



基于 VR 技术的口语能力训练移动应用 “VR 口吃治疗与训练” APP 项目完成情况文档

项目成员：罗佳海 20172131144

吴梓祺 20172131096

李佳蔚 20172131057

谢礼冬 20172131061

目录

一、	设计需求分析	5
(一)	口吃本身	5
(二)	政策方面	5
(三)	市场方面	5
(四)	言语治疗师的角度	6
(五)	口吃患者的角度	6
二、	项目难点及解决方案详述	6
(一)	Unity 3D 构建 VR 场景	7
(二)	手机 VR 框架和实现用户与应用交互	9
(三)	语音评测和语音合成的封装	9
(四)	Unity 项目和 Android 项目的兼容和合并	10
(五)	Java 和 Kotlin 混合编程	12
三、	创新与特色	13
(一)	创新	13
1.	手机显示模式创新	13
2.	社会创新	14
3.	交互形式的创新	14
4.	训练内容的创新	14
5.	训练内容的创新	14
(二)	特色	14
四、	详细实现过程	15
(一)	Unity 3D 方面	16
1.	3D 人物建模	16
2.	3D 人物动画和物体建模	17
3.	3D 人物动画控制	18
4.	手机 VR 框架	20
5.	Unity 3D 调用讯飞语音评测功能	24
6.	Unity 声音控制和光照渲染	25
(二)	Android Studio 方面	26
1.	启动页延时动画	26
2.	自定义 ViewPager+Fragment	26
3.	自定义顶部栏 TopBar	27
4.	轮播图 RecyclerView	27
5.	登录对话框自定义 Dialog	28
6.	侧边导航栏 Material Drawer	28
7.	按钮水波纹	29
8.	全面屏适应	29
9.	可伸缩抽屉 ExpandingDrawerView	30
10.	可伸缩卡片 ExpandingCardView	30
11.	进度展示自定义 CardView	31
12.	堆栈式卡片 CardStackView	31
13.	可滑动的多图表 CombinedChartView	32

14.	可滑动的圆饼图表 PieChartView.....	32
15.	科大讯飞语音模块.....	33
16.	动态权限申请.....	34
17.	Android 和 Unity 3D 交互.....	34
18.	阿里云云数据库 Utils 类.....	34
19.	UI 设计和动效.....	34
(三)	云数据库方面	35
1.	订阅数据库服务.....	35
2.	数据库 CURD 范例.....	35
五、	开发环境.....	36
(一)	开发环境.....	36
六、	软件测试与软件完成情况基本展示	37
(一)	测试方案.....	37
1.	脚本兼容性测试.....	37
2.	深度兼容测试	39
3.	深度性能测试	40
(二)	测试结果分析	42
1.	脚本兼容性分析.....	42
2.	深度性能分析	42
3.	人工测试分析	42
(三)	用户意见和评价	42
(四)	软件完成情况基本展示.....	44
七、	项目分工.....	53
(一)	项目分工.....	53
	参考资料.....	54

“VR 口吃治疗与训练” APP 项目完成情况

摘要

言语障碍治疗在国内尚处于起步阶段，相关的探索仍较少。另外，我国在口吃领域暂无市场监管，市场准入门槛低，大量私人机构孕育而生。患者忙中乱投医，导致口吃治疗效果不佳。而二三线城市的口吃患者去大城市的治疗成本高（时间，金钱），加上口吃的反复几率至多 30%^[1]，难以达到满意的治疗效果。

VR+医疗在近几年发展很快，尤其是在自闭症、抑郁症的治疗上已有广泛的应用。但 VR 在口吃治疗上仍没有先例，值得研究者开拓。口吃所属的言语障碍治疗在大陆尚属于起步阶段，国内还没有言语治疗师的资格考试。而我国患有口吃的人数（严重度不等）比例约为 1%-2%，市场上有大量需求。

项目拟开发一款基于移动端的 VR 应用^[2]，定位于口语基础训练和 VR 虚拟场景训练。为成人口吃患者提供演讲、面试等场景的康复训练，提高其重要情境下的言语流畅度和自信心。并提供字词句的复述训练，给出语音评测分数，帮助口吃患者基础构音上有所提升。

本项目可主要分为三个功能块——“VR 实验室”、“口语卡片”、“进度统计”。目前，项目已经完成“VR 实验室”和“口语卡片”功能的基本框架，实现了这两个功能的基本使用，并且将 Unity3D 项目与 Android 项目做了打包，可以共同运行。开发方面，使用了 Unity 3D（C#）构建 VR 场景，3D MAX、DAZ Studio、Blender 进行物体建模；使用了 Android Studio（Java 和 Kotlin 混合）进行移动应用的开发。和数据库方面，本项目采用了阿里云的云数据库进行数据存储。语音评测方面使用了科大讯飞的 SDK，采用了其中语音评测和语音合成两个模块。软件测试使用了百度 MTC 移动 App 测试，分别进行了脚本兼容性测试、深度兼容测试、深度性能测试等。同时，我们也在线下寻找了数十名同学进行软件使用体验，并将反馈的问题进行记录和改进。最后，我们进行了项目当前进展的总结与回顾。

一、 设计需求分析

（一）口吃本身

口吃是一种言语流畅性障碍,约占全人类总人数 1%(约 7000 万),在我国比例介于 1%~2% (约 1500 万),其中儿童口吃的发生率约为 5%。同时口吃病有较高的反复性, 复发率至多 30%。这使得口吃患者的内心焦虑程度高于常人,而心理压力本身又会加重口吃症状,形成恶性循环。然而,由于口吃成因复杂(遗传因素、生理和心理共同作用、神经损伤),严重程度不一(分为三级),我国起步较晚(约 20 年),口吃的治疗流程还很不完善。

（二）政策方面

党的十八大报告强调指出“注重人文关怀和心理疏导,培育自尊自信、理性平和、积极向上的社会心态”。党的十九大报告也明确提出“加强社会心理服务体系建设,培育自尊自信、理性平和、积极向上的社会心态”。我国目前没有言语治疗师的认证制度,并且,残疾评定中的言语残疾基本不包括口吃(除了极为严重的失语),口吃患者成为我国数量相当庞大的隐形人,影响到社会的和谐稳定和理性平和。

（三）市场方面

理论上口吃患者的正规就医途径如下图。然而我国的综合医院仅一线城市三甲医院的康复科会接诊口吃患者,康复医院和社区医院完全不收纳口吃患者。此外,我国在口吃方面没有任何市场监管,市场准入门槛低,大量私人机构孕育而生。患者忙中乱投医,导致口吃治疗效果不佳。二三线城市的口吃患者想要得到专业治疗只能去大城市,患者也难以达到短期内完全康复的目的。同时,由于治疗过程漫长,加上口吃的反复几率大,需要花费口吃患者大量的时间和金钱。



图 1 口吃患者正规就医途径

（四）言语治疗师的角度

到 2015 年，正式获得康复认证的治疗师大约为 1.4 万人。即使是发达的上海，物理治疗师加上作业治疗师合计也仅仅为 3.28 个/10 万人口（2009 年日本为 42 人/10 万人口）。我国相关专业起步晚，言语治疗从业人员学历普遍偏低，国内言语治疗从业人员专业多样，以康复治疗学专业为主，无培训上岗所占比例较高，年参加培训不多，非医疗系统更明显，取得的证书较杂。然而，医疗系统中康复治疗师为主，非医疗系统中教师为主，故无论在医疗系统还是非医疗系统中的言语从业人员资质都存在很大的缺陷。

