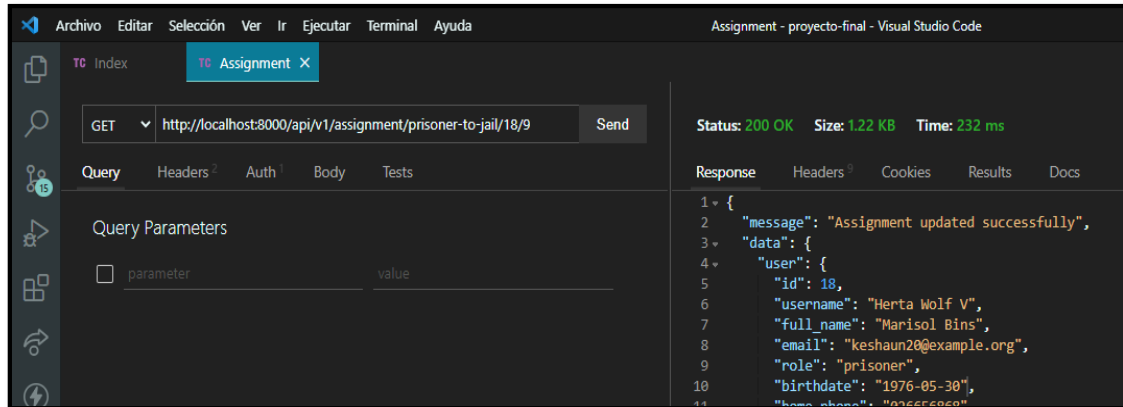


Figura 14

*Endpoint para asignar prisioneros a cárceles*

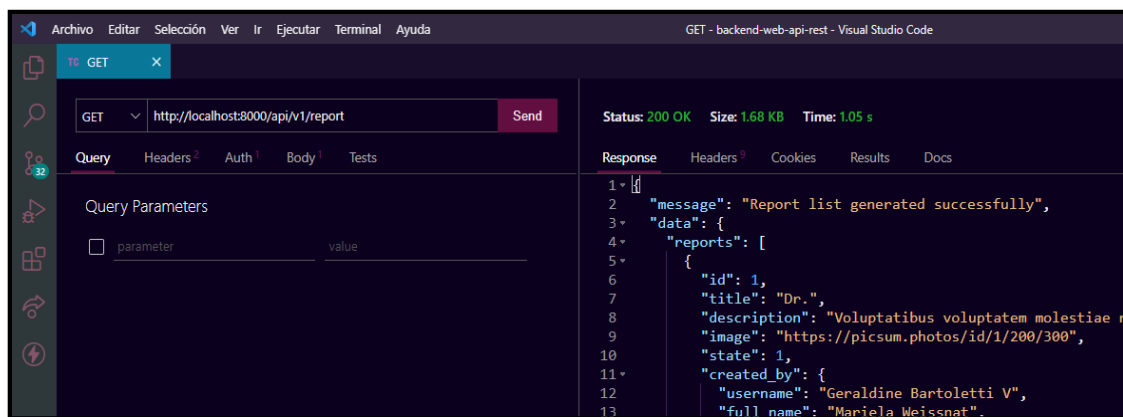


*Sprint 5: Módulo para la gestión de reportes*

Acorde a la planificación del Sprint Backlog el Sprint 5 corresponde al módulo para la gestión de reportes. En ese sentido, se han creado varios endpoints para realizar la gestión de reportes como se ilustra en la figura 15.

Figura 15

*Endpoint para listar los reportes creados*



*Sprint 6: Pruebas y despliegue*

Acorde a la planificación del Sprint Backlog el Sprint 6 corresponde a la sección de pruebas y respectivo despliegue a producción. En ese sentido y acorde a la metodología se procede con el apartado de pruebas para lo cual se ha ejecutado las siguientes pruebas:

### *Pruebas de aceptación*

Este tipo de pruebas permiten validar y aprobar todos los requerimientos funcionales como no funcionales por parte del dueño del producto (Chiluisa, 2014, p.69). Sin embargo, como lo dictamina la metodología cada una de estas pruebas se las fue ejecutando al finalizar cada Sprint para que no exista ningún inconveniente y de esta manera lograr avanzar con las demás iteraciones. A continuación, como se ilustra en la tabla 2 se presenta la plantilla que se ha utilizado para poder evidenciar la aprobación por parte del dueño del producto.

**Tabla 2**

#### *Prueba de aceptación*

Prueba de aceptación	
Identificador (ID): 1	Identificador Historia de Usuario: 1
Nombre prueba de aceptación: endpoint para el inicio de sesión.	
Descripción:	
El usuario administrador tiene la posibilidad de iniciar sesión cuando lo requiera por medio de un endpoint previamente definido.	
Pasos de ejecución:	
Ingresar los datos del usuario.	
Seleccionar el método HTTP.	
Establecer el token de autorización (si lo requiere).	
Presionar el botón enviar.	
Resultado deseado:	
Al ingresar toda la información el usuario puede ver toda su información personal al igual que un token de autorización para acceder a los endpoints protegidos.	
Evaluación de la prueba:	
Se visualiza el resultado deseado y la aprobación respectiva.	

