

## 一、git仓库

### 1.git仓库

## 二、环境配置

### 1.运营平台

- 1.dev
- 2.巴西
- 3.印度
- 4.印尼

### 2.配置平台

- 1.dev

### 3.游戏

- 1.dev
- 2.巴西
- 3.印度
- 4.印尼

### 4.k8s

- 1.dev
- 2.巴西
- 3.印度
- 4.印尼

### 5.日志平台

## 三、工作流程

### 1.添加新表

- 1.从main分支切一个feat分支
- 2.修改对应的sql文件（只用修改mysql文件，不需要修改mongodb文件）
- 3.合到main分支

### 3.添加权限（权限翻译）

- 1.从main分支切一个feat分支
- 2.修改对应的sql文件（参考mala: admin\_permission进行建表）
- 3.合到main分支
- 4.往xlsx文档添加权限翻译
- 5.dev环境需要自己进行翻译（然后配置到配置中心）

### 4.修改mala-core项目（举例）

- 1.设计到的项目工程
- 2.环境配置
- 3.修改工程代码
  - 1.修改mala-idl工程
  - 2.修改mala-core工程
  - 3.修改mala-client工程
  - 4.修改api工程
  - 5.修改db-script工程

## 四、发版

### 1.合并代码

- 1.合并代码时间节点

### 2.MySQL更新

### 3.提单发布

- 1.需要在dev发布，获取version\_id

# 一、git仓库

## 1.git仓库

`https://git.easycodesource.com/users/gavin.g/groups`

# 二、环境配置

## 1.运营平台

### 1.dev

地址: `https://operation.easycodedev.com/`

账号: gavin

密码: 520lanXIA

验证码: 随便填

### 2.巴西

网址: `https://operation.brazil-zenam.com/#/login`

账号: gavin

密码: 520lanXIA

### 3.印度

网址: `https://operation.zenam.net/#/login`

账号: gavin

密码: 520lanXIA

### 4.印尼

网址: `https://operation.idn-zenam.com/#/login`

账号: gavin

密码: 520lanXIA

## 2.配置平台

### 1.dev

网址: <https://conf.easycodedev.com/#/appsSet/appsSet>

账号: gavin

密码: 520lanXIA

## 3.游戏

### 1.dev

### 2.巴西

<https://game.brazil-zenam.com/Cocos-H5-Brazil/473/>

### 3.印度

### 4.印尼

<https://game.idn-zenam.com/Cocos-H5-Idn/236/>

## 4.k8s

### 1.dev

网址: <https://kubernetes-dashboard.easycodesource.com/#/pod?namespace=game-portal-dev>

token:

eyJhbGciOiJSUzI1NiIsImtpZCI6InBHTUdEM1JFNmszYldCV1JUVzBKV1AyNkxzczlRVUViUTRid3YtNl96WXMifQ.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJrdWJlLXN5c3RlbSIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VjcmV0Lm5hbWUiOiJla3MtZWRpdC1zcGVjaWZpYy1ucy10b2t1bi1ubHE5eCIIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VydmljZS1hY2NvdW50Lm5hbWUiOiJla3MtZWRpdC1zcGVjaWZpYy1ucyIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VydmljZS1hY2NvdW50LnVpZCI6IjVkdMDkyMzJjLWM5OWUtNGQzNi1hZWU4LTE5MzUxNDhmZDg3ZSIsInN1YiI6InN5c3RlbTpwZXJ2aWNlYWNjb3VudDprdwJlLXN5c3RlbTpla3MtZWRpdC1zcGVjaWZpYy1ucyJ9.Zjx8XbMUJT54qAM-qe-uVjUcUE7MNMvn7abrLlDBSZT0lOWOIuX46tuqkdEiO6oWy\_5b7U8dh1kN2fFFo3TFCMSRacVwyI7mK\_te2m-Xmqg5dwMg7HngPIkRnc3HrbSXWdBewuu3jVCefNDHdms4EjIe1k4FrGSLjgKBfJLEqmeE5LHPTwDlQ59rXtKy2lnhudx7ZtTS86KkBTDoUc2xe5JIVvA6oz6qtwtlCawAxhGiM2V0BBp2uJIApDGNEFnTBLWV9vzMkxtteaREVKwz0zB-5663EWSjSSL\_njMgtiVhy94BwXHeY9Hon5124LKOVJlHwp6KcI9QHKu7ck4GJdA

