

MADRID, JUNIO 2017

Autor: David Vega Lichacz

Director: Julio García Martín

**Graduado en Ingeniería Informática**

Universidad Politécnica de Madrid

Escuela Técnica Superior de

Ingenieros Informáticos

**TRABAJO FIN DE GRADO**

Desarrollo de la capa de presentación de una plataforma de gestión de formas de pago y cálculo de condiciones en la gestión de créditos al consumo utilizando Angular 2

# **ÍNDICE**

[ÍNDICE 2](#_Toc483938192)

[ÍNDICE DE FIGURAS 3](#_Toc483938193)

[RESUMEN 4](#_Toc483938194)

[SUMMARY 5](#_Toc483938195)

[AGRADECIMIENTOS 6](#_Toc483938196)

[1. INTRODUCCIÓN 7](#_Toc483938197)

[2. MOTIVACIÓN 9](#_Toc483938198)

[3. OBJETIVOS 10](#_Toc483938199)

[4. DESCRIPCIÓN DE LOS CRÉDITOS AL CONSUMO Y LA GESTIÓN DE FORMAS DE PAGO 11](#_Toc483938200)

[5. FRAMEWORK: ANGULAR 12](#_Toc483938201)

[5.1. DESCRIPCIÓN 12](#_Toc483938202)

[6. OTROS FRAMEWORKS Y LIBRERÍAS ACTUALES: REACT, POLYMER, BACKBONE 14](#_Toc483938203)

[6.1. REACT 15](#_Toc483938204)

[6.2. POLYMER 16](#_Toc483938205)

[6.3. BACKBONE 17](#_Toc483938206)

[7. COMPARATIVA DE LOS MEJORES FRAMEWORKS Y LIBRERÍAS FRONT-END 17](#_Toc483938207)

[7.1. COMPARATIVA: ANGULAR, REACT, POLYMER, BACKBONE 17](#_Toc483938208)

[8. OTRAS HERRAMIENTAS Y FRAMEWORKS UTILIZADOS 18](#_Toc483938209)

[8.1. BOOTSTRAP 18](#_Toc483938210)

[8.2. FIREBASE 20](#_Toc483938211)

[9. DESARROLLO 21](#_Toc483938212)

[9.1. DISEÑO DEL CLIENTE WEB 21](#_Toc483938213)

[9.2. DESARROLLO E IMPLEMENTACIÓN DEL CLIENTE WEB 21](#_Toc483938214)

[10. CONCLUSIONES 22](#_Toc483938215)

[11. REFERENCIAS BIBLIOGRÁFICAS 23](#_Toc483938216)

# **ÍNDICE DE FIGURAS**

\*ÍNDICE DE LAS ILUSTRACIONES\*

# **RESUMEN**

Los créditos al consumo son aquellos créditos que, bajo la forma de pago aplazado, préstamo o cualquiera facilidad de pago, se otorgan a las personas físicas, destinados a financiar el gasto en la adquisición de bienes o servicios para uso personal y familiar (no profesional).

El proyecto se centra en el desarrollo de la capa de presentación para un conjunto de servicios (gestión de formas de pago y cálculo de condiciones del crédito) de la plataforma de gestión de créditos al consumo.

La plataforma de gestión de créditos se define como un back-end de servicios web que implementa la contratación de financiación de una entidad financiera.

A lo largo de este documento se hace un repaso de todas las fases del proyecto. Pasando por las tecnologías utilizadas, el análisis, el diseño, el desarrollo y por último las conclusiones. Al finalizar la lectura de esta memoria, se tendrá una idea clara de la arquitectura del sistema y de su implementación.

# **SUMMARY**

\*RESUMEN EN INGLÉS\*

# **AGRADECIMIENTOS**

En primer lugar, me gustaría agradecer a la Escuela Técnica Superior de Ingenieros Informáticos de la Universidad Politécnica de Madrid, por haberme formado en una de las mayores pasiones de mi vida como es la informática. Gracias al conocimiento y a la madurez que he obtenido a lo largo de estos años, he sido capaz de mejorar no solo profesionalmente sino también personalmente.

Me gustaría también darle las gracias a mi tutor de Trabajo de Fin de Grado, Julio García Martín, por el esfuerzo y la dirección prestada a lo largo de este trabajo.

Por otra parte, quería agradecer a BABEL Sistemas de Información por darme la posibilidad de realizar las prácticas curriculares, las prácticas extracurriculares y mi reciente incorporación como empleado, y formarme en algunas de las tecnologías referidas a las aplicaciones web que más me motivan.