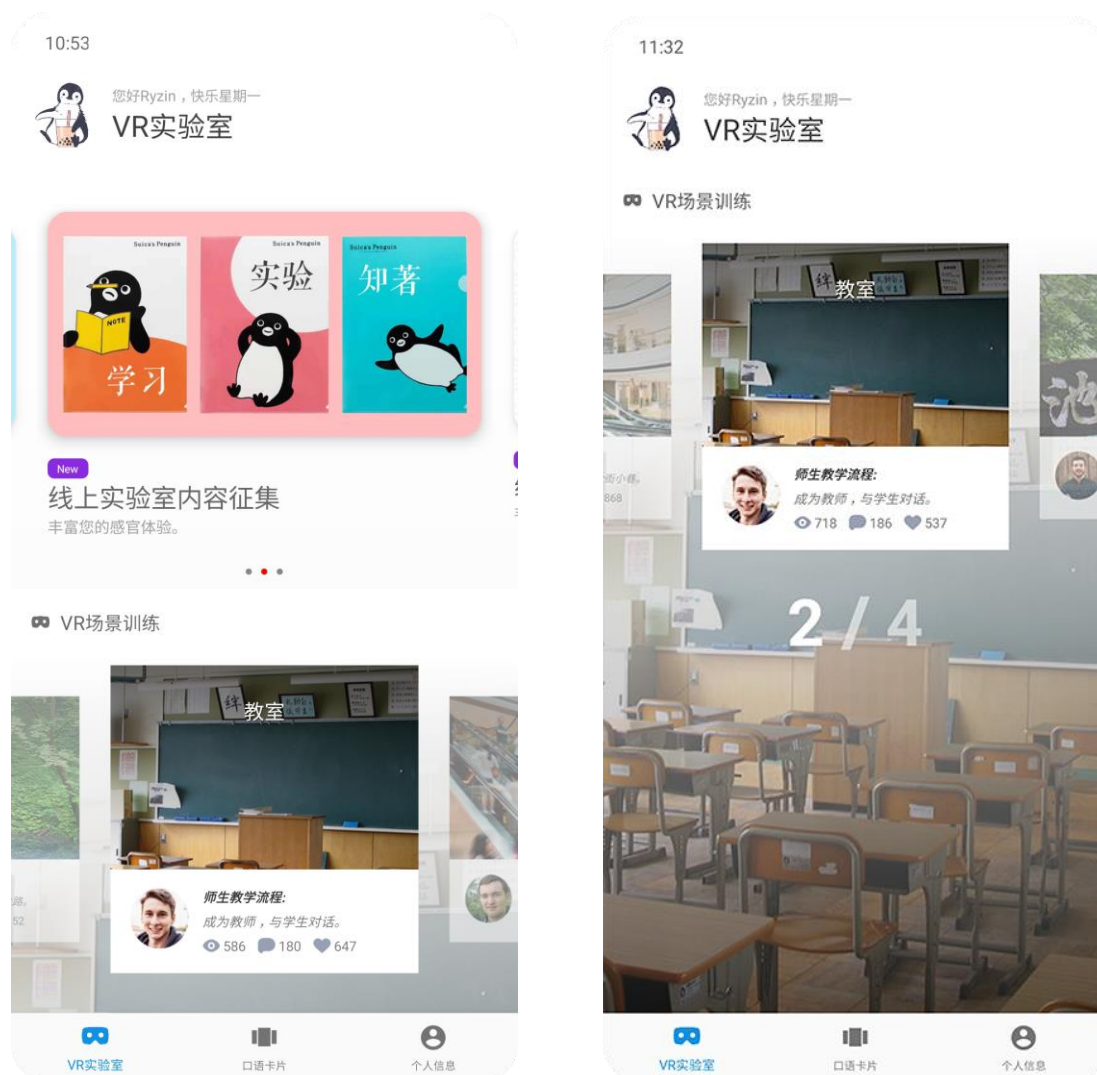
（五）口吃患者的角度

口吃会影响患者对现实感知，即患者对自己口吃的自我认知与现实不符。患者在受到外界的有色眼镜看待后，容易导致出现自卑、逃避、抑郁等症状。以至于口吃患者在正常融入交际圈会有一定难度，对升学，晋升，交友，婚姻等造成一定的困难。因此他们急需可帮助他们治疗口吃的工具。同时，不仅仅是口吃患者，口吃也会对患者家人的心理健康产生长期甚至毕生的影响。所以解决口吃患者的心理问题刻不容缓。

二、 项目难点及解决方案详述

项目拟开发一款基于移动端的 VR 应用，为成人口吃患者提供自主康复训练。口吃患者受心理因素影响，在不同情境下表现出不同的口吃严重度。综合 Ezrati-Vinacour 与 Levin(2004)，Messenger、Onslow、Packman 与 Menzies(2004)的两篇研究，明显看出成人口吃患者的焦虑与情境因素有关，尤其是与说话相关的情境最为明显。项目旨在通过对演讲、面试等场景的 VR 建模，为成人口吃患者提供沉浸式的说话训练，并结合语音识别、语音评

测的技术，创造交互性较强的沟通环境，提高训练的真实度。

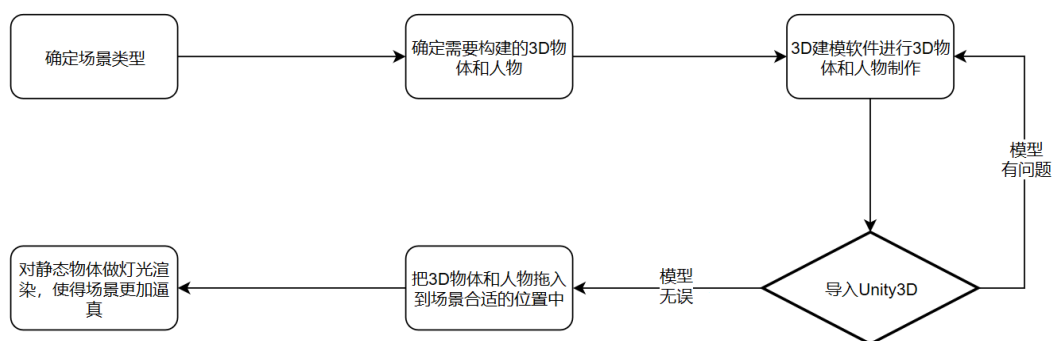


APP 中“VR 实验室”页面

（一）Unity 3D 构建 VR 场景

而构建 VR 场景的步骤如下：首先确定场景类型，从而确定需要构建的 3D 物体和人物，然后通过 3D Max、DAZ Studio、Blender 等热门 3D 建模软件进行 3D 物体和人物的制作，制作完成后以 FBX 格式导出，再导入 Unity 3D 中，并把物体和人物拖动到场景里面，最后对场景里面的静态物体做灯光渲染，使得场景更加真实。

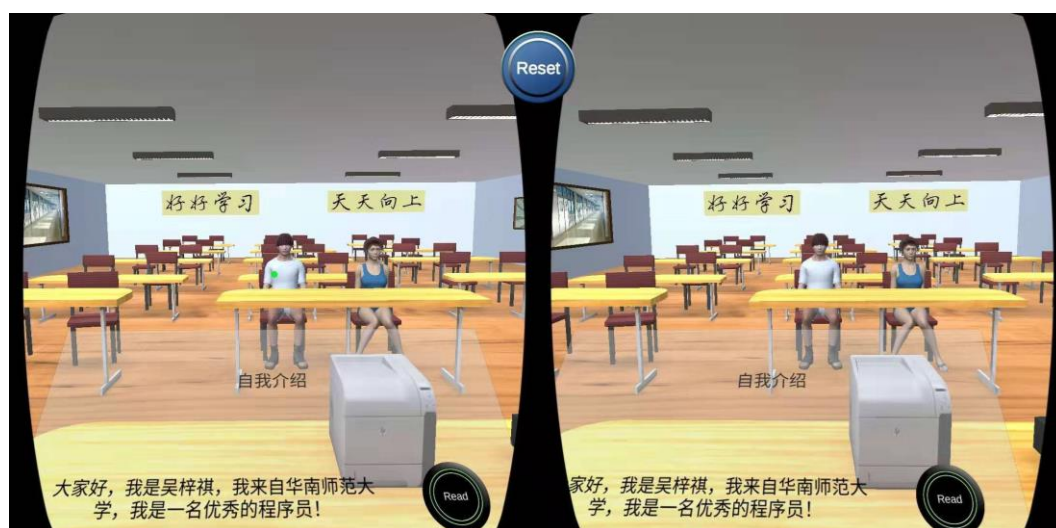
流程如下图：



场景类型细分为室内、室外，室内又分为各种各样的场景，当选择好了场景类型后，之后确定要构建的 3D 物体和人物必须符合场景类型。我们创建的场景是课室，需要的物体就应该是课桌椅、黑板、投影仪等等，需要的人体模型也必须符合学生形象。然而我们并没有 3D 建模的基础。

Unity 3D 支持的 3D 模型格式比较有限，只有 FBX 格式，而且随着 Unity 3D 版本更新，可能会出现一些对旧版本的 FBX 不兼容的情况，最典型的就是当 3D 模型导入到 Unity 3D 后，出现材质丢失的问题，这时只能回到 3D 建模软件检查错误了。

总而言之，Unity 3D 构建 VR 场景的难点有：制作 3D 模型难度较大、Unity 3D 对 3D 模型兼容性比较差。



VR 教室场景



VR 卧室场景

解决方案：寻找网上 3D 模型资源，其中课室场景资源在 Unity Asset Store 上面找到了免费可用的模型；而人体模型资源网上比较难找到，所以我们使用了 Daz Studio 软件制作人体模型，这个软件是一个傻瓜式的捏脸工具，可以高效地构建出人体模型。

（二）手机 VR 框架和实现用户与应用交互

我们制作手机 VR 框架是参考了 HTC Vive VR 设备的，HTC Vive 是由 HTC 与 Valve 联合开发的一款 VR 头显（虚拟现实头戴式显示器）产品，于 2015 年 3 月在 MWC2015 上发布。由于有 Valve 的 SteamVR 提供的技术支持，因此在 Steam 平台上已经可以体验利用 Vive 功能的虚拟现实游戏。

HTC Vive 通过以下三个部分致力于给使用者提供沉浸式体验：一个头戴式显示器、两个单手持控制器、一个能于空间内同时追踪显示器与控制器的定位系统（Lighthouse）。

我们制作手机端的 VR 框架在硬件上就失去了手持控制器和追踪定位系统支持，手机 VR 只能当作一个头戴式显示器，很难实现像 Vive 那样丰富的交互手段，而且由于手机性能参差不齐，也很难运行对设备性能有较高要求的 VR 游戏。所以我们主要利用了手机的陀螺仪和麦克风功能，来实现用户和应用的交互。

（三）语音评测和语音合成的封装

科大讯飞语音评测模块，通过智能语音技术自动对发音水平进行评价、发音错误、缺陷进行定位和问题分析。目前语音评测提供汉语、英语两种语言的评测，支持单字（汉语专有）、

词语 和句子朗读三种题型。通过解析 SDK 返回的结果，得到评测结果，音素（phone），音节，各音节时长等内容，可以综合这些内容给用户相应的训练结果与建议。

```
public class Phone {
    /**
     * 讯飞音标-标准音标映射表 (en)
     */
    public static HashMap<String, String> phone_map = new HashMap<>();

    static {
        phone_map.put("aa", "ɑ:");
        phone_map.put("oo", "ɔ");
        phone_map.put("ae", "æ");
        phone_map.put("ah", "ʌ");
        phone_map.put("ao", "ɔ:");
        phone_map.put("aw", "aʊ");
        phone_map.put("ax", "ə");
        phone_map.put("ay", "aɪ");
        phone_map.put("eh", "e");
        phone_map.put("er", "ɛ:");
        phone_map.put("ey", "eɪ");
        phone_map.put("ih", "ɪ");
        phone_map.put("iy", "i:");
        phone_map.put("ow", "əʊ");
        phone_map.put("oy", "ɔɪ");
        phone_map.put("uh", "ʊ");
        phone_map.put("uw", "u:");
        phone_map.put("ch", "tʃ");
        phone_map.put("dh", "ð");
    }
}
```

