



广东飞企互联科技股份有限公司

凌云中台-前端入门培训

中国智慧园区解决方案领导品牌

凌云中台 2023/02

密级：内部公开



01. 应用套件主要功能点

02. 创建前端工程

03. 业务功能开发

04. 快速定位工程与问题

05. 代码规则

06. 组件库

CONTENTS





01

PART

应用套件主要功能点

COURSE TRAINING

❖ 1.1 套件主要功能点

1. **微应用管理**（工程地址 + 路由）
2. **页面管理**（由pages.json发布）
3. **参照项**（组件：p-dict）
4. **流水号服务**（组件：p-sn）
5. **门户组件**（工作台个性化门户的组件）
6. **表单建模**（创建流程表单-关联）
7. **流程服务**（发布/更新表单、流程图）
8. **数据库管理**（查看数据）
9. **开发沙箱**（给企业沙箱安装套件，快速进行测试）
10. **数联中心**（指标 / 大屏）





02

PART

创建前端工程

COURSE TRAINING

❖ 2.1 PAI 工具创建前端工程

1/ 创建前端: `pai create front -n pai-xxx-ui` (-s 可绑定套件)

2/ 初始化Git: `pai init`

3/ 提交代码到本地Git仓库: `pai commit`

4/ 推送代码: `pai push`

5/ 使用代码编辑器: 项目管理 / Git管理

* `pai -h` 或 `pai xxx -h` 可查看更多命令及语法

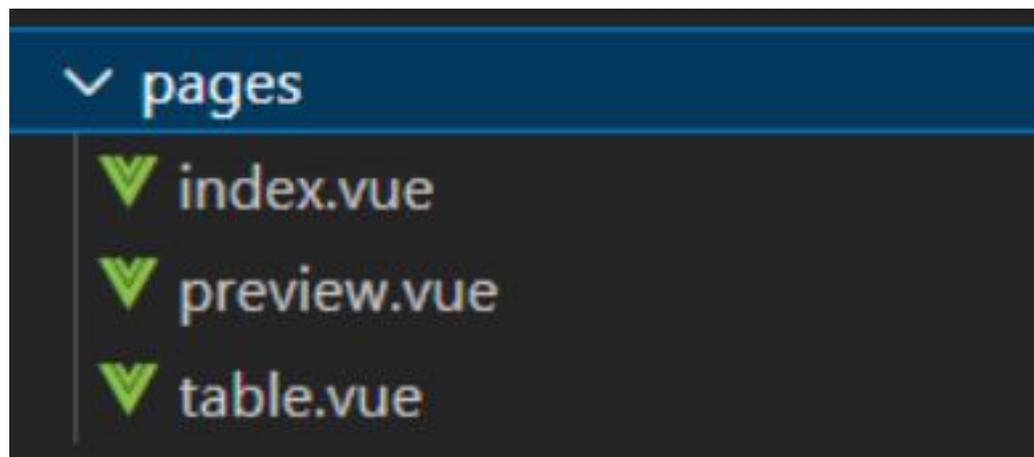


2.2 前端工程介绍

PAI前端脚手架，一个快速开发业务应用页面的框架。基于最新技术组合（**vue + vuex + axios + nuxt + element-ui + paiui**），主要目的让开发者注重专注业务，降低技术难度，从而节省人力成本，缩短项目周期

