

# Desarrollo de una herramienta web para el diseño y aplicación de exámenes basados en la evaluación por competencias para el 5.<sup>o</sup> grado de la Institución Educativa Julio Caicedo y Téllez

Luis Eduardo Cardozo Agredo <sup>1</sup>

Rodolfo Córdoba Santamaría <sup>2</sup>

Ricardo Andrés Almeida Delgado, M. Sc. <sup>3</sup>

César Augusto Gutiérrez Rodríguez, M. Sc <sup>4</sup>

*Recibido: 06-05-2020*

*Aceptado: 16-06-2020*

## Resumen

El Ministerio de Educación Nacional, a través del ICFES, aplica periódicamente unas evaluaciones denominadas pruebas Saber, que tienen como objetivo contribuir al mejoramiento de la calidad de la educación colombiana. En el grado quinto de la Institución Educativa Oficial Julio Caicedo y Téllez de Cali se evidencian bajos resultados en las pruebas en las competencias de lenguaje y matemáticas, por lo que se busca una estrategia que permita

---

1. Ingeniero de sistemas, Universidad Santiago de Cali.  
Correo electrónico: [luis.cardozo01@usc.edu.co](mailto:luis.cardozo01@usc.edu.co)  
ORCID: <https://orcid.org/0000-0002-2060-9791>  
CvLAC: 00018110382020721933  
Google Scholar: [luisedu.cd@gmail.com](https://scholar.google.com/citations?user=luisedu.cd@gmail.com)

2. Ingeniero de sistemas, Universidad Santiago de Cali.  
Correo electrónico: [rodolfo.cordoba@usc.edu.co](mailto:rodolfo.cordoba@usc.edu.co)  
ORCID: <https://orcid.org/0000-0002-7981-0529>  
CvLAC: 000179389720206171859  
Google Scholar: [rocosan1@gmail.com](https://scholar.google.com/citations?user=rocosan1@gmail.com)

3. Magíster en Informática. Docente de tiempo completo de la Universidad Santiago de Cali.  
ORCID: <https://orcid.org/0000-0003-2582-2484>.  
Correo electrónico: [ricardo.almeida00@usc.edu.co](mailto:ricardo.almeida00@usc.edu.co)

4. Magíster en Informática Educativa. Docente de la Universidad Nacional Abierta y a Distancia.  
ORCID: <https://orcid.org/0000-0001-5536-4264>.  
Correo electrónico: [profecesar15@gmail.com](mailto:profecesar15@gmail.com)

mejorarlas y a su vez motivar a los estudiantes. El presente artículo describe el análisis, diseño y desarrollo de una herramienta web para la creación y aplicación de evaluaciones por competencias para grado quinto, desarrollada mediante la metodología Agile Unified Process con algunos instrumentos de la metodología Scrum, la cual dio como resultado un prototipo funcional con una arquitectura de *software* moderna. Tras este resultado, se evidenció una percepción positiva por parte de los docentes y estudiantes de la institución, quienes pilotearon el prototipo final.

**Palabras clave:** evaluaciones por competencias, pruebas Saber, metodología ágil de *software*, TIC educativas, entornos virtuales de aprendizaje, evaluaciones en línea.

## Web tool development for design and application of based on competencies assessments for Julio Caicedo y Téllez Educational Institute's 5th grades

### Abstract

The Ministry of National Education through ICFES, periodically applies evaluations called pruebas Saber, which aim to contribute to the improvement of the quality of Colombian education. In the fifth grade of the Official Educational Institution Julio Caicedo and Téllez de Cali, there are low test results in language and math skills, so a strategy is sought that allows them to improve them and in turn motivate students. This article describes the analysis, design and development of a web tool for the creation and application of competency assessments for fifth grade, developed using the Agile Unified Process methodology with some SCRUM instruments, resulting in a functional prototype with an architecture of modern software As a result, a positive perception was evidenced by the

teachers and students of the institution, who piloted the final prototype.

**Keywords:** competency assessments, pruebas saber, agile software methodology, educational ICT, virtual learning environments, online assessments.

## Introducción

En Colombia, las pruebas Saber constituyen una vía para evaluar periódicamente la educación y valorar el alcance de las metas fijadas por el Gobierno frente al programa Colombia, la Mejor Educada (MEN, 2015). Estas son evaluaciones que deben realizar los alumnos para mostrar los avances logrados en determinados ciclos escolares (3.º, 5.º, 9.º, 11.º). Saber 5.º evalúa a los alumnos de 5.º grado mediante la aplicación de pruebas censales establecidas por el MEN y ejecutadas por el Instituto Colombiano para la Evaluación de la Educación (Icfes), cuyo objetivo es dar a conocer cuáles han sido los alcances en el desarrollo de competencias básicas en lenguaje y matemáticas, y a partir de estos resultados definir planes de mejoramiento en sus respectivos ámbitos de actuación (Icfes, 2017).

Cuando el MEN habla de competencias, hace referencia a “la capacidad de usar los conocimientos en situaciones diferentes de aquellas en las que se aprendieron. Implica la comprensión del sentido de cada actividad, así como de sus implicaciones éticas, sociales, políticas y económicas” (MEN, 2006). Las competencias son medidas a través de la relación presente entre la cantidad de preguntas que responde correctamente el estudiante con el nivel de profundización de estas; a esto se le llama “desempeño” y ha sido especificado por el Icfes en 4 niveles: insuficiente, mínimo, satisfactorio y avanzado, que consisten en una descripción cualitativa

de las habilidades y conocimientos que podrían tener si se ubican en determinado nivel (Icfes, 2017)

La Institución Educativa Julio Caicedo y Téllez de Cali, perteneciente al sector oficial, refleja resultados en las pruebas Saber 5.º que no cumplen con los objetivos trazados dentro de la visión de la institución. Dichas pruebas fueron presentadas por los estudiantes en el año 2017 en las áreas de lenguaje, con un desempeño satisfactorio, y matemáticas, con un desempeño mínimo; en esa aplicación la prueba se realizó a un 70 % de los alumnos de forma escrita y a un 30 % por primera vez de forma virtual. En la tabla 1 se puede apreciar el desempeño alcanzado por la institución en el 2017.

**Tabla 1.** Reporte de resultados de pruebas Saber de estudiantes de grado 5.º de la Institución Educativa Julio Caicedo y Téllez (septiembre de 2017).

| Áreas  | Lenguaje      | Matemáticas |
|--|---------------|-------------|
| Puntaje promedio del colegio                           | 324           | 306         |
| Nivel de desempeño del colegio                         | Satisfactorio | Mínimo      |
| Porcentaje de estudiantes del colegio en insuficiente  | 7 %           | 37 %        |
| Porcentaje de estudiantes del colegio en mínimo        | 41 %          | 31 %        |
| Porcentaje de estudiantes del colegio en satisfactorio | 35 %          | 20 %        |
| Porcentaje de estudiantes del colegio en avanzado      | 17 %          | 12 %        |

**Fuente:** Icfes (2018).

Con el objetivo de fortalecer el modelo educativo y hacer seguimiento del alcance de las competencias desarrolladas por los estudiantes de grado 5.º, la institución educativa ha determinado fortalecer el uso pedagógico de tecnologías de la información y la comunicación (TIC) como parte de su gestión académica. A partir de lo anterior se formula la siguiente pregunta problema:

- ¿Cómo construir una herramienta web que permita la ejecución de evaluaciones basadas en competencias para los alumnos de grado 5.º de básica primaria en las áreas de lenguaje y matemáticas?

