

# **Aplicación web para aprender a usar Git desde el terminal**

Memoria del Proyecto

Autor: Jose Tirado Blázquez

Tutor: Tomás Fernández

Sevilla 28 de mayo de 2016

## The MIT License (MIT)

Copyright (c) 2016 JOSE TIRADO BLÁZQUEZ

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Licencia MIT (versión Española)

Copyright (c) 2016 JOSE TIRADO BLÁZQUEZ

Se concede permiso por la presente, de forma gratuita, a cualquier persona que obtenga una copia de este software y de los archivos de documentación asociados (el "Software"), para utilizar el Software sin restricción, incluyendo sin limitación los derechos de usar, copiar, modificar, fusionar, publicar, distribuir, sublicenciar, y/o vender copias de este Software, y para permitir a las personas a las que se les proporcione el Software a hacer lo mismo, sujeto a las siguientes condiciones:

El aviso de copyright anterior y este aviso de permiso se incluirán en todas las copias o partes sustanciales del Software.

EL SOFTWARE SE PROPORCIONA "TAL CUAL", SIN GARANTÍA DE NINGÚN TIPO, EXPRESA O IMPLÍCITA, INCLUYENDO PERO NO LIMITADO A GARANTÍAS DE COMERCIALIZACIÓN, IDONEIDAD PARA UN PROPÓSITO PARTICULAR Y NO INFRACCIÓN. EN NINGÚN CASO LOS AUTORES O TITULARES DEL COPYRIGHT SERÁN RESPONSABLES DE NINGUNA RECLAMACIÓN, DAÑOS U OTRAS RESPONSABILIDADES, YA SEA EN UN LITIGIO, AGRAVIO O DE OTRO MODO, QUE SURJA DE O EN CONEXIÓN CON EL SOFTWARE O EL USO U OTRO TIPO DE ACCIONES EN EL SOFTWARE.

## Resumen del Proyecto

El proyecto Aprende a Usar Git es una aplicación Web que pretende enseñar, a sus usuarios, por medio de la simulación, a usar Git desde el terminal.

Hoy en día existen numerosos tutoriales, páginas de enseñanza, vídeos en Youtube/Vimeo u otros medios de distribución de contenido audiovisual pero casi todos ellos en inglés. No existe la posibilidad, o al menos no de forma rápida, de encontrar un tutorial de Git actualizado a la última versión y que te ofrezca la posibilidad de aprender usando los mismos comandos que te enseñan.

Aprende a Usar Git, no sólo te muestra los comandos básicos de consola para trabajar con Git, sino que además, tiene una simulación de consola en la que podrás ir probando los comandos paso a paso e ir comprobando el resultado in situ, ayudando así al usuario a que no tenga miedo de probar, ya que no es un entorno real sino simulado.

Por otra parte, a la par que se avanza en el tutorial, dentro de la simulación se irán creando los requisitos ajenos a la aplicación para que ayuden al rápido aprendizaje del usuario, evitando que éste tenga que perder el tiempo en crear tanto un repositorio en un servidor remoto como los diferentes ficheros para ir probando los comandos. Todo ello, con la visión de querer fomentar el aprendizaje rápido y eficaz en el que se basa este proyecto.

## Índice

- 1 Presentación del proyecto
  - 1.1 Introducción
  - 1.2 Objetivos
  - 1.3 Tecnologías y herramientas empleadas
- 2 Análisis de la aplicación
  - 2.1 Definición de la aplicación
  - 2.2 Definición de las interfaces de usuario
    - 2.2.1 Página de introducción
    - 2.2.2 Página del tutorial
  - 2.3 Especificación del plan de pruebas
    - 2.3.1 Pruebas unitarias
    - 2.3.2 Pruebas de integración
    - 2.3.3 Pruebas de implantación
- 3 Diseño de la aplicación
  - 3.1 Identificación de ficheros y directorios
  - 3.2 Necesidades de los equipos
  - 3.3 Necesidades del software
- 4 Desarrollo
- 5 Referencias

# 1 Presentación del proyecto

## 1.1 Introducción

Este proyecto tiene su campo de actuación dentro de la comunidad hispano-hablante ya que el tutorial está enteramente redactado en castellano. No pretende reemplazar a los tutoriales, guías y/o vídeos anglosajones que ya existen (estén más o menos actualizados), sino que la aplicación trata de ser un complemento que permite a aquellos usuarios que no entienden correctamente el inglés a poder utilizar Git sin ningún tipo de problema.