Por último, quería dedicar este trabajo a mi familia y a mis mejores amigos, y darles gracias por todo el apoyo a lo largo de estos años.

De nuevo, gracias por todo.

# **INTRODUCCIÓN**

En la actualidad cada vez son más habituales las transacciones por vías telemáticas, las cuales incluyen la comercialización de bienes y servicios. Esto ha permitido que miles de personas puedan realizar sus compras sin tener que acudir físicamente a los establecimientos de venta de productos.

Al mismo tiempo, la proliferación de estas formas de pago ha impulsado la industria de los negocios online o e-commerce, lo que ha mejorado la economía de cientos de países por todo el mundo.

Por otro lado, gracias a los créditos al consumo, se pueden realizar compras personales y familiares sin que el cliente tenga la necesidad de sufragar su importe en ese mismo momento. El consumidor puede adquirir cualquier bien o servicio sin tener en cuenta sus actividades profesionales. Entre dichas compras se pueden encontrar automóviles, electrodomésticos, muebles, gastos del hogar, viajes, etc.

Mediante esta clase de préstamos, es más sencillo adquirir nuevos productos y por lo tanto, se ven incrementadas las ventas, lo que lleva a un aumento del consumo, y esto a su vez, repercute positivamente en el crecimiento económico global y una mejora en el nivel de vida del ser humano.

En la era de la información, gracias al desarrollo de las comunicaciones y de la transmisión de la información, cada vez es más frecuente utilizar los servicios que ofrecen los negocios online, por lo que se debe asegurar un correcto funcionamiento y un cálculo correcto de las condiciones del crédito que el cliente va a disponer.

Por otra parte, la expansión de internet ha permitido que surjan miles de estos negocios, los cuales pretenden ofrecer al usuario una interfaz amigable, pero que a su vez tenga buena usabilidad y navegabilidad.

Actualmente, existe una gran rivalidad en cuanto a ofrecer al usuario una interfaz que le resulte fácil de entender y a su vez mantenga una buena funcionalidad. Pero en muchos casos los diseños pueden acabar resultando demasiado complicados y pueden resultar en la desorientación del usuario.

Mediante este trabajo, se pretende ofrecer al usuario una interfaz que resulte cómoda, coherente y a su vez, mantenga la funcionalidad de una plataforma de gestión de créditos al consumo. Este proyecto se llevará a cabo en varias etapas. En la primera se dará una descripción y análisis de la plataforma de créditos al consumo, la cual ya ha sido desarrollada e implementada en un trabajo diferente al expuesto.

Tras eso se realizará un estudio y análisis del framework Angular, realizando una comparativa con otros frameworks. Así mismo, se explicarán otras librerías y herramientas utilizadas. Finalmente, se mostrará el desarrollo de la capa de presentación, cuyo diseño e implementación se realizan mediante los frameworks y bibliotecas elegidas.

# **MOTIVACIÓN**

Como motivación, debo decir que me ha influido mi afición a la lectura de temas económicos y finanzas personales. Estoy orgulloso por poder realizar un trabajo sobre una materia tan relacionada con lo que realmente me gusta aparte de la informática.

También, ha sido influyente el iniciar mis prácticas en BABEL, donde he podido familiarizarme con frameworks de front-end y he trabajado para el cliente ING Direct como desarrollador full-stack. Gracias a esto he descubierto otra gran pasión dentro del mundo de la informática, que es la de desarrollar tanto el front-end como back-end de aplicaciones web para el mundo de la banca y las finanzas.

En conclusión, puedo decir que los últimos meses me han hecho reflexionar bastante sobre la importancia de los servicios financieros, y a su vez, la importancia de que el diseño y desarrollo de las páginas web sea cada vez más moderno y adaptable de acuerdo a los requisitos del cliente.

# **OBJETIVOS**

Este proyecto, se va a describir un sistema que sirva como capa de presentación de una plataforma de gestión de créditos al consumo y como representación de la interfaz de usuario, recogiendo y agrupando los datos recibidos por parte de unos servicios web.

Mediante este trabajo, se persigue el objetivo de crear y mantener una aplicación web moderna mediante una tecnología avanzada y con capacidad de modelo vista controlador (MVC).

En este caso en particular, se pretende crear una aplicación web en la cual el usuario pueda obtener financiación por medio de créditos al consumo de forma rápida, accesible y segura.