Para verificar la necesidad de construir la herramienta denotada en el planteamiento del problema, se trabajaron dos actividades:

- Encuesta de argumentos: dirigida a un grupo de diez personas entre docentes y directivos docentes; compuesta por diez preguntas con los objetivos de establecer el conocimiento acerca la estructura de las pruebas Saber 5.º, determinar la visión sobre las oportunidades de mejora, identificar la opinión sobre la inclusión de los padres de familia y comprobar la aceptación y disposición a una herramienta web como nueva alternativa. La encuesta arrojó como resultado una fuerte disposición hacia la posibilidad de mejora en el desempeño de las pruebas Saber, falta de medios suficientes para el seguimiento del desarrollo de las competencias de los estudiantes y una aceptación generalizada para el uso de una nueva herramienta, tal como se presenta en la tabla 2.
- Revisión de herramientas existentes: se tomó una muestra de seis aplicaciones: Plexi (Icfes, s. f.), QuestBase (Fidenia, 2019), ProProfs (ProProfs, 2019) y Easy LMS (Quizworks, 2019). En la figura 1 se observa el número de herramientas que satisfacen cada necesidad inicial referida por la institución.

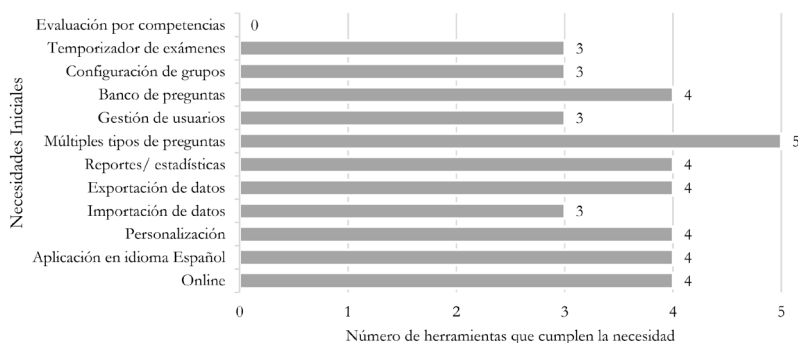
**Tabla 2.** Resultados de encuesta de argumentos.

| Preguntas de la encuesta  | De acuerdo | No de acuerdo |
|---|------------|---------------|
| ¿Conoce usted la estructura de las preguntas con que se realizan las pruebas Saber 5.º?   | 9          | 1             |
| ¿Considera que los estudiantes están familiarizados con la estructura de las preguntas realizadas en las pruebas Saber 5.º?                                     | 7          | 3             |
| ¿Considera que existe una posibilidad de mejora en los resultados de las pruebas Saber de sus estudiantes?  | 10         | 0             |
| ¿Cree usted que realizar simulacros de pruebas saber de manera periódica pueda contribuir a mejorar los resultados de los estudiantes en las pruebas Saber 5.º? | 10         | 0             |
| ¿Cuenta con los medios suficientes para determinar el nivel de preparación de cada estudiante para afrontar las pruebas Saber 5.º?                              | 3          | 7             |
| ¿Le gustaría poder contar con un conjunto de preguntas preestablecidas con estructura de las pruebas Saber 5.º que le ayuden a crear exámenes de forma ágil?    | 7          | 3             |

| Preguntas de la encuesta  | De acuerdo | No de acuerdo |
|---|------------|---------------|
| ¿Considera útil una herramienta web que permita crear evaluaciones con preguntas que tengan la estructura de las pruebas Saber 5.º?   | 10         | 0             |
| ¿Le gustaría poder generar estadísticas del desempeño de los estudiantes por competencias de acuerdo con una evaluación realizada a sus estudiantes?  | 9          | 1             |
| ¿Cree conveniente que los padres de familia puedan acceder a una herramienta web donde puedan hacer seguimiento a los resultados obtenidos por sus hijos en las diferentes pruebas enfocadas en el Saber 5.º?   | 10         | 0             |
| ¿Estaría dispuesto a utilizar un software que le permita crear e implementar evaluaciones que tengan preguntas con la estructura de las pruebas Saber, y que usando esos resultados le permitan realizar el seguimiento del desempeño de cada estudiante? | 10         | 0             |

**Fuente:** elaboración propia.

**Figura 1.** Herramientas que cumplen con el requerimiento.



**Fuente:** elaboración propia.

El análisis refleja que existen varias herramientas en el mercado que cumplen con la mayoría de las necesidades iniciales; sin embargo, ninguna de las herramientas verificadas cumple con la propiedad de “evaluación por competencias”, la cual es característica fundamental en el modelo de evaluaciones definido por el Icfes para las pruebas Saber, y por tanto este se convierte en un rasgo innegociable para la institución.

Con el propósito de proveer dicha solución tecnológica, se define como objetivo general desarrollar una herramienta web que

permita la creación y ejecución de evaluaciones por competencias en las áreas de lenguaje y matemáticas para los alumnos de grado 5.º de básica primaria de la Institución Educativa Julio Caicedo y Téllez; el cual se expresa a través de los siguientes objetivos específicos:

- Recolectar y analizar los requerimientos, con base en las necesidades de la institución educativa, que permitan determinar los módulos a implementar.
- Diseñar y modelar la arquitectura de los módulos, que serán definidos con base en los requerimientos de la institución educativa.
- Ejecutar el desarrollo del prototipo inicial.

Para la construcción del proyecto, se utiliza una metodología de desarrollo de software ágil (Pressman, 2010), la cual permite mayor flexibilidad y está cimentada en los principios establecidos en el Manifiesto ágil (Beck et al., 2019), tal como expresa Laínez (2015):

En las metodologías ágiles los proyectos son divididos en iteraciones cortas que generalmente tienen una o algunas pocas semanas de duración. Cada iteración es realizada en una reunión de planificación para definir lo que será realizado durante la iteración. Su objetivo es identificar junto con el cliente lo que podrá agregar más valor a su negocio. (p. 76)

La metodología seleccionada fue *agil unified process* (AUP), la cual es una versión simplificada del *rational unified process* (RUP); esta metodología proporciona mucha más importancia a la construcción del *software* que a la creación excesiva de documentación y permite una interacción constante con el usuario final. La AUP "es una forma fácil de entender el enfoque del desarrollo de *software* en la construcción de aplicaciones de negocio usando técnicas ágiles y ciertos conceptos que aún se mantiene fieles al RUP" (Ambler, 2005). Adicionalmente, en la ejecución de las diferentes fases de la

AUP se introdujeron herramientas y procesos del marco de trabajo Scrum para reforzar la gestión del proyecto, específicamente en las iteraciones planteadas, las cuales fueron gestionadas con los “sprints” de Scrum (SCRUMstudy, 2017, p. 25).

La AUP cuenta con 4 fases, a saber: inicio, elaboración, construcción y transición. Las características AUP que resaltan dentro su implementación son las siguientes (Ambler, 2002):

- Fragmentación del proyecto en proyectos más pequeños.
- Cada pequeño proyecto que se genere de esa fragmentación es una iteración.
- Las iteraciones deben tener un apropiado seguimiento.
- Cada iteración trata un conjunto de casos de uso.

## Metodología

El presente artículo, según la profundización, se define como un estudio descriptivo (Hernández, Fernández y Baptista, 2014) en el que se busca referir de manera detallada el proceso ejecutado para la obtención del objetivo general.