El tutorial, avanzará paso a paso desde la creación del repositorio local hasta la subida a un servidor remoto que previamente habremos configurado. Todo ello, con sus respectivas explicaciones de cada comando para que el usuario pueda hacerse una idea más clara de lo que realmente está pasando al introducir uno u otro comando.

## 1.2 Objetivos

Aprende a Usar Git pretende ser una aplicación Web en la cual se presente de forma adecuada y fácil de interpretar los comandos básicos para poder trabajar con Git desde el terminal, independientemente del Sistema Operativo del equipo, el navegador o el dispositivo.

Actualmente, las guías, tutoriales y vídeos se encuentran en su gran mayoría dirigidos a personas que entienden el inglés y, el pequeño porcentaje dirigido a la comunidad hispano-hablante o bien está incompleto o bien está desactualizado.

Como se ha indicado anteriormente, no es un objetivo de este proyecto reemplazar lo que ya existe en Internet, sino que trata de ser un complemento de los mismos dirigido particularmente a la comunidad hispano-hablante.

A continuación se describen los objetivos del proyecto en detalle:

- **Proporcionar una aplicación Web accesible para todos.**  
Se trata de una aplicación Web accesible desde cualquier navegador ya sea libre o propietario sin necesidad de contratación. Además, trata de ser responsive para que no tengas que preocuparte de entrar desde un ordenador ya que desde un móvil o una tableta también podrías.
- **Presentar toda la información en castellano.**  
Toda la información que se muestra en la página está enteramente escrita en castellano.
- **Aprender y practicar en un mismo paso.**  
Se ha incluido en nuestra aplicación Web una simulación de una terminal para que el usuario pueda ir probando los diferentes comandos de manera que su aprendizaje se vea reforzado con la práctica al mismo tiempo.

## 1.3 Tecnologías y herramientas empleadas

Repasemos que herramientas y tecnologías usaremos para este proyecto.

Dado que el proyecto no estará subido a Internet (de momento), habrá que crear una simulación local de un servidor; para que la aplicación Web pueda funcionar sin ningún tipo de problema. Para ello, Xampp (o en su defecto Lampp para sistemas Linux) nos ofrece el entorno perfecto en el que podemos hacer toda clase de pruebas de manera local, ideal por ejemplo para realizar una presentación del proyecto.

Para mantener el código accesible desde cualquier dispositivo conectado a Internet y poder trabajar en diferentes líneas al mismo tiempo, se ha empleado un control de versiones, Git, alojado en un servidor remoto gratuito, GitHub.

Respecto a las tecnologías empleadas, cabe destacar que se ha empleado AngularJS, por su potencial con las páginas únicas (o One Pages en inglés), ya que nos ofrece la posibilidad de ir navegando por los diferentes pasos del tutorial de manera que parte del contenido (la información de cada paso) sea dinámico, mientras que por otro lado exista un contenido estático (el fondo, el header, el footer y el terminal) con lo que poder mantener un registro de las acciones que el usuario ha realizado en la terminal y que pueda consultar todos los pasos realizados con anterioridad sin necesidad de ir adelantando o atrasando pasos en la aplicación.

Por otro lado, se ha usado Javascript junto a jQuery para el control total de la simulación de la terminal. Se han añadido algunos comandos útiles de una terminal de Linux que podrían ayudar al usuario, además de los comandos de Git que se enseñan desde la aplicación.

Por último, la página de introducción está construida en HTML5 sustentada por Bootstrap, un canvas como capa superior para dar color y que evita interactuar con un vídeo que se ha incrustado de fondo, originario de Youtube.

## 2 Análisis de la aplicación

### 2.1 Definición de la aplicación

La página de introducción (la que contiene el vídeo de fondo) es la encargada de presentar la página de manera rápida dando la posibilidad de o bien entrar a ver el vídeo o bien redirigirte al comienzo del tutorial.