讯飞音素映射表

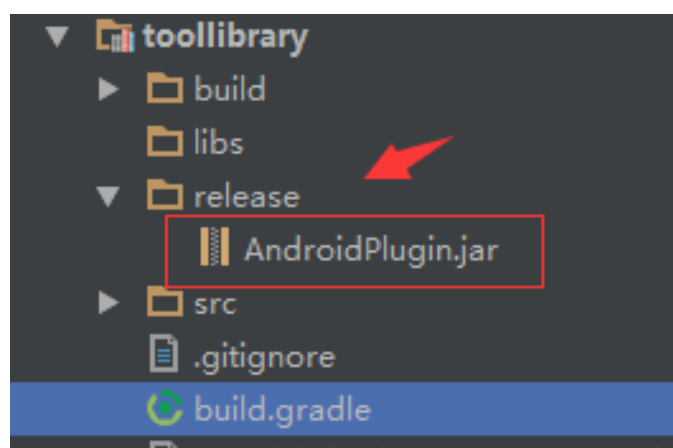
而语音合成是将一段文字转换为成语音，可根据需要合成出不同音色、语速和语调的声音，让机器像人一样开口说话。

在“VR 实验室”和“口语卡片”中我们使用到了科大讯飞的语音 Android SDK，并且做了大量的抽象工作，对该 SDK 做了**高度封装**，开发者甚至只要使用三行代码和回调函数就可以完成语音评测的功能，并且代码在 Unity 项目和安卓项目中通用。

（四）Unity 项目和 Android 项目的兼容和合并

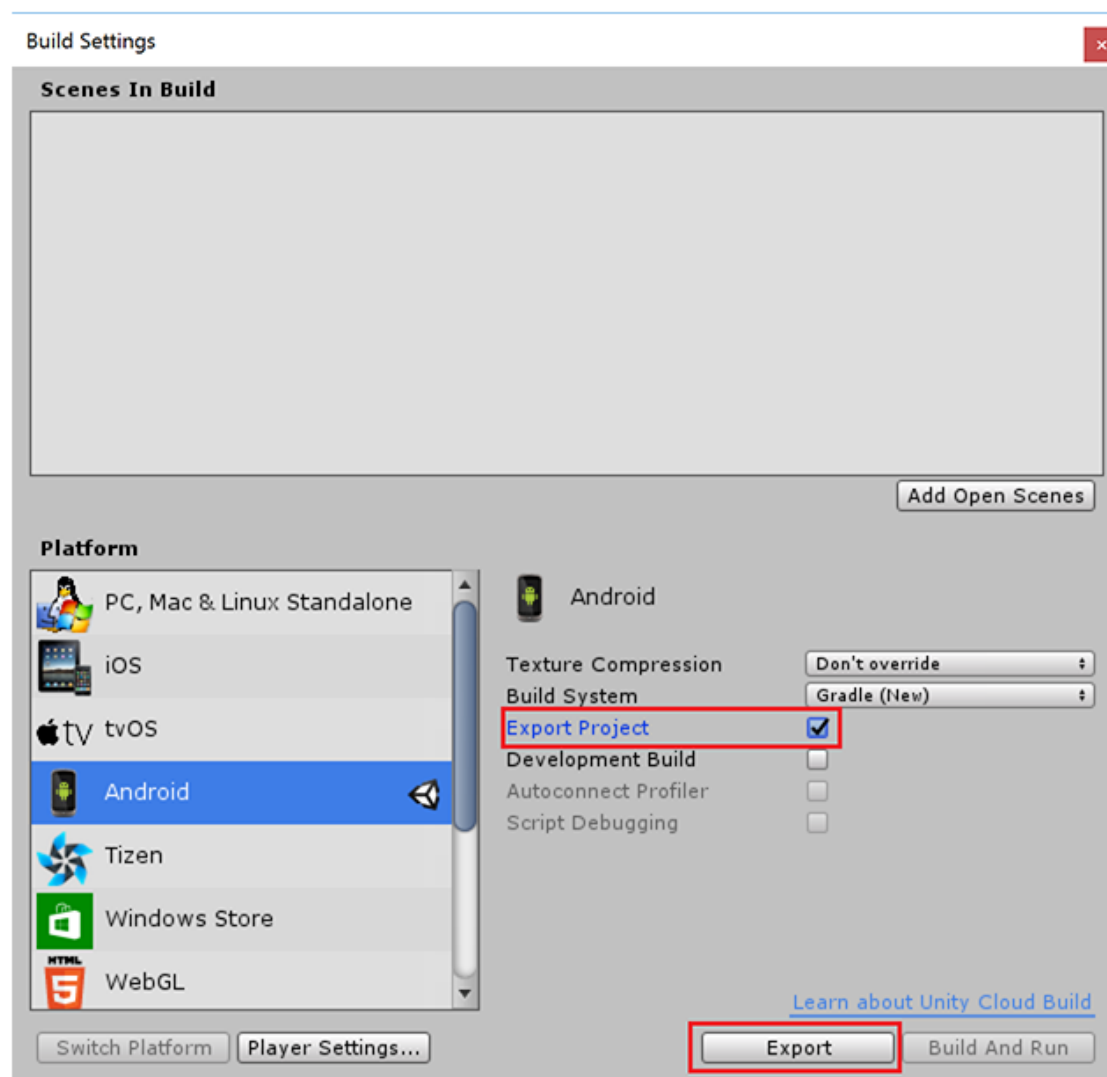
Unity 3D 和 Android 合并的具体方法分为以下两种。

第一种方法是在 Android Studio 中建立 library, 并打包成 jar 包或者打包成 aar 包, 然后放入 Unity 3D 中使用, 这种方法的特点在 Unity 3D 中可以直接调用 Android 里面写好的 Public 方法, 实现简单的功能, 而且最后在 Unity3D 中导出产品, 需要有较深的 Unity 3D 方面的技术要求, 适合把项目重心落在 Unity 3D 上, 缺点是 Android Studio 和 Unity 都要做好配置, 最重要的是两边的包名必须保持一致, 而且在 Unity 3D 导出最终成品的时候非常容易出错。



图中显示了 Android Studio 导出的 jar 包

第二种方法是把 Unity 3D 项目打包成一个 Module, 然后导入到 Android Studio, 作为 Android 项目的一部分, 这种方法特点是需要有较深的 Android Studio 方面的技术要求, 适合把项目重心落在 Android Studio 上。



图中显示了 Unity 3D 导出 Module

我们需要完成的治疗口吃 APP, VR 场景只是作为训练的一部分, 所以需要使用到 Android 方面的开发, 所以我们经过讨论研究, 最后采用了第二种方法完成 Unity 3D 和 Android 的交互。

(五) Java 和 Kotlin 混合编程

在项目中, 我们使用了许多开源组件, 如 CardStackView^[3]、Material Drawer^[4]等。其中某些组件是使用 Kotlin 编程的, 所以本项目在修改其代码时, 必然也要接触到 Kotlin 编程语言, 增加了编程难度。在“口语卡片”模块, 主要使用 Kotlin 语言进行编程, 实现了对卡片左滑, 将顶部卡片加入栈底, 表示未掌握该卡片内容; 右滑从栈中退出, 表示已掌握

卡片内容，不再显示。进行录音后，还会直接显示评测分数，点击详情按钮，可以展示朗读的具体分析结果，有**可拉伸，放大，左右滑动**等操作的图表。点击喇叭按钮，还能听自己朗读的录音，检查朗读问题。



APP “口语卡片” 界面

三、 创新与特色

(一) 创新

1. 手机显示模式创新

传统应用都是用全屏幕来显示用户界面的，而我们把屏幕一分为二，让两个眼睛分别对应一个屏幕，实现立体画面的效果，提高了沉浸感。更高的维度会让口吃患者接受到的信息

更多，就好比文字、音频、视频的关系，文字要想表现出情绪、环境和故事情节需要大量的铺垫和深刻的文笔能力，音频通过语气就能表现出很多内容，更容易带入到故事当中，它的维度比文字要高，有声音和语言两种，视频的维度就更高了，还多了影像，可以更迅速直观的表现场景。所以虚拟现实技术可以为口吃患者最大程度地还原真实场景，使得口吃患者得到更好的治疗。

2. 社会创新

利用 VR 设备的便携性优势，可以实现患者在家治疗、在家康复，缓解国内言语治疗师数量匮乏的现状。

3. 交互形式的创新

针对口吃严重度在压力场景中容易增加的事实，通过语音识别进行交互，给予患者掌声、嘘声、笑声等正向、负向反馈。

4. 训练内容的创新

已有的系统脱敏训练内容单调，而在 VR 场景中可以自由建模、组合场景。此外，可以通过可视化的时间限制、流畅度提示、观众满意度提示，让训练内容更加充实。

5. 训练内容的创新

融合了心理学、计算机科学、医学等多个学科的内容。VR 在医学已有广泛应用，而在口吃治疗领域还没有先例。系统脱敏需要提供场景支撑，而 VR 恰好适合。

（二）特色

高质量的 UI 设计	本项目花费了大量的时间在调整动效和界面设计方面，可以说，在某些方面，我们甚至超过了一些商业作品。目的是为了用户有更好的体验，能够以较低门槛上手本 APP。
针对性强	该 APP 针对口吃患者进行针对性的治疗。

语音测评返回结果丰富	不仅返回测评综合得分，而且对每一个字的语音、语调都有对应的评价，评价包括：正确、漏读、回读、重复读等等。
语音测评时间快，使用内存小	使用讯飞语音测评，几乎不需要使用内存，而且测评时间非常短。
测评内容丰富	测评内容包括字、词语、句子。
测评内容可扩展性强	测评内容可以通过阿里云的云数据库获取，只需要在数据库中插入信息，即可扩展 APP 中测评内容。
沉浸感强	使用虚拟现实技术，可以 360 度观看场景里面的物体，就像走进了游戏场景里面一般。
无需昂贵的治疗费	医院专业的口吃患者治疗需要昂贵的费用，而使用 APP 治疗几乎不需要任何费用
解决治疗地域限制	具备口吃治疗资格的医院较少，很多口吃患者远离口吃治疗医院，使用 APP 治疗只需要拿起手机即可，实现了对所有口吃患者零距离治疗。