Por lo tanto, el objetivo es la creación de una capa de presentación preparada para las exigencias del consumidor para la financiación por medio del pago aplazado de productos y servicios a adquirir en empresas de e-commerce y otros negocios de internet.

# **DESCRIPCIÓN DE LOS CRÉDITOS AL CONSUMO Y LA GESTIÓN DE FORMAS DE PAGO**

En la actualidad, cada vez es más importante la financiación de productos y servicios para el consumidor. A menudo se presentan ocasiones en las que el cliente prefiere elegir otras modalidades de pago, y una de las que mayor impulso están adquiriendo es la de crédito al consumo, que se obtiene en el momento de la compra y solo para esa operación. Gracias a que se le ofrece al consumidor esta forma de financiación, la empresa vendedora aumenta sus ventas y obtiene mayores beneficios.

Es importante, en primer lugar, explicar lo que es un crédito al consumo. Esta modalidad de pago se define como el contrato que una persona física o jurídica, que regularmente realiza una actividad comercial, concede al consumidor con el pretexto de ofrecerle la financiación de la adquisición de bienes o servicios para uso personal o familiar, mediante la forma de pago aplazado, préstamo o cualquier medio equivalente de empréstito. No entran dentro de esta definición la prestación continuada de servicios o el suministro de bienes de un mismo tipo.

Dentro del concepto de crédito al consumo, se pueden diferenciar las siguientes formas de pago: préstamo clásico (o pago a plazos), crédito de libre inversión, crédito revolving, tarjetas de crédito y crédito de vehículo, entre otros.

En la gestión de las formas de pago por medio de una aplicación web, una de las partes más importantes es el front-end, donde el usuario pueda realizar pagos de forma rápida, cómoda y segura utilizando únicamente sus datos personales y un medio de pago tradicional, como una cuenta bancaria o tarjeta de crédito.

Una de las principales ventajas que tiene pagar mediante una plataforma que gestiona los pagos, es la de que el vendedor no obtiene información en ningún momento de la tarjeta de crédito o cuenta bancaria del comprador. La única autorizada a procesar los datos del comprador es la entidad financiera que presta este tipo de servicio a través de la aplicación.

Se puede concluir entonces que, una plataforma gestora de formas de pago que ofrece la modalidad de pago de crédito al consumo, no sólo puede aumentar las ventas de los negocios de internet, sino que también permite que un cliente pueda pagar los productos a plazos y realizar compras de forma cómoda y segura.

# **FRAMEWORK: ANGULAR**

Hasta hace poco, se procuraba que la mayoría de la lógica y las vistas de las aplicaciones web residieran en el servidor, utilizando el cliente únicamente para obtener dichas vistas e interactuar con ellas. Sin embargo, con la aparición de HTML5 y ECMAScript (lenguaje que sirve como mejora de JavaScript), han cogido impulso las páginas web que se comunican con el servidor mediante servicios REST y que cuentan con una sola página (single page application).

Gracias a esta clase de tecnologías, han surgido poderosos frameworks como Angular, el cual se va a describir a continuación.

# DESCRIPCIÓN

Angular, también llamado Angular 2, es una tecnología del lado del cliente, un framework TypeScript de código abierto, desarrollado por Google y utilizado principalmente para la creación y el mantenimiento de aplicaciones web de una sola página. El marco de trabajo funciona con las tecnologías web HTML, CSS y JavaScript, impulsando de este modo las aplicaciones basadas en el Modelo Vista Controlador (MVC). De esta forma, se utiliza el navegador web como cliente ligero sin necesidad de que el usuario tenga que descargar o instalar más software del que su propio navegador incluye.

El framework extiende de HTML con etiquetas propias, permitiendo construir aplicaciones web interactivas y modernas, y aumentando el nivel de abstracción entre el desarrollador y el proceso de desarrollo. Al mismo tiempo, Angular cuenta con la compatibilidad necesaria para ser ejecutado en todos los navegadores de última generación.

A diferencia de las aplicaciones web tradicionales, las aplicaciones desarrolladas con Angular tienen plantillas actualizadas en tiempo real, lo que da lugar a vistas completamente dinámicas. Otra de las ventajas fundamentales de Angular es la de separar cada componente de la aplicación, adoptando el estándar de componentes web. Esto permite que se puedan reutilizar porciones de código de otros proyectos con bastante facilidad.