La primera opción redirigirá a la página de Youtube donde se podrá visualizar el vídeo que se muestra de fondo en la aplicación. La segunda opción, redirigirá al comienzo del tutorial de Aprende a Usar Git.

Una vez dentro del tutorial, podemos dividir la página en 4 partes:

**1. El Header**

Se compone de la barra horizontal de la parte superior de la página. Es meramente estética.

**2. Las Instrucciones / Contenido dinámico**

El siguiente es un bloque de contenido dinámico que se refresca gracias a AngularJS que mediante el uso de AJAX es capaz de generar contenido dinámico sin necesidad de refrescar la página al completo. En él, se muestra un pequeño resumen del funcionamiento del siguiente comando que usará el usuario así como de las instrucciones necesarias para avanzar al siguiente paso del tutorial.

**3. La simulación del terminal**

La terminal de nuestra aplicación está adaptada a las necesidades del proyecto. Como es de esperar, no incluye todas las opciones de una terminal real, aunque se han incluido algunos comandos útiles procedentes de una terminal de Linux como pueden ser el *help*, *ls*, *cat* o *echo*. Además, incluye todos los comandos que se mostrarán en el tutorial, con el fin de que el usuario pueda probarlos sin ningún tipo de restricción.

**4. El Footer**

El pie de página se compone de una pequeña línea de texto con un enlace que redirige al código fuente alojado en GitHub.

## 2.2 Definición de interfaces de usuario

Al ser una aplicación Web, todas las funciones las presta a través de páginas Web a las que se acceden por medio de un navegador.

La toma de datos por parte del usuario se harán con el uso de un input incrustado en la simulación del terminal. La información de respuesta a la entrada de datos se mostrará a continuación dentro de la propia simulación. En caso de introducción de un comando que no corresponda, la aplicación comprobará si el comando es correcto y devolverá información acorde a dicho comando o bien, si no es un comando correcto devolverá un mensaje de error predeterminado.

La ayuda sólo mostrará la información de los comandos de Linux a los que se tiene acceso.

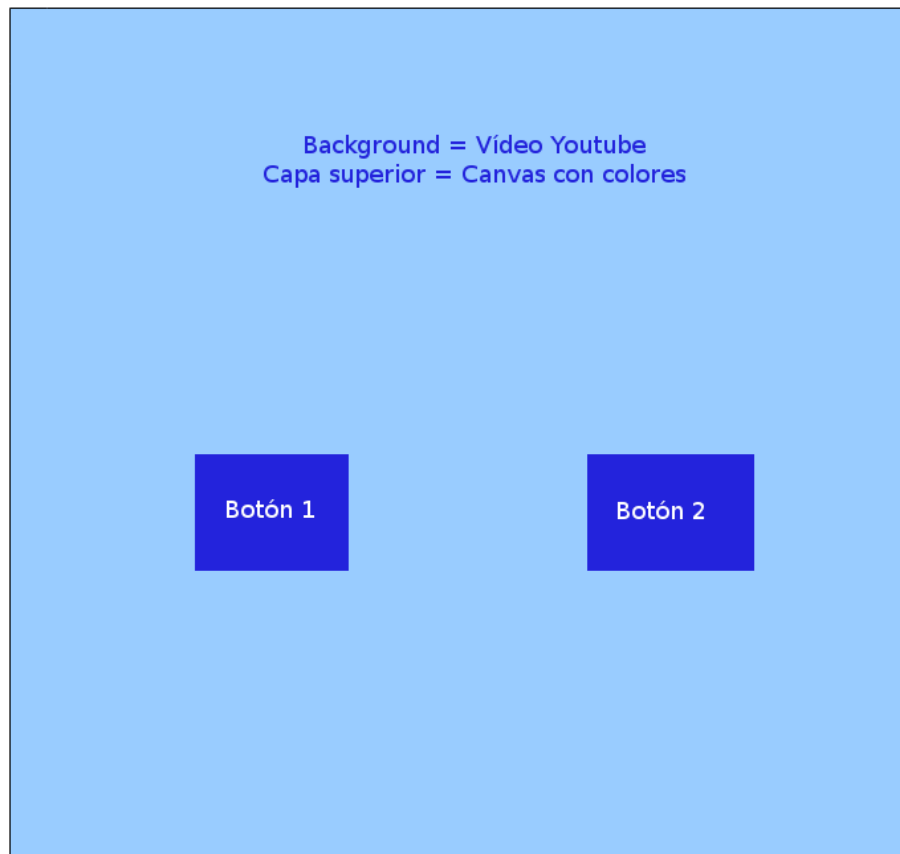
La aplicación se compone de 2 páginas estáticas y 16 fragmentos dinámicos.

