

Diseño de una aplicación para preparar estudiantes en la Prueba Saber Pro - Módulo de razonamiento cuantitativo

Design of an application to prepare students in the Saber Pro Test: Quantitative reasoning module

Píta Pedraza, Diego Alexander¹, Ochoa, Juan Sebastián²; y Galvis Barrera, Luis Fernando³
Fundación Universitaria de San Gil - UNISANGIL, Facultad de Ciencias Naturales e Ingeniería
Departamento de Ciencias Básicas
Yopal, Casanare, Colombia

dpita@unisangil.edu.co
juansebastian@unisangil.edu.co
lgalvis@unisangil.edu.co

Fecha de Recepción: marzo 02 de 2020
Fecha de Aceptación: julio 08 de 2020

Resumen - La Unidad de Ciencias Básicas de la Facultad de Ciencias Naturales e Ingeniería de *UNISANGIL*, en la Sede Yopal (Casanare), prepara a los estudiantes para la realización de la *Prueba Saber Pro* mediante un simulacro de dicha prueba, enfocado en el módulo de razonamiento cuantitativo. Sin embargo, el trámite para obtener, evaluar y analizar los resultados del simulacro es largo y tedioso, por lo que el objetivo del presente trabajo es reducir el tiempo y facilitar la presentación de los resultados. Como refuerzo, se ha diseñado un software que muestra una encuesta a desarrollar por el estudiante. La información recogida en dicha encuesta se almacena en una base de datos y se analiza por técnicas multivalentes para presentar la información a los estudiantes y a los docentes; donde se permitirá definir acciones en el desarrollo académico de los temas evaluados.

Palabras clave: - aplicación, razonamiento cuantitativo, *Saber Pro*, simulación.

Abstract: - The Basic Sciences Department of the Faculty of Natural Sciences and Engineering of *UNISANGIL*, at Yopal campus (Casanare), prepares its students for the completion of the *Saber Pro* Test through a simulation of the mentioned test, focused on the quantitative reasoning module. However, the process to obtain, evaluate and analyze the results of the simulacrum is long and tedious, so the objective of this work is to reduce the time and facilitate the presentation of the results. As reinforcement, it has designed software that displays a survey carried out by the student. The information collected in this survey is stored in a database and analyzed by multivalent techniques for presenting information to students and teachers; where it is possible to define actions in the academic development of the subjects evaluated.

Keywords - application, quantitative reasoning, *Saber Pro*, simulation.

¹ Ingeniero Electromecánico, Docente de la Facultad de Ciencias Naturales e Ingeniería de UNISANGIL

² Estudiante de Ingeniería de Sistemas, UNISANGIL

³ Ingeniero de Minas, Docente de la Facultad de Ciencias Naturales e Ingeniería de UNISANGIL

I. INTRODUCCIÓN

Dentro del soporte brindado a los estudiantes de últimos semestres, la Fundación Universitaria de San GIL – UNISANGIL [1] planteó realizar un simulacro de la Prueba *Saber Pro* de tipo escrito, del cual podrían extraerse datos que, analizados, permitieran definir acciones a llevar a cabo tanto por parte de los estudiantes como de los docentes, en lo referente al área de razonamiento cuantitativo.

En el presente trabajo se ha desarrollado un software para que los estudiantes de pregrado de UNISANGIL realicen un simulacro de la Prueba *Saber Pro* en el Módulo de Razonamiento Cuantitativo, de modo virtual, que les permita conocer las condiciones del examen y definir acciones de mejora. Además, los resultados se guardan en una base de datos y a partir de ellos se pueden realizar diferentes estudios para mejorar la docencia relacionada con el razonamiento cuantitativo en UNISANGIL.

El Examen de Estado de Calidad de la Educación Superior, Saber Pro, establecido por el Instituto Colombiano para el Fomento de la Educación Superior ICFES está compuesto por módulos de competencias genéricas, las cuales son entendidas como aquellas que deben desarrollar todos los estudiantes en cualquier área de conocimiento, y de competencias las específicas, que son aplicadas según los grupos de programas con características de formación similares. [2]

Los módulos que componen el examen, basado en los “Módulos de Competencias Genéricas, ICFES – 2018”, son los siguientes [3]:

- Lectura Crítica
- Razonamiento Cuantitativo
- Competencias Ciudadanas
- Comunicación Escrita
- Inglés

Además de los anteriores módulos de competencias genéricas, hay 40 módulos asociados a temáticas y contenidos específicos en los que los estudiantes tienen la posibilidad de presentar el examen de acuerdo a su área de formación profesional.

El presente trabajo se centra en las pruebas del módulo de Razonamiento Cuantitativo.