De igual manera, se aborda como una investigación de tipo aplicada (Vargas, 2009), por cuanto se enfoca en encontrar mecanismos, metodologías y estrategias para lograr un objetivo en concreto, dentro de un ámbito muy específico y bien delimitado, en este caso, la utilización de metodologías de desarrollo de software que permitirán diseñar una herramienta web de evaluaciones alineadas con las especificaciones de las pruebas Saber 5.º.



## Metodología de software ágil: AUP y Scrum

En esta sección se desarrolla el análisis, diseño y desarrollo de la herramienta web haciendo uso de la metodología seleccionada.

### Fase de inicio

Inicialmente, se elaboraron las primeras historias de usuario donde se representaron las funcionalidades de alto nivel conocidas en Scrum como “épicas” (SCRUMstudy, 2017, p. 39); es decir que en esta primera parte se identificaron los módulos de la herramienta web, como se observa en la tabla 3.

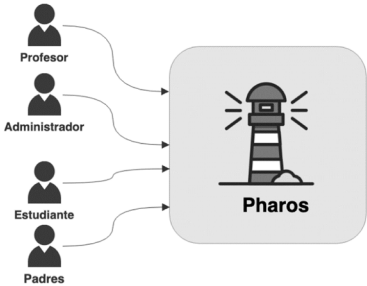
**Tabla 3.** Listado de épicas.

| ID | Épica                                    |
|----|--|
| 1  | Crear módulo de gestión de exámenes      |
| 2  | Crear módulo de gestión de preguntas     |
| 3  | Crear módulo de presentación de exámenes |
| 4  | Crear módulo de gestión de usuarios      |
| 5  | Crear módulo de parametrización          |

**Fuente:** elaboración propia.

Se continuó con la identificación de los usuarios, que son quienes hacen uso de la herramienta web; como se puede apreciar en la figura 2 se identificaron 4 tipos de usuarios.

**Figura 2.** Usuarios de la herramienta web.



**Fuente:** elaboración propia.

Respecto a la interacción de los usuarios (figura 1) con la herramienta web:

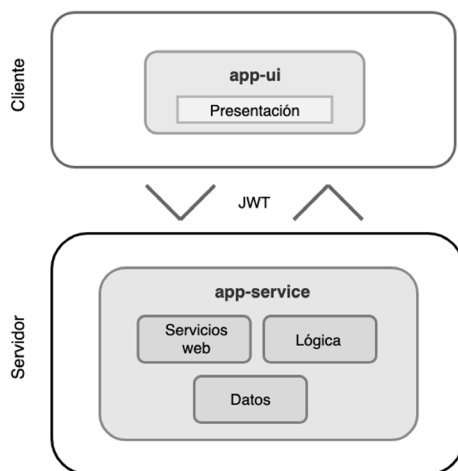
- Profesor: son los docentes registrados de la institución, quienes dentro de la herramienta pueden crear evaluaciones y preguntas, ejecutar las evaluaciones creadas, consultar estadísticas de resultados y gestionar los alumnos de sus cursos.
- Administrador: es el usuario administrador del sistema, el cual se encarga de manejar la parametrización (cursos, competencias, componentes, etc.) y la administración de usuarios.
- Estudiante: son los alumnos registrados, los cuales están habilitados para presentar las evaluaciones creadas por los profesores y pueden consultar sus resultados.
- Padres: son los tutores legales de los estudiantes; estos usuarios pueden consultar los resultados de las pruebas presentadas por sus tutelados.

Finalmente, se define la arquitectura del sistema como una API-REST, debido a que permite construir aplicaciones que mantienen una separación entre el cliente y el servidor, proporcionando fiabilidad, escalabilidad y flexibilidad en el desarrollo (Richardson y Ruby, 2007). Adicionalmente, para la construcción de este tipo de aplicaciones se considera el uso de interfaces uniformes, diseño de recursos y diseño de URI que no manejan estado y son cacheables (Allamaraju, 2010).

Tomando en cuenta la separación mencionada, se construyó una single page application (SPA) para el front-end, que contiene la vista de usuario, proporcionando un menor uso de recursos, el uso de un solo lenguaje para su programación y, lo más importante, permite una página más fluida e interactiva (Mikowski y Powell, 2014); y para el back-end se desarrolló una aplicación compuesta

por tres capas: los servicios web, la lógica de negocio y la capa de datos. En la figura 3 se pueden apreciar de manera general los elementos que componen la arquitectura definida.

**Figura 3.** Arquitectura inicial propuesta.



**Fuente:** elaboración propia.

La aplicación orientada al cliente contiene los componentes con los que interactúan los usuarios a través de una interfaz gráfica; de igual manera, consumen la lógica de negocio. La sigla APP-UI designa es la aplicación del lado del cliente, donde los usuarios acceden a la vista de la aplicación.

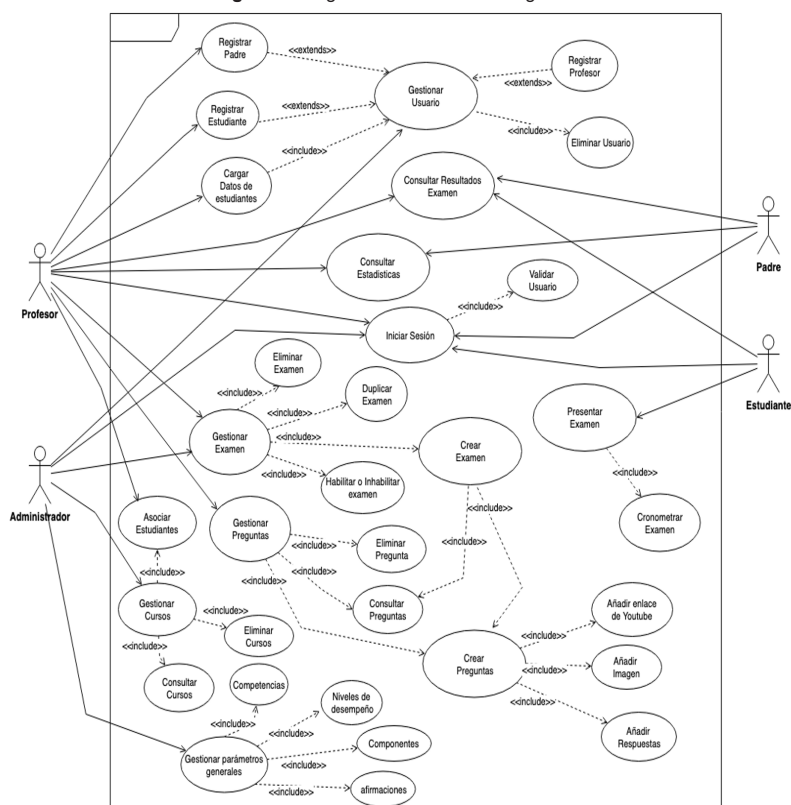
Por otro lado, el servidor contiene la herramienta web. La APP-service es la aplicación del lado del servidor, en el cual están las funcionalidades de negocio que son expuestas mediante servicios web.

- **Lógica:** capa de negocio: define la lógica necesaria para el funcionamiento esencial del sistema.
- **Datos:** capa que contiene los modelos y entidades.
- **Servicios web:** capa donde están los servicios web que exponen las funcionalidades desarrolladas en la capa de negocio.

