

Proyecto de Software

Informe Trabajo Final

Grupo 02

Borrelli Franco Martin
Brost Pedro

Presentación del documento

Este documento se presenta como un informe sobre Trabajo de Final de Promoción del grupo 2 de la materia Proyecto de Software. El mismo se encuentra estructurado de acuerdo al índice que se encuentra a continuación:

Información del grupo	3
Fundamentación sobre la elección del framework elegido.	3
Descripción de los módulos desarrollados en el trabajo de la cursada que pudieron ser aprovechados para ser usados por el framework.	4
El mecanismo provisto para el manejo de seguridad y routing	4
La forma de manejar el MVC	5
Material de Referencia (libros, tutoriales, cursos, etc).	7

Información del grupo

Los dos integrantes del grupo son:

- Borrelli Franco Martín (Legajo: 13726/5, DNI: 39831178)
- Brost Pedro (Legajo: 13775/5, DNI: 40289549)

El grupo no sufrió modificaciones respecto al grupo conformado durante la cursada. Como nota de cursada obtuvimos un 6.

Fundamentación sobre la elección del framework elegido.

Gracias a varios trabajos realizados en otras materias de la facultad ya contábamos con cierta experiencia trabajando con frameworks. En particular, trabajamos con el framework PHP Symfony¹ en Ingeniería de Software 2 y con Ruby on Rails² en el taller de Ruby.

Debido a este motivo queríamos optar por hacer algo diferente. Aprovechar la oportunidad que este trabajo proporcionaba para aprender algo en lo que teníamos poco conocimiento. Fue así como decidimos profundizar nuestro conocimientos en Javascript, y las buenas prácticas de su utilización tanto del lado del cliente, como del lado del servidor (Node.js). Javascript hoy en día ha cobrado gran protagonismo, siendo una de las tecnologías más utilizadas en aquellos proyectos que buscan innovar.

Decidimos enfatizar también en un concepto nuevo para nosotros, el de las SPA (Single Page Applications). Si bien este concepto se mencionó brevemente durante la cursada, nos pareció la oportunidad perfecta para ver como es la dinámica de su desarrollo en comparación a la MPA (Multiple Page Application) que desarrollamos durante la cursada. A su vez gracias a esta decisión pudimos poner en práctica otros conceptos vistos tales como request de tipo AJAX y manejo del DOM.

De acuerdo a todo lo antes mencionados, desarrollamos dos proyectos por separado: una para el lado del cliente, utilizando la librería **React** y otra para el servidor, utilizando el framework **Express** para el desarrollo de una API Rest.

React es una librería javascript para la creación de interfaces de usuario desarrollada por Facebook. Hoy en día se posiciona dentro de las tecnologías más utilizadas para el desarrollo de SPA, junto a otras como Vue.js y Angular.

Por el lado del servidor decidimos utilizar Express ya que es uno de los frameworks javascript más utilizados actualmente, contando un amplia comunidad. Express cuenta con una documentación que facilita su aprendizaje de manera rápida y sencilla. Este framework tiene la capacidad de poder utilizar de manera sencilla otras librerías en conjunto que aportan funcionalidades más específicas. Entre estas podemos encontrar librerías que permiten la utilización de cors, de operaciones de log, de debug y de health-check.

¹ <http://symfony.es>

² <https://rubyonrails.org>

Para la base de datos, se decidió utilizar MongoDB por su fácil integración con las tecnologías javascript antes mencionadas. Esto se debe a que Mongo no es una base de datos relacional, sino una no-SQL basada en documentos.

Una gran ventaja de la elección de este stack denominado MERN (Mongo - Express - React - Node) es que requiere el aprendizaje de un único lenguaje de programación.

Descripción de los módulos desarrollados en el trabajo de la cursada que pudieron ser aprovechados para ser usados por el framework.

El módulo de reportes de usuario pudo adaptarse casi totalmente con facilidad. Esto se debe a que ciertas funcionalidades relacionadas con este módulo como por ejemplo la forma en que los gráficos se generan y los mecanismos de exportación en formato pdf, ya se resolvían con Javascript mediante la utilización de la librería HighCharts.