Angular permite, además, crear un enrutamiento para cada uno de los componentes, y de esta forma poder asegurar el acceso y la seguridad de los mismos. Por otra parte, el entorno de trabajo proporciona todo lo necesario para realizar peticiones HTTP a servicios REST, por lo que se trata de un framework muy potente y que ofrece muchas posibilidades de desarrollo.

Una de las características principales de Angular, y la diferencia fundamental respecto a su predecesor AngularJS, es el estándar de utilización del lenguaje TypeScript para realizar el desarrollo de las aplicaciones web. La mayor ventaja de utilizar este lenguaje es el nivel de documentación disponible y la consistencia que ofrece frente a ECMAScript.

TypeScript incluye todas las características de ECMAScript, introduciendo además funcionalidades extra y correcciones de sintaxis. Una importante particularidad de TypeScript y que lo sitúa por encima de JavaScript es que permite definir el tipo de las variables a declarar. Por otro lado, dispone del patrón de diseño decorador, que se utiliza para añadir funcionalidad a un objeto de forma dinámica. Otro detalle importante es el de que TypeScript permite declarar datos tanto públicos como privados, ya sea en las propias clases o en el constructor de las mismas.

Un apunte importante sobre TypeScript, es que Angular, convierte en tiempo de compilación el lenguaje TypeScript a JavaScript.

En la figura 1, se puede observar un código de ejemplo de un componente en Angular, nótese el decorador @Component y la clase con código en el lenguaje TypeScript.

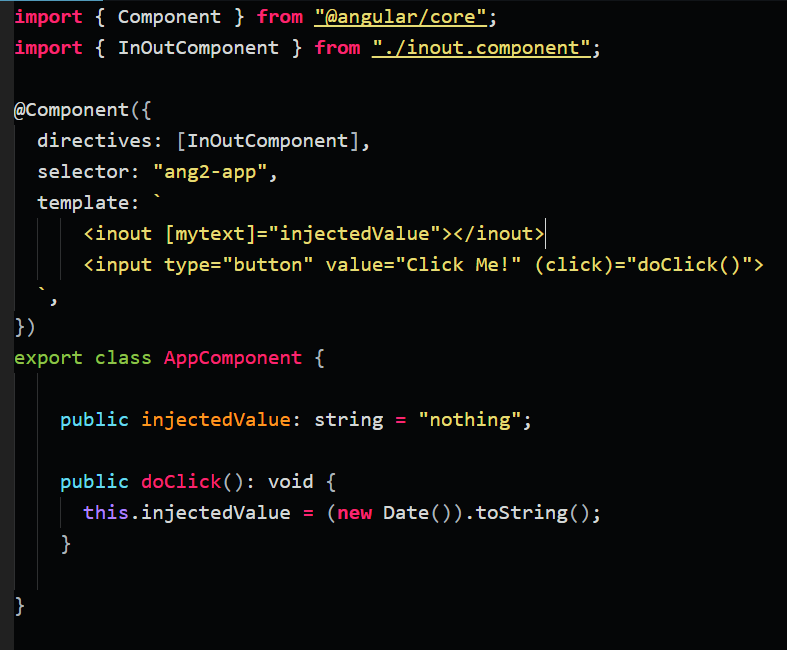


Fig. 1. Angular

# **OTROS FRAMEWORKS Y LIBRERÍAS ACTUALES: REACT, POLYMER, BACKBONE**

En los últimos años se han desarrollado numerosos frameworks para simplificar el diseño y el desarrollo de páginas web utilizando el lenguaje JavaScript. En este apartado, se explican otras soluciones existentes en el mercado para la creación de la capa de presentación de aplicaciones web.

# REACT

React, también conocido como ReactJS, es una biblioteca JavaScript en código abierto, mantenido por Facebook e Instagram y utilizado para el desarrollo y mantenimiento de aplicaciones web de una sola página. Permite desarrollar aplicaciones con menos código que usando JavaScript puro o librerías como jQuery. Esta potente librería está construida bajo el patrón de diseño Modelo Vista Controlador (MVC).

La principal característica de React es que permite que las vistas sean actualizadas en tiempo real para reflejar los cambios de estado de la aplicación web. Esta biblioteca utiliza el patrón de diseño por componentes para modular el código fuente de la aplicación web. La biblioteca está escrita en ECMAScript 5 y las aplicaciones pueden ser creadas con ECMAScript 5 o 6.