## Fase de elaboración

Dentro de esta fase se realizó la recolección y análisis de requerimientos, lo que se llevó a cabo con la ayuda de los docentes y directivos de la institución. Las “historias de usuario”, recogidas en la fase de inicio, permitieron conocer de forma ordenada sus necesidades y expectativas. Con base en estas historias de usuario se realizó el listado de requerimientos siguiendo el estándar de la IEEE-830 (IEEE, 1998) para la creación del documento Especificación de Requisitos de Software (ERS), que sirvió como base para el diseño y modelado del software. Posteriormente, se elaboró el diagrama de casos de uso global, que sirve como la representación de las funcionalidades generales que desea el usuario dentro de la herramienta web (Seidl, Scholz, Huemer y Kappel, 2012), tal como se observa en la figura 4.

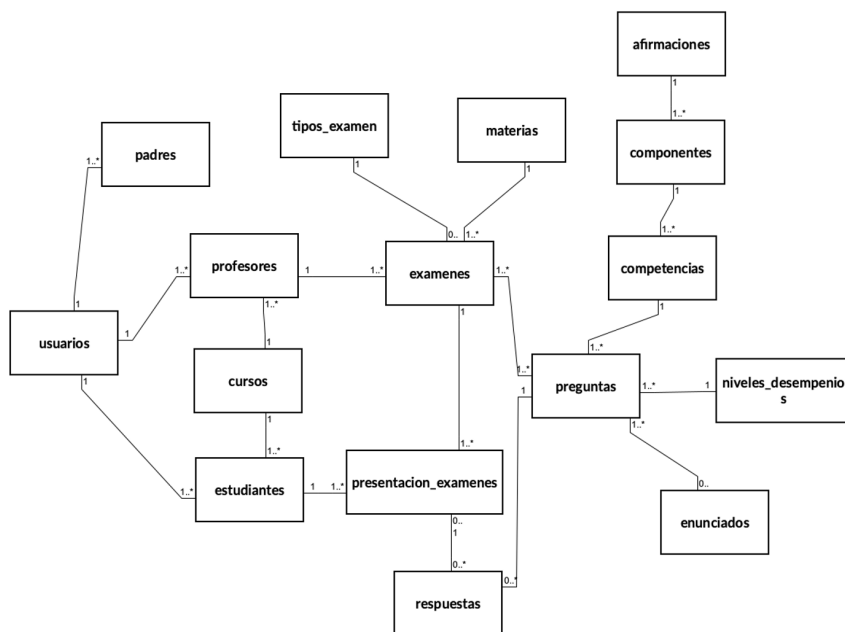
Figura 4. Diagrama de casos de uso general.



Fuente: elaboración propia.

Para el modelamiento de datos se crea, inicialmente, un modelo conceptual de alto nivel, el cual “proporciona los conceptos para presentar los datos de una manera más cercana a la perspectiva de las personas” (Watt y Eng, 2002). En la figura 5 se visualiza de manera general la relación entre las diferentes entidades de datos de la herramienta web, donde la concentración del flujo de datos se encuentra en las entidades: exámenes, preguntas y presentación\_exámenes.

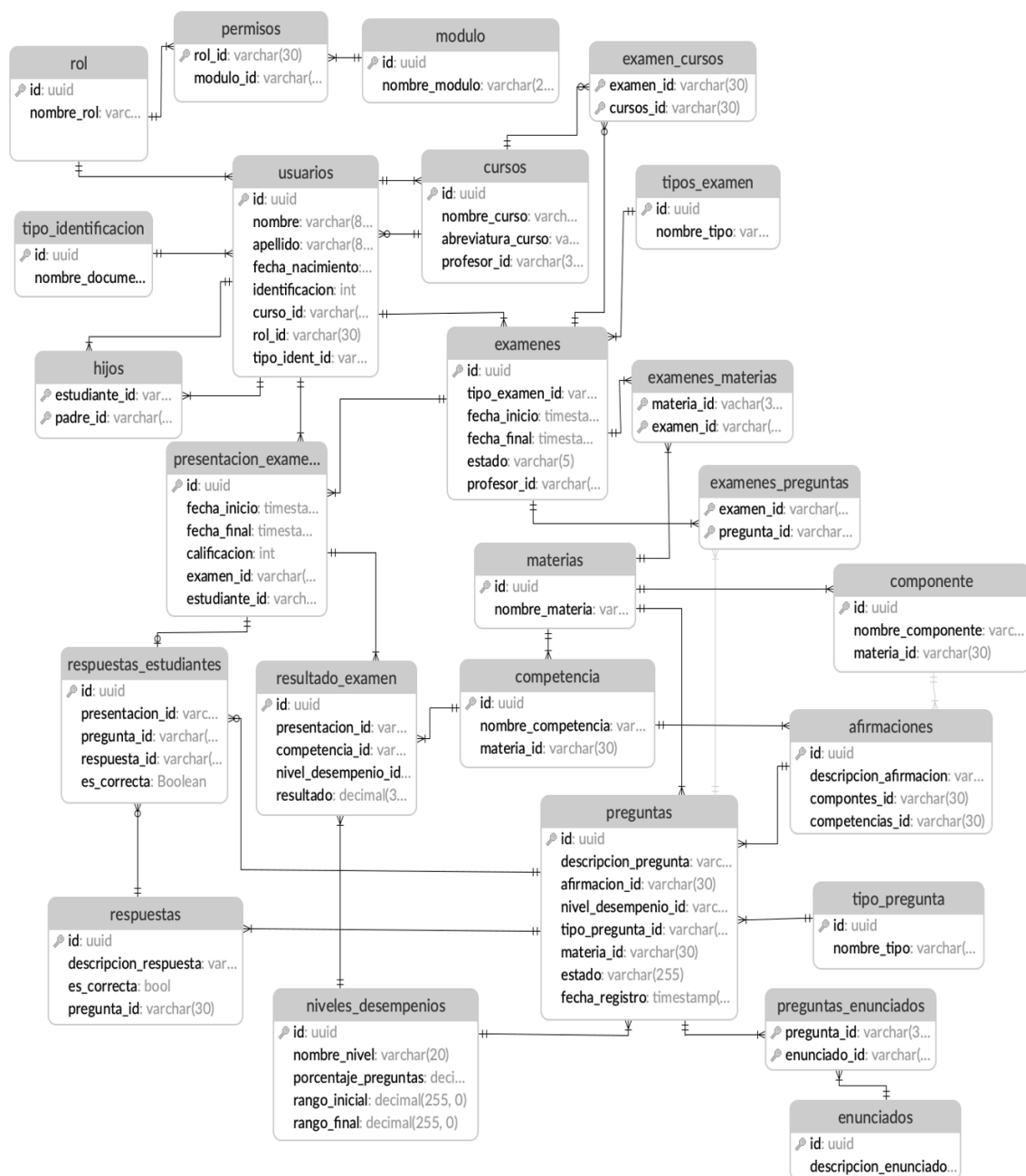
**Figura 5.** Diagrama de casos de uso general.



**Fuente:** elaboración propia.

Tomando como referencia el modelo conceptual de datos (figura 5) y siguiendo la guía de estilo SQL (Holywell, 2019), se diseña el modelo entidad-relación (MER) (Silberschatz, Korth y Sudarshan, 2002), tal como se observa en la figura 6, el cual permite visualizar las entidades de datos, sus atributos asociados y las relaciones entre ellas (Sommerville, 2011).

