

Instituto Tecnológico de Costa Rica

Escuela de Computación

IC-4302 Bases de Datos II

Proyecto I - Etapa 1

Profesor: Ing. José Dolores Navas Sú

Esteban Chavarría Fallas - 2016021924  
estebanchavarriacr@gmail.com

Randall Delgado Miranda - 2016238520  
rdelgadam23@gmail.com

José Luis Rodríguez Hernández - 2016093725  
joserh1206@gmail.com

Grupo de trabajo #6

I Semestre 2018

# Introducción

Las aplicaciones Web han ido creciendo poco a poco conforme la tecnología avanza, debido a las facilidades que ésta permite respecto a aplicaciones más estáticas o hechas para trabajar sin conexión a Internet. El simple hecho de poder utilizarlas desde cualquier parte del mundo y en cualquier sistema operativo hace que la mayoría de ellas se estén desarrollando en este ambiente, e incluso se han visto ya varios casos en los que aplicaciones más antiguas migran a un ambiente Web para facilitar el uso a sus clientes.

Para el aprovechamiento de los desarrolladores, se han formado “stacks” o infraestructuras que ayudan enormemente a organizar un desarrollo óptimo, ya que utilizan aplicaciones diferentes para capas específicas del software, además de caracterizarse por ser open-source en todas sus partes. Un ejemplo muy utilizado de un stack es LAMP, que combina Linux, Apache, MySQL y (PHP, Perl o Python) como sistema operativo, servidor web, gestor de BD y lenguaje de programación respectivamente.

En este trabajo, se hará uso del stack MEAN (compuesto por MongoDB, Express, Angular y NodeJS), para desarrollar una aplicación Web, dedicada a brindar un sistema de registros académicos, para usuarios estudiantes y profesores. En esta primera etapa, el sistema proveerá las funciones necesarias para el mantenimiento de instituciones académicas, escuelas, programas académicos, mallas curriculares, materias, cursos y grupos, y por supuesto el registro de estudiantes y profesores. El sistema manejará la transferencia de datos entre el usuario y la base de datos de MongoDB, con tal de recrear además un sistema de matrícula y administración de grupos.

## Decisiones de diseño

Para el planeamiento del diseño de la base de datos en MongoDB, en un inicio podría tomar dos rumbos: Uno en donde se manejan varios documentos incrustados, de acuerdo con las dependencias respectivas (conformando al final un solo documento conteniendo todos los datos), u otro en donde se tienen varias colecciones, en donde los documentos se relacionan entre sí mediante sus identificadores o códigos, manejando arreglos con los diferentes valores asociados a cada documento o colección.

Luego de considerar cada posibilidad, se decidió que era más conveniente usar varias colecciones, debido a que son más flexibles que los documentos incrustados para manejar consultas/queries, usualmente ocupando sólo el comando \$lookup. Además, se consideró que este diseño es más manejable para su implementación en el stack MEAN, ya que todos son al final documentos planos, en vez de seguir una jerarquía y realizar búsquedas por profundidad.

Se implementaron ID's secuenciales en vez de los tradicional ObjectId() que genera automáticamente MongoDB, debido a su fácil interpretación y a la posibilidad de organizar los documentos en orden de inserción. Además, para asociar las colecciones entre sí, se vuelve más legible para los usuarios, que necesitarán conocer constantemente algunos códigos o identificadores para indicar estas asociaciones dichas anteriormente.

Para la interfaz, se tiene una pantalla de inicio de sesión y registro, que permite la creación de cuentas y acceso a los estudiantes y profesores. Sin embargo, las inserciones “en cadena” no están implementadas, en principio porque sólo se muestran las instituciones, escuelas y programas académicos ya existentes; luego se podrán crear automáticamente si el usuario a registrarse ingresa que pertenece a alguna entidad que aún no esté incluida.

Se permitirá que los estudiantes sean los encargados de administrar sus propios datos, es decir, de actualizar o borrar su usuario de la base de datos, en caso de que cambien de programa o se den de baja en algún curso o de la institución, entre otros casos. Al eliminarse de la base, tendrán la opción de crear de nuevo un usuario estudiante si así lo desean.

Los profesores tendrán la administración de las materias, cursos y grupos, por el momento. Todavía está en decisión si se les permitirá administrar otros aspectos como la escuela o la institución, o si esto se dejará a manos de algún nuevo usuario administrador. Para la creación de cursos y grupos, se tendrá la opción de ingresar el número de grupos a ingresar, para que luego el profesor pueda definir el horario de clases y las evaluaciones, que se agregarán en la segunda etapa de este proyecto.

## Bibliografía y Enlaces de Referencia

Videos:

“MEAN Stack Application Tutorial For Beginners”

<https://www.youtube.com/watch?v=fJ3MMIv3AQY>

“Mean Stack Tutorial - MongoDB, Express, Angular CLI and Node”

[https://www.youtube.com/watch?v=vLvNr3Wa5YI&index=1&list=PLC3y8-rFHvwj2o0LLotCYum\\_9wmGeTJx9](https://www.youtube.com/watch?v=vLvNr3Wa5YI&index=1&list=PLC3y8-rFHvwj2o0LLotCYum_9wmGeTJx9)

Páginas web:

“Mongoose CRUD (Create, Read, Update, Delete)”

<https://coursework.vschoool.io/mongoose-crud/>

Libros:

Lloyd, H. & Sevilleja, C. "MEAN Machine - A beginner's practical guide to the JavaScript stack", 2015.

Dayley, B. "Node.js, MongoDB and AngularJS Web Development", 2014.

Elrom, E. "Pro MEAN Stack Development", 2016.