### 2.2.1 Página de introducción

Es la pantalla por defecto de la aplicación y mostrará un vídeo de fondo que se reproducirá automáticamente al entrar en la aplicación y dos botones los cuales redirigirán al usuario o a la página de Youtube para poder ver el vídeo que se encuentra de fondo o bien al comienzo del tutorial de Aprende a Usar Git.



Este es el aspecto de la página:



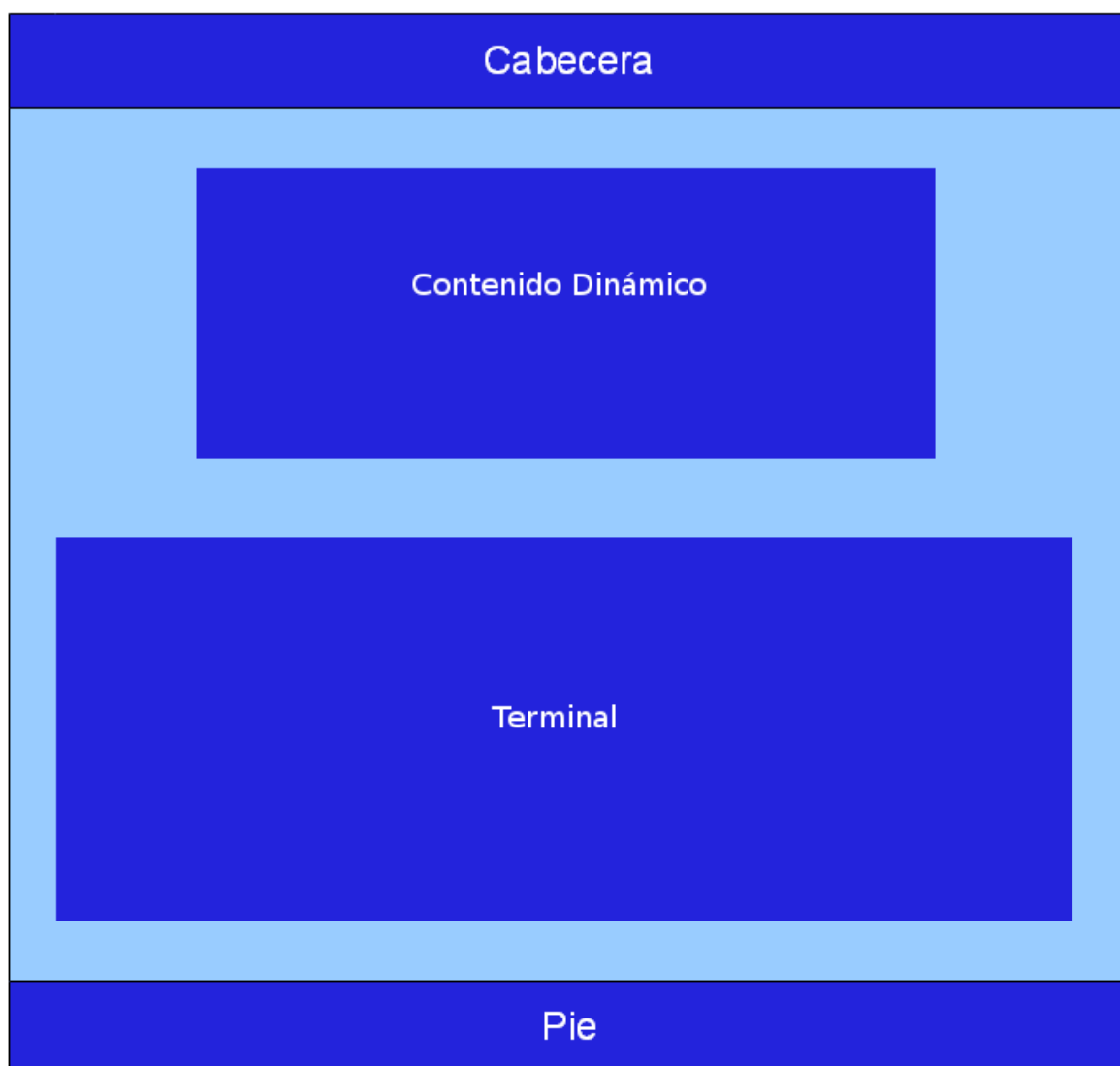
Consta de:

- Fondo con un vídeo de Youtube
- Canvas con colores
- 2 botones

### 2.2.2 Página del tutorial

Se accede a través de la página de introducción de la aplicación. En ella encontraremos los ya mencionados 4 bloques.

El aspecto de la página del tutorial sería:



## 2.3 Especificación del plan de pruebas

En general se realiza un plan de pruebas que permita la detección temprana de errores. Empezando por cada componente individual y progresivamente ir probando la integración con otros componentes.

### 2.3.1 Pruebas unitarias

Se realizan pruebas de los componentes individuales a nivel de función dentro de cada uno.

- Para cada función se comprobará que con unos parámetros de entrada correctos devuelve el resultado esperado. Se deben probar parámetros válidos, del tipo y valor que espera la función, y parámetros inválidos de tipo o valor incorrectos.
- Para probar las páginas Web, se comprobará que muestra correctamente los contenidos para cada posible opción que el usuario pueda introducir.
- Aquí también debemos comprobar que no siendo el comando que se especifica en las instrucciones, la salida por la terminal es correcta.

### 2.3.2 Pruebas de integración

Se deben probar cómo interactúan todos los componentes entre sí. Se partirá de los casos válidos, haciendo caso a las instrucciones del tutorial y se continuará probando cada posible comando para cada página, verificando que sólo se continúa con el tutorial cuando se introduce el comando correcto.

### 2.3.3 Pruebas de implantación

Esta prueba tendrá lugar una vez que las pruebas anteriores hayan concluido con éxito. Se realizará una prueba general para verificar que la interfaz de usuario se mantiene, independientemente del navegador y/o dispositivo.

## 3 Diseño de la aplicación

Comenzamos con el diseño de la aplicación, aquí vamos a definir “como” está estructurado el código de nuestra aplicación y “como” se comunican unos componentes con otros.

### 3.1 Identificación de ficheros y directorios

Nuestra aplicación cuenta con un total de 45 ficheros organizados en 5 directorios.

Cada directorio cumple con una función y recoge los ficheros de la misma temática, con ello se logra que la aplicación esté debidamente estructurada y sea de fácil acceso.

El primer directorio que nos encontraremos, llamado *CSS*, contiene 10 hojas de estilos. Ocho de las cuales corresponden a las incluidas por Bootstrap; las dos restantes son hojas de estilos creadas desde la necesidad de definir algunos parámetros de la interfaz de forma personalizada (una para la página de introducción y la otra para la página del tutorial).

El segundo directorio, llamado *Fonts*, contiene 5 ficheros con las fuentes e iconos importados por Bootstrap.

El tercer directorio, llamado *Images*, contiene tanto el favicon de la página como la imagen que se emplea en el pie de página del tutorial.

El cuarto directorio, llamado *Js*, contiene 5 ficheros javascript. Tres de los cuales son importados por Bootstrap; uno es el config de AngularJS (*script.js*); otro es el controlador de la simulación del terminal que se encuentra en el tutorial (*terminal.js*).

El quinto y último directorio, llamada *Partials*, contiene los 16 ficheros *.html* que componen el contenido del bloque dinámico del tutorial. En cada fichero se encuentra tanto la información y/o explicación del comando, como el propio comando correspondiente a ese paso del tutorial.

Por último, podemos encontrar 4 ficheros más:

- La licencia del proyecto.
- Un documento con información relevante sobre el proyecto.
- La página de introducción.
- La página del tutorial.

## 3.2 Necesidades de los equipos

Dado que nos encontramos ante una aplicación Web, el único requisito es tener una conexión a internet estable y un navegador.