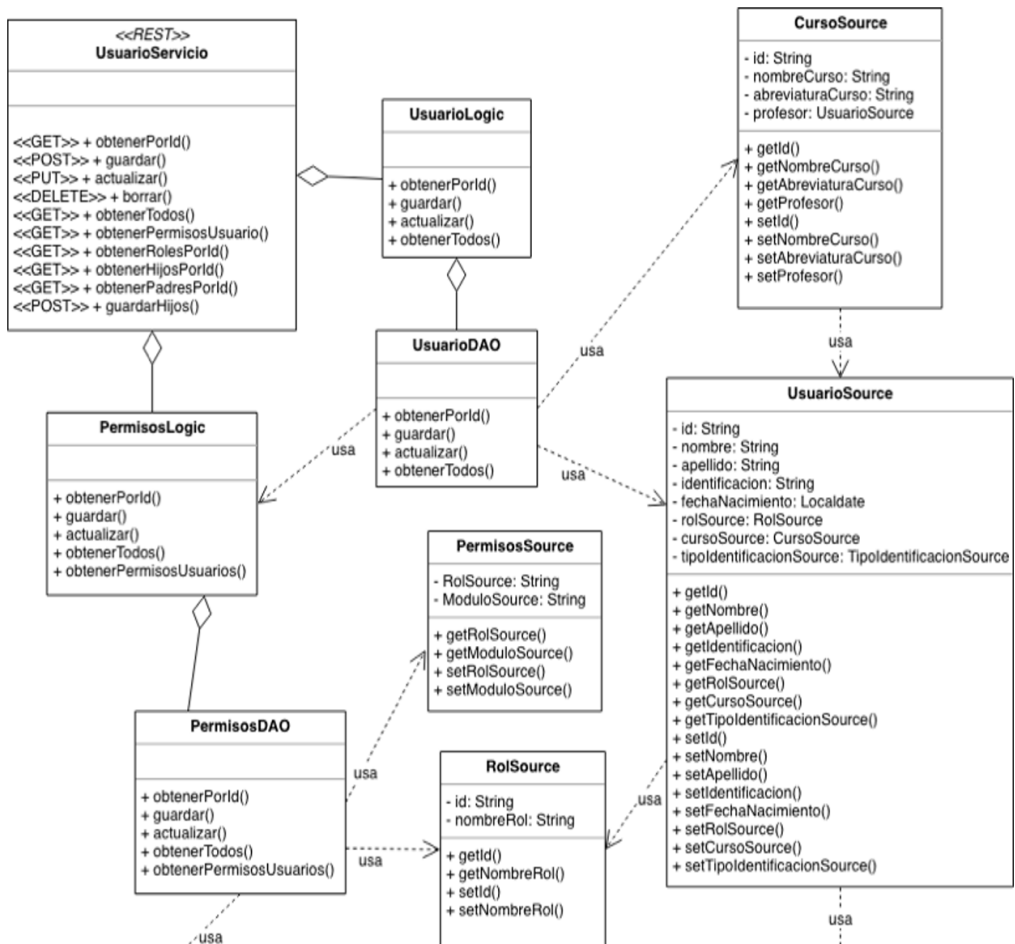
**Figura 6.** Modelo entidad-relación.

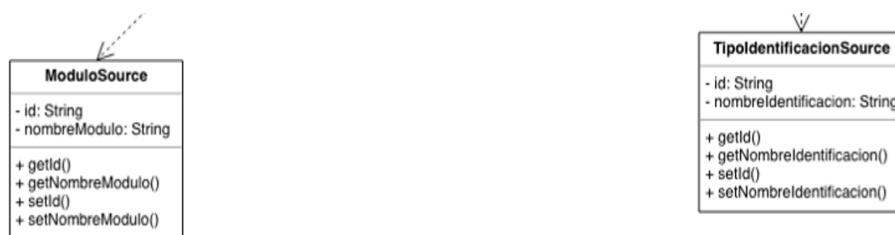


**Fuente:** elaboración propia.

Una vez finalizado el MER, se elaboraron los diagramas de clases, los cuales se utilizaron para especificar las estructuras de datos y las estructuras de objetos de la herramienta web (Seidl *et al.*, 2012). En la figura 7 se observa el diagrama de clases del servicio web de gestión de usuarios, donde se puede apreciar la arquitectura de tres capas planteada en la arquitectura inicial (figura 2) que soporta el *software*: la primera capa contiene los servicios web, la segunda capa contiene las clases lógicas del negocio, y una última capa se describe con un patrón de diseño *data access object* (DAO), con el que se controla la manipulación y acceso a los datos.

Figura 7. Diagrama de clases del servicio de gestión usuarios.

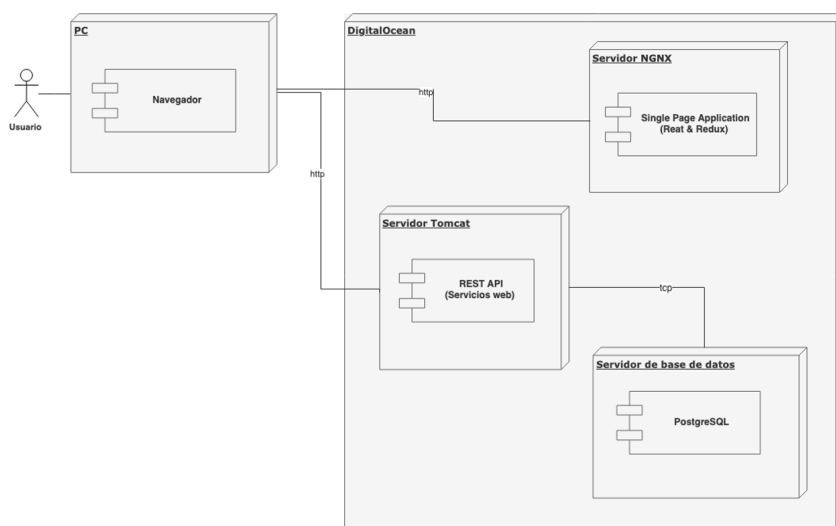




**Fuente:** elaboración propia.

Continuando con la fase, se plantea el diagrama de despliegue (figura 8), donde se visualizan los nodos del *hardware* y la interacción de los componentes del *software* (Seidl et al., 2012), lo que provee una vista más detallada de la infraestructura seleccionada para la herramienta web.

**Figura 8.** Diagrama de despliegue.



**Fuente:** elaboración propia.

En la figura 8 se advierte que fue necesario un servidor donde se contenerizaron las dos aplicaciones y se instaló la base de datos. Se crearon dos contenedores que son una encapsulación de cada aplicación (*front-end* y *back-end*) con sus respectivas dependencias (Mouat, 2016); cada contenedor tiene su propia configuración, donde el destinado para el *back-end* cuenta con Java 10 y Tomcat como servidor de aplicaciones. Para el *front-end* se configuró



Nginx como servidor web y, finalmente, se eligió PostgreSQL como servidor de base de datos. Esta infraestructura está soportada sobre un “droplet” que cuenta con Ubuntu 18, 1 GB de RAM, 25 GB de disco duro y un procesador de 1 núcleo, dentro del servicio de computación en la nube DigitalOcean, que adicionalmente ofrece servicios de *firewall*, inteligencia artificial, *reporting*, etc.

## Fase de construcción

Esta fase se inició mediante la selección de las herramientas tecnológicas (lenguajes de programación, base de datos, librerías) utilizadas para la construcción del prototipo funcional.

Teniendo en cuenta la exigencia de desarrollar dos aplicaciones independientes debido a la arquitectura API-REST establecida en el diseño, fue necesario garantizar la seguridad de la conexión entre ambas aplicaciones, para lo cual se implementa un *token* que se transmite por URL como medio seguro para las peticiones entre las dos partes, como lo define el estándar JSON Web Token (JWT) (Jones, Bradley y Sakimura, 2005), certificando la transmisión de la información.