En términos generales también se adaptaron aspectos relacionados con la vista, tales como la disposición de las diferentes pantallas que componen el sitio y los elementos que se presentan en cada una de estas (como mostrar la información, donde poner botones, que información útil es importante agregar para el usuario). El código HTML y CSS utilizado durante el proyecto de la cursada no pudo reutilizarse completamente debido al cambio que implicó pasar de Bootstrap a Ant.Design. Pantallas tales como la pantalla principal, la página del login y las pantallas de error y mantenimiento se mantuvieron completamente.

Más allá de estas cuestiones, pasar de PHP a Javascript conllevó tener que arrancar casi desde cero en lo referido a las funcionalidades realizadas del lado del backend.

El mecanismo provisto para el manejo de seguridad y routing

Routing

Cliente

La manera de realizar routing dentro de una SPA es muy distinto a como se trabaja en una página web tradicional (MPA). Mientras que estas últimas suponen un recargo de página ante cada cambio de ruta, en una SPA esto se maneja desde el lado del cliente, siempre actualizando la página actual. En particular, para React existe una librería independiente denominada React Router que facilita los mecanismos de ruteo. Lo que destaca a React Router es que provee un mecanismo de ruteo dinámico, es decir que el ruteo se lleva a cabo mientras la aplicación se renderiza en lugar de utilizar por ejemplo un archivo de configuración.

Servidor

Para la realización de los distintos endpoints que conforman a la API REST desarrollada, se utilizaron las herramientas que provee Express para la realización de

rutas haciendo uso de distintos middlewares los cuales son uno de los principales conceptos dentro de Express.

Seguridad

Autenticación

Para la implementación de la autenticación no se utilizaron los mecanismos tradicionales, es decir mediante el uso de cookies y sesiones. Por lo contrario, se utilizó una tecnología denominada JWT (JSON Web Tokens) la cuál es la que se suele elegir a la hora de realizar una SPA. Esta técnica permite que ante el logueo de un usuario se cree un token firmado del lado del servidor el cual le servirá al cliente como evidencia de que es quien realmente dice ser. Esto se logra enviando el JWT al cliente, el cual lo almacenará y enviará a través de headers en todos los requests que requieran autenticación.

Autorización

Una de las particularidades de los JWT es que no pueden ser modificados por el cliente. Esto permite almacenar dentro de ellos cierta información relacionada con el usuario logueado. Dentro de esta podemos encontrar los distintos roles y permisos que el usuario posee. De esta manera, es fácil determinar para la aplicación si el usuario puede realizar ciertas operaciones. Hay casos en los que igualmente es necesario acceder a la base de datos para verificar los permisos que tiene un usuario en un determinado momento.

XSS (cross-site-scripting)

Uno de los principales beneficios de utilizar React es que provee mecanismos que permiten evitar la ocurrencia de XSS. React garantiza que nunca se podrá inyectar nada que no esté escrito explícitamente en la aplicación. Todo valor pasado a un elemento JSX se convierte en un string antes de ser renderizado.

El mecanismo provisto para operaciones CRUD

Express al ser un framework muy simple, en sí mismo no posee ningún mecanismo para la automatización de la generación de operaciones CRUD. Existen ciertas librerías independientes complementarias que proveen esta funcionalidad pero optamos por no utilizar ninguna de ellas. Más allá de esto, se decidió utilizar una denominada “boilerplate application”, la cual es una aplicación de referencia que sirve para comenzar a desarrollar de manera fácil y sencilla siguiendo buenas prácticas.

La forma de manejar el MVC

(explicando detalladamente el árbol de directorios)

Material de Referencia (libros, tutoriales, cursos, etc).

React:

- Documentación oficial de React : www.reactjs.org/docs
- Curso de React de Udemy: www.udemy.com/react-the-complete-guide-incl-redux
- Doc. de React Router: www.reacttraining.com/react-router/core/guides/philosophy
- Documentación oficial de Ant.Design: www.ant.design
- Documentación de gráficos HighChart: www.github.com/whawker/react-jsx-highcharts
- Documentación de Redux: www.redux.js.org/introduction

Express:

- Página Oficial de Express: www.expressjs.com
- Boilerplate Application: www.github.com/kunalkapadia/express-mongoose-es6-rest-api
- Documentación de Mongoose: www.mongoosejs.com