

《移动应用软件开发》期末软件设计报告

年级、专业、班级	2021 级计卓 1 班			姓名	李宽宇	学号	20215279
提交时间	2023. 12. 10	2023 学年 第二学期	2023-2024（1）		指导教师	杨瑞龙	
软件名称	大学迎新 APP						
<p>说明：</p> <p>1. 每人独立完成移动应用软件设计开发，并完成设计报告。</p> <p>2. 自拟软件题目。不准抄袭，应当是自己独立编写的软件。抄袭计 0 分。</p> <p>3. 软件开发平台选择 HarmonyOS，优先使用 ArkUI 开发。</p> <p>4. 从界面、技术、功能等方面对软件进行评价。程序有必要的注释。</p> <p>5. 从文档条理性、文档规范性、内容完整性等方面对设计报告进行评价。</p> <p>6. 所开发的程序源代码压缩成 ZIP 格式，命名为：学号姓名-期末软件设计.zip。报告命名为：学号姓名-期末软件设计报告.docx。演示视频清晰，MP4 格式，大小不超过 20M，命名为：学号姓名-期末软件设计演示.mp4。请在谷歌或者搜狗浏览器预览视频是否能够正常播放。三个文件分别提交到云班课。</p>							
<p>报告正文</p> <p>注：格式要规范，正文汉字一律使用宋体小四，文字行间距使用 1.5 倍行距。段首缩进 2 个汉字，图片居中，每个图片在底部有标题和编号。表格上面有表格标题和编号。版面漂亮、整洁、统一。不超过 20 页。</p>							
<p>一、软件设计意图、功能介绍和特色</p> <p>1.设计意图:</p> <p>学校迎新 APP 的设计意图主要是为了帮助新生更好地适应校园生活，提供各种便利的功能和服务。以下是一些可能的设计目标：</p> <p>（1）信息获取：新生可以通过 APP 获取学校的基本信息，如校园地图、教学楼位置等。</p> <p>（2）报到流程：新生可以通过 APP 获取当前报到进度，如宿舍入住办</p>							