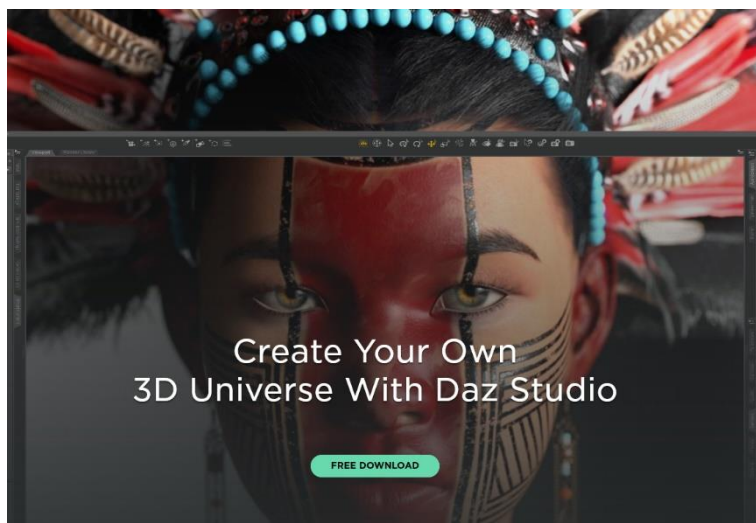
四、 详细实现过程

本项目当前主要实现了“VR 实验室”和“口语卡片”两个模块，还有一个模块“进度统

计”将会在未来完善。

（一）Unity 3D 方面

1. 3D 人物建模



Daz Studio 软件

1.1 使用过程

本内容需要 Daz Studio4.11 软件、dim 安装经理 64 位和基础资源包。首先先安装 dim 安装经理的 64 位版本和 Daz Studio4.11 软件,使用安装经理安装基础资源包到 Daz Studio 中,然后就可以进行人物模型创建了,创建过程不需要任何 3D 建模基础,就像玩游戏一样轻松装扮人物角色,然后把创建的人物模型以 FBX 格式导出。

我们根据课室场景需要,构建了两个基础学生模板,可以在两个模板基础上通过改变高矮肥瘦和着装创造出不同的学生模型。



学生模型

1.2 遇到的问题

在人物建模过程中也会遇到一些问题，使用 Daz Studio 导出的人物模型面数较多，而模型面数越多，就越消耗手机 CPU 资源，所以导入到 Unity3D 场景的人物模型个数受到很大的制约，现在只能最多放入两个模型，才能保证应用在手机中运行时保持可以接受的帧率。

1.3 解决方案

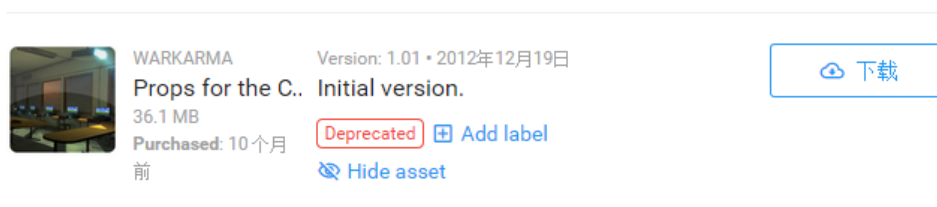
①用图片代替人物模型，这样做的优点是保证了运行时有较高的帧率，缺点是使用者可以感知到人物都是二维的，影响了使用者的体验，而且由于都是图片，很难实现人物 3D 动画的制作。

②用图片代替场景后面的人物模型，这样做可以较好地解决前一个方案中使用者感知到人物是二维的问题，因为由于手机像素的限制，使用者不能看到坐在后排的学生的具体细节，也就无法感知到坐在后排的学生是否是二维的，而且也可以加入人物的 3D 动画了，但是缺点是前排一共有六个位置，而由上文描述，最多只能放入两个模型，所以第一排还是有四个位置无法加入 3D 人体模型，这样使用者只要足够细心，还是可以发现人物是二维的。

③对人物模型做减面处理，使得其不会消耗太多 CPU 资源，从而增加可以导入到场景中的人物数量，结合着方案②的使用，可以说是最佳的处理方案。

2. 3D 人物动画和物体建模

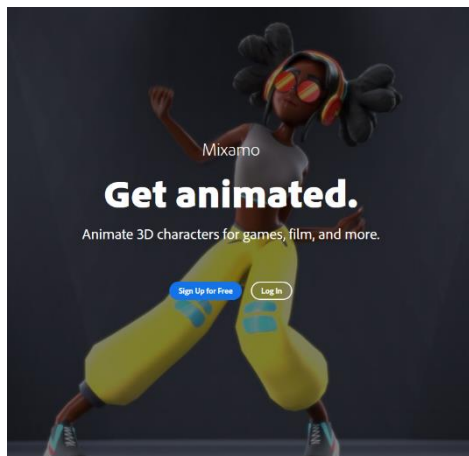
在网上搜索资源，最后发现通过 Unity 3D Asset Store 官网，可以直接下载到课室场景需要的绝大部分物体模型。



课室场景

我们使用的 3D 建模软件主要是 3D Max，对于场景需要但是又难以从网上获取的模型资源，我们使用 3D Max 进行构建。

动画资源是通过 Unity 3D Asset Store 官网和 Mixamo 官网获取的。对于在网上下载的 3D 动画资源，大多无法直接应用到 3D 人物模型上，我们使用 Unity 3D 插件 UMotion Pro 编辑人物动画，使得动画更好地应用到 3D 人物模型上。



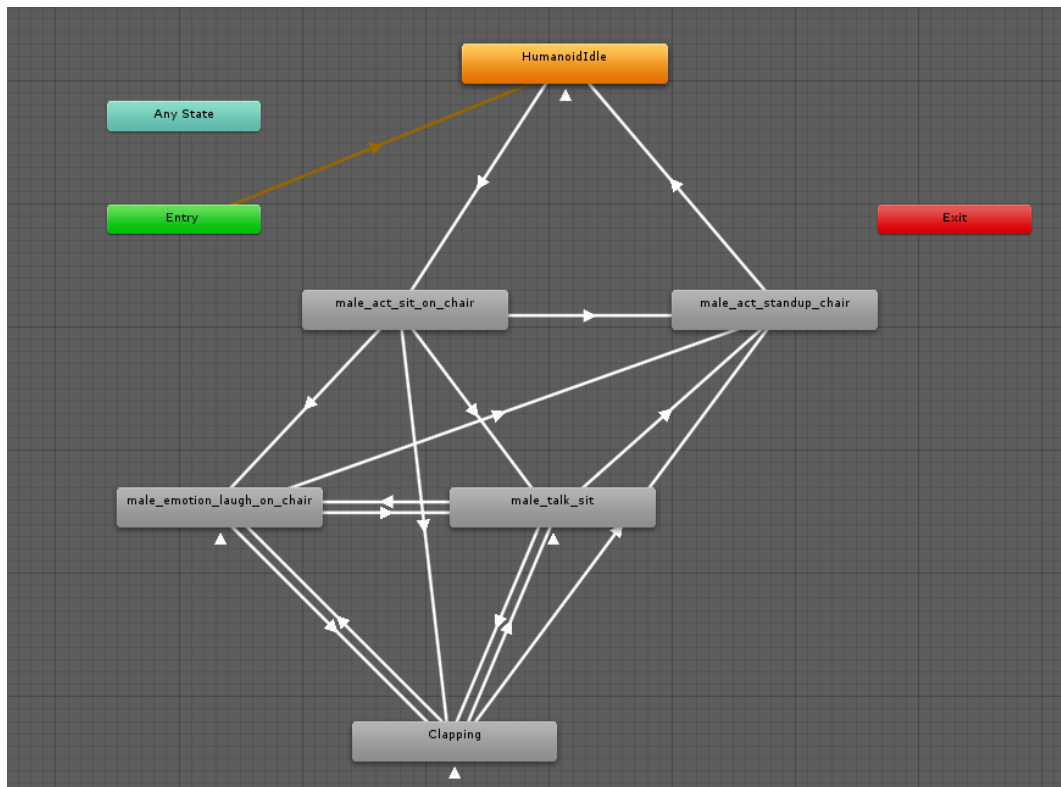
SOXWARE INTERACTIVE
UMotion Community - Ani...
★★★★★ (67)
FREE

Mixamo 官网和插件

3. 3D 人物动画控制

使用 Unity 3D 自带的动画控制器功能，首先新建 Animation Controller 并显示在 Animator 窗口中，把已经完成的动画资源导入到 Animator 窗口中，建立动画播放的逻辑，在 Animator 窗口中每个动画状态之间会以连线的方式展示从播放一个动画转移到播放另一个动画的逻辑。

然后还要提供一个 Public 函数，测评结果返回后，触发对应的人物动作。



Unity 3D Animator 窗口

```

1 个引用
public void sitDownOrStandUp()
{
    string str = "SitDown";

    animator.SetBool(str, !animator.GetBool(str));
}
0 个引用
public void startLaugh()
{
    animator.SetTrigger("Laugh");
}
0 个引用
public void startTalk()
{
    animator.SetTrigger("Talk");
}
0 个引用
public void startClap()
{
    animator.SetTrigger("Clap");
}

```

动作触发 Public 函数

4. 手机 VR 框架

3.1 通过手机陀螺仪获取头部转动数据

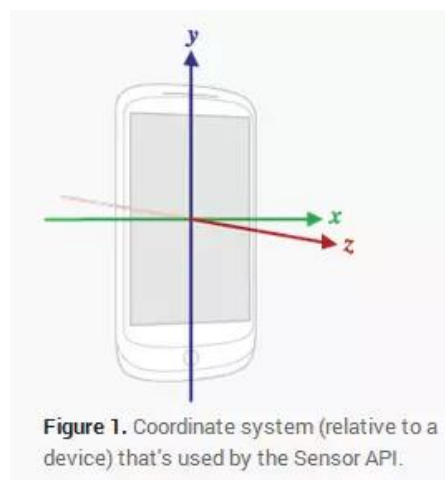
我们使用 Unity 3D 提供的 Gyroscope 类获取头转数据，其中可以获取头转数据的函数有 `rotationRate`, `rotationRateUnbiased`, `attitude`, `rotationRate`, 他们使用解释如下：
`Input.gyro.attitude`: `Vector3` 类型，返回手机当前的姿态信息，即与手机初始姿态相比在 `x`、`y`、`z` 轴上的偏离情况(单位:弧度)，初始姿态指的是手机水平正向放置(此时 `Input.gyro.attitude` 为 `(0, 0, 0)`)。

