

3. Frontend

- [.env](#)
- [.env.dabory](#) Gate Token
- [Frontend](#)
- [Theme](#)
- [gate-token-get](#)

.env 파일

“

.env 파일

파일

Laravel .env <https://laravel.kr/docs/5.7/configuration>)

Laravel .env Example(https://zetawiki.com/wiki/.env,_env.example)

```
# [Dabory]
ERP_THEMES=테마 이름 (themes/erp 폴더)
PRO_THEME=PRO 테마 이름 (themes/pro 폴더)
LOCALE_SEQUENCE=언어 코드
APP_MOBILE_NO=앱 번호
TAG_LINE=태그
CDN_TYPE=CDN (aws-s3 or local)
MEDIA_URL=미디어 base url (local or a3)
FAVICON_PATH=favicon path
IS_SKIP_DBUPDATE=DBUPDATE or
IS_ON_MEMBER_SIGNUP=ON/OFF

## [MetaMask]
CHAIN_ID=0xE67B
CHAIN_NAME='Dang Smart Chain'
RPC_URLS=http://3.38.62.211:8545
NATIVE_CURRENCY_NAME=DGT
NATIVE_CURRENCY_SYMBOL=DGT
NATIVE_CURRENCY_DECIMALS=18
BLOCK_EXPLORER_URLS=https://explorer-staging.dangnn.co.kr

### [asset & cache]
BROWSER_CACHE=ON/OFF (false)
```

CSSJS_URL=local or asset url (local)

[Printer]

REPORT_SERVER_URL= base url

[Laravel]

APP_NAME=

APP_ENV=.env

APP_KEY=base64:.env

APP_DEBUG=.env

APP_URL=url

LOG_CHANNEL=stack

LOG_LEVEL=debug

DB_CONNECTION=x

DB_HOST=x

DB_PORT=x

DB_DATABASE=x

DB_USERNAME=x

DB_PASSWORD=x

BROADCAST_DRIVER=x

CACHE_DRIVER=x

FILESYSTEM_DRIVER=x

QUEUE_CONNECTION=x

SESSION_DRIVER=x

SESSION_LIFETIME=x

MEMCACHED_HOST=x

REDIS_HOST=x

REDIS_PASSWORD=x

REDIS_PORT=x

MAIL_MAILER=.env

MAIL_HOST=.env

MAIL_PORT=.env

MAIL_USERNAME=.env

MAIL_PASSWORD=.env

MAIL_ENCRYPTION=.env

MAIL_FROM_ADDRESS=.env

MAIL_FROM_NAME=.env

AWS_ACCESS_KEY_ID=.env

AWS_SECRET_ACCESS_KEY=.env

AWS_DEFAULT_REGION=.env

AWS_BUCKET=.env

AWS_USE_PATH_STYLE_ENDPOINT=.env

PUSHER_APP_ID=□□x
PUSHER_APP_KEY=□□x
PUSHER_APP_SECRET=□□x
PUSHER_APP_CLUSTER=□□x

MIX_PUSHER_APP_KEY=□□x
MIX_PUSHER_APP_CLUSTER=□□x

ANALYTICS_UA_ID=
ANALYTICS_VIEW_ID=

ANALYTICS_G_ID=
ANALYTICS_PROPERTY_ID=

.env.dabory □ Gate Token

2 Rest Api (<https://manual.dabory.com/books/frontend-tutorial/chapter/restfull-api>) Gate Token

dabory Gate Token(DGT)

- [KeyPair/PublicKey] PHP Code

<https://www.php.net/manual/en/function.sodium-crypto-box-seal.php>).

- [Sodium Ajax (260~262 Lines)]

<http://git.daboryhost.com:10880/dabory/dbrrp/src/master/routes/web.php>).

- [DB ☐ ☐ how to use it in PHP (254~263 Lines)]

<http://git.daboryhost.com:10880/dabory/dbrrp/src/master/resources/views/front/dabory/pro/my-app/popup/popup-form1/form-a/client-app-form.blade.php>).

- `[secret_key] [key_pair]`
- `[erp_center_db, erp_memory_db, erp_individual_dbs]`
- `(driver, host, username, database, passwd, public_key) -> client config`
- `[...]`

[DGT 1111 11]

- `api` `DGT` `status`.
- `api` `client config` `20` `Sodium` `status`.
- `DGT` `status 200`, `DGT` `status`.
- `DGT` `status`.
- `DGT` `DGT` `status`.