理、学籍注册、物资领用、证件办理、体检、领取医保卡。

(3) 报到指南：新生可以通过 APP 获取报到的详细信息，如某过程的详细时间，地点。

(4) 校园建筑介绍：新生可以通过 APP 获取校园建筑的位置、用途、历史渊源。

(5) 校园文化介绍：新生可以通过 APP 学习校歌、宣传视频，来了解学校。

2.功能介绍：

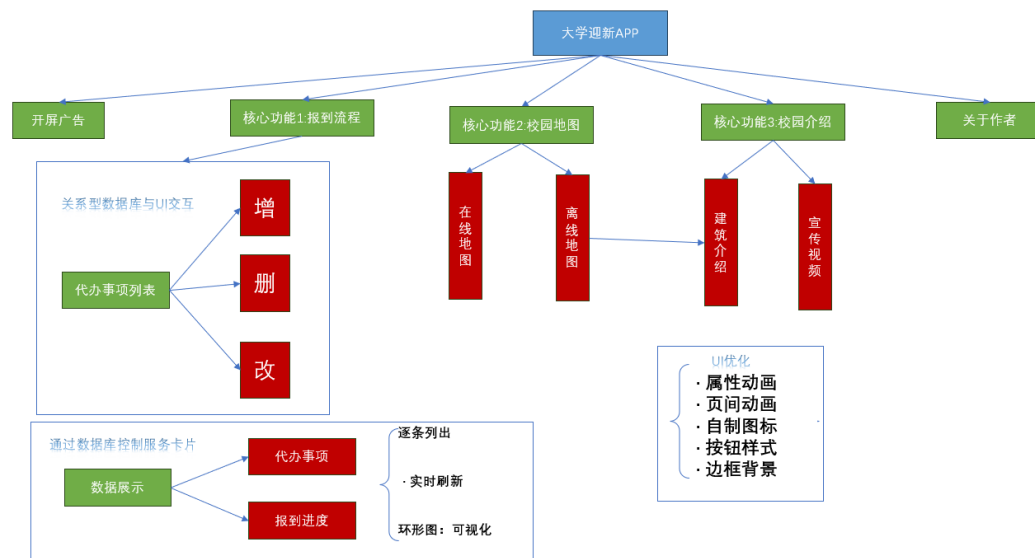


图 1：学校迎新 APP 功能介绍图

学校迎新 APP 主要承担协助学生报到和熟悉校园的功能。针对新生报到，实现报到流程提示的功能、校园地图的功能；针对熟悉校园，实现了校园地图、校园介绍的功能。

3.特色

技术方面：服务卡片、离线地图绘制、在线地图访问、视频播放、动画效果；

设计方面：解决了传统报到单与地图分离的问题，使得新生在不了解校园地理的情况轻松报到；解决了网络质量差时，无法看在线地图的问题，离线地图附带详细的介绍，帮助同学快速了解校园。

进一步而言，未来报到功能由服务端统一发布，及时更新服务卡片，将解决纸质报到单的不灵活、消息不能及时触及新生的问题。

二、开发和运行环境（可能包括开发平台、运行平台、开发工具、第三方组件使用情况、其他支撑软件运行的条件等）

开发平台：Deveco 3.1

运行平台：Api9 的本地鸿蒙手机模拟器

开发工具：Arkts

第三方组件：百度地图 <https://map.baidu.com/>

三、设计说明（写明设计思想、程序的结构、功能设计、界面设计、模型设计、程序主要执行流程图，最后是核心源代码，截图等）

1. 设计思想

学校迎新 APP 主要承担协助学生报到和熟悉校园的功能，将地图和报到流程整合起来，极大的方便了新生的使用。针对新生报到，实现报到流程提示的功能、校园地图的功能；针对熟悉校园，实现了校园地图、校园介绍的功能。

用户体验：设计直观、易用的用户界面，使用户能够轻松地理解和操作。数据可视化以帮助人们更直观、更易于理解地理解数据背后的含义和关系。在当今信息爆炸的时代，大量的数据被不断产生和积累，而数据可视化作为一种强大的工具，对于理解复杂数据、发现趋势、交互分析以及支持决策制定都具有重要的意义。将地图和报到流程整合起来，极大的方便了新生的使

用；服务卡片方便快捷，无需打开 App，开屏即用。

2. 程序的结构

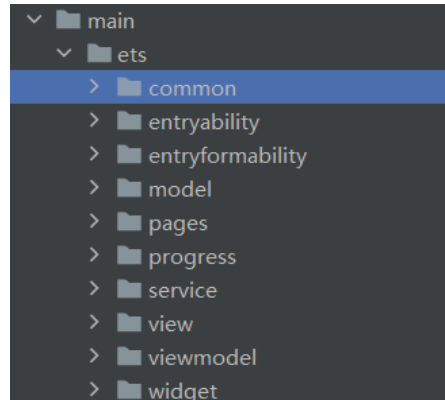


图 2：源代码的结构

Common 路径下定义了常量、数据库及其工具，其中 formutil 用于更新服务卡片，accoutTabele.ets 封装了对 rdb 表格报到流程的 aduq 操作。

