房信 ERP 项目本地环境搭建步骤：

1、克隆项目代码：

仓库地址：

SSH方式：git@gitlab.1fangxin.cn:fangxin-tech/erp.git

HTTPS方式：<http://gitlab.1fangxin.cn/fangxin-tech/erp.git>

利用 Source Tree 来克隆

打开 Source Tree ，点击克隆，输入房信 ERP 的仓库地址：

<http://gitlab.1fangxin.cn/fangxin-tech/erp.git>

会弹窗让输入账号密码，

选择目标文件夹，E:\code\fangxin

点击克隆按钮，克隆项目到本地。

也可以看到一个主分支：master

2、初始化仓库，很重要的一步

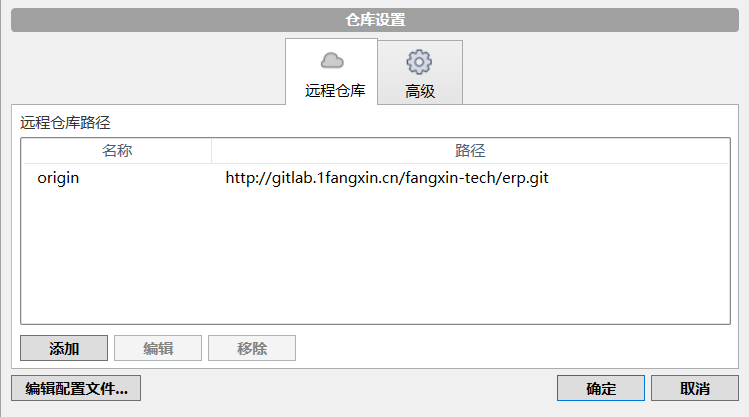
Souce Tree 点击 git 工作流，弹出如图窗口



点击确定，初始化完成后，会出现 develop 的分支

3、修改仓库配置文件

Souce Tree 点击设置，选择编辑配置文件



修改[gitflow "prefix"]如下，把自己名字设为分支目录，便于团队合作：

[gitflow "prefix"]

feature = feature/wangtaishan/

bugfix = bugfix/wangtaishan/

release = release/

hotfix = hotfix/wangtaishan/

support = support/

versiontag =

4、基于线上某个稳定的基础分支创建一个自己的开发分支

例如目前我房用的是“真实房源接口”的分支，我们基于该分支创建一个自己的 test 分支。

先在 Souce Tree 远程里检出基础分支

切换到该分支

点击分支

在新分支输入框输入 feature/wangtaishan/test

点击创建分支

标准的 gitflow 流程应该是下面这样的

feature->develop（开发分支）在这上面开发和测试->release（发布分支）产品验证->通过工作流的“完成当前项目“操作，会自动同步到->master（主分支）运维来操作该分支进行发布

由于项目需求多，开发任务多，无法将每个分支及时回归 master，我们目前采用非标准的方式

5、laragon 面板设置

点击右上角的齿轮图标（参数设置）

常规选项卡：

文件根目录改为项目的 public 目录，E:\code\fangxin\erp\public

主机名改为{name}.local ，因为我房信本地域名用的是 member.fx.local

服务&端口选项卡：

勾选MySQL、Nginx、Redis

6、修改Nginx配置

Nginx配置文件在 laragon 安装目录 E:\laragon\etc\nginx\sites-enabled 下的 00-default.conf 文件，是默认项目的Nginx配置

我们新建一个 member.fx.local.conf 文件，作为房信项目的配置文件，配置如下：

server {

listen 80;

server\_name member.fx.local erp.fx.local api.fx.local website.fx.local;

root "E:/code/fangxin/erp/public/";

index index.html index.htm index.php;

# Access Restrictions

allow 127.0.0.1;

deny all;

include "E:/laragon/etc/nginx/alias/\*.conf";

location / {

try\_files $uri $uri/ /index.php$is\_args$query\_string;

}

location ~ \.php$ {

include snippets/fastcgi-php.conf;

fastcgi\_pass php\_upstream;

#fastcgi\_pass unix:/run/php/php7.0-fpm.sock;

}

location ~ .\*\.(gif|jpg|jpeg|png|bmp|swf|flv|mp4|ico)$ {

expires 30d;

access\_log off;

}

location ~ .\*\.(js|css)?$ {

expires 7d;

access\_log off;

}

charset utf-8;

location = /favicon.ico { access\_log off; log\_not\_found off; }

location = /robots.txt { access\_log off; log\_not\_found off; }

location ~ /\.ht {

deny all;

