Manual de Instalación

----------------------------------------------------- Sistema CENADE



Tabla de contenido

[1. Windows 3](#_Toc458044867)

[1.1. PostgreSQL 3](#_Toc458044868)

[1.2. NodeJS 3](#_Toc458044869)

[1.3. Ruby 4](#_Toc458044870)

[1.4. DevKit 4](#_Toc458044871)

[1.5. wkhtmltopdf 5](#_Toc458044872)

[1.6. Despliegue de la aplicación 5](#_Toc458044873)

[1.7. NGINX 6](#_Toc458044874)

[1.8. Apache 7](#_Toc458044875)

[1.9. Script de Arranque 8](#_Toc458044876)

[1.10. Ejecución de la Aplicación 9](#_Toc458044877)

# Windows

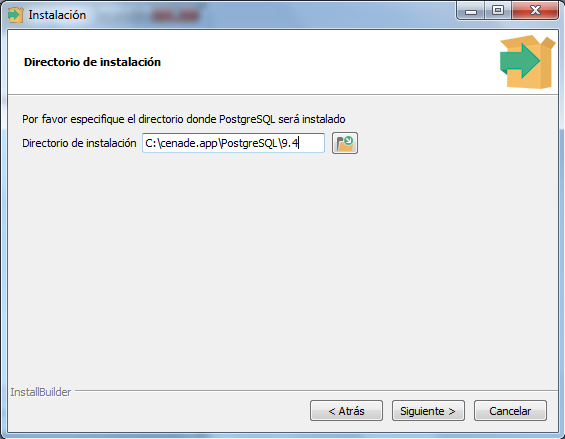
Se describe la instalación de los componentes necesarios para la ejecución del Sistema CENADE dentro del sistema operativo Windows 7.

Para establecer una convención, dentro del directorio C:\cenade.app serán instalados todos los componentes del sistema.

# PostgreSQL

Se describe la instalación del motor de base de datos PostgreSQL 9.4.8.

1. Descargar el instalador de <https://www.postgresql.org/download/> . Se debe descargar la versión de 32 bits o 64 bits, acorde al sistema operativo destino. Una versión de 32 bits será funcional tanto para un sistema de 32 y 64 bits.
2. Ejecutar el archivo descargado.

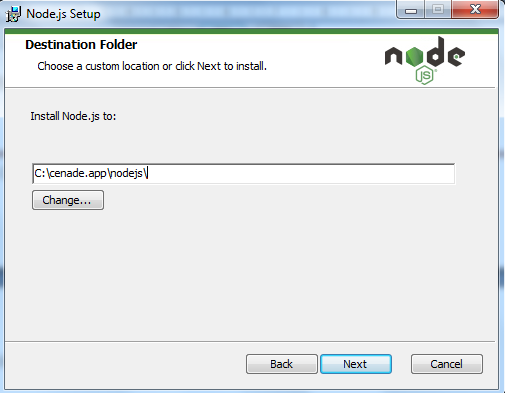
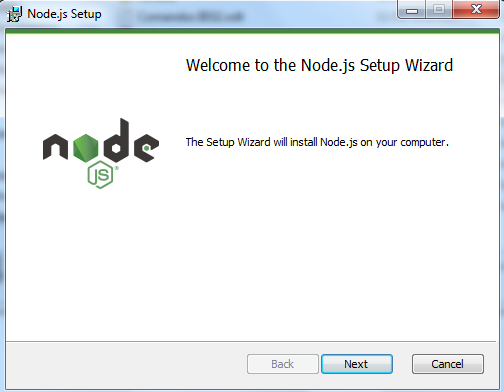


1. Especificar C:\cenade.app\PostgreSQL\9.4 como el directorio de instalación.
2. En la siguiente pantalla especificar C:\cenade.app\PostgreSQL\9.4\data como el directorio de almacenamiento de datos.
3. En la siguiente pantalla especificar la contraseña para el superusuario postgres. La contraseña utilizada será cenadeappdb.
4. En la siguiente pantalla especificar 5432 como puerto de escucha del servidor Postgres.
5. En la siguiente pantalla dejar la configuración por defecto.
6. Proceder con la instalación.
7. Ejecutar el programa PGAdmin III y conectarse al motor de base de datos, para comprobar que esta funcionando.