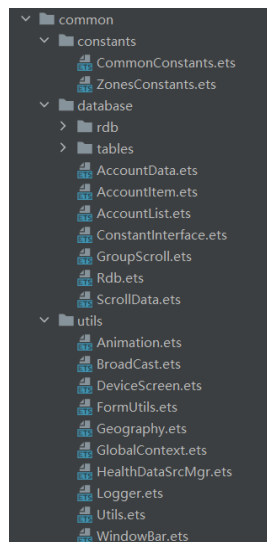


图 3：源代码的结构

开发充分利用了面向对象的特性，将页面中的组件抽象出来，通过组件的拼接实现页面的结构

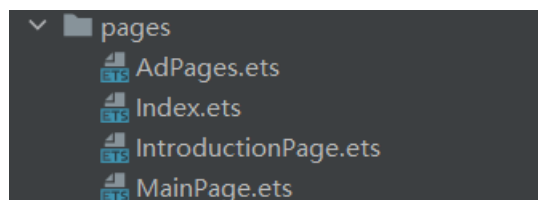


图 4：源代码的结构

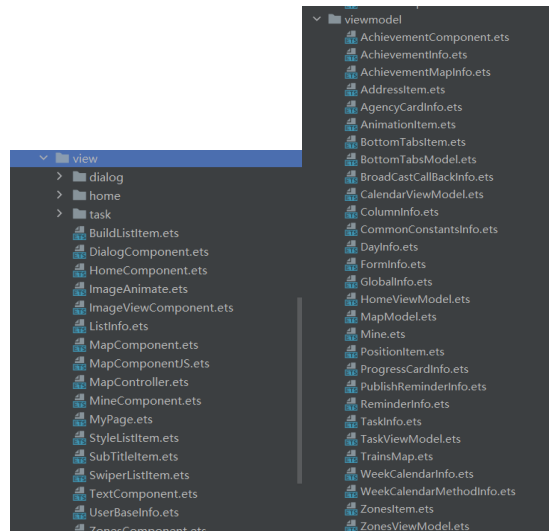


图 5：源代码的结构

3. 功能设计

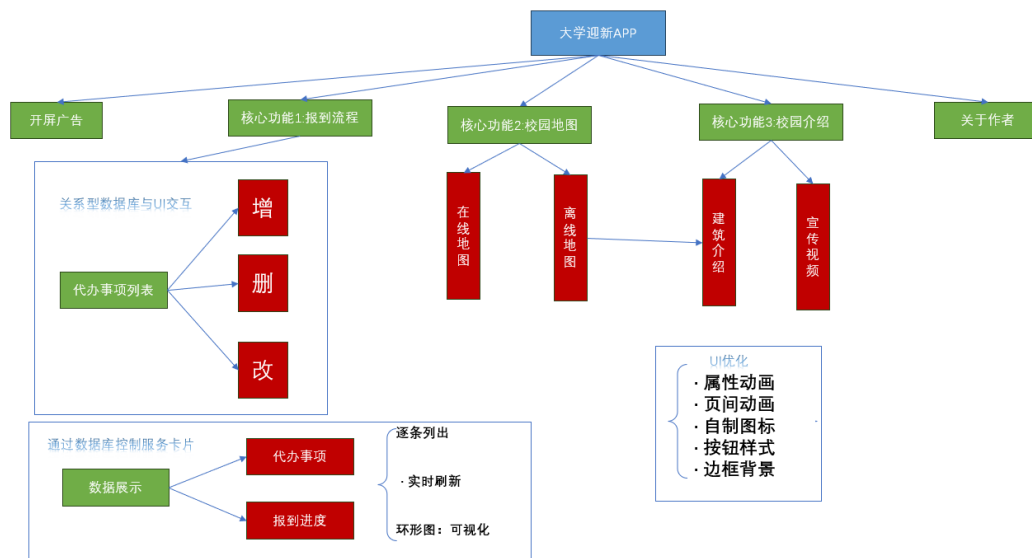


图 6：学校迎新 APP 功能介绍图