`Input.gyro.rotationRate`: `Vector3` 类型，返回陀螺仪测量的各轴上的旋转速率的原始值，单位为“弧度/秒”。

`Input.gyro.rotationRateUnbiased`: `Vector3` 类型，返回陀螺仪测量的各轴上的旋转速率的修正值，单位为“弧度/秒”。

我们经过测试，发现使用 `rotationRate` 会出现画面抽搐的现象，使用 `attitude`，由于初始姿态是手机水平正向放置，需要做转换，而且当手机已经完成旋转后，画面还会出现旋转，而使用 `rotationRateUnbiased`，解决了使用 `rotationRate` 会出现的抽搐现象，而且由于获取的是速率，所以得到的是旋转的变化量，不需要像 `attitude` 做转换才可以应用到摄像机上。

所以，最后我们采用了 `rotationRateUnbiased` 获取手机转动数据。



手机坐标系

3.2 手机显示双画面

首先在 Unity 3D 中，一个画面的呈现需要一个摄像机，现在需要两个画面，当然是需要两个摄像机啊了，而且两个画面应该是一左一右呈现，所以 Camera 的 Depth 属性同为 0，而 View Rect 的 w 只能是 0.5，x 也要对应做修改。



双画面

3.3 摄像机位置重置

手机获取头转动数据并不是非常准确，需要不时做修正，修正的方法就是触发按钮后让摄像机位置重置，根据观察，手机 VR 眼镜上面会有一个在手机放入后也可以按下的按钮，我们就把这个按钮设置为摄像机位置重置按钮。



重置按钮

3.4 游标交互

由于手机输入设备的局限，对 VR 场景产生交互的能力较弱，我们从爱奇艺 VR 中发现，他们在屏幕正中间显示一个绿色游标，当游标落在场景内的按钮上时，会出现一个读条，当读条结束，按钮就会被触发，于是我们也学爱奇艺 VR 的交互方式，我们使用 Unity 3D 提供的函数 Raycast，获取到画面正中心的物体，然后判断这个物体是否为可以触发的物体，如果是可以交互的物体，便调用交互函数进行交互。

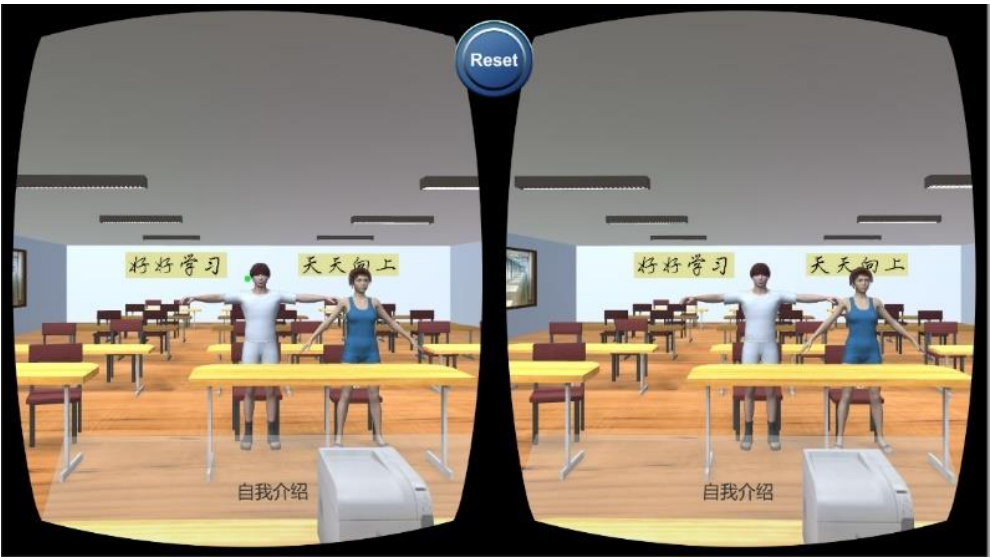
现在设置了按钮交互和学生交互，当游标落到学生身上，学生会做出交谈的动作，当游标落在按钮上并持续一段时间，则会触发按钮响应事件。



游标

3.5 画面四周遮罩

这个我们也是参考了爱奇艺 VR 的设计，因为单纯显示两个画面，看上去比较丑，如果在四周加上一个遮罩，会比较美观。这个我们采用了 Screen Space-Camera 模式的 Canvas, 这样就可以把遮罩应用于单个画面。



最终效果

5. Unity 3D 调用讯飞语音评测功能

讯飞语音测评 SDK 分为四种 Android SDK、IOS SDK、Linux SDK，Windows SDK，还有一种是通过网络请求的方式调用该功能，前期我们使用的是网络请求的方式，这个方式的优点是：对所有语言，各种设备都百分百地支持，只要可以发出网路请求就可以了，缺点是：测评返回时间较长，而且需要设置白名单。

最后我们使用的是 Android SDK，这种方法优点是：测评速度快，延迟较小，不需要设置白名单，所以非常适合我们开发 APP 使用。

下面，我们来介绍如何用 Unity 3D 调用 Android SDK 测评功能。

语音评测SDK		
SDK名称	版本	操作
Android MSC	1138	下载 文档
iOS MSC	1174	下载 文档
Linux MSC	1126	下载 文档
Windows MSC	1126	下载 文档

讯飞语音评测 SDK

4.1 使用 Android Studio 导出的 arr 包

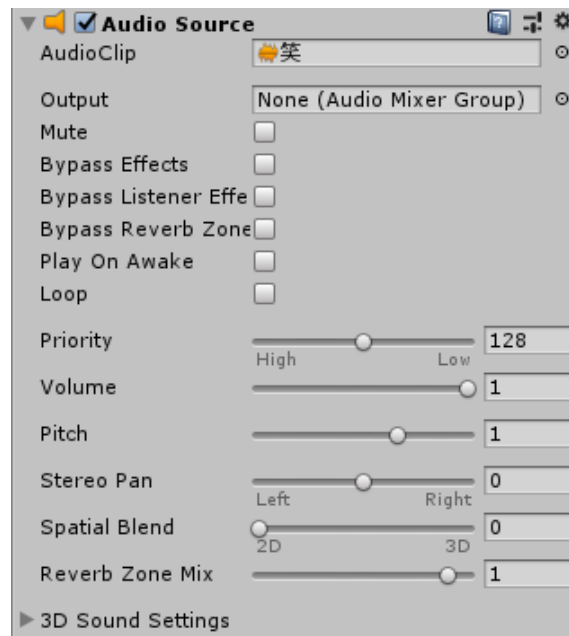
这个过程的缺点是需要配置的东西比较多，比较容易出错，而且异常都是把问题说得很模糊，几乎不可能了解到出错的根源是什么，更不知道如何修改，项目初期为了解决这个方法出现的问题，花费了大量的时间，最后发现把 Unity 3D 打包项目设置中的 Build System 从 Gradle 改为 Internal, 便可以解决问题。

4.2 导出项目成为 Android 开发的一部分

由于项目需要 Android studio 开发其他界面，所以上一个方法最终不能被采用，所以我们又通过网上搜索发现了另一种方式，就是把 Unity 3D 项目导出到 Android Studio 上，然后作为 Android Studio 上面的一个模块，这种方法配置较简单，操作较容易，相比于上面的方法，可以较轻松的实现调用讯飞 SDK 的功能。

6. Unity 声音控制和光照渲染

Unity 3D 提供播放声音的组件 Audio Source，然后还要提供一个 Public 函数，测评结果返回后，发出对应声音。



Audio Source 组件

光照渲染的目的是让场景更接近真实，首先要把场景中的不需要改变位置的物体设置为静态物体，然后在场景中加入 Area Light，然后点击渲染就可以了，由于场景物体较多，每次渲染的时间较长，所以我们把 Unity 灯光渲染设置为不自动做渲染，等到把所有物体位置

确定后，再做渲染。



灯光渲染后，场景阴影更加真实

（二）Android Studio 方面

由于本项目即将申请软件著作权，不再进行具体代码展示。此处仅基本展示实现的功能模块。

1. 启动页延时动画

启动页使用了延时、位移、缩放动画，可以让用户感觉到应用的专业性。



VR口吃治疗 0.1.3

2. 自定义 ViewPager+Fragment

为了避免“VR 实验室”页面的卡片左右滑动和 ViewPager 左右滑动冲突，我们使用了

不可滑动的 ViewPager。但点击底部导航栏依旧可以进行页面切换。

```
/**
 * 可设置是否可滑动的自定义ViewPager
 *
 *
 */
public class NoScrollViewPager extends ViewPager {
    private boolean isScroll = false;
```

3. 自定义顶部栏 TopBar

顶部栏可以根据页面不同切换标题，并且点击头像可以呼出侧边导航栏。在点击头像时，会有水波纹动效。



您好Ryzin，快乐星期一

VR实验室

4. 轮播图 RecyclerView

此 RecyclerView 的 Adapter 支持三种方式获得轮播图图片：本地数据、url 网络数据、本地数据和网络数据同时。左右显示的轮播图大小不同，中间为放大展示。