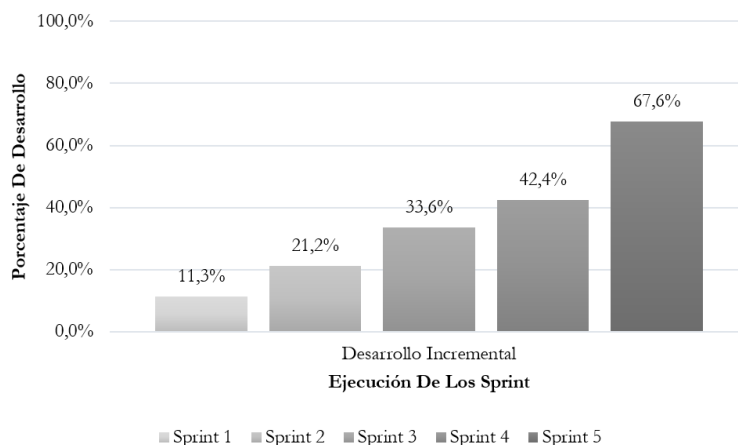
Para el desarrollo del front-end se construyó una single page application (SPA), que se define como una aplicación web completa de una sola página, la cual contiene todas las demás páginas web de la aplicación (Fink y Flatow, 2014). Para su construcción se usa la biblioteca de JavaScript React.js; Redux para el manejo de los estados de la SPA, y Redux-Saga para el consumo y gestión de las respuestas de los web services (WS). La construcción del back-end se llevó a cabo en el lenguaje de programación Java con Spring Framework, usando los módulos Spring Data, Spring Security y Spring Web; por otra parte, para agilizar la construcción de clases se usó la biblioteca Lombok y, finalmente, para la gestión de la base de datos se empleó PostgreSQL 10.

Para la construcción de este prototipo funcional se definió el siguiente alcance:

- Construcción del inicio de sesión de la aplicación para los usuarios profesor y estudiante.
- Construcción del registro de usuarios con rol estudiante y profesor.
- Construcción de la creación de exámenes.
- Construcción de la presentación de exámenes.
- Construcción de consulta de estudiantes.

Tal como se tenía previsto, el desarrollo de la herramienta web se llevó a cabo a través de “sprints”, como se muestra en la figura 9, donde se ve la representación del avance de las tareas del bac-klog una vez finalizada cada iteración, cumpliendo con el alcance definido para el prototipo funcional.

**Figura 9.** Desarrollo incremental de la herramienta web.



**Fuente:** elaboración propia.

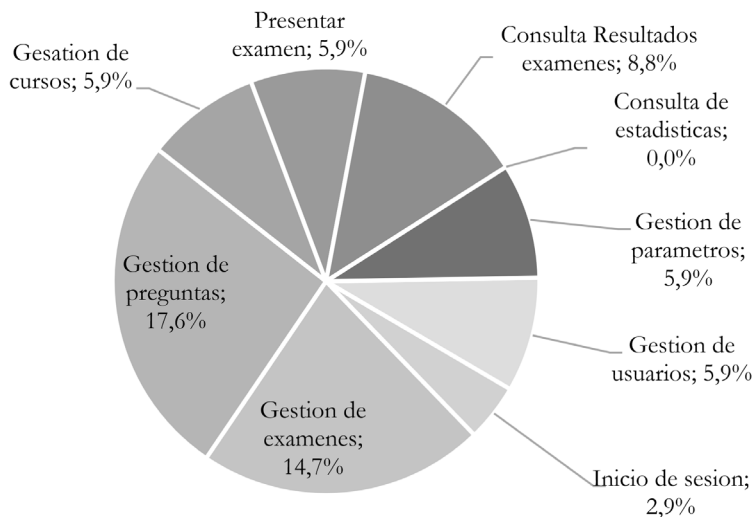
## Resultados

### Resultados del desarrollo de la metodología

#### Desarrollo vs. requerimientos funcionales

A la fecha de presentación de este artículo, el desarrollo del prototipo muestra un avance del 67,6 % con respecto a los requerimientos funcionales totales, distribuido en los módulos como se observa en la figura 10. Los datos corresponden al porcentaje de contribución de cada módulo al porcentaje completado del prototipo. El mayor aporte se nota en Gestión de preguntas y Gestión de exámenes, con un 17,6 % y 14,7 % respectivamente, alcanzando el cumplimiento de requerimientos como creación, actualización y eliminación tanto de preguntas como de exámenes, temporización de los exámenes y habilitación o inhabilitación de estos.

**Figura 10.** Desarrollo por módulos vs. requerimientos funcionales.

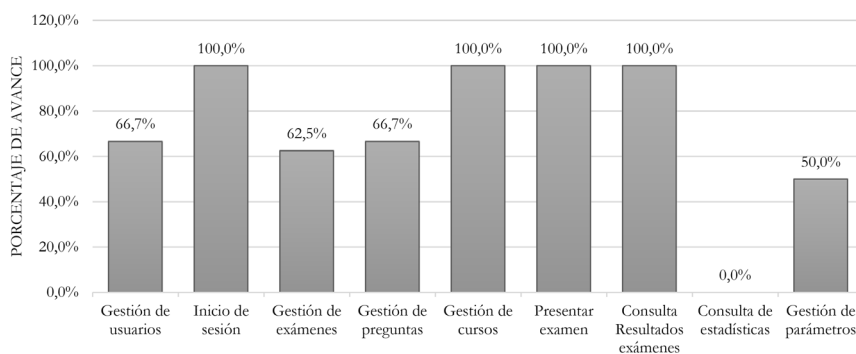


**Fuente:** elaboración propia.

En la figura 11 se puede observar el porcentaje de desarrollo de cada módulo de acuerdo con los requerimientos alcanzados.

Se pueden observar varios módulos finalizados al 100 %, como el Inicio de sesión y Presentar examen.

**Figura 11.** Cumplimiento de requerimientos por módulo.

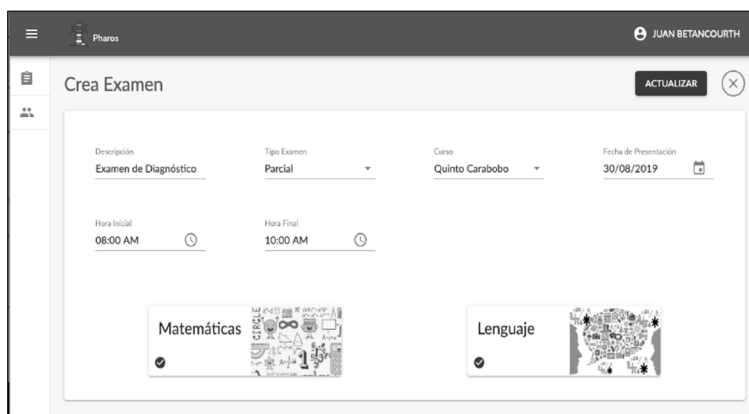


**Fuente:** elaboración propia.

A continuación, se presentan vistas resultantes del desarrollo del prototipo funcional inicial.

La figura 12 muestra la pantalla donde los docentes pueden iniciar la creación de los exámenes y posteriormente adicionar las preguntas a las materias de lenguaje y matemáticas.