## 2.巴西

网址: <https://kubernetes-dashboard.easycodesource.com/#/pod?namespace=game-portal-brazil-test>

token: 同上

## 3.印度

网址: <https://kubernetes-dashboard.easycodesource.com/#/pod?namespace=game-portal-india-test>

token: 同上

## 4.印尼

网址: <https://kubernetes-dashboard.easycodesource.com/#/pod?namespace=game-portal-idn-test>

token: 同上

## 5.日志平台

```
https://kibana-logs.easycodesource.com/app/discover#/?_a=(columns:!(message),filters:!(
),hideChart:!t,index:fbbb3ce0-2cdd-11ed-b9a5-5dfa4c4d40eb,interval:auto,query:
(language:kuery,query:''),sort:!(('@timestamp',desc))&_g=(filters:!(
),refreshInterval:(pause:!t,value:0),time:(from:now-3d,to:now))
```

# 三、工作流程

## 1.添加新表

### 1.从main分支切一个feat分支

#### 1. git仓库

```
https://git.easycodesource.com/platform/db-script
```

### 2.修改对应的sql文件（只用修改mysql文件，不需要修改mongodb文件）

#### 1. 文件名称

```
mala-init.sql
mala-v1.3.0.sql
```

### 3.合到main分支

### 3.添加权限（权限翻译）

#### 1.从main分支切一个feat分支

##### 1. git仓库

```
https://git.easycodesource.com/platform/db-script
```

#### 2.修改对应的sql文件（参考mala： admin\_permission进行建表）

##### 1. 修改sql文件

```
admin-v1.3.0.sql  
mala-data-init.sql
```

##### 2. 修改版本sql, admin-v1.3.0.sql

参考数据表： mala.admin\_permission

```
INSERT INTO admin_permission VALUES  
  (7250, '在线抽奖配置', 'online_raffle_conf', 1, NULL, '7000/7200/7250/', 0, NULL,  
7200, NOW(), NOW()),  
  (7251, '在线抽奖配置查询', 'online_raffle_conf_get', 1, NULL,  
'7000/7200/7250/7251/', 1, NULL, 7250, NOW(), NOW()),  
  (7252, '在线抽奖配置新增', 'online_raffle_conf_add', 2, NULL,  
'7000/7200/7250/7252/', 1, NULL, 7250, NOW(), NOW()),  
  (7253, '在线抽奖配置修改', 'online_raffle_conf_mod', 3, NULL,  
'7000/7200/7250/7253/', 1, NULL, 7250, NOW(), NOW()),  
  (7254, '在线抽奖配置删除', 'online_raffle_conf_del', 4, NULL,  
'7000/7200/7250/7254/', 1, NULL, 7250, NOW(), NOW()),  
  (7260, '在线抽奖记录', 'online_raffle_record', 1, NULL, '7000/7200/7260/', 0,  
NULL, 7200, NOW(), NOW()),  
  (7261, '在线抽奖记录查询', 'online_raffle_record_get', 1, NULL,  
'7000/7200/7260/7261/', 1, NULL, 7260, NOW(), NOW()),  
  (7262, '在线抽奖记录导出', 'online_raffle_record_download', 2, NULL,  
'7000/7200/7260/7262/', 1, NULL, 7260, NOW(), NOW());
```