React no puede ser considerado un framework, puesto que no cuenta con soporte para enrutamiento o comunicación con el servidor. Para ello, tiene que ser combinado con otros frameworks o librerías para lograr dicho funcionamiento. Por otro lado, no trabaja bien combinado con librerías como jQuery.

En la figura 2 se puede contemplar un fragmento de código de React donde se crea un componente “botón”.

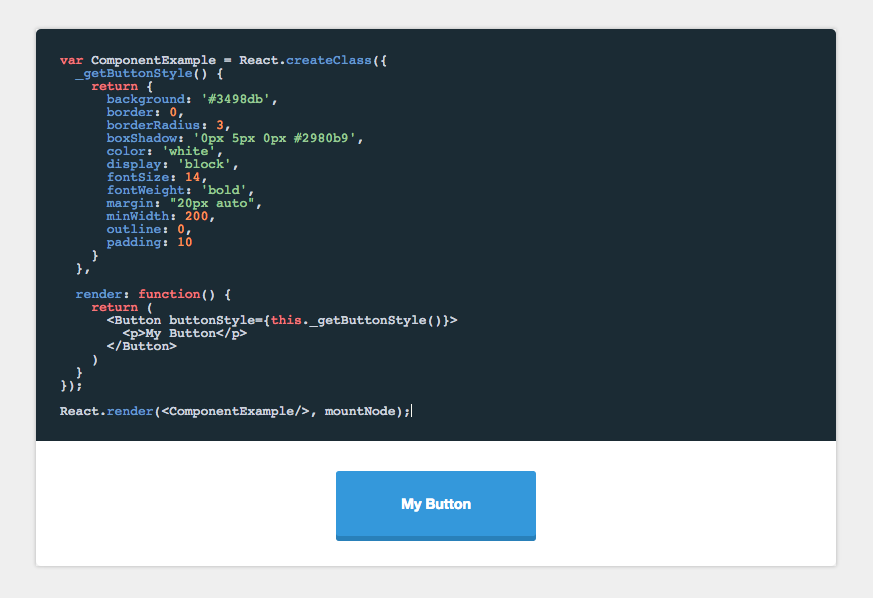


Fig. 2. React

# POLYMER

Polymer es una librería JavaScript en código abierto para la construcción y manutención de aplicaciones web usando el patrón de diseño de componentes web. Esta librería está siendo desarrollada por Google y otros desarrolladores independientes, y su código fuente puede ser visualizado a través de la plataforma GitHub.

Esta librería permite simplificar la creación de componentes web, y la reutilización de estos de forma muy sencilla. Estos componentes, tan solo necesitan de ser importados dentro de Polymer para poder ser utilizados.

La principal ventaja de Polymer es la modularidad de sus aplicaciones web. Otra de sus principales características es que sus componentes utilizan el estándar de Google de material design. En contraste, esta tecnología tiene mayores problemas para realizar tareas como el enrutamiento de los componentes.

En la figura 3 se pueden ver numerosos componentes de Polymer que pueden ser añadidos a una aplicación web que utiliza esta librería.

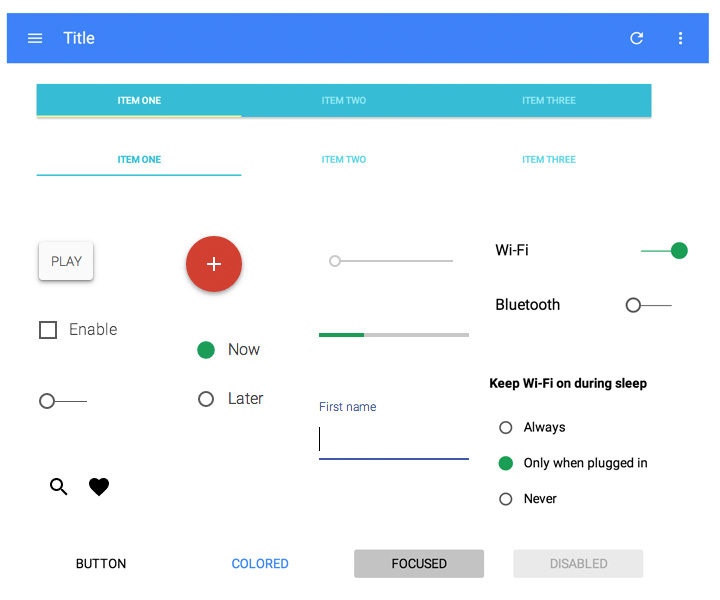


Fig. 3. Polymer

# BACKBONE

Backbone, también conocido como Backbone.js, fue creado en 2010 por un desarrollador llamado Jeremy Ashkenas. A día de hoy, esta librería es usada por grandes corporaciones como ING Direct para el desarrollo de la capa de presentación de sus aplicaciones web.

