**前端开发说明书**

**1 需求概要**

根据项目要求，要实现手机端页面开发，包括页面构造，设备适配，接口接入。主要的功能有：登录、显示个人信息、请假、查看请假记录、审批下属请假、查看下属请假记录、查看假期相关规定、登出

**2 技术选择**

本前端项目使用React做为框架，使用Redux保存数据，同时使用Immutable固化数据。使用React-router来实现单页面应用的路由切换，使用ant-design作为基本样式模板。使用webpack作为js库依赖版本控制器和打包工具。此外还用了很多如moment.js这样的小的js库完成一些琐事

**3 具体页面构建及实现**

**3.1 整体结构**

1）单页面应用以组件为单位划分开，整个页面为Base组件，base组件包含leftmenu（左引入菜单），header，footer。

1. leftmenu是一个隐藏在主页面左侧的组件，宽度为主页面的70%，通过base组件的对应的store中的值判断显示还是隐去（使用left：-70%）

3）footer是一个黑色固定高度比（5%）的黑色条状，始终位于页面底部，写有项目github地址。

4）header是一个同样黑色固定在顶部的黑色块状，定高10%，其中包括，用户头像（使用天津大学校徽代替）、用户名（登录前为ant-design的 user Icon）。点击头像触发leftmune的显示操作，container的任意点触都将隐藏leftmenu

5）在footer和header之间是一个85%定高的空区域container，用以填充子组件。header、container、footer使用相对定位（position：ralative），固定在页面的对应位置上。

6）login是base的一部分，由store中的属性控制显示和隐藏，当未登录时，显示login的界面（由ant-design的modal组件修改而成），登录后将用户名保存并显示在header上

7）container包含了几个子组件，当rooter不同是加载不同的组件显示。

**3.2 ask组件**

ask组件是请假和修改请假内容的组件，主要内容为填写请假类别，时间，原因，表单采用了ant-design的form组件，在此基础上加入了一些条件判断，新增了一些动态变化属性，如当选择加班时，要选择调休或双薪的单选按钮，当选择调休时，还要选择调休的日期。ask组件有两个按钮，分别是提交和重置，在提交时，会对已填写的表单项做简单验证，比如日期选择项不可在当前日期之前等。验证符合后，触发action，发送异步请求（本项目所有请求都通过isomorphic-fetch发送异步请求）待请求结果返回，若fail将报错信息显示给用户，若succ则通过modal的方式告诉用户并通过设置react-rooter的pathname实现跳转到请假记录页

**3.3 history组件**

history组件是显示历史请假记录的页面，可以看到历史请假的时间，类型，批复情况，批文等。并添加了搜索框进行模糊匹配。在当前页渲染前，组件的生命周期函数componentWillMount会请求接口获取该用户的请假列表，并将结果保存在store中。页面渲染时判断store中是否已有数据，若有则通过array.map的方式列举并排版显示出来。判断每条请假记录是否已经被批复，如果领导未批复则用户可以看到和点击修改按钮，回到ask组件来修改该次申请。而此时ask组件会获得额外的props以便区分请假和修改的不同操作。

**3.4 me组件**

me组件是个人信息页的组件，构造简单，渲染前获取个人信息接口，获取到后渲染出来。主要的信息有用户名、性别、年龄、电话、婚姻状态、子女数、入职时间

**3.5 audit组件**

audit组件是显示下属请假列表和批复的界面。该界面结构与history组件基本相同，有筛选输入框，可以根据需要筛选记录，每条记录上有申请的类型，时间，同时显示申请状态，若未批复，则有批准按钮和拒绝按钮，同时还有批文框，无论批准和拒绝都可以写批文

**3.6 detail组件**

detail组件是对各种假期的说明，通过菜单或ask组件中的类型边上的问号❓都可以到达该页面。

该页面以文字为主，详细的显示了请假的各种要求和加班调休或请求双薪的规定。

**4 API管理**

项目将api统一管理在/src/api/下，每个组件有独立的文件夹来存写接口，以便日后维护更换url。个人文件都引入同目录下的index.js 在这个文件中配置了host目录，方便域名变化或本地调试。

**5 项目打包和发布**

在项目根目录下有webpack的配置文件package.json，在其中已经在sricpt部分写好了一些指令，同过terminal的npm来运行这些指令，如需要重新打包时 运行 npm run build，即可根据新的代码生成一份新的静态文件。其中，为了方便将打包后代码发布，项目根目录外放置了另一个git项目的文件夹class-design， 当执行npm run build 的时候会自动用新生成的dist文件夹中的静态资源包替换class-design中的旧资源包。在此之后只需将class-design给push到仓库，再从阿里云服务器根目录下的clone代码pull最新的修改，项目上线就完成了。

**6 本地调试**

在类unix系统下，需先安装npm，进入项目frontend根目录，运行 npm install加载依赖的包文件，具体时间视网速决定。待加载完成后即可运行npm run start，当terminal中显示 webpack built xxxxxxxxxx（一串MD5） in xxxms（一个毫秒数）的字样后，可通过浏览器http://localhost:5555访问。请尽量使用chrome浏览器的近新版本。