##### 3. 修改总的sql, mala-data-init.sql

```

(7250, '在线抽奖配置', 'online_raffle_conf', 1, NULL, '7000/7200/7250/', 0, NULL,
7200, NOW(), NOW()),
(7251, '在线抽奖配置查询', 'online_raffle_conf_get', 1, NULL,
'7000/7200/7250/7251/', 1, NULL, 7250, NOW(), NOW()),
(7252, '在线抽奖配置新增', 'online_raffle_conf_add', 2, NULL,
'7000/7200/7250/7252/', 1, NULL, 7250, NOW(), NOW()),
(7253, '在线抽奖配置修改', 'online_raffle_conf_mod', 3, NULL,
'7000/7200/7250/7253/', 1, NULL, 7250, NOW(), NOW()),
(7254, '在线抽奖配置删除', 'online_raffle_conf_del', 4, NULL,
'7000/7200/7250/7254/', 1, NULL, 7250, NOW(), NOW()),
(7260, '在线抽奖记录', 'online_raffle_record', 1, NULL, '7000/7200/7260/', 0,
NULL, 7200, NOW(), NOW()),
(7261, '在线抽奖记录查询', 'online_raffle_record_get', 1, NULL,
'7000/7200/7260/7261/', 1, NULL, 7260, NOW(), NOW()),
(7262, '在线抽奖记录导出', 'online_raffle_record_download', 2, NULL,
'7000/7200/7260/7262/', 1, NULL, 7260, NOW(), NOW());

```

4. dev环境执行sql, 往数据库添加权限
5. 什么环境需要我们执行, 什么环境需要运维执行

### 3. 合到main分支

#### 4. 往xlsx文档添加权限翻译

1. 往xlsx文档添加翻译

网址: <https://docs.google.com/spreadsheets/d/1A-2M9XfMnqXVnfGzy0Vcq9FSc6LfspW/edit#gid=1017811402>

需要新增: msg\_id、zh, 然后通知PM, 等PM去翻译其他语言

#### 5. dev环境需要自己进行翻译 (然后配置到配置中心)

1. 修改配置中心翻译

网址: <https://conf.easycodedev.com/#/appsSet/appsSet>

修改: i18n-resource-portal.csv

1. 复制到本地, 新建一个csv文档
2. 添加新权限翻译 (这时候可以借助第三步的sql进行添加)

2. 发布
3. dev需要自己翻译

## 4.修改mala-core项目（举例）

### 1.设计到的项目工程

●

项目工程	仓库地址	备注
mala-core	<a href="https://git.easycodesource.com/game-portal/mala-core">https://git.easycodesource.com/game-portal/mala-core</a>	gRPC服务端
mala-idl	<a href="https://git.easycodesource.com/game-portal/mala-idl">https://git.easycodesource.com/game-portal/mala-idl</a>	定义proto文件（供客服端和服务端掉用）
mala-client	<a href="https://git.easycodesource.com/game-portal/mala-client">https://git.easycodesource.com/game-portal/mala-client</a>	gRPC客服端
api	<a href="https://git.easycodesource.com/int/api">https://git.easycodesource.com/int/api</a>	定义proto文件（供前端掉用） 这个proto文件里的内容和mala-idl中的proto内容不一致，该文件中提供给前端使用，mala-idl是客服端和服务端之间的数据交互