Puesto que la aplicación aún no se ha lanzado, en el caso de realizar las pruebas de manera local, además de los requisitos anteriores, será necesario tener un servidor local (por ejemplo, Xampp/Lampp).

Por último, dado que el código está alojado en un servidor remoto de GitHub, o bien se descarga el código desde el navegador, o bien se usa un programa que gestione el control de versiones (como pueden ser TortoiseGit para Windows o RabbitVCS para Linux) o bien se descarga vía comandos gracias a Git para el terminal.

### 3.3 Necesidades del software

Dado que AngularJS requiere de AJAX y éste a su vez de Internet, el único requisito es el mencionado.

## 4 Desarrollo

Esta es la fase en la que se codifican los diferentes componentes de la aplicación. En la fase de diseño ya se ha definido lo que hace cada página dentro de nuestra aplicación y cómo lo hace. Aquí crearemos las funciones para llevar a cabo cada una de las operaciones que deben realizar los diferentes componentes de la aplicación.

Todos los módulos o componentes son de nueva creación, la única excepción son las funciones generales de la simulación de la terminal que se usará una ya creada y con licencia MIT a la que se le añadirán algunos cambios para adaptarlas a las necesidades de la aplicación.

La técnica de desarrollo será la de refinamientos sucesivos. Crear un prototipo lo antes posible e ir añadiendo funcionalidades a la aplicación de forma secuencial. Todo ello, gestionado desde el control de versiones que nos facilitará en gran medida la aceptación o rechazo de diferentes funcionalidades de una forma rápida y sencilla.

Cada fichero de código fuente debe comenzar por un comentario que constará de:

- Proyecto
- Nombre del fichero
- Autor
- Licencia, que en todos debe ser MIT
- Explicación del cometido del fichero

Cada función debe comenzar con un comentario que constará de:

- Explicación de lo que hace la función
- Parámetros de entrada
- Valores devueltos, en caso de existir

Cada función debe finalizar con otro comentario que indicará la finalización del mismo.

Todo el código estará justificado correctamente.

A continuación se describe cada una de las tareas a realizar en detalle:

**1. Preparación del entorno de desarrollo.**

1. Instalación y configuración del sistema operativo.
2. Instalación del servidor Apache.
3. Instalación del control de versiones.

**2. Desarrollo de la página del tutorial.**

1. Inclusión de AngularJS.
2. Conexión con los ficheros .html mediante AngularJS.
3. Incrustación del simulador del terminal.
4. Conexión con el controlador del simulador de la terminal.

**3. Desarrollo de AngularJS (script.js).**

1. Configuración del Route.

**4. Desarrollo del script que controla la simulación de la terminal (terminal.js).**

1. Desarrollo de funciones genéricas de un terminal de Linux.
2. Desarrollo de funciones específicas de Git para terminal.

**5. Desarrollo del contenido del tutorial.**

**6. Desarrollo de la página de introducción.**

1. Incrustación del vídeo de Youtube como fondo.
2. Desarrollo del canvas.
3. Inclusión de Bootstrap.

**7. Creación de estilos.**

1. Página de introducción.
2. Página del tutorial.

**8. Realización de pruebas de integración.**

1. En diferentes navegadores.
2. En diferentes dispositivos.

## 5 Referencias

Licencia MIT ([https://github.com/bl0z/Aprende\\_git\\_app/blob/master/LICENSE](https://github.com/bl0z/Aprende_git_app/blob/master/LICENSE))

Bootstrap (<http://getbootstrap.com/>)

AngularJS (<https://angularjs.org/>)

AJAX: Asynchronous JavaScript And XML (<http://www.w3schools.com/Ajax>)

Javascript (<http://www.w3schools.com/js/>)

Apache (<http://www.apache.org/>)

Xampp/Lampp (<https://www.apachefriends.org/>)

HTML: HyperText Markup Language (<https://www.w3.org/html/>)

CSS: Cascading Style Sheets (<https://www.w3.org/Style/CSS/>)

Youtube (<https://www.youtube.com/>)

Git (<https://git-scm.com/>)

GitHub (<https://github.com/>)

TortoiseGit (<https://tortoisegit.org/>)

RabbitVCS (<http://rabbitvcs.org/>)