学校迎新 APP 主要承担协助学生报到和熟悉校园的功能。针对新生报到，实现报到流程提示的功能、校园地图的功能；针对熟悉校园，实现了校园地图、校园介绍的功能。

4. 界面设计

UI 设计直观、易用的用户界面，使用户能够轻松地理解和操作。融合学校的特色元素，如校徽、校训；

动画流畅，页面跳转自然优美；

服务卡片方便快捷，无需打开 App，开屏即用，包括列表和环形图。

5. 程序主要执行流程图

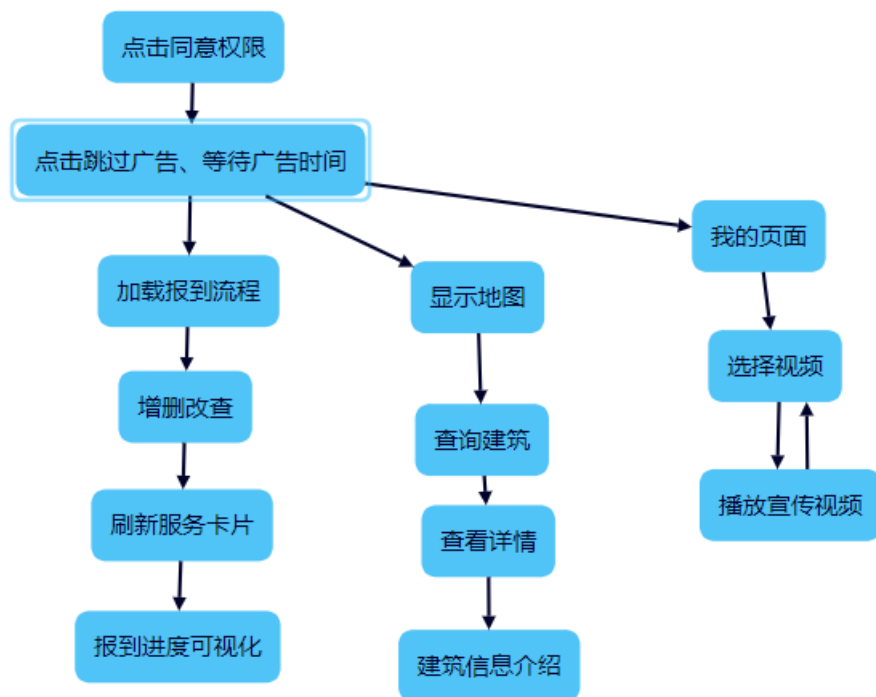
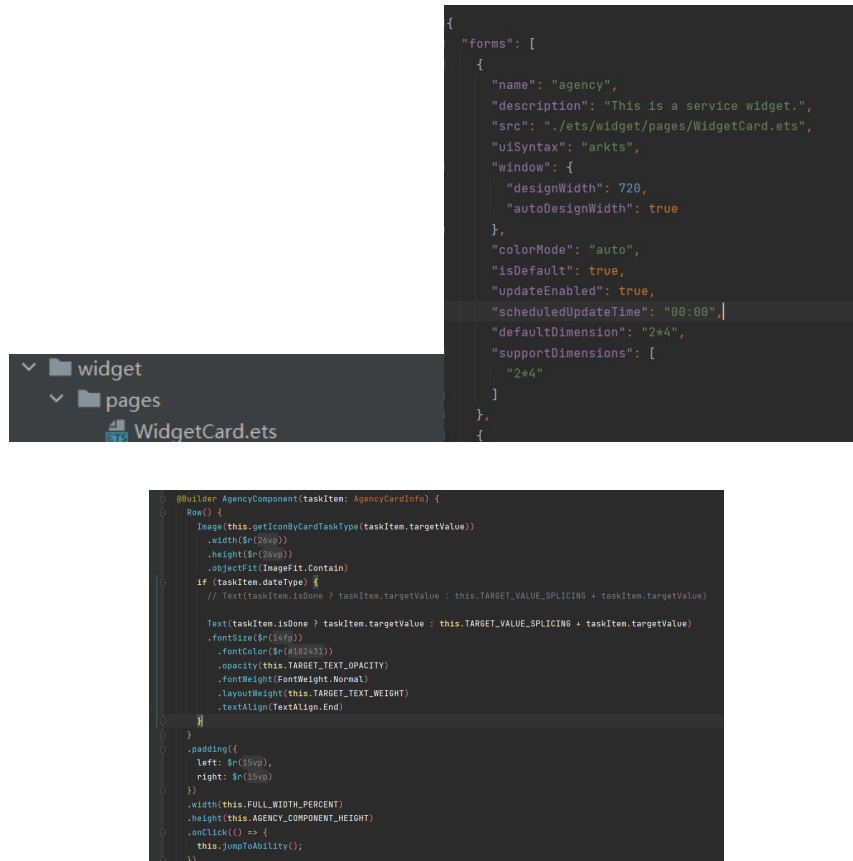