# NodeJS

Se describe la instalación del entorno de ejecución para Javascript NodeJS 4.4.7. NodeJS es una instalación previa, necesaria para la aplicación Rails.

1. Descargar el instalador de <https://nodejs.org/en/download/> . Se debe descargar la versión de 32 bits o 64 bits, acorde al sistema operativo destino. Una versión de 32 bits será funcional tanto para un sistema de 32 y 64 bits.
2. Ejecutar el archivo descargado.

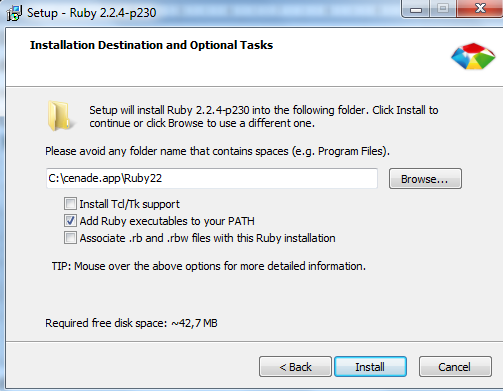
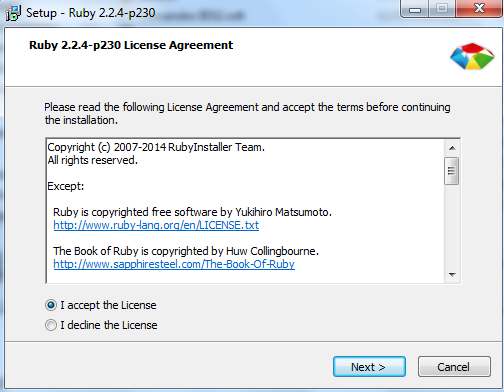


1. En la siguiente pantalla, aceptar las condiciones de la licencia.
2. En la siguiente pantalla, especificar C:\cenade.app\nodejs\ como el directorio de instalación.
3. En la siguiente pantalla, dejar la configuración por defecto.
4. Proceder con la instalación.

# Ruby

Se describe la instalación del lenguaje Ruby 2.2.4.

1. Descargar el instalador de <http://rubyinstaller.org/downloads/> . Se debe descargar la versión de 32 bits o 64 bits, acorde al sistema operativo destino. Una versión de 32 bits será funcional tanto para un sistema de 32 y 64 bits.
2. Ejecutar el archivo descargado.

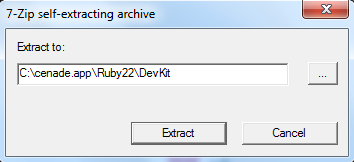


1. Aceptar las condiciones de la licencia.
2. En la siguiente pantalla, especificar C:\cenade.app\Ruby22 como directorio de instalación y Habilitar el agregado del ejecutable de Ruby al PATH del sistema operativo.
3. Proceder con la instalación.

# DevKit

Se describe la instalación del Kit de desarrollo DevKit. DevKit es un requerimiento para poder instalar más adelante algunas gemas de Ruby, por ejemplo, la gema json.

1. Descargar el instalador de <http://rubyinstaller.org/downloads/> . Se debe descargar la versión de 32 bits o 64 bits, acorde al sistema operativo destino. Una versión de 32 bits será funcional tanto para un sistema de 32 y 64 bits.
2. Ejecutar el archivo descargado.