II. DESARROLLO DEL SOFTWARE

A. Almacenamiento de la información

La base de datos almacena la información necesaria para trabajar en el software de aplicación del simulacro de la Prueba *Saber Pro*. Para ello se diseñó un modelo de relación entre las diferentes entidades con sus atributos, como:

estudiante, pregunta y respuesta, en MySQL Workbench [4]. La base de datos se desarrolló sobre el sistema gestor de datos MariaDB [5], uno de los más conocidos. Dichas herramientas facilitaron el proceso de almacenamiento de la información.

La base de datos se diseñó de acuerdo al modelo relacional, tal como se puede observar en la Figura 1. Mediante la consola, se ejecuta el código (backup base de datos), el cual permite que se cree cada entidad con sus respectivos atributos, tipo de datos y las relaciones que hay entre las entidades.

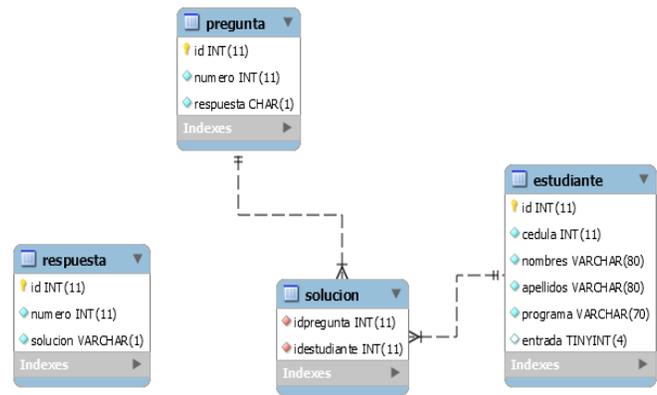


Fig. 1. Modelo relacional del software

Se realiza el registro de los estudiantes a realizar el simulacro, junto con las preguntas, la solución y respuesta, para usar la información mediante lenguaje de programación PHP.

El resultado para los estudiantes y docentes se realiza por consulta dentro de la base de datos, para obtener información de los atributos de las diferentes entidades.

B. Programación de la prueba

El diseño del software se realizó en el lenguaje de programación PHP [6], dentro del editor de texto Sublime Text. Se planteó un modelo de desarrollo estático y fijo, es decir que una parte de contenido no cambia y la otra permite manipular la información. El tipo de programación que se utilizó para el desarrollo del software es embebida, lo cual permite capturar los datos utilizando el lenguaje de programación PHP.

La aplicación comienza con la realización de una conexión a la base de datos desde PHP. Se emplea programación orientada a objetos POO para definir la clase y las funciones para conectar la base de datos, llamando a: el servidor localhost, el usuario (estudiante y docente), la contraseña y el nombre de la base de datos.

Se crea el módulo de sesión (archivo index.php), que permite que el servidor lo encuentre y lo muestre al usuario. En el módulo de sesión, el usuario ingresa con el número de documento de identidad siempre que esté registrado en la base de datos y no haya ingresado anteriormente al sistema para realizar la prueba de simulacro.

El módulo de prueba permite al usuario visualizar cada una de las preguntas y dar la selección de la respuesta (pregunta con opción de 4 respuestas) del cuestionario, la cual se guarda en la base de datos, identificando al estudiante que desarrolló el simulacro.

Al finalizar de desarrollar el cuestionario, el módulo de resultados le presenta al estudiante las siguientes tablas:

- Tabla de resultados detallando el número de pregunta, su respuesta y el resultado, respectivamente de la respuesta de cada pregunta.
- Tabla de totales: que define la cantidad de las preguntas correctas e incorrectas del simulacro realizado por el estudiante, además presenta los aciertos y errores porcentuales del total de preguntas del cuestionario.
- Tabla de verificación: en las preguntas incorrectas, donde se puede acceder a través del módulo de justificación y detallar cada pregunta errónea para que el estudiante evalúe el resultado y comprenda el modo de desarrollo de la pregunta.

El módulo de justificación muestra la respuesta correcta además de su defensa.

El módulo de salir de la aplicación, tanto para el estudiante como para el profesor, cierra la sesión inicializada, quedando en la base de datos la información hasta donde el estudiante desarrolló el simulacro. El estudiante solo tiene autorizado el ingreso una vez a la aplicación.

El módulo consulta docente es habilitado para los docentes y permite revelar los resultados de las preguntas de cada estudiante, el acceso a la información es ilimitado, aunque depende si el administrador retira la información de la base de datos.

El diagrama de estados permite definir el estado donde se encuentra el usuario a medida que va avanzando en la aplicación, se ve en la Figura 2.