Backbone es una herramienta de desarrollo mediante el lenguaje JavaScript, y que cuenta con una interfaz REST por JSON. Al igual que otras librerías actuales, está construida bajo el paradigma de diseño Modelo Vista Controlador, sincroniza las partes de sus aplicaciones en tiempo real y permite crear aplicaciones web de una sola página.

Sin embargo, Backbone por sí solo no contiene suficientes patrones de diseño como para guiar al desarrollador a la hora de crear una aplicación. Para solventar este problema en parte, se creó la librería Marionette.js, la cual añade una serie de funciones adicionales a Backbone y patrones de diseño como el poder modular los componentes de la librería o tener el control sobre las rutas de las vistas de la aplicación.

En la figura 4 se observa un fragmento de código en Backbone, donde se puede apreciar la utilización de los patrones Collection, View o Router que proporciona la librería.

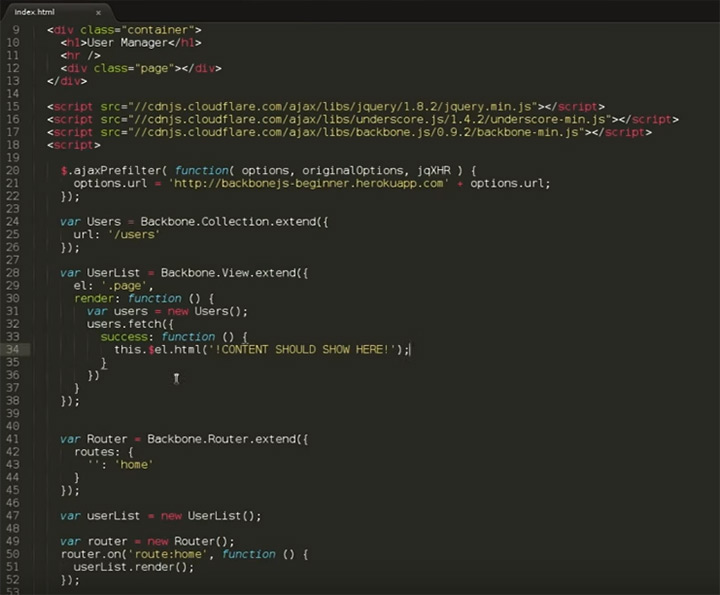


Fig. 4. Backbone

# **COMPARATIVA DE LOS MEJORES FRAMEWORKS Y LIBRERÍAS FRONT-END**

# COMPARATIVA: ANGULAR, REACT, POLYMER, BACKBONE

Las cuatro herramientas son muy similares en cuanto a funcionalidad básica y la utilización de patrones de diseño y paradigmas utilizados en la actualidad del desarrollo de aplicaciones web.

Podemos observar que todas tienen en común que utilizan el lenguaje JavaScript o ECMAScript, que sirven para crear aplicaciones web de una sola página y que siguen el patrón de diseño MVC. Pese a estas similitudes, es necesario notar que cada una de estas herramientas son completamente diferentes, y cada una tiene sus pros y sus contras.

Por un lado, Angular es un framework muy maduro que cuenta con múltiples funcionalidades que permiten hacer prácticamente cualquier cosa dentro de una aplicación web, pero manteniendo importantes patrones de diseño y respetando muchos paradigmas de la programación orientada a objetos. Aunque este entorno de trabajo ofrezca una potencia abrumadora, su curva de aprendizaje puede ser demasiado lenta y costosa debido a su tamaño. Sin embargo, a medida que los desarrolladores obtienen la experiencia necesaria, Angular se puede convertir sin duda en una de las mejores opciones actuales para desarrollar una aplicación web de una sola página.

Por otro lado, React ofrece mucha flexibilidad a la hora de integrarse con otro tipo de arquitecturas y librerías. No proporciona un marco de trabajo sumamente amplio, sin embargo resulta ser sencillo y fácil de aprender. Al utilizar un lenguaje universal como JavaScript, permite que un usuario experimentado en este lenguaje pueda obtener un amplio conocimiento de React en un breve periodo de tiempo. Pero su mayor ventaja frente a otras librerías, es su facilidad para debugear y encontrar errores en tiempo de ejecución.