1. Especificar C:\cenade.app\Ruby22\DevKit como directorio de extracción de los archivos. Es ideal extraer los archivos dentro del directorio donde está instalado Ruby.
2. Proceder con la extracción.
3. Abrir la línea de comandos y ubicarse en la carpeta DevKit. Se puede usar el siguiente comando:

> cd C:\cenade.app\Ruby22\DevKit

1. Ejecutar los siguientes comandos:

> ruby dk.rb init

> ruby dk.rb review

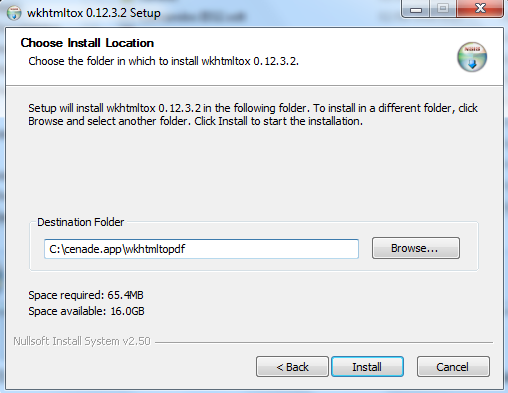
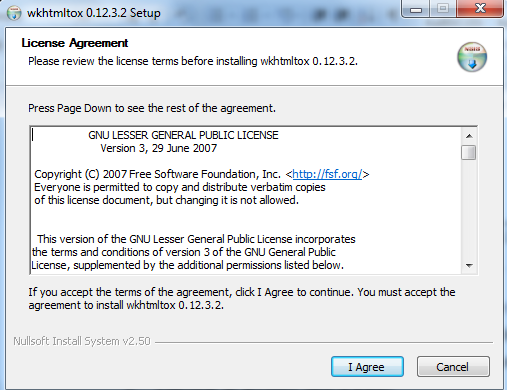
> ruby dk.rb install

1. Cerrar la terminal.

# wkhtmltopdf

Se describe los pasos, para instalar wkhtmltopdf 0.12.3.2. Wkhtmltopdf es una herramienta de línea de comandos que permite convertir html a pdf. Esta herramienta es utilizada por la gema WickedPDF, para poder imprimir listados y registros en formato pdf.

1. Descargar el instalador para wkhtmltopdf de <http://wkhtmltopdf.org/downloads.html> . Se debe descargar la versión de 32 bits o 64 bits, acorde al sistema operativo destino. Una versión de 32 bits será funcional tanto para un sistema de 32 y 64 bits.
2. Ejecutar el archivo descargado.



1. Especificar C:\cenade.app\wkhtmltopdf como directorio de instalación.
2. Proceder con la instalación.

# Despliegue de la aplicación

Se describe los pasos, para desplegar la aplicación.

1. Copiar todo el código de la aplicación en C:\cenade.app\rails-app
2. Abrir la línea de comandos y ubicarse en la carpeta rails-app. Se puede usar el siguiente comando:

> cd C:\cenade.app\rails-app

1. Instalar la gema bundler (Nota: Si se copia y se pega en la línea de comandos, es posible que no instale la versión especificada).

> gem install bundler -v 1.11.2

1. Instalar las gemas de la aplicación Rails (Nota: Instala las gemas especificadas en el Gemfile, incluyendo la gema Rails que también está especificada, los grupos test, development y production se ignoran, por no ser necesarias para un entorno de ejecución en producción local).

> bundle install --without test development production

Un camino alternativo es usar las gemas empaquetadas con

> bundle package

Con esto se podrá evitar la conexión a internet para bajar las gemas

> bundle install –local

Este camino alternativo no funciona si bundle package se ejecutó en un entorno Linux y la aplicación se desea desplegar en un entorno Windows.

1. Agregar al archivo C:\cenade.app\rails-app\config\initializers\wicked\_pdf.rb la siguiente línea:

:exe\_path =>'C:/cenade.app/wkhtmltopdf/bin/wkhtmltopdf.exe'

1. Establecer el usuario (postgres) y contraseña (cenadeappdb) en el archivo C:\cenade.app\rails-app\config\database.yml
2. Crear la base de datos, correr las migraciones y el seed. Todo para el entorno de producción.