- 内置引入 **pai-ui**
- 无需配置路由表
- 国际化支持
- 编译 / 预览：流程表单 | WEB | APP小组件

- 按需引入 **pai-sp-ui**



前端介绍文档：<http://api.flyrise.cn:9099/docs/open-docs/open-docs-1cu0bc8t83ha0>



03

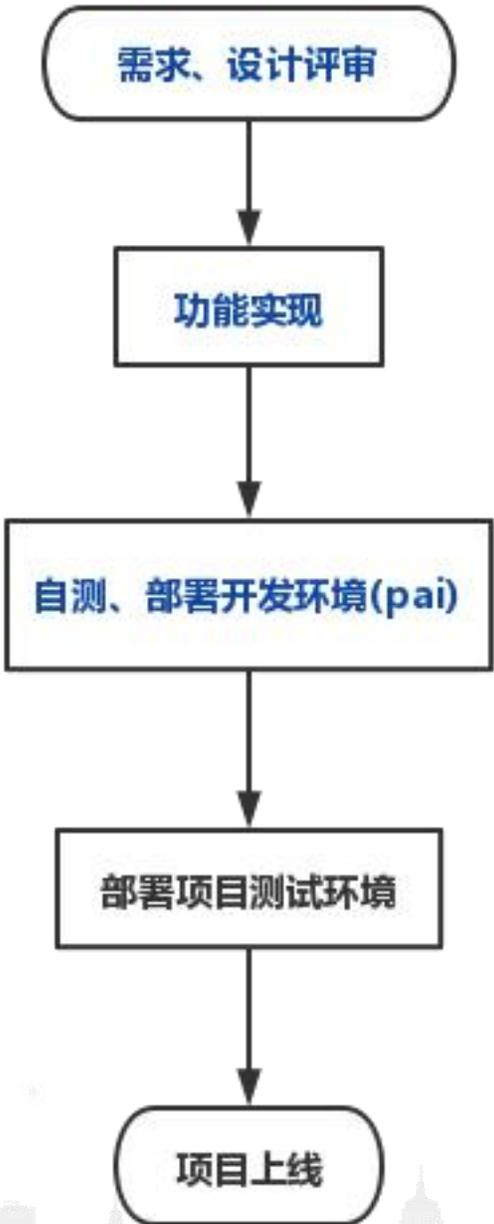
PART

业务功能开发

COURSE TRAINING

3.1 业务开发周期

- 1/ 需求、设计评审
- 2/ 功能实现
- 3/ 自测、部署开发环境(pai)
- 4/ 部署项目测试环境
- 5/ 项目上线



3.2 前端开发—设置OSS

```

JS nuxt.config.js 1, M X ①
JS nuxt.config.js > [?] <unknown>
72   env: {
73     // 开发过程的接口域名
74     SERVER_URL,
75     // 接口前缀
76     REQUEST_BASE_URL,
77     // 存储token到cookie的key
78     TOKEN_KEY: `${package.name}-token`,
79     // 基础路由
80     BASE_ROUTE,
81     // 业务参数
82     version: 'v1', // 版本
83     isOss: true, // 是否开启OSS-需要后端业务接口对接OSS, 请修改/api/user.js => getOssToken
84     isAuth: true, // 是否开启登录验证
85   },
86   proxy: {

```

```

JS user.js M X ②
api > JS user.js > ...
48
49  /**
50   * 获取oss-token
51   * 根据实际后端提供的oss接口修改url
52   */
53   export const getOssToken = () => {
54     return request({
55       url: '/work-engine-api/oss/token',
56       method: 'GET',
57     })
58   }

```

```

JS pai-ui.js M X ③
plugins > JS pai-ui.js > ...
12   export default ({ store }) => {
13     // 初始化
14     Vue.use(PaiUI, {
15       // 接口地址
16       baseURL: process.env.REQUEST_BASE_URL,
17       // 获取token方法
18       token: () => {
19         return Cookies.get(process.env.TOKEN_KEY)
20       },
21       cacheKey: process.env.TOKEN_KEY + '-vuex',
22       // 获取ossToken方法
23       // ossToken: () => {
24         // 知一
25         // return Cookies.get(`...`)
26       },
27       // 国际化
28       locale: store.state.locale,
29     })
30   }

```

1. 开启 OSS

2. 配置业务 OSS 的接口

3. 配置 paiui 的 OSS

4. 使用上传组件（提交需确认文件）

```

<p-uploader v-model="form.attachId" :updating="true"></p-uploader>
</div>
</template>

<script>
export default {
  data() {
    return {
      form: {
        attachId: '',
      },
    }
  }
}