### 2.环境配置

1. env.yaml（每个项目使用的env.yaml文件不同，不过数据库配置相同）

```
i18n:
  lang: "en-us"
  tz: "UTC"
  src_path: "/app/mnt/i18n-resource-portal.csv"

#mysql:
#  name: "mala"
#  user: "root"
#  password: "123456"
#  host: "127.0.0.1"
#  port: 3307

mysql:
  name: "mala"
  user: "root"
  password: "abc@123456A"
  host: "middleware-2.easycodedev.com"
  port: 3306

redis:
#  url: "redis://127.0.0.1:6379/"
```

```
url: "redis://middleware-2.easycodedev.com:6379/"

mongo:
# url: "mongodb://127.0.0.1:27017/"
url: "mongodb://middleware-2.easycodedev.com:27017/"

data_platform_mongo:
url: "mongodb://127.0.0.1:27017/"

db_aes_key: "ZmtkZiomZmQ4NzZmZGZLZg=="

kafka:
bootstrap_servers: "127.0.0.1:9092"

mala_grpc_server:
host: "mala-core-grpc:50051"
token: "36E81134-79EF-478F-844E-0580C3D9D8EE"

turbo_grpc_server:
host: "turbo_glo_srv:50051"
token: "36E81134-79EF-478F-844E-0580C3D9D8EE"

gconf:
server_host: ""
app_key: ""
app_secret: ""
conf_data_dir: ""
cluster: ""
debug: true

s3:
aws_access_key_id: ""
aws_secret_access_key: ""
region_name: ""
bucket_name: ""
domain: ""

report:
server_host: "http://report-server-api:8000/api/"
app_key: "c93ntiZ4dmQkOc8MWmrSMhMVwPjp4D2AW4szuv5V"
app_secret: "i5aIg-04VTKKAUpgv4cqQNFdnb4KrXOmA-I2wZqS"
temp_file_dir: "/tmp/export_data/"
debug: false

channel_rpc_server:
host: ""
api_key: ""

ip_db:
```



```

path: "/app/mnt/IP2LOCATION-LITE-DB3.BIN"

log:
  level: "DEBUG"

debug: true

maestro:
  enable: false
  kafka_broker_servers: ""

safe_dog:
  api_domain: ""
  app_key: ""
  app_secret: ""

sso:
  host: ""
  secret_key: ""
  verify_token:
    uri: ""
    enable_login_by_token: true
    enable_login_by_password: true
  verify_vcode:
    uri: ""
    enable: false

otel:
  trace:
    enable: false
    service_name: "mala-op"
    endpoint_host_env: "OTEL_AGENT_HOST" # OTel agent host 环境变量名
    endpoint_port: 4317
    enable_grpc_client: true # 是否启用 grpc client 自动检测
    enable_grpc_server: false # 是否启用 grpc server 自动检测
    debug: false

```

## 2. 修改hosts文件

### 1. mala-core项目需要配置grpc服务启动的host和端口

mala-core项目env.yaml文件里面有如下配置:

grpc\_server:

host: "0.0.0.0:50051"

token: "36E81134-79EF-478F-844E-0580C3D9D8EE"

mala-core项目settings文件里面有如下配置:

GRPC\_SERVER\_LISTEN\_HOST = ENV["grpc\_server"]["host"]

GRPC\_SERVER\_TOKEN = ENV["grpc\_server"]["token"]

然后mala-core项目下有一个脚本, 名字叫做start\_open\_server

所以mala-core项目启动时候, 就是启动grpc服务, 使用如下命令

python manage.py start\_open\_server

## 2. mala-core:start\_open\_server.py (启动gRPC服务)

```
class Command(BaseCommand):
    def handle(self, *args, **options):
        # MemoryProfiler
        # profiler = MemoryProfiler(interval_period=20, top_files=30)
        # profiler.start()

    def _handle_signal(signalnum, frame):
        pid = os.getpid()
        _LOGGER.info(
            "open server process {} receives signal {}, exit.".format(
                pid, signalnum
            )
        )
        server._server.stop(3)

    signal.signal(signal.SIGINT, _handle_signal)
    signal.signal(signal.SIGTERM, _handle_signal)

    _LOGGER.info("starting mala-core rpc server ... ")
    # 启动服务端
    server = SecureServer(
        settings.GRPC_SERVER_LISTEN_HOST,
        settings.GRPC_SERVER_CREDENTIALS,
        settings.GRPC_SERVER_TOKEN,
        max_workers=30,
    )

    for service in _PROVIDER_SERVICE_LIST:
        server.register_service(service)

    server.startup()
```