> rake db:create RAILS\_ENV=production

> rake db:migrate RAILS\_ENV=production

> rake db:seed RAILS\_ENV=production

1. Ejecutar la aplicación

> rails s -e production

**Notas:**

Con Ngnix se debe usar

> rails s -e production

Con Apache se debe usar

> bundle exec puma -C config/puma.rb -e production

# NGINX

Se describe los pasos para instalar el servidor http y proxy Nginx 1.10.1. Nginx se utiliza como proxy reverso, para dar acceso a la aplicación.

1. Descargar el instalador de <https://nginx.org/en/download.html> .
2. Extraer el archivo descargado dentro del directorio C:\cenade.app\nginx-1.10.1
3. Modificar el archivo ngnix.conf que está dentro de la carpeta conf, agregando la siguiente línea:

include C:/cenade.app/nginx-1.10.1/conf/sites-enabled/\*.conf;

1. Dentro de la carpeta conf de NGINX crear las carpetas sites-available y sites-enabled.
2. Dentro de la carpeta sites-available crear el archivo cenade.app.conf con el siguiente contenido:

server {

listen 80;

server\_name cenade.app;

access\_log C:/cenade.app/nginx-1.10.1/logs/cenade.app.access.log;

error\_log C:/cenade.app/nginx-1.10.1/logs/cenade.app.error.log;

location / {

proxy\_pass http://[::1]:3000/;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header Host $http\_host;

}

}

**Notas**:

En vez de puerto 80, se puede usar otro puerto xx en cuyo caso, para acceder a la aplicación se usa cenade.app:xx

En el archivo hosts del sistema operativo, se debe agregar la línea:

127.0.0.1 cenade.app

Para linux, el proxy\_pass <http://localhost:3000/>; funciona, pero en Windows, hace que la carga de la página tarde aproximadamente 60 s, en la primera petición, y, casi nada en la segunda.

1. Dentro de la carpeta sites-enabled crear un acceso directo al archivo cenade.app.conf recién creado. En la línea de comandos, ejecutar:

> mklink /H C:\cenade.app\nginx-1.10.1\conf\sites-enabled\cenade.app.conf C:\cenade.app\nginx-1.10.1\conf\sites-available\cenade.app.conf

**Nota:**

El comando anterior crea un link simbólico para cenade.app.conf de la carpeta sites-available. En otra máquina con Windows 7 funcionó crear un acceso directo, pero en otra, también Windows 7, se tuvo que usar el comando, caso contrario, nginx no tomaba la configuración de la carpeta sites-enabled

1. Establecer una dirección DNS para la aplicación, en la máquina servidor

Ingresar a C:\Windows\System32\drivers\etc y agregar al archivo hosts, la siguiente línea:

localhost cenade.app

**Notas:**

En vez de cenade.app se puede usar otro nombre de host, pero el nombre debe coincidir con lo especificado en cenade.app.conf (Configuración de ngnix). Una vez hecho esto se accede a la aplicación con la URL: [http://cenade.app](http://cenade.app/)

Es posible que deba abrir el archivo hosts en un editor de texto como Administrador, para editar el archivo hosts, por cuestiones de permisos.

1. Establecer una dirección DNS para la aplicación, en la máquina cliente

Ingresar al directorio C:\Windows\System32\drivers\etc y agregar al archivo hosts, la siguiente línea:

192.168.0.1 cenade.app

**Notas:**

Este mapeo corresponde a una aplicación que se está ejecutando en la dirección IP especificada en el puerto 80 por defecto.