```

3.3 前端开发—本地代理

1. 配置代理后端、本地IP（与后端本地调试）

```
/**
 * 本地服务器代理配置（生产环境无效）
 * 根据本地后端服务修改，或添加到proxies
 */
const LOCAL_PROXY_OPTIONS = {
  enable: true, // 是否启用
  target: 'http://10.62.20.91:8080', ←
  proxies: [
    {
      path: 'park-property-api', // 替换成业务后端api
    },
    // {
    //   path: 'test-api',
    //   target: 'http://127.0.0.1:8888'
    // }
  ],
}
```

2. 需与①中的对应

```
/**
 * 获取列表
 * @param query
 */
export const ApiGetTableList = (query) => {
  return request({
    url: '/park-property-api/table/page',
    method: 'GET',
    params: query,
  })
}
```



3.3 前端开发—参照项

参照项		参照项选项													
<p>扩展参照项 系统参照项</p> <hr/> <p>请输入关键字 <input type="text"/></p> <table border="1"> <thead> <tr> <th>参照项标识</th> <th>参照项名称</th> </tr> </thead> <tbody> <tr> <td>okr_test_type</td> <td>请假类型</td> </tr> </tbody> </table>		参照项标识	参照项名称	okr_test_type	请假类型	<table border="1"> <thead> <tr> <th>选项值</th> <th>选项名</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>事假</td> </tr> <tr> <td>2</td> <td>年假</td> </tr> <tr> <td>3</td> <td>病假</td> </tr> </tbody> </table>		选项值	选项名	1	事假	2	年假	3	病假
参照项标识	参照项名称														
okr_test_type	请假类型														
选项值	选项名														
1	事假														
2	年假														
3	病假														

请假类型:

- 事假
- 年假
- 病假

```
<p-dict v-model="form.dict" suite-code="cn.flyrise.okr" dict-type="okr_test_type"></p-dict>
```

- * 只读：不显示在控制台
- * 编辑：企业可自定义修改（控制台）
- * 企业需安装该套件

参照项文档：<http://api.flyrise.cn:9099/docs/open-docs//2062>

3.3 前端开发—流水号

流水号服务
提供给开发者统一管理流水号的生成和使用

模板编号	模板描述
okr_order_no	测试流水号

规则管理

模板内容预览: QJD202302220001

规则类别	规则内容	模拟内容
文本 <input type="text" value="QJD"/>	QJD	QJD
日期 <input type="text" value="YYYYMMDD"/>	YYYYMMDD	20230222
流水位 <input type="text" value="4"/>	4	0001

请假单号: QJD202302220002

```

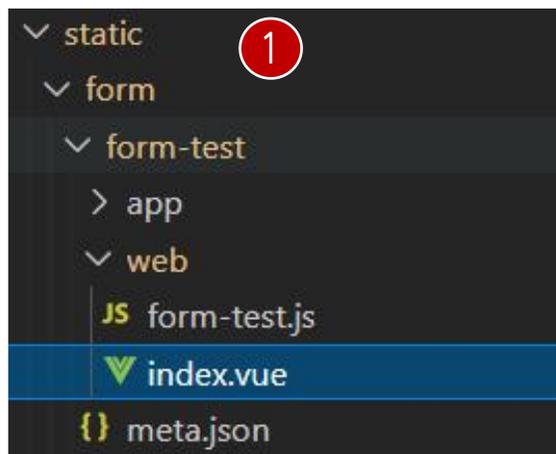
<p-sn v-model="form.sn" temp-name="okr_order_no"></p-sn>
```

*** 企业可自定义修改 (控制台)**

流水号文档: <http://api.flyrise.cn:9099/docs/open-docs/open-docs-1ctpsffrfilbn>

3.4 前端开发—表单开发

1. 新建 /static/form/表单



2. 创建 /绑定表单 (meta.json)

表单建模 ②

通过可视化拖拽组件生成前端表单代码

pai-okr-ui pai-djh-demo-ui

表单ID	页面名称
1628292642316361728	总部请假单

VS Code 文件(F) 编辑(E) 选择(S) 查看(V) 转到(G) 运行(R)

```

{} meta.json M X
static > form > form-test > {} meta.json > ...
You, a minute ago | 2 authors (djh and others)
1  {
2    "info": {
3      "desc": "总部请假单",
4      "formId": "1628292642316361728"
5    }
6  }
7
                    
```