图 7：程序主要执行流程图

6. 核心源代码

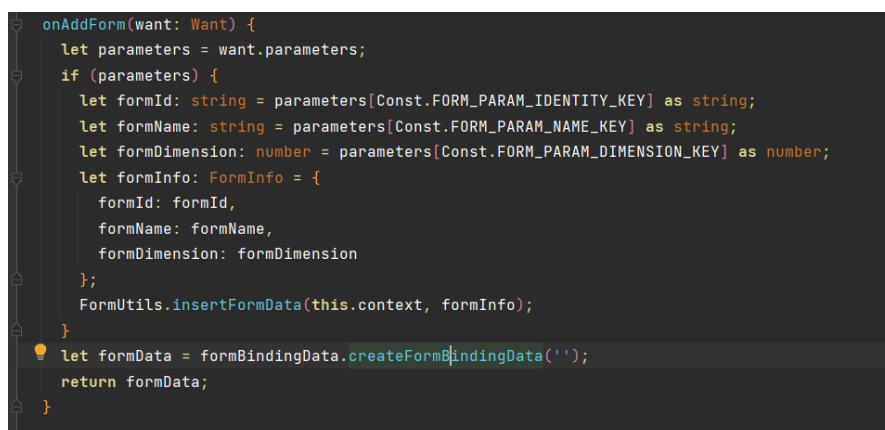
核心源代码主要包括服务卡片的实现，服务卡片的刷新，在线地图访问，离线地图实现，页面间切换的动画，视频播放等关键技术

(1) 服务卡片的实现（列表卡片），在 form_config.json 中注册



(2) 服务卡片的刷新

首先 entryFormAbility.ets 获取卡片的标识 formId，交给 FormUtils 存储 formId



Formutil.ets 中，如果切到后台（服务卡片显示出来），调用下面的函数，检查全局上下文变量 taskListChange，如果改变，就查询当前的报到数

数据库，刷新卡片。

```
public backgroundUpdateCard(taskListChange: boolean): void {
    Logger.info(Const.RDB_TAG, '进入后台循环查询')
    if (taskListChange) {
        Logger.info(Const.RDB_TAG, '进入后台循环查询，列表更新')
        GlobalContext.getContext().setObject('taskListChange', false);
        let timeId = setTimeout(() => {
            this.queryForms();
            clearInterval(timeId);
        }, Const.TIMES_100);
    }
}
```

因此，只需要在插入、删除数据时将全局上下文变量 taskListChange 设为 true，就可以实现刷新卡片的功能。

```
accept(isInsert: boolean, newAccount: AccountData): void {
    Logger.info(CommonConstants.RDB_TAG, 'insertData() accept');
    if (isInsert) {
        Logger.info(`${CommonConstants.INDEX_TAG}`, 'The account inserted is: ${JSON.stringify(newAccount)}`);
        AccountTable.insertData(newAccount, (id: number) => {
            newAccount.id = id;
            this.accounts.push(newAccount);
            this.accountData2ScrollData(this.accounts);
            GlobalContext.getContext().setObject('taskListChange', true);
            Logger.info(CommonConstants.RDB_TAG, '修改了taskListChange为真');
        });
    } else {
        AccountTable.updateData(newAccount, () => {
        });
        let list = this.accounts;
        this.accounts = [];
        list[this.index] = newAccount;
        this.accounts = list;
        this.index = -1;
        this.accountData2ScrollData(this.accounts);
        GlobalContext.getContext().setObject('taskListChange', true);
    }
}
```

(3) 在线地图访问

```
import web_webview from '@ohos.web.webview'

web_webview.once("webInitied", () => {
    console.log("setCookie")
    web_webview.WebCookieManager.setCookie("https://map.baidu.com/@11837574.16400545,3429243.1373088,15z", "a=b")
})

@Preview
@Component
export struct MapComponentJS {
    controller: web_webview.WebviewController = new web_webview.WebviewController();
    build() {
        Column() {
            Web({ src: 'https://map.baidu.com/@11837574.16400545,3429243.1373088,15z', controller: this.controller })
        }
    }
}
```