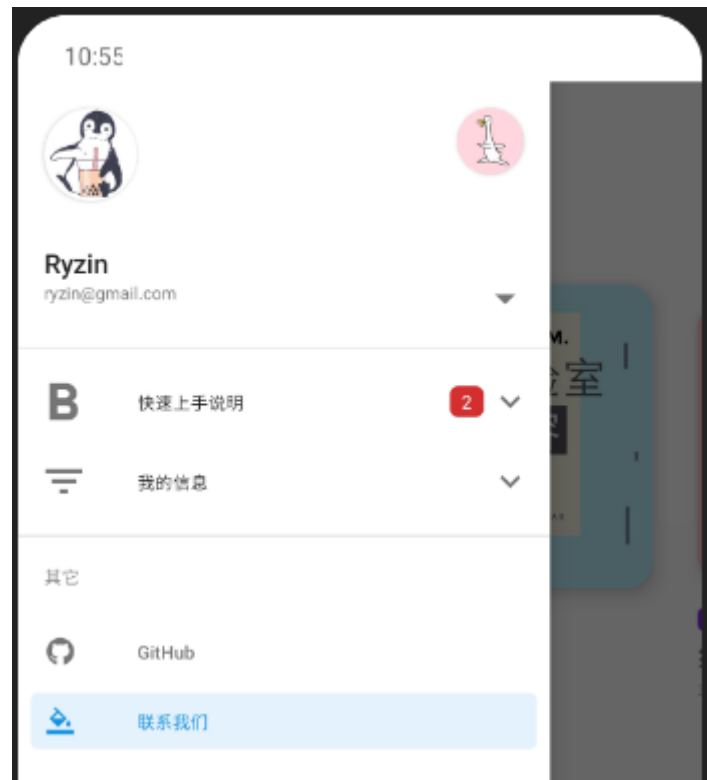
5. 登录对话框自定义 Dialog

该对话框实现了弹出和弹回的动画，并且点击对话框以外的区域可以关闭，十分人性化。



6. 侧边导航栏 Material Drawer

侧边导航栏使用了符合 Material Design 的 Material Drawer。



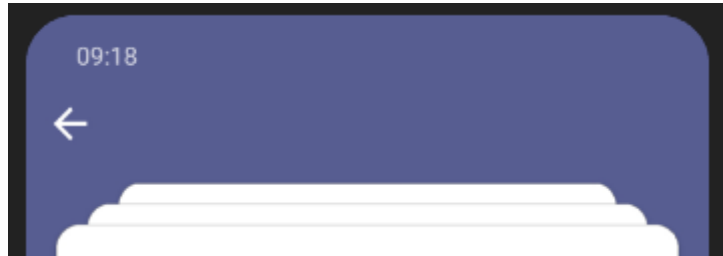
7. 按钮水波纹

我们特别喜欢水波纹动效和圆角矩形，所以几乎所有的按钮和卡片都是有这两个特性，在经过一段时间的调优，我们实现了同时具有这两个特性的按钮，但后来我们发现了更好的实现方法——用 CardView 这个组件。



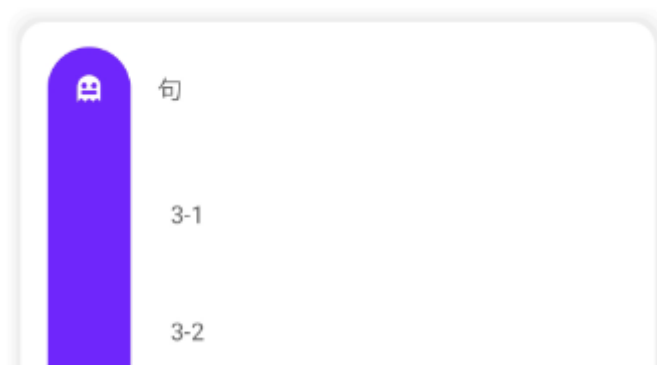
8. 全面屏适应

我们针对应用各个页面都做了沉浸式导航栏的处理，对一些全面屏的手机比较友好，也更加美观，让用户更加沉浸。



9. 可伸缩抽屉 ExpandingDrawerView

在口语卡的课程模块，我们使用了可垂直展开的组件，供用户具体选择课程，并且在展开过程也有延时、位移和缩放的动画，让人感觉十分自然。



10. 可伸缩卡片 ExpandingCardView

该组件可以水平展开卡片，达到显示更多内容的目的。



11. 进度展示自定义 CardView

此 CardView 着重展示了数据，并且颜色也很美观。



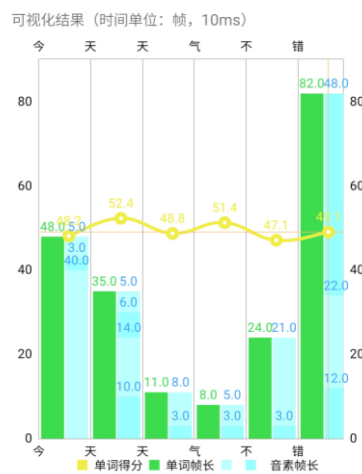
12. 堆栈式卡片 CardStackView

这种栈式卡片为用户提供了更加新颖的交互方式，方便了用户使用口语卡片的功能。

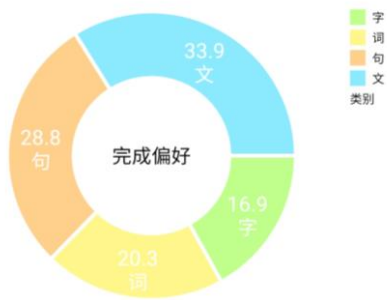


13. 可滑动的多图表 CombinedChartView

此图表结合了普通条形图，栈式条形图和折线图，密集地展示了用户的录音结果信息，便于用户掌握自己的朗读情况。并且，此图表还允许放大、点击和滑动，便于用户查看。

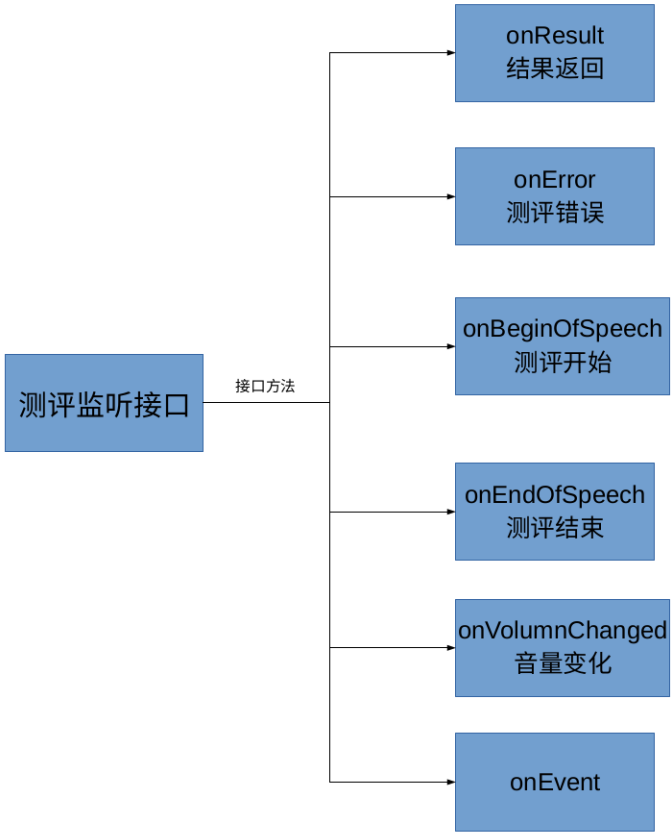


14. 可滑动的圆饼图表 PieChartView

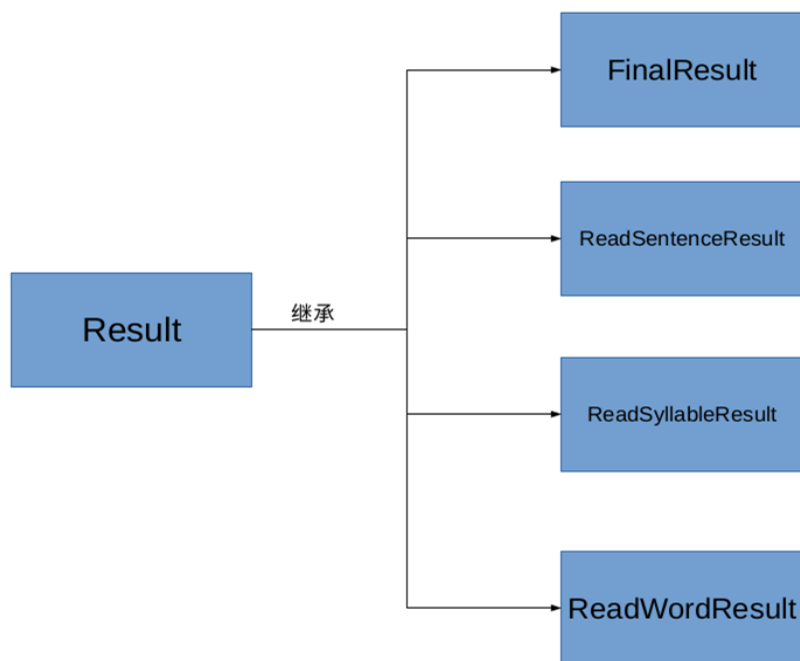


15. 科大讯飞语音模块

通过理解官方范例，我们重载了测评监听接口，封装了语音测评方法，提供了测评结果回调函数接口。



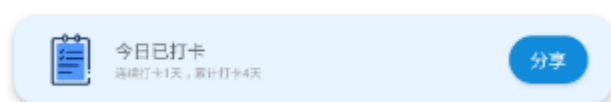
测评监听接口结构



Result 类继承结构

16. 动态权限申请

我们十分憎恨应用启动时强制要求权限申请的应用，所以我们将所有的权限申请，都设置在用户主动点击按钮时请求。同时，我们通过存储权限实现了文件分享的功能。



17. Android 和 Unity 3D 交互

为了在 Android 项目能够对 Unity 项目进行修改，我们编写了 Android 调用 Unity 3D 场景物体的组件函数。并提供 Public 函数，供 Unity 3D 调用。

18. 阿里云云数据库 Utils 类

为了获取和对云数据库数据进行处理，我们创建了相应 Utils 类。

19. UI 设计和动效

为了不让用户感到突兀，我们对很多细节上的效果进行了大量的尝试。如在“口语卡片”

页面，当焦点在搜索栏时，底部导航栏会立刻收起，软键盘会弹出。当点击键盘外区域时，能够主动收起软键盘，并且搜索栏会失去焦点，同时，为了等待软键盘完全收起，底部导航栏会延时弹出。这样就实现了一个完整的操作过程。