3. 编译表单 / 发布表单 ③

```

npm run build:form-web -n xxx
pai form deploy web -n xxx
                    
```

4. 创建流程及绑定:

④ * 关联表单:

pai-djh-demo-ui / 总部请假单

pai-okr-ui	>	✓ 总部请假单
pai-djh-demo-ui	>	



❏ 3.4.1 前端开发—表单/组件编译与发布

- 请求的方法

1/ this.\$REQUEST

2/ fetch()

注：不能使用axios；不能import @/api文件里的。部分ES6语法不可用

- **创建表单建模 / meta绑定formId / 创建流程服务 / 绑定表单js**

- **预览表单 — /xx路由/preview**

- **流程服务 编辑 / 更新js / 沙箱安装 / 控制台 / 应用套件 / 编辑流程 / 发布**

流程表单文档：<http://api.flyrise.cn:9099/docs/open-docs/open-docs-1cu3aqm3qoiec>

❖ 3.4.2 前端开发—表单事件API

- 事件API

<code>submit('staging')</code>	暂存
<code>submit('start')</code>	送办一点击送办触发-未选人, 需 <code>this.\$emit('next', form)</code> 往下执行
<code>startSubmit()</code>	开始后—送办后-已选人
<code>handleSubmit()</code>	办理一点击办理触发-未选人, 需 <code>this.\$emit('handle', form)</code> 往下执行
<code>handleAfterSubmit({ idea, audit})</code>	办理后—送办后-已选人
<code>endSubmit()</code>	结束—最后的节点且送办后触发-已选人
<code>beforeSubmit(params, callback)</code>	在 (转办 / 退回 / 加签) 之前时触发, 往下执行需 <code>callback()</code>



3.4.3 前端开发—表单/门户组件：预览调试

1. 打开 (/工程名/preview)

2. 组件开发/预览



3. 表单开发/预览

localhost/djh-demo/preview?id=111

WEB小组件预览 APP小组件预览 流程表单预览

模拟流程状态:

模拟流程事件:

* 姓名:

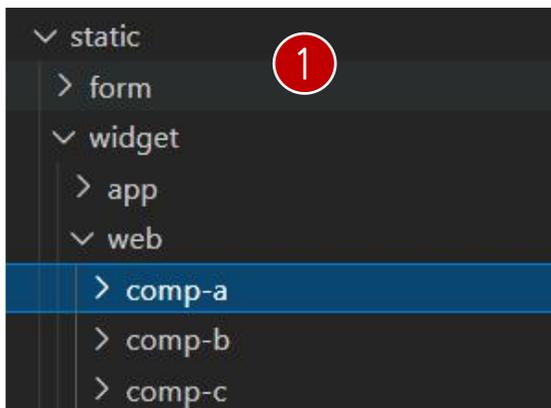
* 公司名称:

3.4.4 前端开发—门户组件

可支持桌面挂件：

FE 飞企互联

1. 新建/static/widget/web/组件



2. 编译组件 / 发布组件 2

```
npm run build:widget -n xxx
```

```
pai widget deploy web -n xxx
```



门户组件文档：<http://api.flyrise.cn:9099/docs/open-docs/open-docs-1ctpscsnrf94s>

❖ 3.5 后端API文档（swagger文档）

- <https://pai.flyrise.cn/后端服务-api/doc.html>
- 后端工程-api（park-property-api）
- 登录、调试

The screenshot displays the API documentation for the 'work-engine-api' project. The browser address bar shows the URL: <https://pai.flyrise.cn/work-engine-api/doc.html#/default/>. The page title is '凌云中台 - 工单引擎应用'. The main content area shows a GET request for the endpoint `/work-engine-api/work/order/type/v1/list`. The request parameters are listed as follows:

参数名称	值
parkId	

The interface also includes a sidebar with navigation options like '文档', '调试', and 'Open'. On the right side, there is a '后端工程' (Backend Project) section with a list of projects, including 'pai-work-engine' which is highlighted with a red box. A red arrow points from the 'pai-work-engine' entry in the sidebar to the API endpoint in the main content area.