(4) 页面间切换的动画，TOP 代表从下到上

type 为 RouteType.None 表示对页面栈的 push、pop 操作均生效，type 的默认值为 RouteType.None。

```
pageTransition() {
  PageTransitionEnter({ duration: Const.SHARED_DURATION })
    .slide(SlideEffect.Top)
  PageTransitionExit({ delay: Const.EXIT_DELAY })
    .opacity(0)
}
```

(5) 点击按钮，显示在地图上相应位置

```
Image(this.data.icon)
  .width(Const.MAP_LANDMARKS_SIZE)
  .height(Const.MAP_LANDMARKS_SIZE)
  .offset({
    x: item.positionX,
    y: item.positionY
  })
```

(6) 视频播放

```
Video({
  src: this.videoSrc,
  previewUri: this.previewUri,
  currentProgressRate: this.curRate,
  controller: this.controller
})
  .width('90%')
  .height(210)
  .autoplay(this.isAutoPlay)
  .controls(this.showControls)
  .onStart(() => {
    console.info('onStart')
  })
  .onPause(() => {
    console.info('onPause')
  })
  .onFinish(() => {
    console.info('onFinish')
  })
  .onError(() => {
    console.info('onError')
  })
  .onPrepared((e) => {
    console.info('onPrepared is ' + e.duration)
  })
  .onSeeking((e) => {
    console.info('onSeeking is ' + e.time)
  })
  .onSeeked((e) => {
    console.info('onSeeked is ' + e.time)
  })
```

```
Row() {
  Button('下一条视频').onClick(() => {
    if (this.viedoNum == 0) {
      this.videoSrc = $rawfile('bbb.mp4') // 切换视频源
      this.previewUri = $r('app.media.ic_B')
      this.viedoNum = 1;
    }
    else {
      this.videoSrc = $rawfile('aaaa.mp4') // 切换视频源
      this.previewUri = $r('app.media.ic_A')
      this.viedoNum = 0;
    }
  }).margin(5).width(120).backgroundColor(Color.White).fontColor(Color.Black)
  Button('显示控制栏').onClick(() => {
    this.showControls = !this.showControls // 切换是否显示视频控制栏
  }).margin(5).width(120).backgroundColor(Color.White).fontColor(Color.Black)
}
}.height(Const.FULL_PERCENT)
.backgroundColor(Color.White)
```