Por otra parte, Polymer ofrece lo suficiente para manipular componentes web creados por la comunidad y además añade otras características que siguen los patrones de diseño de material design. No se trata de un framework como tal, pero puede servir como una solución suficiente para crear aplicaciones web de forma rápida y sencilla.

Por último Backbone, se trata de una herramienta que permite programar una aplicación web sin apenas una curva de aprendizaje. Aunque sea una librería menos potente que otras soluciones en el mercado, proporciona mucho control, ya que se puede crear y modificar el comportamiento de los elementos de una aplicación de prácticamente cualquier forma posible.

En conclusión, para proyectos enfocados en otro tipo de aplicaciones web, es posible que se busque una solución más sencilla y rápida de realizar, sin necesidad de tener una curva de aprendizaje tan pronunciada. Dependiendo del tamaño de la aplicación y del conocimiento del desarrollador, se puede elegir utilizar React, Polymer o Backbone. Sin embargo, para la realización de un proyecto de tamaño considerable, se debería elegir Angular debido a la enorme funcionalidad y potencia que ofrece.

# **OTRAS HERRAMIENTAS Y FRAMEWORKS UTILIZADOS**

Una vez decidido el entorno de trabajo, a la hora de diseñar y desarrollar una aplicación web surgen problemas como que la aplicación sea ajustable a todos los dispositivos, que presente un diseño sencillo y amistoso, o que cuente con sincronización y almacenamiento de datos en tiempo real.

En este apartado se presenta un framework de diseño (Bootstrap) y una API de base de datos (Firebase), ambos de última generación, que se utilizan en numerosas aplicaciones web actuales para solventar esta clase de retos, y cuyo uso está directamente ligado con el desarrollo de este proyecto.

# BOOTSTRAP

Bootstrap es un framework de diseño web, desarrollado y mantenido por Twitter desde el año 2011, y utilizado para crear páginas web de diseño adaptable, lo que quiere decir que se puedan ajustar a cualquier clase de dispositivo, independientemente de la resolución del aparato. Además, Bootstrap es completamente compatible con la mayoría de navegadores disponibles.

El entorno de trabajo funciona principalmente con las tecnologías CSS, HTML y JavaScript, es de código abierto, y permite que las aplicaciones web tengan un diseño amigable, agradable y atractivo. El framework se encuentra bajo licencia MIT (código abierto) y su desarrollo puede ser seguido a través de su repositorio de GitHub.

La principal ventaja de utilizar Bootstrap es que permite simplificar el proceso de maquetación de las aplicaciones web, utilizando buenas prácticas y estándares de material design. Este entorno de trabajo permite crear cualquier clase de página web mediante la utilización de los estilos y elementos que ofrecen sus librerías.

Su sencilla instalación y utilización dentro del mundo de las aplicaciones web ha conseguido extender su uso. Otra de las principales razones de su extendida utilización es la de disponer de una fuerte comunidad prestando soporte, creando componentes y ofreciendo plugins y librerías en JavaScript.

En la figura 4 se pueden contemplar el diseño y los componentes a utilizar en una aplicación web que utiliza Bootstrap.



Fig. 4. Bootstrap

# FIREBASE

Firebase es una potente API, desarrollada por Google, y utilizada para almacenar y sincronizar datos en tiempo real utilizando una base de datos de tipo NoSQL. Este servicio permite, además, que la aplicación web esté completamente sincronizada con los dispositivos que navegan dentro de la aplicación, permite el funcionamiento offline y la sincronización de cuando el dispositivo vuelva a tener acceso a internet.

Una de las principales características de Firebase es que proporciona la velocidad, escalabilidad y seguridad necesarias para poder hacer funcionar cualquier aplicación web con un almacenamiento de datos y una sincronización en tiempo real.

La ventaja de utilizar una base de datos de tipo NoSQL es la de que ofrece un mayor rendimiento frente al procesamiento de datos que las tradicionales bases de datos SQL, sobre todo cuando el volumen de datos puede crecer muy rápidamente en momentos puntuales o las necesidades del proceso no se pueden prever. Esto significa que es adecuado para aplicaciones las cuales esperar picos de uso del sistema por parte de múltiples conexiones simultáneas de una gran cantidad de usuarios.