3.6 部署工程、发布微应用

1/ 配置pages.json

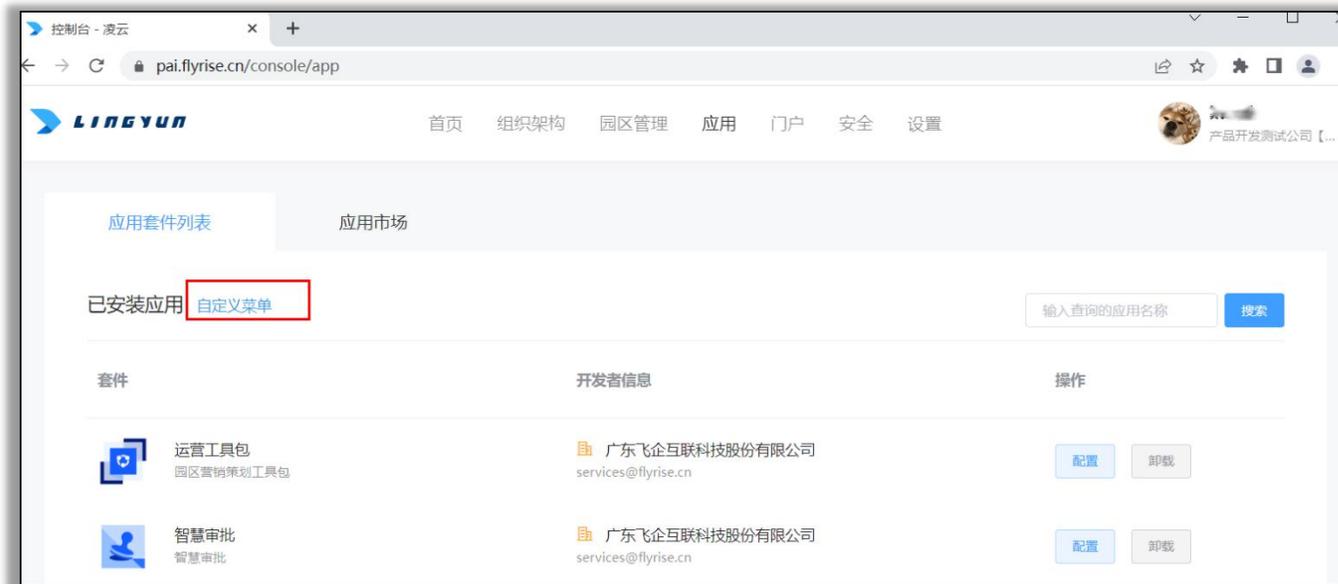
2/ 部署工程 pai deploy front

3/ 页面管理—增加微应用

4/ 沙箱安装套件

5/ 控制台添加菜单（控制台 / 应用 / 自定义菜单 / 导入应用）

6/ 控制台添加权限（控制台 / 组织架构 / 角色管理 / 菜单权限）



前端介绍及部署文档：<http://api.flyrise.cn:9099/docs/open-docs/open-docs-1cu0bc8t83ha0>

❏ 3.7 镜像

前端工程打镜像：

```
pai image publish -n xxx
```

查看工程运行状态：

```
pai pods -N test -l xxx
```

得到镜像（推送到边端 / 项目环境）：

```
dev.flyrise.cn:8082/pi-dev/pai-xxxxx-ui:latest-20230223141253
```