}

}

7、修改 host 文件

将127.0.0.1指向项目域名，配置如下：

127.0.0.1 member.fx.local

8、命令行进入项目目录：

通过laragon的“终端”打开命令行窗口，通过cd命令进入房信项目的根目录。

9、复制配置文件

将 .env.example 文件复制一份，命名为 .env ，这是 laravel 项目的配置文件，命令如下：

cp .env.example .env

参考配置

APP\_ENV=local

APP\_DEBUG=true

APP\_KEY=base64:P4NtH6HBWFNoC3WZYxXBw2Feh8FC2JNhdOUjlINVHmc=

APP\_LOCAL=wf

APP\_OPEN\_WEB\_SYNC=false

ADMIN\_URL = erp.fx.local

MEMBER\_URL = member.fx.local

API\_URL = api.fx.local

WEB\_URL = website.fx.local

DC\_URL = dc.fx.local

WEB\_SYNC\_URL = sync.fx.local

M\_URL =m.fx.local

LIFE\_TIME=60

DB\_CONNECTION=mysql

DB\_HOST=192.168.101.250

DB\_PORT=3307

DB\_DATABASE=wofangerp2

DB\_USERNAME=wofangerp

DB\_PASSWORD=wofang\_erp

DB\_PREFIX=erp\_

DB\_WWW\_HOST=192.168.101.250

DB\_WWW\_PORT=3307

DB\_WWW\_DATABASE=wofangerp2

DB\_WWW\_USERNAME=wofangerp

DB\_WWW\_PASSWORD=wofang\_erp

DB\_WWW\_PREFIX=erp\_

CACHE\_DRIVER=redis

SESSION\_DRIVER=redis

QUEUE\_DRIVER=redis

REDIS\_HOST=127.0.0.1

REDIS\_PASSWORD=null

REDIS\_PORT=6379

REDIS\_NO\_0=0

REDIS\_NO\_1=1

REDIS\_NO\_2=2

SCOUT\_DRIVER=null

SCOUT\_QUEUE=null

SCOUT\_ELASTIC\_HOST=null

MAIL\_DRIVER=smtp

MAIL\_HOST=mailtrap.io

MAIL\_PORT=2525

MAIL\_USERNAME=null

MAIL\_PASSWORD=null

MAIL\_ENCRYPTION=null

AUDITING\_QUEUE=false

BUGSNAG\_API\_KEY=

PUSHER\_APP\_ID=

PUSHER\_APP\_KEY=

PUSHER\_APP\_SECRET=

TEST\_ADMIN\_ID=910

TEST\_STORE\_AGENT\_ID=56

TEST\_AGENT\_ID=8

API\_CRYPT\_KEY=ZZ8cUkhZZbDXwoWF

启动laragon服务，这时浏览器访问，会抛出异常：

**Warning**: require(E:\code\fangxin\erp\bootstrap/../vendor/autoload.php): failed to open stream: No such file or directory in **E:\code\fangxin\erp\bootstrap\autoload.php** on line **17**

**Fatal error**: require(): Failed opening required 'E:\code\fangxin\erp\bootstrap/../vendor/autoload.php' (include\_path='.;E:/laragon/etc/php/pear') in **E:\code\fangxin\erp\bootstrap\autoload.php**on line **17**

意思是缺少一些依赖文件，稍后我们需要利用 composer 和 yarn 把依赖文件搞下来

10、安装php的依赖库的

安装命令如下：

首先设置composer为国内镜像，不然下载速度很慢，有VPN可以忽略，执行：

composer config -g repo.packagist composer <https://packagist.phpcomposer.com>

安装命令：

composer install

#install 命令从当前目录读取 composer.json 文件，处理了依赖关系，并把其安装到 vendor 目录下。

可能报错如 php 版本低：

这是因为不匹配composer.json要求的版本。php版本要求7.1.3以上

建议安装一个新一点的php版本

<https://windows.php.net/download/>

下载一个7.2.10版本，(下载官方版本名称中不带 nts 的php版本文件)

下载后将解压的文件复制到 bin/php 目录下

参考帖子：<https://blog.csdn.net/hj960511/article/details/51693225>

或者忽略版本composer设置忽略版本匹配

参考帖子：<https://blog.csdn.net/zmzwll1314/article/details/74091788>

composer可以设置忽略版本匹配，命令是：

composer config -g repo.packagist composer <https://packagist.phpcomposer.com>

然后

composer install --ignore-platform-reqs

参考帖子：<https://blog.csdn.net/zmzwll1314/article/details/74091788>

如果报错如下：

[Composer\Exception\NoSslException]

The openssl extension is required for SSL/TLS protection but is not available. If you can not enable the openssl extension, you can disable this error, at your own risk, by setting the 'disable-tls' option

to true.

解决方法：在php.ini文件中打开php\_openssl扩展

extension=openssl

参考帖子：<https://blog.csdn.net/Luke__/article/details/79509977>

可能报如下一系列的错误：

laravel/framework v5.4.36 requires ext-mbstring \* -> the requested PHP extension mbstring is missing from your system.

将 php.ini 对应的服务开启即可，把对应 extension=mbstring 前面的分号去掉

类似的报错，开启对应的服务即可

11、yarn install

yarn 是Node JS的一个包管理工具，比NPM执行效率高

NPM的全称是Node Package Manager，是随同NodeJS一起安装的包管理和分发工具，它很方便让JavaScript开发者下载、安装、上传以及管理已经安装的包。

12、打包前端资源代码（也可以通过执行该命令检查是否安装了gulp）

gulp

13、执行数据库迁移（如果需要）

php artisan migrate