四、源程序调试过程（运行、调试截图和文字）

1. 界面截图

(1) 初始界面和广告



图 8：初始界面和广告

(2) 报到列表，点击条目可以查看和修改，



图 9：报到列表

(3) 点击加号，选择自定义事项，具体可以写在备注里，可以看到已经添加。



图 10：自定义事项

(4) 服务卡片，报到进度环和当前剩余事项列表



图 11：服务卡片

(5) 侧滑体检一栏，点击完成



图 12：完成事项

(6) 点击离线地图，点击第一教学楼

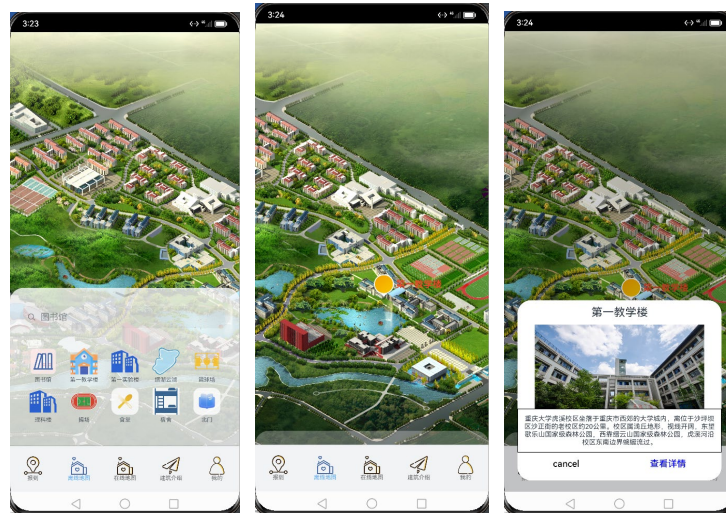


图 13: 离线地图

(7) 点击查看详情

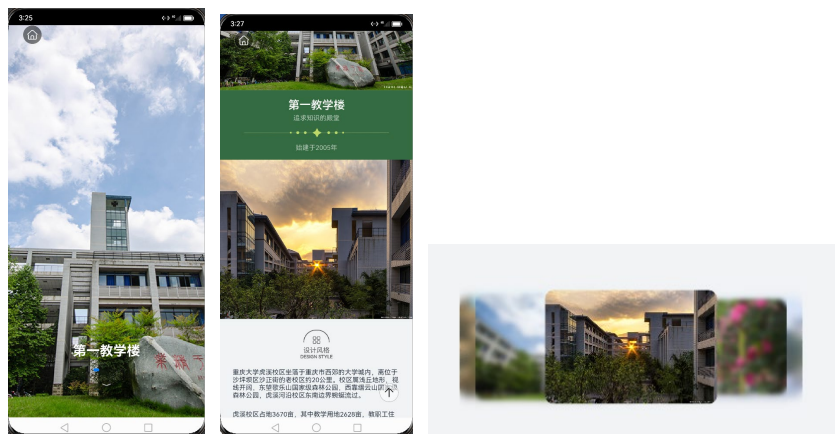


图 14: 离线地图

(8) 在线地图，访问百度地图网站

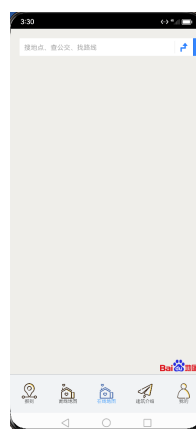


图 15: 在线地图

(9) 建筑介绍，可以侧滑

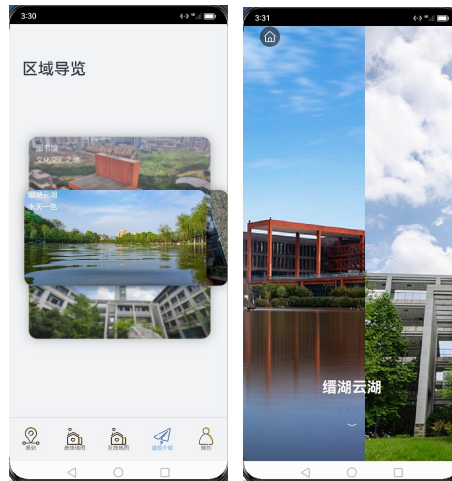


图 16：建筑介绍

(10) 我的页面，底部是重庆大学宣传视频，点击可以播放、切换，视频来源哔哩哔哩



图 17：我的和视频播放

五、总结及分析（完成过程中的心得体会，成功与失败之处，经验，收获，建议、将来计划和展望）

1. 心得体会

（1）服务卡片的实现

首先 entryFormAbility.ets 获取卡片的标识 formId，交给 FormUtils 存储 formId（数据库存储）Formutil.ets 中，如果切到后台（服务卡片显示出来），调用下面的函数，检查全局上下文变量 taskListChange，如果改变，就查询当前的报到数据库，刷新卡片。因此，只需要在插入、删除数据时将全局上下文变量 taskListChange 设为 true，就可以实现刷新卡片的功能。

（2）动画设置

```
pageTransition() {
  PageTransitionEnter({ duration: Const.SHARED_DURATION })
    .slide(SlideEffect.Top)
  PageTransitionExit({ delay: Const.EXIT_DELAY })
    .opacity(0)
}
```