3.8 用户角色/场景

- 理解P B C端用户

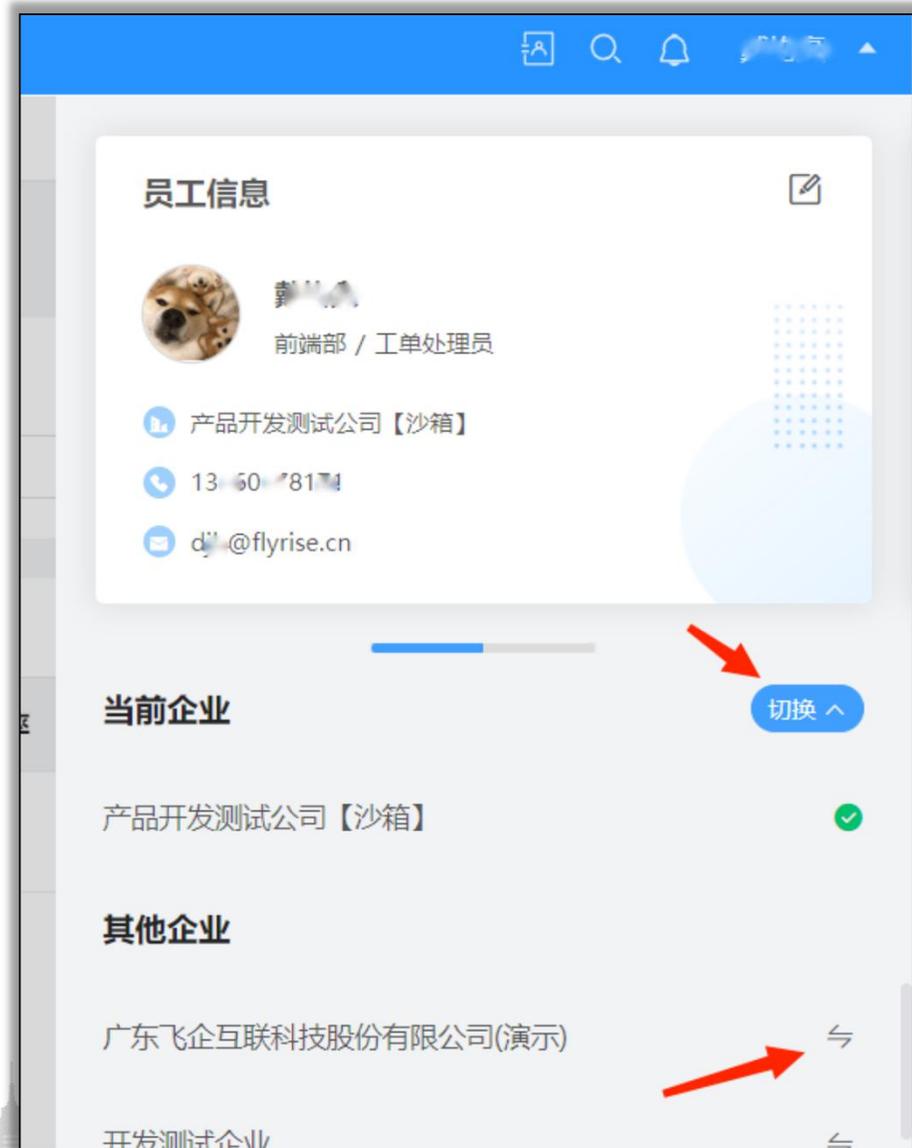
P 园区用户（运营企业）

B 企业用户

C 无企业用户（APP端）

- 切换企业（工作台）

- 切换园区（web端业务切换：sp-ui的园区组件）





04

PART

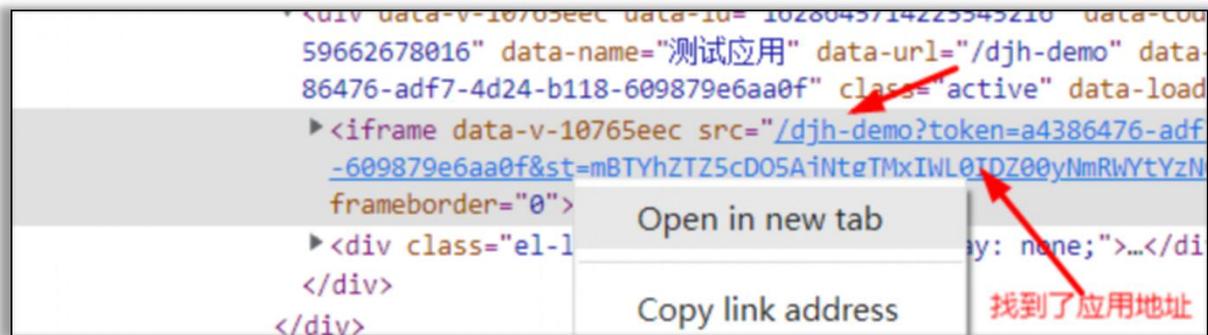
快速定位工程与问题

COURSE TRAINING

4.1 快速定位工程与问题

测试有BUG？团队别人做的功能怎么找？太多工程(微应用)如何定位？

1. 工作台菜单搜索
2. 开发中心-微应用管理
3. 开发中心-页面管理
4. pages.json
5. F12 - iframe地址



```
<div data-v-10765eec data-id= 1028045714225545210 data-coo
59662678016" data-name="测试应用" data-url="/djh-demo" data-
86476-adf7-4d24-b118-609879e6aa0f" class="active" data-load
▶ <iframe data-v-10765eec src="/djh-demo?token=a4386476-adf
-609879e6aa0f&st=mBTYhZTZ5cD05AīNteTMxIWL0TDZ00yNmRwYtYzN
frameborder="0">
▶ <div class="el-l
ay: none;">...</di
</div>
</div>
```

Open in new tab

Copy link address

找到了应用地址



05

PART

代码规范

COURSE TRAINING

5.1 前端开发规范

- 命名规范：

1/ 工程命名 (pai-portal-ui)

2/ 样式命名 (page-header、page_header、page-header_title、ph-icon)

3/ 变量命名 (驼峰式命名 constiableName)

4/ 函数命名 (驼峰式命名，带实际意义)

5/ 文件命名 (file-name、room-detail.vue)

- 前端规范：<http://api.flyrise.cn:9099/docs/open-docs/code-rules>