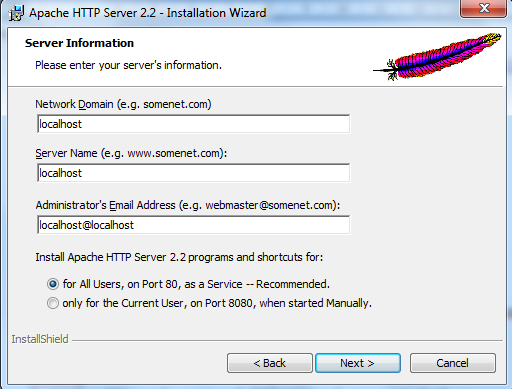
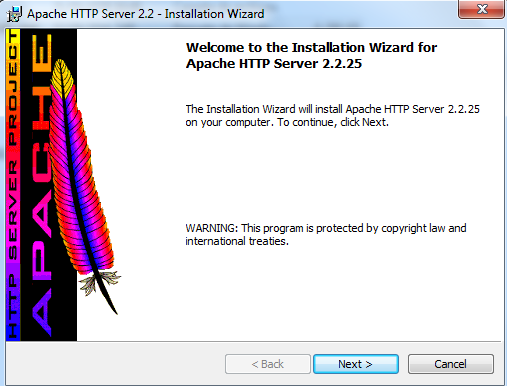
La IP especificada debe corresponder a la de la máquina servidor. cenade.app es el nombre que se la da al host (puede usarse otro nombre). Una vez hecho esto se accede a la aplicación con la URL: [http://cenade.app](http://cenade.app/)

Es posible que deba abrir el archivo hosts en un editor de texto como Administrador, para editar el archivo hosts, por cuestiones de permisos.

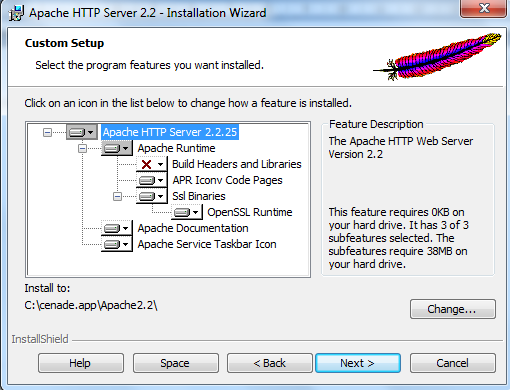
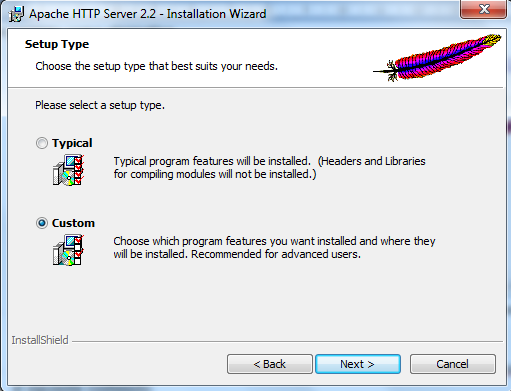
# Apache

Se describe los pasos para instalar el servidor Apache 2.2.25. Se utiliza Apache como proxy reverso, para dar acceso a la aplicación. Apache tiene un mejor rendimiento que NGNIX, por lo que se presenta como alternativa.

1. Descargar el instalador de <http://archive.apache.org/dist/httpd/binaries/win32/> .
2. Ejecutar el archivo descargado



1. Aceptar los términos de la licencia y especificar localhost como nombre de dominio y servidor, y, localhost@localhost como dirección de correo.



1. Seleccionar la opción de instalación personalizada.
2. Establecer C:\cenade.app\Apache2.2\ como directorio de instalación.
3. Proceder con la instalación.
4. Dentro de C:\cenade.app\Apache2.2\conf\extra\ crear el archivo cenade.app.conf con el siguiente contenido:

<VirtualHost \*:80>

ErrorLog "C:\cenade.app\Apache24\logs\cenade.app.error.log"

CustomLog "C:\cenade.app\Apache24\logs\cenade.app.access.log" common

ServerName cenade.app

ProxyRequests Off

ProxyPreserveHost On

ProxyPass / http://localhost:3000/

ProxyPassReverse / http://localhost:3000/

</VirtualHost>{

1. Descomentar las líneas correspondientes a los módulos proxy y proxy\_http del archivo C:\cenade.app\Apache24\conf\httpd.conf

LoadModule proxy\_module modules/mod\_proxy.so

LoadModule proxy\_http\_module modules/mod\_proxy\_http.so

1. Agregar la siguiente línea en la sección para includes de httpd.conf

Include conf/extra/cenade.app.conf

1. Abrir la línea de comandos y ubicarse en la carpeta Apache24. Se puede usar el siguiente comando:

> cd C:\cenade.app\Apache2.2

1. Iniciar apache

> httpd.exe

1. Iniciar apache como servicio

> httpd.exe -k install

Notas:

No se usó una versión superior a 2.2.25, porque apache ya no provee de ejecutables, además que para las nuevas versiones, se requiere que Windows esté actualizado y se debe descargar la siguiente actualización específica para la ejecución de apache:

<https://www.microsoft.com/en-us/download/details.aspx?id=52982>

# Script de Arranque

Para facilitar la ejecución del sistema CENADE, se creó un script que se ejecuta al arrancar el sistema operativo. El nombre del script es start\_background\_cenade\_app.bat

**Para establecer los scripts de la aplicación**

**Opción 1:**

Se puede copiar el archivo .bat en la carpeta Inicio > Todos los programas > Inicio, esto iniciará el script al iniciar el sistema operativo y mostrará la línea de comandos con los pasos de la ejecución de la aplicación

**Opción 2:**

Ejecutar

> gpedit.msc

Se abrirá una ventana, en la sección configuración de Windows, se podrá especificar scripts de inicio y apagado. No se muestra el proceso en una terminal. Se comprobó que funciona para iniciar Nginx, pero no, para iniciar la aplicación Rails.

**Detalles del script de arranque:**

**start\_cenade\_app.bat:** Inicia Nginx si se habilita y también ejecuta rails s en primer plano, permitiendo cancelarlo con CTRL + c.

**start\_background\_cmd\_script.vbs:** Ejecuta el script start\_cenade\_app.bat en segundo plano, por lo que la ejecución de Rails no podrá ser cancelada.

**start\_background\_cenade\_app.bat:** Se encarga de ejecutar el script start\_background\_cmd\_script.vbs. Muestra una ventana cmd que dura 60 segundos, antes de cerrarse, con el objetivo de informar que el sistema se ejecutó.

# Ejecución de la Aplicación

Se describe los comandos que se pueden usar para ejecutar la aplicación. El tema de los puertos, es un tanto complicado, porque se tienen varias alternativas.

> rails s -e production

> rails s -e production -b localhost

1. Escucha en localhost:3000

2. Con ngnix funciona si proxy\_pass http://[::1]:3000/;

3. Con Apache no funciona: No se puede establecer una conexión ya que el equipo de destino denegó expresamente dicha conexión. : proxy: HTTP: attempt to connect to 127.0.0.1:3000 (localhost) failed

> rails s -e production -b 127.0.0.1

1. Escucha en 127.0.0.1:3000

2. Con ngnix funciona muy lento si proxy\_pass <http://localhost:3000/>; (Necesita una doble petición para reaccionar más rápido)

3. Con ngnix funciona muy bien si proxy\_pass <http://127.0.0.1:3000/>;

4. Con ngnix no funciona si proxy\_pass <http://[::1]:3000/>;

5. Con Apache funciona si ProxyPass / <http://localhost:3000/>

> bundle exec Puma -e production

> bundle exec puma -C config/puma.rb –e production

> rails s -e production –b 0.0.0.0

1. Escucha en 0.0.0.0:3000

2. Con ngnix funciona muy lento si proxy\_pass <http://localhost:3000/>; (Necesita una doble petición para reaccionar más rápido)

3. Con ngnix funciona bien si proxy\_pass <http://127.0.0.1:3000/>;

4. Con ngnix no funciona si proxy\_pass <http://[::1]:3000/>;

3. Con Apache funciona si ProxyPass / <http://localhost:3000/>

4. Desde la red, se puede acceder especificando la IP de la máquina servidor y el puerto correspondiente, sin necesidad de usar un proxy. Se puede usar el puerto 80, si no se usa un proxy.

**Notas:**

En fin, el que mejores resultados da con los proxies es:

> rails s -e production -b 127.0.0.1

Pero si no se usa proxies, y se desea acceder directamente a puma, se usa:

> rails s -e production –b 0.0.0.0

O cualquiera del último grupo.