### 3. mala-client里面需要配置grpc对应的host和端口

mala-client项目env.yaml文件里面有如下配置:

```
mala_grpc_server:  
  host: "mala-core-grpc:50051"  
  token: "36E81134-79EF-478F-844E-0580C3D9D8EE"
```

mala-client项目settings.py有如下代码:

```
MALA_RPC_HOST = ENV["mala_grpc_server"]["host"]  
MALA_RPC_AUTH_TOKEN = ENV["mala_grpc_server"]["token"]
```

mala-client直接通过python manage.py runserver启动

前端发起HTTP调用, 然后进入mala-client后, 进入view函数, 然后使用RpcCaller调用mala-core gRPC服务

### 4. mala-client里面需要连接mala-core gRPC服务地址

#### 1. 请求

```
path("activity_center/list/", activity_center_view.get_all_activities)
```

#### 2. 调用view函数

```
@validate_parameters(api_pb2.ListActivityReq, api_pb2.ListActivityResp)
```

```
def get_all_activities(request, req, resp):
```

```
    resp = get_caller().act_activity_center_stub.ListActivity(  
        activity_pb2.ListActivityReq(  
            user_id=req.header.user_id, p=data_to_message(request.p_dict,  
common_pb2.P)  
        )  
    )  
    validate_response(resp)
```

```
    return dict(  
        ongoing_activities=_get_act_data(resp.ongoing_activities),  
        unopened_activities=_get_act_data(resp.unopened_activities),  
        expired_activities=_get_act_data(resp.expired_activities),  
        user_act_stats=[  
            dict(  
                act_id=i.act_id,  
                available_bonus_amount=i.available_bonus_amount,  
            )  
            for i in resp.user_act_stats  
        ],  
    )
```

### 3.使用caller调用gRPC服务

```
def get_caller() -> RpcCaller:
    global _caller
    if _caller is not None:
        return _caller
    _caller = RpcCaller()
    return _caller
```

### 4.RpcCaller()连接到mala-core: gRPC服务

```
class RpcCaller:
    def __init__(self):
        self._channel = insecure_channel(
            settings.MALA_RPC_HOST,
            settings.MALA_RPC_AUTH_TOKEN,
        )
```

### 5. 修改hosts文件，添加域名映射

```
sudo vim /etc/hosts
```

添加如下配置：

```
127.0.0.1      mala-core-grpc
```

## 3.修改工程代码

### 1.修改mala-idl工程

- 仓库地址：<https://git.easycodesource.com/game-portal/mala-idl>
- 从dev分支切一个个人分支
- 然后将改动合到dev分支，推送到远端
- dev和test环境都是用dev分支进行部署
- 到发版时候，会有负责人将dev合到main分支上（不需要将个人分支合并到main分支），然后打上tag

### 2.修改mala-core工程

- 仓库地址：<https://git.easycodesource.com/game-portal/mala-core>
- 从对应的版本分支切一个个人分支
- 将代码改动合并到dev分支，推送到远端，部署进行测试
- 在gitlab上将代码合并到版本分支，提MR，将合并后的版本分支部署到test各个市场

### 3.修改mala-client工程

- 仓库地址：<https://git.easycodesource.com/game-portal/mala-client>
- 从对应的版本分支切一个个人分支
- 将代码改动合并到dev分支，推送到远端，部署进行测试
- 在gitlab上将代码合并到版本分支，提MR，将合并后的版本分支部署到test各个市场

## 4.修改api工程

- 仓库地址: <https://git.easycodesource.com/int/api>
- 从dev分支切一个个人分支
- 然后将改动合到dev分支, 推送到远端
- dev和test环境都是用dev分支进行部署
- 到发版时候, 会有负责人将dev合到main分支上 (不需要将个人分支合并到main分支), 然后打上tag

## 5.修改db-script工程

- 仓库地址: <https://git.easycodesource.com/platform/db-script>
- 从main分支切一个个人分支
- 修改下面4个文件

```
admin-v1.3.0.sql:      权限相关
mala-data-init.sql:   权限相关
mala-init.sql:        MySQL数据表
mala-v1.3.0.sql:      MySQL数据表
```

- 在gitlab上将代码合并到main分支, 提MR, ALin审核 (什么时候提这个MR可以问问, 发版前一两天还是什么时候)

# 四、发版

## 1.合并代码

### 1.合并代码时间节点

## 2.MySQL更新

## 3.提单发布

### 1.需要在dev发布, 获取version\_id