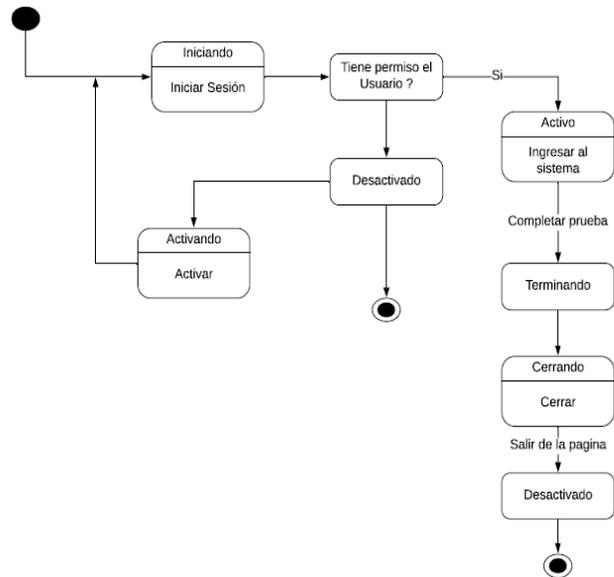


Fig. 2 Diagrama de estados en la aplicación.

C. Diagrama de actividades o flujo

En la Figura 3 se muestra el diagrama de actividades o flujo del software, el cual muestra el funcionamiento cuando el usuario usa la aplicación. El primer paso es la validación de datos, desde que el usuario esté registrado en la base de datos por el administrador con su número de documento y el programa en curso del estudiante. Luego debe definirse si el usuario es estudiante o docente. Para docente, verifica los resultados de los estudiantes que realizaron la prueba, donde se muestra el nombre completo del estudiante, el programa, la tabla de resultados y la tabla de totales del estudiante en consulta. Para el estudiante, presenta el simulacro de la prueba hasta que la termina o desea salir; al finalizar, se visualiza sus resultados en las tablas mencionadas en la programación de la prueba y tiene la posibilidad de verificar los resultados de las preguntas erróneas, solo se le permite realizar el simulacro una vez.

Los datos quedan almacenados en la base de datos, a la cual solo tiene acceso el administrador. Además, dichos datos se pueden extraer para realizar un análisis de los mismos de acuerdo al interesado en la información.

