

INETInfoSystem Manual Installation

Server requirements

Nodejs

```
root@billing-dataconnect#curl -o-  
https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash  
root@billing-dataconnect#export NVM_DIR="$([ -z "${XDG_CONFIG_HOME-}" ] && printf %s  
"${HOME}/.nvm" || printf %s "${XDG_CONFIG_HOME}/nvm")"  
root@billing-dataconnect#"$[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  
root@billing-dataconnect#nvm install v14.10.1 && nvm use v14.10.1
```

Laravel

For **ubuntu 20.04**, pkg php7.4 is available, if you are using another os, please check the php pkg available

```
root@billing-dataconnect#sudo apt update  
root@billing-dataconnect#sudo apt-add-repository ppa:ondrej/php  
root@billing-dataconnect#sudo apt-get install php7.4 && apt-get install php7.4-fpm  
root@billing-dataconnect#sudo service php7.4-fpm restart  
root@billing-dataconnect#sudo apt-get install libapache2-mod-php7.4  
root@billing-dataconnect#sudo a2dismod mpm_event; sudo a2enmod mpm_prefork ; sudo  
a2enmod php7.4
```

Clone repository in Gitlab

```
git clone https://gitlab.apollo.com.ph/dataconnect-ph/inet-infosystem.git
```

Go to inet-infosystem folder and navigate to front to build the frontend

```
root@billing-dataconnect#cd ~/inet-infosystem/front  
root@billing-dataconnect#mv package-lock.json package-old  
root@billing-dataconnect#nvm use v14.10.1  
root@billing-dataconnect#npm install && npm run build
```

Copy inet-infosystem in /var/www/html directory

```
root@billing-dataconnect#cp -r inet-infosystem/* /var/www/html
```

Change directory to /var/www/html. All contents of inet-infosystem is in /var/www/html

```
root@billing-dataconnect:/var/www/html# ls
README.md  back-old  dist-old  imgPath.txt  inetinfosystem.pdf  node_modules  package-old  settings.env  webpack.config.js
back      dist      front-old  index.html    maintenance1.gif    package-lock.json  package.json  src
```

Create settings.env

```
root@billing-dataconnect#touch /var/www/html/settings.env
root@billing-dataconnect#nano /var/www/html/settings.env
```

Paste these lines and save by ctrl + x

```
root=http://<ip_address_or_fqdn>/back/
img_path=http://<ip_address_or_fqdn>/back/public/
attachment_path=http://<ip_address_or_fqdn>/back/public/attachments/
keycloak_client=Development
keycloak_redirect=https://apiauth.dctechmicro.com:8443/auth/realms/DctechH%20APPS/protocol/openid-connect/auth?client_id=Infosystem&redirect_uri=https%3A%2F%2F
inetinfosystem.dctechmicro.com%2F&state=d3384902-3b8e-4fbe-9a45-7f4ab7270a45&response_mode=fragment&response_type=code&scope=openid&nonce=744ed245-1f24-464d-88d9a-0a4b77cd5294
```

Create .env for backend

```
root@billing-dataconnect#cd /var/www/html/back
root@billing-dataconnect#touch .env
root@billing-dataconnect#nano .env
```

Paste these lines and edit base on your configuration

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:xDKyz3mVwkazcbRsyyIjvI5DOITJ6+cjeci3rFcM8KE=
APP_DEBUG=true
APP_URL=http://<ip_address_or_fqdn>
LOG_CHANNEL=stack

DB_CONNECTION=mysql //mysql or postgresql
DB_HOST=127.0.0.1 //ip address of the db server
DB_PORT=3306 //default port of the mysql server
DB_DATABASE=c4_infosystem //database name
DB_USERNAME=laravel_user //username to access the db
DB_PASSWORD=<password> //password to access the db

BROADCAST_DRIVER=log
```

```
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_DRIVER=smtp
MAIL_HOST=smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null