像这样细节的地方还有很多，因为我们最终希望用户能够理所当然的使用本应用的功能模块，而不是在某些地方会感觉莫名其妙。

（三）云数据库方面

1. 订阅数据库服务

我们使用的是阿里云云数据库 RDS 版，使用云数据库的好处是，可以让 APP 的测试内容更新，提高了 APP 的扩展性和灵活性，而且使用云数据可以让所有移动设备都获取到测试数据。当然使用云数据库是付费的，还好阿里云提供了学生版，只要通过了学生认证，仅仅 10 元就可以使用三个月的云数据库。

基本设置

标签设置

实例名称

请输入实例ID进行检索

搜索

标签

<input type="checkbox"/>	实例名称	运行状态(全部)	创建时间	实例类型(全部)	数据库类型(全部)	所在可用区	网络类型(全部)	付费类型
<input type="checkbox"/>	rm-wz96v85n0w43655jy rm-wz96v85n0w436...	运行中	2019-10-23 13:36	常规实例	MySQL 5.7	华南1 (深圳) 可用区D	经典网络	包月 171 天后到期
<input type="checkbox"/>	<div>编辑标签</div> <div>实例授权</div>							

共 1 页

阿里云云数据库 RDS 版

2. 数据库 CURD 范例

刷新 单行详情 总行数 导出数据 模板SQL ▼		
	id ▼	content ▼
1	31	室
2	30	纸
3	29	壳
4	28	哭
5	27	菇
6	26	倒
7	25	袍
8	24	杯
9	23	西
10	22	粗
11	21	紫
12	20	肉
13	19	书
14	18	出
15	17	猪
16	16	吸
17	15	七
18	14	鸡
19	13	河
20	12	拷

构音基础课程数据库内容

五、 开发环境

（一）开发环境

项目数据库开发环境

序号	项目	详细信息
1	操作系统	Wondos-10-x64-desktop-Home
2	编程语言	MySQL
3	编程工具	MySQL 5.5 Command Line Client
4	依赖工具	阿里云 云数据库 RDS 版

项目安卓开发环境

序号	项目	详细信息
1	操作系统	Ubuntu-18.04.3-desktop-amd Wondos-10-x64-desktop-Home
2	编程语言	Java、Kotlin
3	Android 开发平台	Android Studio 3.0.1 64 位 Android Studio 3.4.1 64 位

4	调试设备	1、模拟器: Nexus 5X API 28 (Android 7.0, API 24) Samsung S10 5G API 29 Android 9.0, API 29) 2、真机: HUAWEI P10 (Android 9.0, API 29) Samsung S10 5G API 29 (Android 9.0, API 29)
---	------	---

项目 VR 开发环境

序号	项目	详细信息
1	操作系统	Windows-10-x64-desktop-Home
2	编程语言	C#
3	开发平台	Unity 2018.3.2f1 (64-bit)
4	调试设备	1、模拟器: Nexus 5X API 28 (Android 7.0, API 24) 2、真机: HUAWEI P10 (Android 9.0, API 29)

项目建模环境

序号	项目	详细信息
1	操作系统	Windows-10-x64-desktop-Home
2	平台 1	DAZ Studio
3	平台 2	3D MAX
4	平台 3	Blender

实机测试环境

序号	测试系统
1	HUAWEI P10 VTR-TL00 Android 版本 9 EMUI 版本 9.0.1 运行内存 4G 屏幕 1920*1080
2	Samsung S10 5G SM-G977N Android 版本 9.0 One UI 版本 1.0 运行内存 8G 屏幕 2280*1080

六、 软件测试与软件完成情况基本展示

(一) 测试方案

我们使用的测试方案是百度 MTC 移动 App 测试方案, 其中包括以下四个方面的测试内容。

1. 脚本兼容性测试

本测试覆盖市场主流机型, 测试安装、启动、主要功能、卸载; 可定制测试脚本, 覆盖所需页面和核心功能; 测试报告包含用例细节、截图、日志、性能数据和 BUG 详情。

性能结论

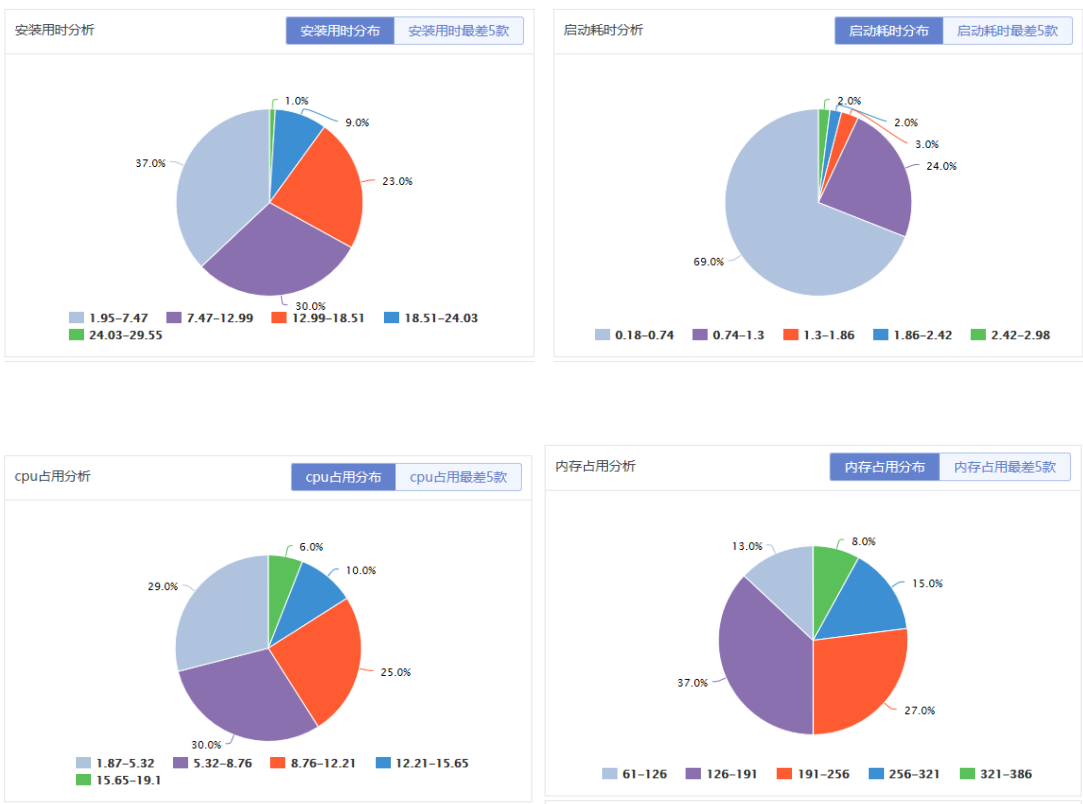
安装用时 10.16s	启动耗时 0.7s	CPU占用 8.21%	内存占用 203MB
行业最优: 0.04s 行业均值: 3.84s	行业最优: 0.19s 行业均值: 0.63s	行业最优: 1.81% 行业均值: 5.96%	行业最优: 5MB 行业均值: 132MB
华为 荣耀V9 1.95s	华为 荣耀畅玩5C 0.18s	华为 畅享6S 1.87%	三星 三星2016版GALAXY A9(S... 61MB
OPPO OPPO R9s Plus(OPPO ... 29.55s	三星GalaxyOn7 2.98s	金立M2017 19.1%	三星 GALAXY S6 Edge(SM-G9... 386MB

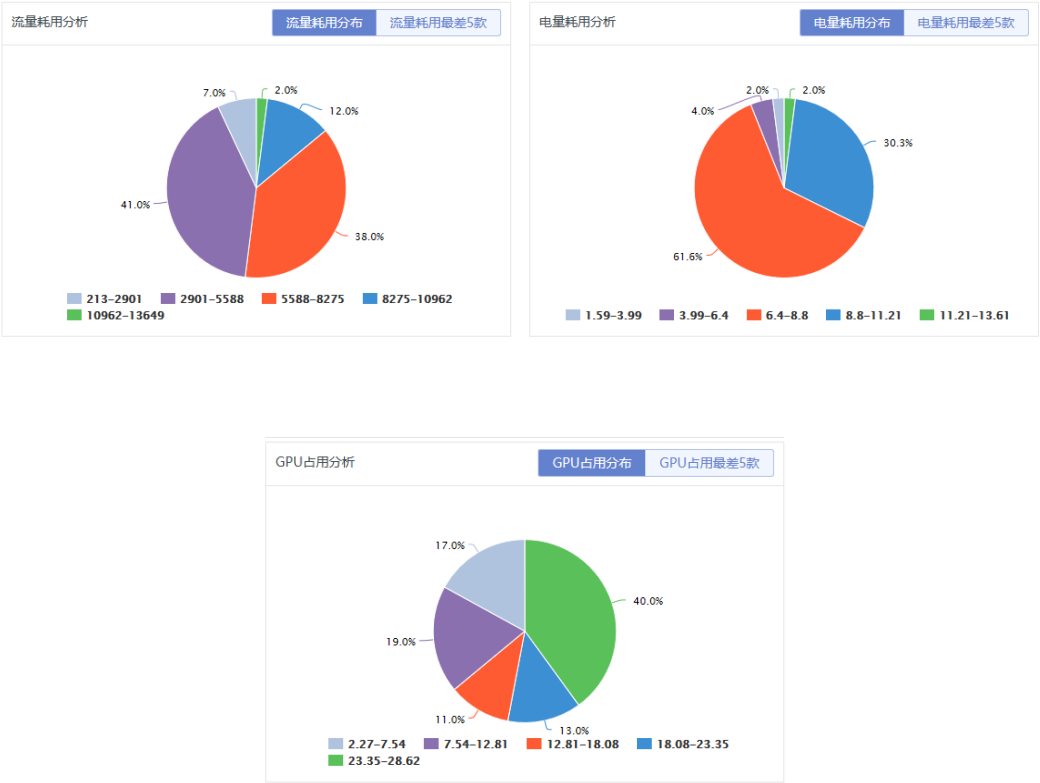
数据工具

流量耗用 5848KB	电量耗用 8.22mAh	GPU占用 18.28%
行业最优: 0KB 行业均值: 4,681KB	行业最优: 0.32mAh 行业均值: 4.13mAh	行业最优: 2.85% 行业均值: 16.97%
华为Mate 8 213KB	华为Mate 8 1.59mAh	vivo X3L 2.27%
索尼Xperia Z3+ Dual 13649KB	小米MiPad2 16GB 13.61mAh	三星 SM-A7000 28.62%