También es importante mencionar que apenas no es necesario tener conocimiento en bases de datos NoSQL para su gestión y almacenamiento de datos.

En la figura 5 se puede observar el cuadro de mandos de Firebase con una serie de datos en formato NoSQL.

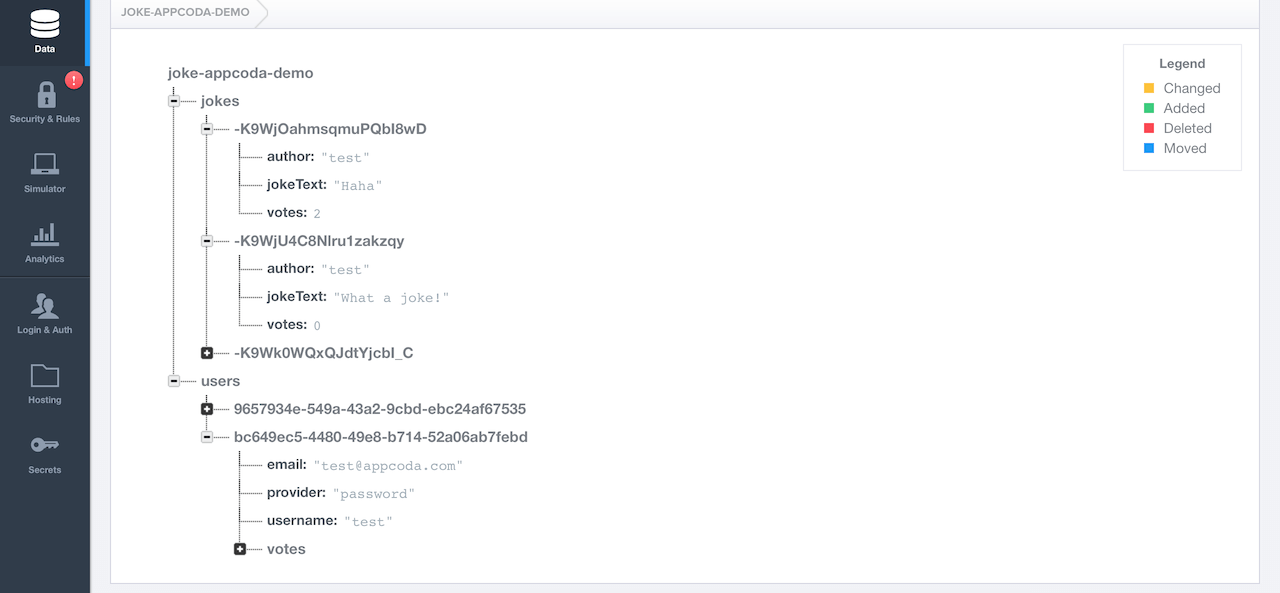


Fig. 5. Firebase

# **DESARROLLO**

# MODELO DE DATOS

TOKEN DE ENTRADA (PREDEF. LO ENVIA TIENDA A APP) (IDUSER, IDCESTA{PROD…})

USUARIO -> A BD

DATOS PERSONALES -> A BD

TARJETA -> A BD

CONDICIONES -> CALCULO INVENTADO (TAE, COMIS.) MIRAR PAGINA ING

TOKEN DE SALIDA -> (IDUSER, IDCESTA{PROD…}, CONF. PAGO)

# DISEÑO DEL CLIENTE WEB

\*HOW YOU STARTED TO MAKE THE FRONT END, THINGS, COMPONENTS, CLASSES, DIAGRAMS, ETC\*

It could be good to mention how you connected the front to the back (aka firebase), class diagrams and some kind of shit. IMAGES, POSSIBILITIES, ALL THINGS YOU CAN DO.

(ABOUT 12-20 MORE PAGES FOR THIS SECTION, INCLUDE IMAGES OF THE FRONT-END, COMPONENT DIAGRAMS AND THAT KID OF SHIT)

MAXIMUM OF 40-45 PAGES

35 YELLOW LINE, 40 ORANGE LINE, 45 REDLINE

22ACTUAL + 15 DESARROLLO = 37 PAG

# **CONCLUSIONES**

\*BLABLABLA\*

# **REFERENCIAS BIBLIOGRÁFICAS**

1. Roberto García Martínez, ADICAE. *El crédito al consumo en España.* Disponible en: <http://asp.adicae.net/proyectos/internacionales/dosierlegislacion/dosierpdf/A%20Documentos%202.pdf>