[Main Request]

- 11 111111 11 11 11 DGT 1111 1111.

[DCS 1111 11]

- DGT 和 DCS 的 区别。
- memory_db 和 DGT 的 区别 DCS 的 区别。
- DCS 和 DB(erp databases) 的 区别。
- db 和 data 的 区别。

Frontend

.Frontend

- └─ 0.0.0
- └─ app
- └─ app.examples
- └─ appslabke
- └─ artisan*
- └─ composer
- └─ config
- └─ **dabory**
 - └─ para
 - └─ themes
 - (...)
- └─ **public**
- └─ ralouphe
- └─ ramsey
- └─ resources └─ robots.txt
- (...)

- dabory

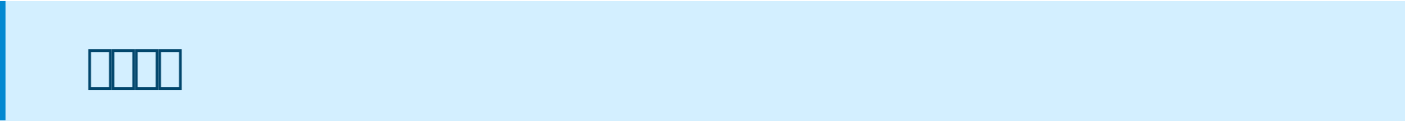
- dabory 是一个 JSON 文件，它定义了项目的主题（themes）。它包含以下信息：
 - para 是一个 ERP 系统的名称。
 - themes 是一个包含主题名称的列表。

- public

- public 是一个包含项目文件的目录。它包含以下文件：
 - public 文件，包含项目的名称、主题、CSS 文件、JavaScript 文件、以及

*Symbolic Link (Symbolic Link)

dabory 是一个 public 的 symbolic link，dabory 是一个 public 的 dabory 的 symbolic link。它包含以下信息：



它包含以下信息：

- dabory 是一个 public 的 symbolic link，它包含以下信息：

Theme

“

Theme

Theme is a set of styles and components that can be used to create a consistent look and feel for your application. It is a collection of CSS classes and components that can be used to create a consistent look and feel for your application.

Pro ()

Pro is a set of styles and components that can be used to create a consistent look and feel for your application. It is a collection of CSS classes and components that can be used to create a consistent look and feel for your application.

- pro

themes

point2u()

erp

shell

pro

app

Http

Services

para

resources

routes

shell

이 단락을 보십시오

1. app/Http

- 이 단락을 보십시오 Controller 클래스, 이 클래스는 API 클래스와 view 클래스를 포함합니다.

2. resources

- views:** Pro 클래스는 view 클래스를 포함합니다. 이 Controller 클래스는 이 클래스를 포함합니다, Controller 클래스는 이 클래스를 포함합니다

2. erp (이 단락을 보십시오)

ERP 클래스는 erponly 클래스와 ERP/pro 클래스를 포함합니다. 이 클래스는 이 클래스를 포함합니다.

- erp 클래스는 이 클래스를 포함합니다

└─themes

└─erponly

└─erp

└─shell

└─pro

└─routes

└─shell

└─qry

○ 이 단락을 보십시오

local PC: \$ cd pro newerp-php73

\$ cd /Users/EricKim/docker-works/php/newerp-
php73/dbrrerp/public/themes/pro

server : \$ cd pro newerp cd /home/newerp/public_html/public/themes/pro

local PC: \$ cd erp newerp-php73

```
$ cd /Users/EricKim/docker-works/php/newerp-  
php73/dbrrerp/public/themes/erp  
server : $ cd pro newerp cd /home/newerp/public_html/public/themes/erp
```

gate-token-get

- api gate-token

rest api PostMan ThunderClient gate-token response

.env.dabory MAIN_API_URL, MAIN_API_CLIENT_ID, MAIN_API_BEFORE_BASE64

Thunder Client MAIN_API_URL clientId BeforeBase64 gate-toek
store() test() post
body json

```
{
  "ClientId": "MAIN_API_CLIENT_ID value",
  "BeforeBase64": "MAIN_API_BEFORE_BASE64 value"
}
```

api server gate-toekn