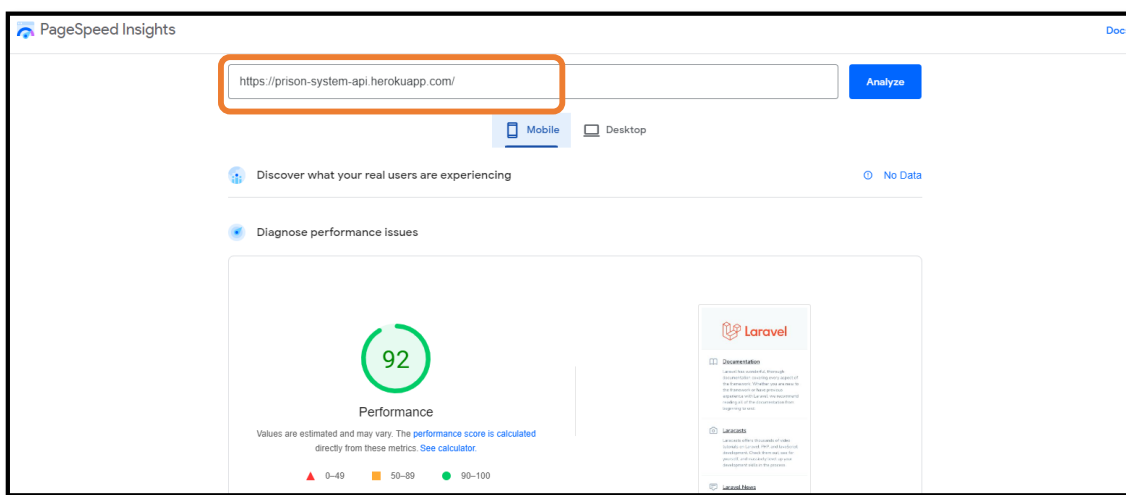
### *Pruebas de rendimiento*

A través de este tipo de pruebas se busca determinar el rendimiento del sistema software bajo una carga de trabajo definida utilizando diferentes tipos de pruebas de rendimiento tales como pruebas de carga, de estrés y de estabilidad (Valdez, 2022, p.26). En ese sentido, se ha medido el rendimiento de cada endpoint para determinar la respuesta frente

a diferentes condiciones y determinar si existe fallas hacer los cambios necesarios, caso contrario se puede pasar a la siguiente etapa. A continuación, se presenta un resultado de la ejecución de la prueba como se ilustra en la figura 16.

**Figura 16**

*Resultado de la prueba de rendimiento*

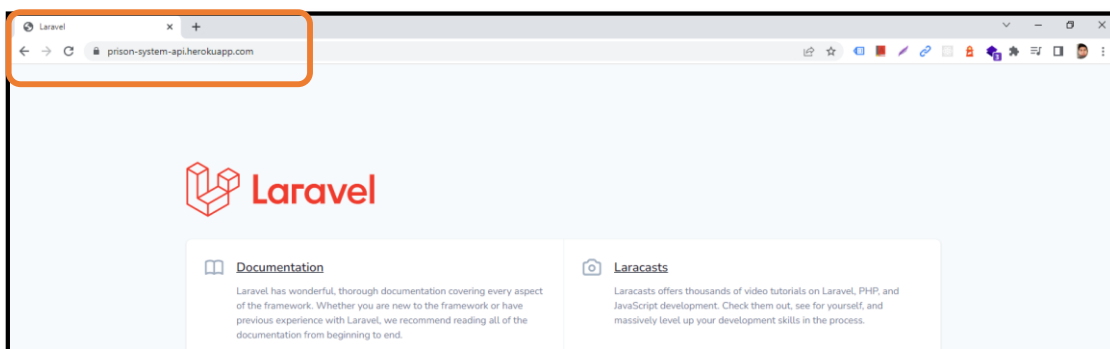


### Despliegue

Una vez que se ha finalizado con éxito la ejecución de pruebas y la aprobación del dueño de producto en esta sección se procede a realizar el despliegue a producción del backend para que de esta manera pueda ser consumido por cualquier aplicación del lado de cliente o móvil si se lo requiere. Es por esta razón, que se ha utilizado la plataforma Heroku para realizar el despliegue a producción como se ilustra en la figura 17 la cual puede ser accedida desde cualquier navegador web o cliente HTTP.