**Figura 12.** Toma de pantalla de creación de examen.



**Fuente:** elaboración propia.

En la figura 13 se observa donde los docentes configuran la pregunta de acuerdo con la estructura de competencias que establece el Icfes.

**Figura 13.** Toma de pantalla de creación de preguntas.

**Fuente:** elaboración propia.

También se puede advertir, en la figura 14, la vista de pantalla de donde los alumnos presentan las evaluaciones diseñadas por los docentes.

**Figura 14.** Ejemplo de inicio de presentación de examen.

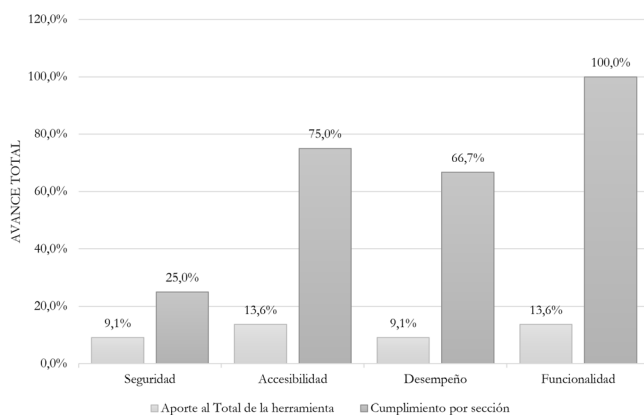
**Fuente:** elaboración propia.

Las imágenes presentadas permiten dar un vistazo a las principales funcionalidades y avalan el cumplimiento de los requerimientos funcionales establecidos en el alcance del prototipo inicial.

## Desarrollo vs. requerimientos no funcionales

A la fecha de presentación de este artículo, el desarrollo del prototipo muestra un avance del 71,4 % con respecto a los requerimientos no funcionales, distribuido en los módulos como se observa en la figura 15. Los datos corresponden al porcentaje de contribución de cada módulo al porcentaje completado del prototipo. El mayor aporte se nota en el desarrollo de la Funcionalidad y la Accesibilidad, con un 13,6 % cada uno, alcanzando el cumplimiento de requerimientos como sistema de control de versiones, implementación en navegadores, manejo de formularios y desarrollo incremental. En la misma figura se observa el cumplimiento de requerimientos por sección, donde de momento únicamente se alcanza el 100 % de los requerimientos de Funcionalidad.

**Figura 15.** Desarrollo por módulos vs. requerimientos no funcionales.

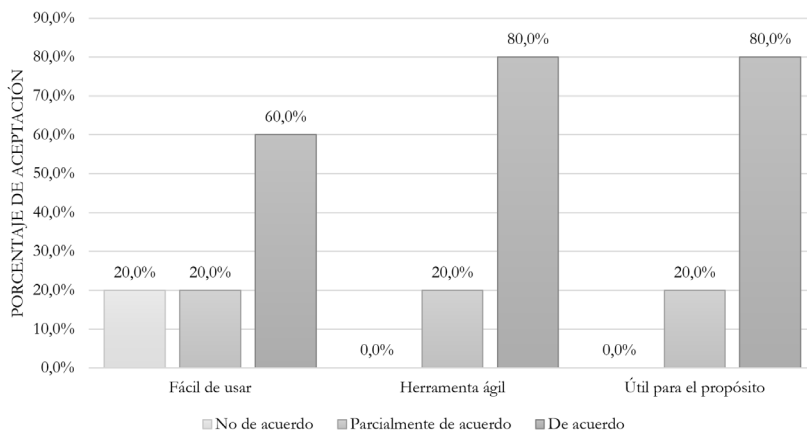


**Fuente:** elaboración propia.

A continuación, en la figura 16, se observan los resultados de una encuesta realizada a los profesores después de haber interactuado con la herramienta, donde probaron el ingreso a la plataforma, la navegación a través de ella y la elaboración de preguntas

y exámenes. Con este elemento se pudo medir la percepción del usuario en tres componentes: facilidad de uso, la cual tiene una aceptación del 60 %; agilidad, con un 80 % de aceptación, y utilidad de la herramienta también con un 80 % de aceptación.

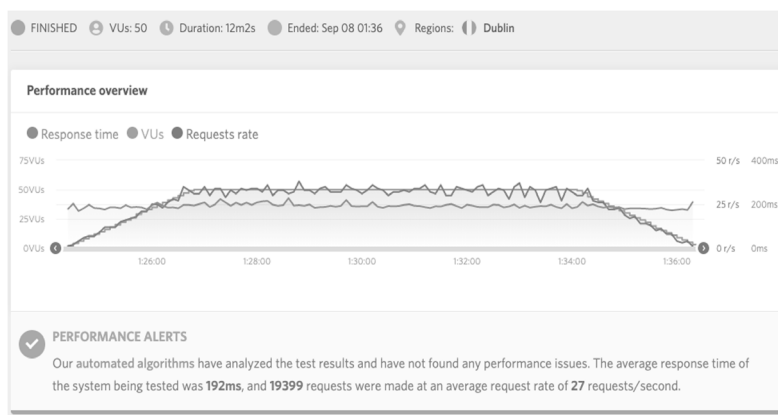
**Figura 16.** Percepción de los usuarios.



**Fuente:** elaboración propia.

Una vez concluido el desarrollo del prototipo funcional se procedió con el montaje en la infraestructura propuesta y se realizó una prueba de estrés para evaluar el rendimiento del servidor de DigitalOcean. La prueba consistió en simular el ingreso a la aplicación de 50 usuarios virtuales usando la aplicación web LoadImpact (Load Impact, 2019). Los resultados se muestran en la figura 17, en la que se observa que el tiempo de respuesta de aplicación se mantiene adecuadamente en 192 milisegundos en promedio durante los 8 minutos que dura la prueba, usando los 50 usuarios al tiempo y un total de 19 399 peticiones durante ese periodo.

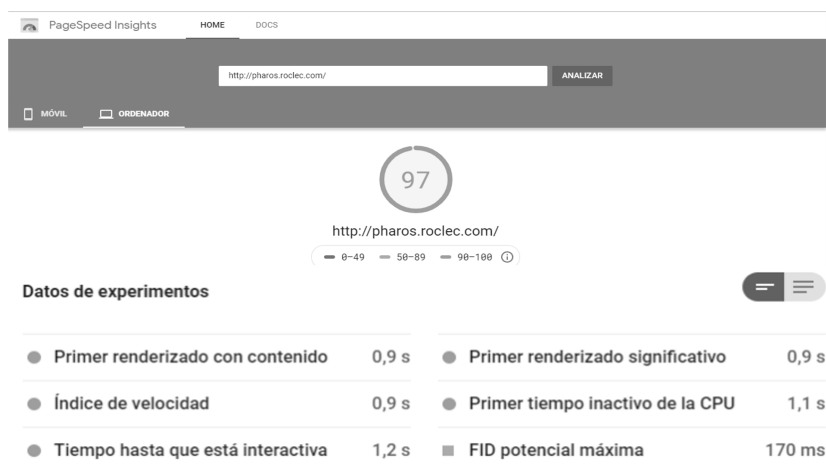
**Figura 17.** Prueba de estrés de la infraestructura propuesta.



**Fuente:** Load Impact (2019).

Finalmente, con el objetivo de medir el desempeño de la aplicación se ejecutó una prueba en la página PageSpeed Insights de Google (Google, s. f.), que analiza el contenido y genera sugerencias para mejorar la velocidad de las páginas web. Se obtuvo 97 de calificación, como lo muestra la figura 18, contando con un primer renderizado en 0,9 s y demorando tan solo un 1,2 s en la carga total de la página.

