



UNIVERSIDAD SIMÓN BOLÍVAR



Manual de Instalación

Juego de memoria del Archivo Nacional de Arte Rupestre

Versión del documento: 1.0

Última actualización: 22 de julio de 2016



UNIVERSIDAD SIMÓN BOLÍVAR



Índice de contenido

Introducción.....	3
Requerimientos.....	4
Instalación.....	5
Instalando el juego en el ambiente definitivo.....	5
Descarga e instala Git y Node.js.....	5
Descarga el código fuente.....	5
Instala las dependencias del servidor para multijugador.....	5
Descarga, instala y configura CouchDB.....	5
Instala y configura tu servidor web.....	6
Habilita la comunicación para las partidas multijugador.....	7
Instalando el juego en un ambiente de desarrollo.....	8
Instala el manejador de dependencias de front-end.....	8
Cambia a una base de datos local (opcional).....	8



UNIVERSIDAD SIMÓN BOLÍVAR



Introducción

Este documento explica, paso por paso, como instalar el juego de memoria, bien sea en un ambiente definitivo o de desarrollo. Su intención es ayudar a los nuevos desarrolladores para que puedan tener un ambiente de trabajo útil lo más rápido posible.

Aunque el juego puede ser instalado en varios sistemas de operación, los desarrolladores actuales solo hemos usado sistemas Linux. Se intentó que las instrucciones fueran aplicables a todas las plataformas, pero si quieres agregar un dato o una corrección sobre tu sistema de operación de preferencia, eres bienvenido a incluirlo en una nueva versión de este documento.

Aunque la explicación es lo más detallada posible, para entender este documento es mejor estar familiarizado con la línea de comandos de tu sistema de operación.



UNIVERSIDAD SIMÓN BOLÍVAR



Requerimientos

El juego está desarrollado usando tecnologías multiplataforma, por lo que puede instalarse en múltiples sistemas de operación, incluyendo Mac OS X, Windows y Linux. Necesitarás lo siguiente:

- Node.js (<https://nodejs.org/en/download/>): ambiente de ejecución para Javascript que se usa en el servidor.
- Git (<https://git-scm.com/downloads>): sistema de control de versiones con el que se está desarrollando el proyecto.
- CouchDB Versión 1 (<http://couchdb.apache.org/>): base de datos para documentos JSON que se utiliza en la aplicación.
- El servidor web de tu preferencia, que pueda servir contenido estático. Por ejemplo Apache (<https://httpd.apache.org/>) o Nginx (<https://nginx.org/>).

Si puedes instalar eso en tu computador, podrás ejecutar el juego de memoria. El código está alojado de forma gratuita en GitHub (<https://github.com/>), así que también necesitarás acceso a internet.



UNIVERSIDAD SIMÓN BOLÍVAR



Instalación

Instalando el juego en el ambiente definitivo

Si tu sistema cumple los requerimientos, puedes instalar la aplicación siguiendo estos pasos.

Descarga e instala Git y Node.js

En sus respectivos sitios web encontraras los instaladores disponibles para la descarga, y los instrucciones.

Descarga el código fuente

Lo primero es descargar el código desde el repositorio de GitHub donde se encuentra. Para ello, desde la línea de comandos ejecuta:

```
$ git clone https://github.com/SC-ANAR/ANAR\_MEMORIA
```

Eso creará una carpeta ANAR_MEMORIA con el contenido del juego. La versión más actualizada (aunque no necesariamente estable) del juego estará en la rama develop, así que entra en la carpeta y cambia de rama:

```
$ cd ANAR_MEMORIA  
$ git checkout develop
```

Instala las dependencias del servidor para multijugador

Deberás entrar a la subcarpeta anar_gameserver, e instalar las dependencias especificadas en el archivo package.json:

```
$ cd anar_gameserver  
$ npm install
```

Esto instalará el framework Loopback (<https://loopback.io/>), que se uso para generar el API con que se comunican el cliente y el servidor, y la librería Socket.io (<http://socket.io>), que se usa para la comunicación en tiempo real.

Descarga, instala y configura CouchDB

Conseguirás instaladores para la base de datos en su página web (<http://couchdb.apache.org/>). Si



UNIVERSIDAD SIMÓN BOLÍVAR



tu sistema de preferencia no está listado, deberás descargar el código fuente y compilar la base de datos. En particular, si usas Debian Jessie, te recomendamos estas instrucciones:

<https://forum.cozy.io/t/how-to-install-couchdb-manually-on-debian-jessie/1230>

Una vez que hayas instalado CouchDB, deberás asegurar la instalación (<http://guide.couchdb.org/draft/security.html>) y habilitar CORS, el mecanismo que permite compartir recursos desde distintos dominios (<http://docs.couchdb.org/en/1.3.0/cors.html>). Para la versión 1 de CouchDB, puedes habilitar CORS con estos comandos:

```
$ HOST=http://adminname:password@localhost:5984 # aquí tus datos de administrador
$ curl -X PUT $HOST/_config/httpd/enable_cors -d '"true"'
$ curl -X PUT $HOST/_config/cors/origins -d '"*"'
$ curl -X PUT $HOST/_config/cors/credentials -d '"true"'
$ curl -X PUT $HOST/_config/cors/methods -d '"GET, PUT, POST, HEAD, DELETE"'
$ curl -X PUT $HOST/_config/cors/headers -d '"accept, authorization, content-type, origin, referer, x-csrf-token"'
```

Si usas un sistema Windows, probablemente debas instalar curl (<https://curl.haxx.se/>) para poder ejecutar estos comandos, ya que no está instalado por defecto.

Para crear la base de datos ejecuta el siguiente comando con los datos apropiados:

```
$ curl -X PUT $HOST/tu_base_de_datos
```

Para facilitar el desarrollo, la aplicación estará configurada para comunicarse con una instancia de CouchDB instalada en Cloud9 (<https://c9.io/>) o para guardar los datos de forma local en el navegador. Una vez que vayas a implantarla en el ambiente definitivo, deberás hacer que la variable `db_user` del archivo `database.js` apunte a la base de datos adecuada, con una línea como la siguiente al comienzo del archivo:

```
var db_user = new PouchDB('http://host_definitivo:puerto/tu_base_de_datos', {ajax: {timeout: 10000}});
```

Instala y configura tu servidor web

Una vez que hayas instalado un servidor web de tu preferencia, deberás configurarlo para que sirva el archivo `index.html` que está en la carpeta `app`. Dependiendo del servidor que escogiste, la configuración será distinta. Por ejemplo si usas Nginx en Linux, tu archivo de configuración podría



UNIVERSIDAD SIMÓN BOLÍVAR



tener algo así:

```
server {
    listen      0.0.0.0;
    server_name test;

    location / {
        root    /home/user/ANAR_MEMORIA/app;
        index   index.html;
    }

    location = /50x.html {
        root    /usr/share/nginx/html;
    }
}
```

Para otros servidores, deberás consultar la documentación. Es importante que **no sirvas la aplicación por el mismo puerto por el que sirves CouchDB** si ambos servidores estarán en la misma máquina. De igual manera, **si el servidor de comunicación para las partidas multijugadores está corriendo en la misma máquina, no sirvas la aplicación por el puerto 3000.**

Habilita la comunicación para las partidas multijugador

Si ya llegaste hasta este punto, podrás acceder al juego usando un navegador web y el nombre de dominio o dirección IP de la máquina donde instalaste tu servidor web (0.0.0.0 si es tu máquina local). Dependiendo de como hayas configurado tu servidor, podrías necesitar un número de puerto, con un URL final como `http://<tu_ip_o_nombre>:puerto/`. Si CouchDB está bien configurado, ya podrás crear nuevos usuarios e ingresar al juego, pero solo funcionará el modo de un solo jugador. El servidor que comunica a los jugadores en las partidas multijugador se ejecuta sobre Node.js. Para activarlo, ejecuta `node` en el directorio `anar_gameserver`:

```
$ cd anar_gameserver
$ node .
```

Este servidor aceptará peticiones por el puerto 3000, y podrás consultar el API entrando a un URL parecido a `http://<host_donde_se_ejecuta>:3000/explorer`.

Con esto el juego debería quedar instalado.



UNIVERSIDAD SIMÓN BOLÍVAR



Instalando el juego en un ambiente de desarrollo

Con la excepción de la sección sobre la base de datos, **deberás llevar a cabo todos los pasos de la sección anterior para poder instalar el juego**. Además, podrías querer agregar los que se listan a continuación.

Instala el manejador de dependencias de front-end

En el repositorio están incluidas todas las dependencias que necesita descargar la aplicación cliente. Sin embargo, si quieres modificarlas, te podría ser útil el manejador de paquetes Bower. Si ya instalaste Node.js y Git, basta con ejecutar:

```
$ npm install -g bower
```

Ten en cuenta que podrías necesitar permisos de administrador. Para más información sobre Bower, consulta su página web: <https://bower.io/>

Cambia a una base de datos local (opcional)

No deberías utilizar la misma base de datos definitiva para el desarrollo. Si no consigues donde alojar una base de datos para el desarrollo (te recomendamos intentar con Cloud9, <https://c9.io/>), puedes hacer que los datos se guarden directamente en el navegador, cambiando la variable `anar_user` en el archivo `database.js`. Basta con poner algo similar a lo siguiente:

```
var db_user = new PouchDB('anar_user');
```

Los datos se guardarán ahora usando PouchDB (<https://pouchdb.com/>), una base de datos que se ejecuta en el cliente. Serán persistentes solo en la misma máquina y con el mismo navegador web, y se borrarán si limpias el caché del navegador.