- 自动化格式插件：[pai-beautify \(https://k8s.flyrise.cn/pai-ui/docs/webpack\)](https://k8s.flyrise.cn/pai-ui/docs/webpack)



5.2 代码提交规范

- 代码提交

IMP: 提升改善正在开发或者已经实现的功能

FIX: 修正BUG

REF: 重构一个功能, 对功能重写

ADD: 添加实现新功能

REM: 删除不需要的文件

- 冲突处理

逐个对比 / 使用远端, 接受传入 / 使用本地, 接受当前

- 注意分支管理 / 切换

Gitea 中台私有仓库: <http://dev.flyrise.cn:8090>



06

PART

组件库

COURSE TRAINING

❖ 6.1 组件库

凌云UI组件库

- pai-ui 文档：示例账号: guest，密码: 123456

<https://k8s.flyrise.cn/pai-ui>

业务UI组件库

- pai-sp-ui 文档：

<https://pai.flyrise.cn/pai-sp-ui>

移动端组件库

- pai-mp-ui





**THANK
YOU**

❖ 课后练习：

1. 创建套件（部门或小组共用）
2. 创建项目并提交仓库
3. * 前端工程连接本地后端服务（协助后端调试）
4. 如何配置oss，使用上传组件
5. 使用参照项（数据字典组件）
6. 使用流水号
7. 部署前端工程到pai环境
8. 发布一个微应用

加入培训开发团队：

<https://pai.flyrise.cn/developer/2.0/invite/4abe55769b8a4433ad7313808c72eedc/7c28868e67df79bbd753eb57377c8106>

加入沙箱企业：

<https://pai.flyrise.cn/console/register/invite/W2JM6Y1V>



该二维码7天内(3月2日前)有效，重新进入将更新



❖ 常用地址

凌云中台开发者文档: <http://api.flyrise.cn:9099>

开发者中心: <http://pai.flyrise.cn/developer>

控制台: <http://pai.flyrise.cn/console>

工作台: <http://pai.flyrise.cn/workbench> (pai 验证码: 默认 1234)