**Figura 18.** Prueba de velocidad de respuesta



**Fuente:** Google (s. f.).



## Conclusiones

La Institución Educativa Julio Caicedo y Téllez en la actualidad no tiene herramientas que le permita determinar el nivel de los estudiantes en cada competencia de las que se evalúan en las pruebas Saber.

A partir de las encuestas realizadas, se concluye que el personal docente ve muy útil la implementación de una herramienta web que atienda la necesidad de generar información en los procesos concernientes al sistema de evaluación y contar con esta información en un tiempo que permita el análisis y ejecución de planes de mejoramiento. Igualmente, se acreditó la viabilidad de implementar una herramienta propia, ya que teniendo en consideración las características de diferentes herramientas disponibles en el mercado se encontró que, si bien se obtienen opciones muy variadas, no se observa la posibilidad de elaborar exámenes que cumplan las especificaciones de las preguntas como las define el Icfes.

En cuanto al desarrollo de la metodología, se logró el diseño de una solución alineada con el modelo de evaluación por competencias, y que aborda las necesidades planteadas por la institución educativa. Se obtuvo la definición de una arquitectura moderna con un enfoque que facilita desarrollos y mejoras futuras. El prototipo funcional desarrollado se encuentra dentro de los estándares de desempeño del mercado de acuerdo con las validaciones primarias ejecutadas, y se asegura su correcta funcionalidad en los principales navegadores del mercado (Chrome, Edge y Firefox). De acuerdo con las pruebas realizadas al software, la infraestructura planteada se adapta a los requerimientos y soporta el funcionamiento principal del sistema.

La respuesta de los usuarios al uso del prototipo mostró una percepción favorable después de interactuar con los diferentes módulos construidos. Se prevé una fácil adaptación a la herramienta para elaborar evaluaciones tipo Saber, gracias a que los docentes y directivos tienen un conocimiento claro de cómo son diseñadas

las pruebas, la manera como se desarrollan, el tipo de preguntas que se encuentran y las competencias que se evalúan en ellas. Con los estudiantes se requiere establecer una etapa de adaptación y aprendizaje tanto sobre la forma en que se les evalúa como en el uso de la herramienta web.

Los trabajos futuros en la aplicación a corto plazo estarán representados en la construcción de una plataforma *web responsive*, el rediseño y personalización de la interfaz de usuario y el lanzamiento de la aplicación en la institución. A mediano plazo se proyecta la construcción de la aplicación móvil dirigida a sistemas iOS y Android, la inclusión en la herramienta de los demás ciclos escolares y la formación de alianzas estratégicas con otras instituciones educativas, la Secretaría de Educación Municipal de Cali y el Ministerio de Educación de Colombia. Será necesario un replanteamiento de la infraestructura para que pueda soportar el incremento de usuarios una vez definida la expansión de la herramienta dentro de la institución y concretadas las diferentes alianzas.

## Referencias bibliográficas

Allamaraju, S. (2010). *RESTful web services cookbook*. Sebastopol (EE. UU.): O'Reilly Media.

Ambler, S. (2002). *Agile modeling*. Nueva York: John Wiley & Sons.

Ambler, S. (2005). *The agile unified process (AUP)*. Recuperado de <http://www.ambysoft.com/unifiedprocess/agileUP.html#Serial>

Beck, K. et al. (2001). *Manifiesto por el desarrollo ágil de software*. Recuperado de <http://agilemanifesto.org/iso/es/manifesto.html>

Fidenia. (2019). *Questbase*. Recuperado de <https://www.questbase.com>

Fink, G. y Flatow, I. (2014). *Pro single page application: Using Backbone.js and ASP.NET*. Nueva York: Apress.

Google. (s. f.). *PageSpeed Insights*. Recuperado de <https://developers.google.com/speed/pagespeed/insights/?hl=es>

Hernández, R., Fernández, C. y Baptista, P. (2014). *Metodología de la investigación* (6.ª ed.). Ciudad de México: McGraw Hill.

Holywell, S. (19 de junio de 2019). *SQL Style Guide*. Recuperado de <https://www.sqlstyle.guide/>

Icfes. (2017). *Saber 5.º: guía de orientación, 2017*. Recuperado de <https://www.icfes.gov.co/documents/20143/1353827/Guia+de+orientacion+saber+5+2017.pdf>

Icfes. (2018). *Saber 3.º, 5.º, 9.º* (aplicaciones antes de 2018). Recuperado de <https://www.icfes.gov.co/web/guest/resultados-historicos-saber-35>

Icfes. (s. f.). *Demoplexi*. Recuperado de <https://demoplexi.icfes.gov.co/login>

IEEE. (1998). IEEE 830-1998: *IEEE Recommended Practice for Software Requirements Specifications*. Recuperado de <https://standards.ieee.org/findstds/standard/830-1998.html>

Jones, M., Bradley, J. y Sakimura, N. (2005). *JSON Web Token (JWT)*. Recuperado de <https://tools.ietf.org/html/rfc7519>

Laínez, J. R. (2015). *Desarrollo de software ágil: extreme programming y scrum*. Vigo: IT Campus Academy.

Load Impact. (Agosto de 2019). *Performance testing for DevOps teams*. Recuperado de <https://loadimpact.com/>

Mikowski, M. y Powell, J. (2014). *Single page web applications*. Shelter Island: Manning Publications.

MEN (Ministerio de Educación Nacional). (2006). *Estándares básicos de competencias en lenguaje, matemáticas, ciencias y*

*ciudadanías*. Recuperado de [https://www.mineduacion.gov.co/1621/articles-340021\\_recurso\\_1.pdf](https://www.mineduacion.gov.co/1621/articles-340021_recurso_1.pdf)

MEN. (Ministerio de Educación Nacional). (2015). *Colombia, la mejor educada en el 2025*. Recuperado de [https://www.mineduacion.gov.co/1759/articles-356137\\_foto\\_portada.pdf](https://www.mineduacion.gov.co/1759/articles-356137_foto_portada.pdf)

Mouat, A. (2016). *Using Docker. Sebastopol* (EE. UU.): O'Reilly Media.

Pressman, R. S. (2010). *Ingeniería del software: un enfoque práctico* (7.ª ed.). Ciudad de México: MacGraw Hill.

ProProfs. (2019). *Proprofs*. Recuperado de <https://www.proprofs.com>

Quizworks. (2019). *Easy LMS*. Recuperado de <https://www.easy-lms.com/es/>

Richardson, L. y Ruby, S. (2007). *RESTful web services. Sebastopol* (EE. UU.): O'Reilly Media.

SCRUMstudy. (2017). *Cuerpo de conocimiento de Scrum (Guía SBOK)*. Avondale: SCRUMstudy.

Seidl, M., Scholz, M., Huemer, C. y Kappel, G. (2012). *UML@ classroom: An introduction to object-oriented modeling*. Heidelberg: Springer.

Silberschatz, A., Korth, H. y Sudarshan, S. (2002). *Fundamentos de bases de datos* (4.ª ed.) Madrid: McGraw Hill.

Sommerville, I. (2011). *Ingeniería de software* (9.ª ed.). Madrid: Pearson Educación.

Vargas, Z. R. (2009). La investigación aplicada: una forma de conocer las realidades con evidencias. *Revista Educación*, 33, 155-165.

Watt, A. y Eng, N. (2002). *Database design* (2nd ed.). Victoria (Canadá): BCCampus Open Education.