AWS_ACCESS_KEY_ID=
AWS_SECRET_ACCESS_KEY=
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=

PUSHER_APP_ID=
PUSHER_APP_KEY=
PUSHER_APP_SECRET=
PUSHER_APP_CLUSTER=mt1

MIX_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
MIX_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"
```

Install mysql server

```
root@billing-dataconnect#sudo apt update
root@billing-dataconnect#sudo apt install mysql-server
root@billing-dataconnect#systemctl status mysql
```

Create database and user

```
root@billing-dataconnect#mysql -u root -p
mysql>create database c4_infosystem;
mysql>create user 'info_user'@'localhost' identified by '<password>'
mysql>grant all privileges on c4_infosystem.* to 'info_user'@'localhost';
mysql>flush privileges;
```

Install apache2

Make sure the apache2 service is running, upon checking the status, please note that the service is **Active: Running**.

```
root@billing-dataconnect#sudo apt update
root@billing-dataconnect#sudo apt install apache2
root@billing-dataconnect#systemctl status apache2
```

Edit /etc/apache2/apache2.conf

```
root@billing-dataconnect#nano /etc/apache2/apache2.conf
```

Add this in the last line:

```
SetHandler application/x-httpd-php
```

Ctrl + w and find this block. **Replace the highlighted blue into 'All'**

```
<Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride None
  Require all granted
</Directory>
```

Edit 000-default.conf and add this on the last line:

```
<VirtualHost *:443>
  ServerName pilot-billing.dataconnect.com.ph #ip address_or_fqdn
  DocumentRoot /var/www/html

  SSLEngine on
  SSLCertificateFile /letsencrypt/live/dataconnect.com.ph/cert.pem #path of the cert
  SSLCertificateKeyFile /letsencrypt/live/dataconnect.com.ph/privkey.pem #path of the privkey

  ErrorLog ${APACHE_LOG_DIR}/zabbix_error.log
  CustomLog ${APACHE_LOG_DIR}/zabbix_access.log combined
</VirtualHost>
```

Enable apache2 ssl and rewrite

```
root@billing-dataconnect#sudo a2enmod rewrite
root@billing-dataconnect#sudo a2enmod ssl
```

Before restarting the apache2 service, navigate to /var/www/html/back and run these commands

```
root@billing-dataconnect#php artisan config:cache
root@billing-dataconnect#php artisan clear:cache
root@billing-dataconnect#php artisan route:clear
root@billing-dataconnect#php artisan optimize:clear
root@billing-dataconnect#php artisan route:list
```

And if you run the `php artisan route:list`, you will see all the applications routes

Change owner and permission in /var/www/html

```
root@billing-dataconnect#sudo chown -R www-data.www-data /var/www/html/back/storage
root@billing-dataconnect#sudo chmod -R 777 /var/www/html/
```

Enable mysql pdo service

```
root@billing-dataconnect#vi /etc/php/7.4/apache2/php.ini
```

Comment out and save

```
;extension=pdo_mysql
```

Finally restart apache2 service

```
root@billing-dataconnect#systemctl restart apache2
```

Troubleshooting

If application key is missing, you can generate:

```
php artisan key:generate
```

Run `php artisan config:cache` for all your configuration changes

References:

<https://www.digitalocean.com/community/questions/why-do-my-laravel-routes-not-work>

<https://www.techrepublic.com/article/how-to-fix-apache-2-not-executing-php-files/>

<https://www.arubacloud.com/tutorial/how-to-enable-https-protocol-with-apache-2-on-ubuntu-20-04.aspx>

<https://www.digitalocean.com/community/tutorials/how-to-secure-apache-with-let-s-encrypt-on-ubuntu-20-04>

<https://www.digitalocean.com/community/tutorials/how-to-configure-the-apache-web-server-on-an-ubuntu-or-debian-vps>

<https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-laravel-with-nginx-on-ubuntu-20-04>

<https://laravel.io/forum/06-11-2014-not-found-but-route-exists>