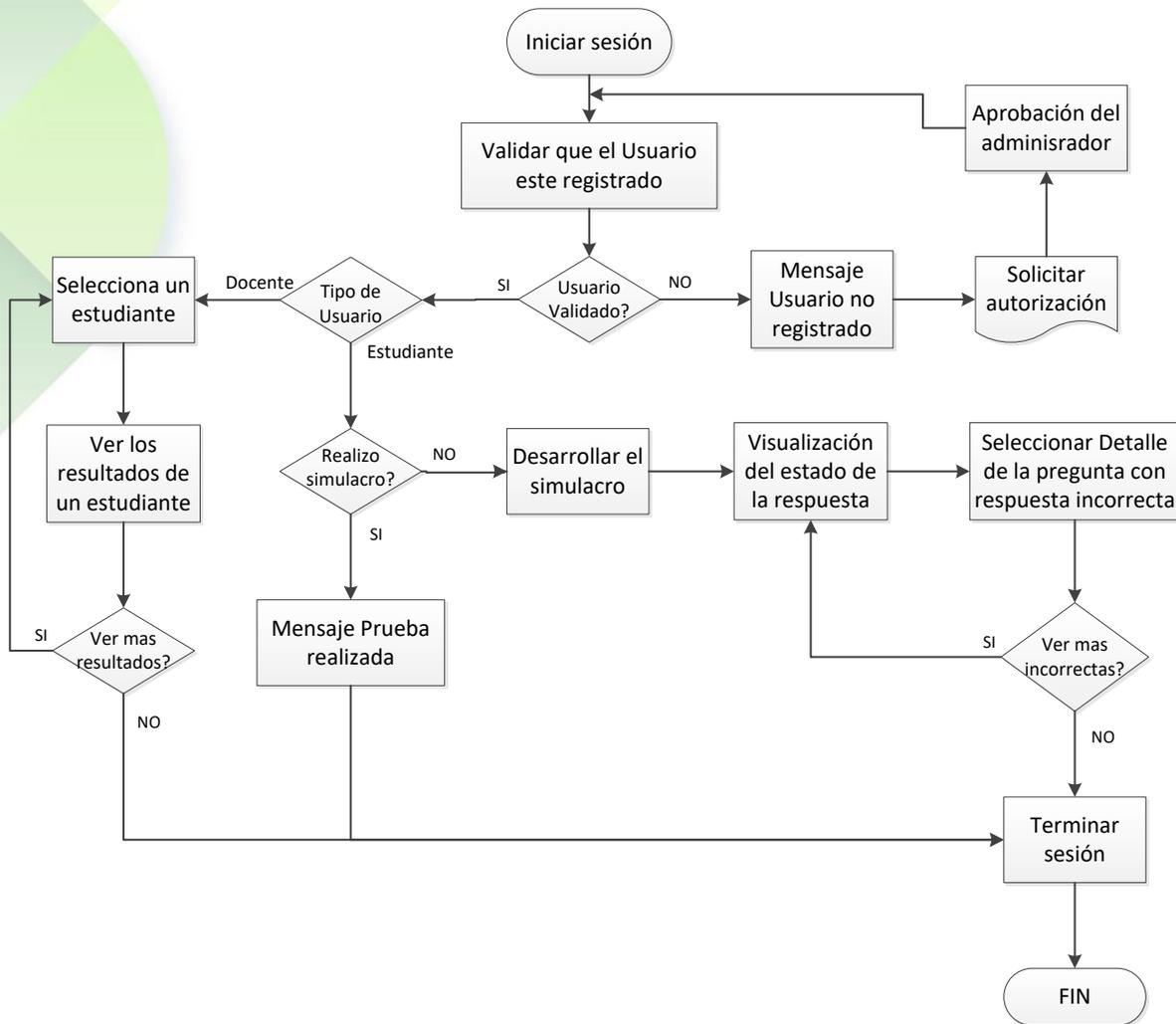


Fig.3 Diagrama de actividades o flujo

D. Instalación, verificación y validación

Instalación

Primero se registró en un servidor de *POST* [7] gratuito una página web para que el usuario pudiera acceder desde un dispositivo que tuviera conexión a internet. Luego se definió el nombre de la página y la contraseña y se subieron los archivos a la carpeta del servidor, especificando el archivo (*index.php*) que abre automáticamente. Al final, se editó la conexión de la base de datos en la aplicación para que la información dada por el usuario se guardase a través de la aplicación.

Verificación

Se realizó una prueba piloto para determinar que la información dada si se registraba en la base de datos, y si se mostraba al estudiante y al docente de acuerdo al diagrama de actividades planteado.

Validación

La aplicación del simulacro desarrollado se verificó con los docentes del Departamento de Ciencias Básicas y a un grupo de 20 estudiantes de últimos semestres del programa de enfermería de *UNISANGIL*, sede Yopal. El resultado fue que funcionó respondiendo a las necesidades y que se desarrolló según lo programado en el diagrama de caso de estudio formulado. Los estudiantes desarrollaron el simulacro, observaron sus resultados y revisaron sus errores para corregir. Los docentes ingresaron a ver los resultados de la prueba realizada por cada estudiante de acuerdo a lo mencionado en el diagrama de actividades. No obstante, se plantearon mejoras en la forma de presentación de la aplicación para que la información brindada fuese más fácil de entender por parte del usuario.

III. CONCLUSIONES

El desarrollo de aplicaciones virtuales para el apoyo de las pruebas saber PRO en los estudiantes es una excelente aplicación tanto para el estudiante como herramienta de aprendizaje, como para el docente como herramienta de conocer el avance del estudiante.

Respecto al diseño del software del simulacro, se deben tener en cuenta el modelo relacional del software y el diagrama de flujo para iniciar a definir los parámetros de programación.

En este caso el lenguaje usado PHP, facilitó la gran mayoría de procesos, además se identificó que es un lenguaje que tiene varias alternativas y facilitan la escritura del código.

En la selección del servidor y buscando un beneficio para la aplicación del software se definió un servidor libre, aunque esta condición puede ser cambiada debido a los parámetros de seguridad de la aplicación.

Después de realizada la validación, se evidenció que la prueba de simulación virtual desarrollada funcionó de acuerdo al objetivo esperado y se puede seguir aplicando en todos los programas de *UNISANGIL*, con los estudiantes a realizar la Prueba saber PRO.

REFERENCIAS

- [1] Fundación Universitaria de San GIL – UNISANGIL [En línea]. Disponible en: <http://www.unisangil.edu.co/>
- [2] Instituto Colombiano para la Evaluación de la Educación - ICFES, “Saber Pro. Módulos de Competencias Genéricas - 2017,” pp. 11–41, 2017.
- [3] Instituto Colombiano para la Evaluación de la Educación - ICFES, “Módulos de Competencias Genéricas,” pp. 9–19, 2018.
- [4] MCLAUGHLIN, Michael. MySQL Workbench: Data Modeling & Development. McGraw Hill Professional, 2013.
- [5] MariaDB Foundation (2020). MariaDB Documentation. [En línea]. Disponible en: <https://mariadb.org/documentation/>
- [6] M. Á. T. Remon, Desarrollo de aplicaciones web con PHP. Lima, 2014.
- [7] A. Ramos, Aplicaciones Web (Novedad 2011). Editorial Paraninfo, 2011.