**Figura 17**

*Despliegue a producción del backend*





## Conclusiones

- Al finalizar el desarrollo del backend se dispone de un medio tecnológico para que las máximas autoridades del sistema penitenciario puedan implementarlo en cada una de las cárceles del país y con ello llevar un mejor control sobre la información que se genera diariamente.
- Gracias a la implementación de una metodología de desarrollo ha permitido que el backend pueda ser completado etapa por etapa y validando las funcionalidades en los distintos periodos de tiempo que se han planificado.
- Una adecuada implementación de los artefactos de la metodología Scrum ha permitido que el desarrollo del backend se lo realice de forma ordenada, modular y centrándose siempre en el objetivo que se ha propuesto.
- Gracias al uso de diferentes herramientas, librerías y un Framework de desarrollo moderno como lo es Laravel ha permitido que el desarrollo de cada uno de los endpoints se lo pueda realizar sin tanto esfuerzo ya que los desarrolladores únicamente tienen que centrarse en la lógica del negocio y no desperdiciar su esfuerzo en configuraciones adicionales.
- La incorporación de una Base de datos relacional es un factor muy importante donde se tiene proyectado que la gestión de la información se realice a través de una gran cantidad de registros. Es por esta razón, que se ha utilizado este tipo de Base de datos garantizado las propiedades ACID de las bases de datos.
- El disponer de una serie de endpoints por parte del backend, permite que cualquier aplicación del lado del cliente o móvil pueda consumir esta información sin ningún problema, al igual que el backend está a disposición de ser escalable a futuro si se lo requiere para poder integrar base de datos NoSQL, microservicios y requerimientos de las actuales tecnologías.
- Las pruebas que se han ejecutado para el backend han permitido verificar la eficiencia tanto en módulos como en funcionalidades y comprobar si realmente se cumple con los requerimientos que se han establecido al inicio del proyecto.

## Referencias bibliográficas

- Bermeo Panchi, N. L. (2022). *Desarrollo de un sistema para la interacción de los miembros de la comunidad de la Iglesia IFGF*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/22678>
- Cajamarca, B. G. (2019). Desarrollo de una aplicación web y móvil en tiempo real, una evolución de las aplicaciones actuales. *Ciencia Digital*, 3(1), 201-216. <https://doi.org/10.33262/concienciadigital.v3i1.282>

- Cando Obaco, J. R. (2020). *Desarrollo de sistema web y aplicación móvil para la gestión del barrio San Isidro de Puengasí-Quito*. Bachelor's thesis, Quito. <https://bibdigital.epn.edu.ec/handle/15000/21051>
- Chiluisa Pallo, A. P. (2014). *Desarrollo e implantación del sistema de control de inventarios y gestión de laboratorios para la de la facultad de Ciencias*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/7732>
- Cola Chicaiza, G. L. (2022). *Desarrollo de sistema para reportar el acoso sexual en Instituciones de Educación Superior-Quito*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/22040>
- Córdova Córdova, M. F. (2020). *Desarrollo de sistema web y aplicación móvil para la gestión de información odontológica en el centro “Jema Dental”*. Bachelor's thesis, EPN. <https://bibdigital.epn.edu.ec/handle/15000/21093>
- Díaz Erazo, A. D. (2021). *Análisis comparativo de rendimiento en operaciones de escritura para bases de datos SQL y NoSQL*. Bachelor's thesis, Universidad Central del Ecuador, Quito. <http://www.dspace.uce.edu.ec/handle/25000/24230>
- Espinosa-Hurtado, R. (2021). Análisis comparativo para la evaluación de frameworks usados en el desarrollo de aplicaciones web. *CEDAMAZ*, 11(2), 133-141. <https://doi.org/10.54753/cedamaz.v11i2.1182>
- Guanoluisa Quiroz, M. A. (2022). *Desarrollo de un sistema para la interacción de los miembros de la comunidad de la iglesia IFGI*. Bachelor's thesis, Escuela Politécnica Nacional, ESFOT, Quito. <https://bibdigital.epn.edu.ec/bitstream/15000/22678/1/CD%2012161.pdf>
- Guapi Mullo, F. D. (2022). *Desarrollo de aplicación móvil de realidad aumentada para el aprendizaje de la materia de ecología*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/22670>
- Hernández, L. M. (2021). Arquitectura REST para el desarrollo de aplicaciones web empresariales. *Revista Electrónica sobre Tecnología, Educación y Sociedad*, 8(15). <https://www.ctes.org.mx/index.php/ctes/article/view/748>
- Iza Viracocha, J. A. (2021). *Desarrollo de sistema web para el control de mantenimientos y equipos del laboratorio de tecnología industrial de la ESFOT*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/21405>
- Ruiz, A. P. (2021). *Soporte de aplicaciones de tiempo real en dispositivos móviles*. Doctoral dissertation, Universidad de Cantabria. <https://repositorio.unican.es/xmlui/handle/10902/21642>

Suárez, G. L. (2021). *Un análisis comparativo de herramientas ORM*. Doctoral dissertation, Universidad Nacional de La Plata. <http://sedici.unlp.edu.ar/handle/10915/127331>

Valdez, D. J. (2022). *Desarrollo de un Sistema Web informativo de trámites académicos y administrativos para la ESFOT*. Bachelor's thesis, EPN, Quito. <https://bibdigital.epn.edu.ec/handle/15000/22115>

Velasco, M. V. (2021). Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software. *Dominio de las Ciencias*, 7(4), 54. <http://dx.doi.org/10.23857/dc.v7i4>



El artículo que se publica es de exclusiva responsabilidad de los autores y no necesariamente reflejan el pensamiento de la **Revista Conciencia Digital**.



El artículo queda en propiedad de la revista y, por tanto, su publicación parcial y/o total en otro medio tiene que ser autorizado por el director de la **Revista Conciencia Digital**.



#### Indexaciones