2. 成功之处

（1）技术方面：服务卡片、离线地图绘制、在线地图访问、关系型数据库、视频播放、动画效果、安装了中文输入法；

（2）设计方面：

解决了传统报到单与地图分离的问题，使得新生在不了解校园地理的情况轻松报到；

解决了网络质量差时，无法看在线地图的问题，离线地图附带详细的介绍，帮助同学快速了解校园。

宣传文章详细真实，设计元素有学校特色。

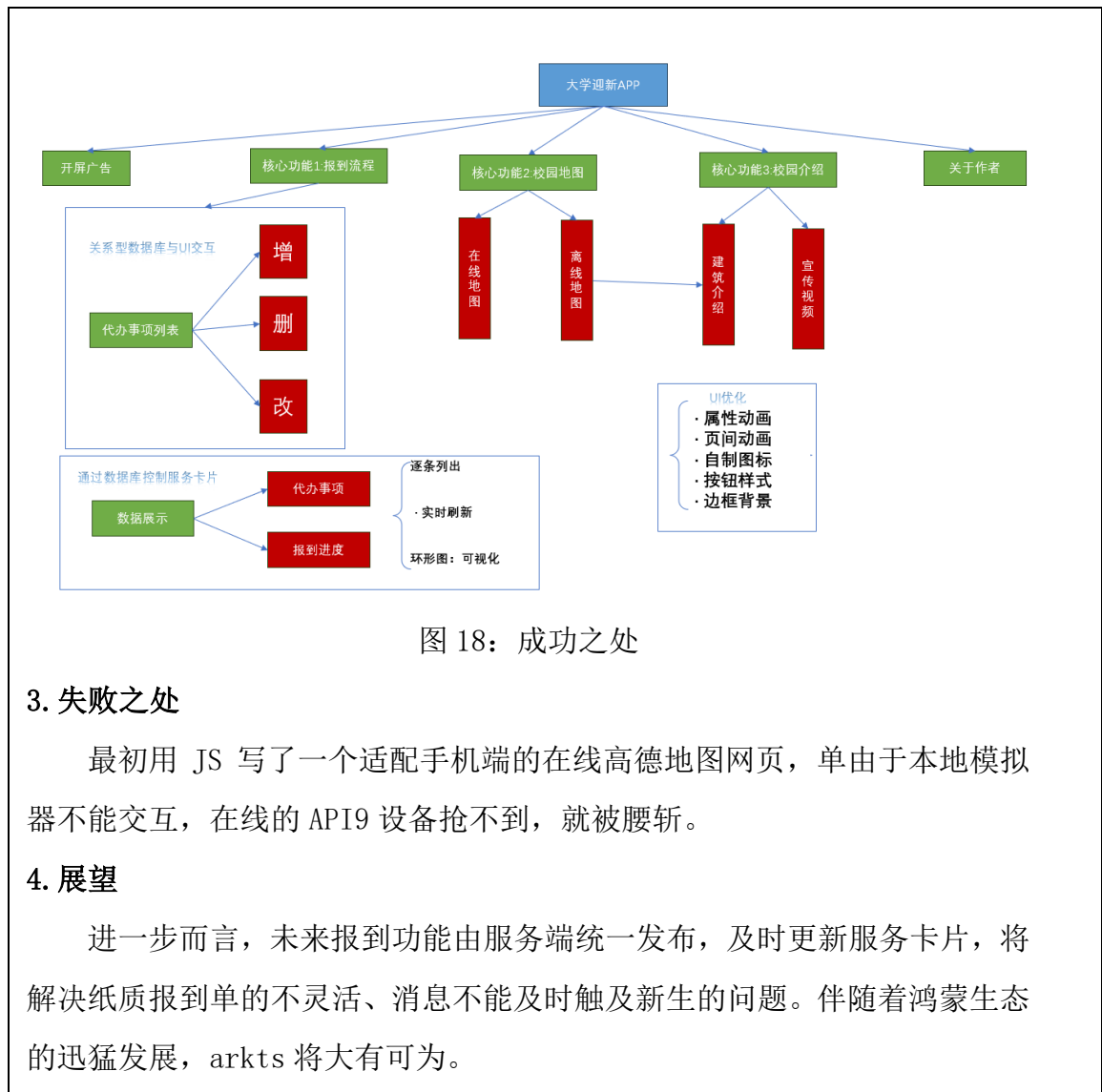


图 18: 成功之处

3. 失败之处

最初用 JS 写了一个适配手机端的在线高德地图网页，单由于本地模拟器不能交互，在线的 API9 设备抢不到，就被腰斩。

4. 展望

进一步而言，未来报到功能由服务端统一发布，及时更新服务卡片，将解决纸质报到单的不灵活、消息不能及时触及新生的问题。伴随着鸿蒙生态的迅猛发展，arkts 将大有可为。