性能结论

性能分析

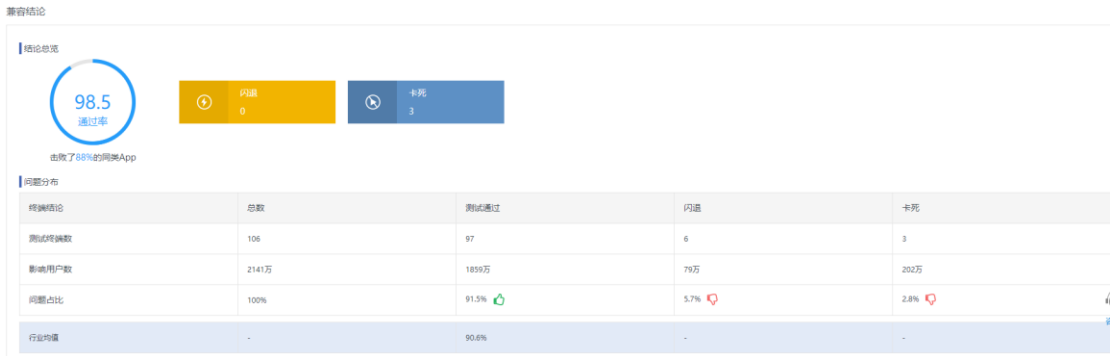




性能分析

2. 深度兼容测试

在选定机型上安装、启动、monkey、卸载等；记录测试过程中完整日志、截图、录像捕获 CPU、内存、流量、电量等性能数据助您定位闪退、crash、ANR 等问题



兼容结论

问题列表

出错定位筛选 全部出错定位

问题机型测试

问题类型	问题级别	问题描述	出错定位	发生机型数	通用解决方案	问题机型
闪退	致命级 (P0)	非法状态错误	java.lang.IllegalStateException	6	一般需要具体问题具体分析,例如,对于一些调用摄像头功能的程序,Java层和Native层的状态不同可能会导致这个错;再比如,是兼容对客户端的响应已经结束,但是仍然尝试向客户端(实际上是缓冲区)输出内容。	小米 MI 3W 华为 P9 Plus 中国移动N2 更多

问题列表



问题终端分布

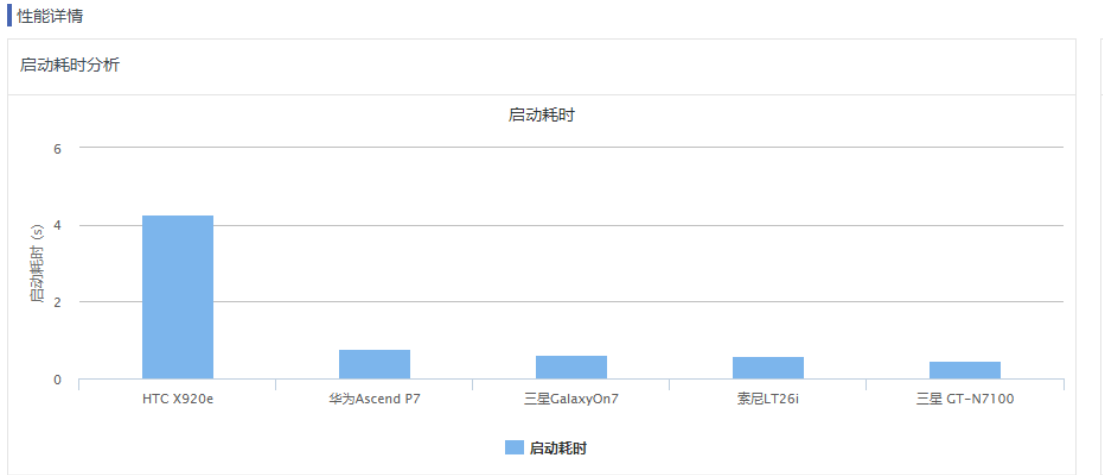
3. 深度性能测试

拟典型使用场景及状态；全面获得启动时长、电量、流量、CPU 等。

性能结论

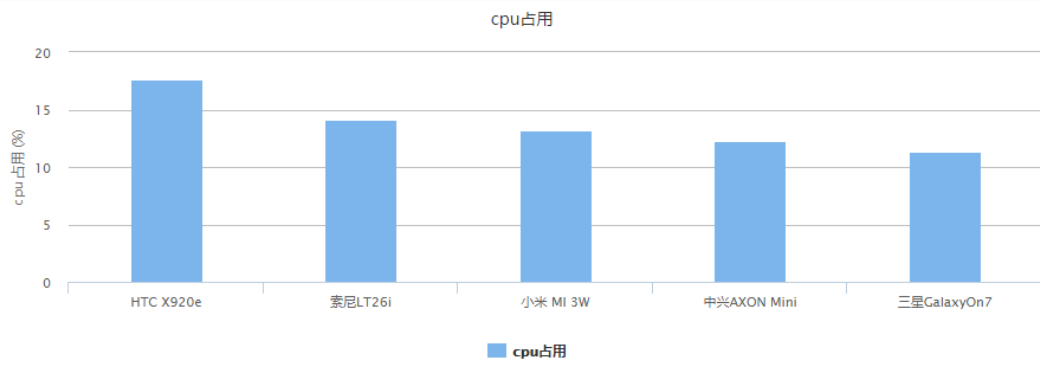
启动耗时	cpu占用	内存占用	流量耗用	电量耗用	GPU占用
0.58s	6.91%	66MB	502KB	5.81mAh	0%
表尼 L39h(android 4.2.2) 0.16s	华为 P9标准版(android 6.0) 1.68%	三星 GT-N7100(android 4.1.2) 42MB	HTC X920e(android 4.1.1) 19KB	三星 GT-N7100(android 4.1.2) 1.64mAh	-
HTC X920e(android 4.1.1) 4.26s	HTC X920e(android 4.1.1) 17.65%	三星 Galaxy S6 Edge(android 4.1.2) 135MB	索尼 L39h(android 4.2.2) 4251KB	索尼LT26i(android 4.0.4) 14.18mAh	-

性能结论



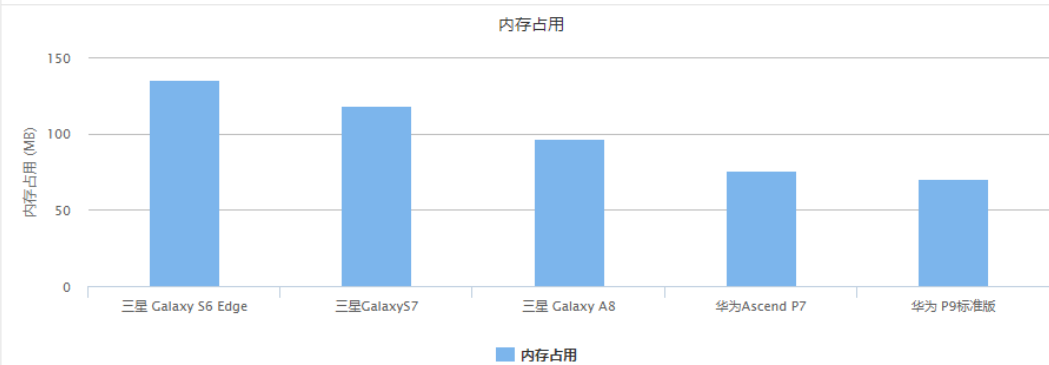
启动耗时

cpu占用分析



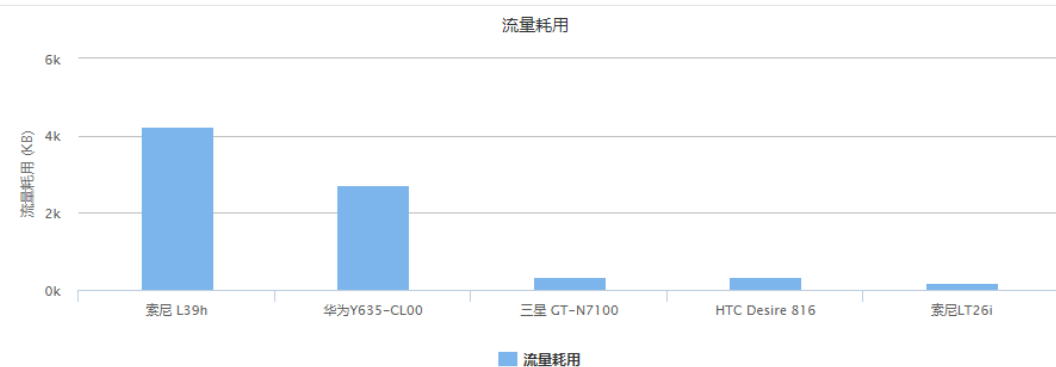
CPU 占用

内存占用分析



内存占用

流量耗用分析



流量耗用

（二）测试结果分析

我们对前面三方面的测试结果一一作分析。

1. 脚本兼容性分析

安装时间最多高达 29 秒的，可能原因是我们 APP 比较大，超过了 100MB，其中最占存储的应该是 Unity 3D 方面的资源。同时可以看到内存占用率和电量消耗都是比较大的。

2. 深度性能分析

从结论总览中可以看到我们 App 还是比较良好的，闪退次数为零，但是出现了三次卡死，我们猜测是讯飞 api 获取测评对象时出现了错误，根本原因还是我们测评是需要联网的，如果网络环境太差，就要可能出现卡顿甚至卡死。

3. 人工测试分析

功能还是太少了，建议 VR 场景的个数，并使用动态加载的方式加载 VR 场景，这样可以极大的减少 App 存储大小；VR 场景没有兼容全面屏，若使用有不完整屏幕的手机，会出现缺陷区域部分黑暗的情况；VR 场景使用过程中会出现轻微卡顿，建议对模型进行减面操作。

（三）用户意见和评价

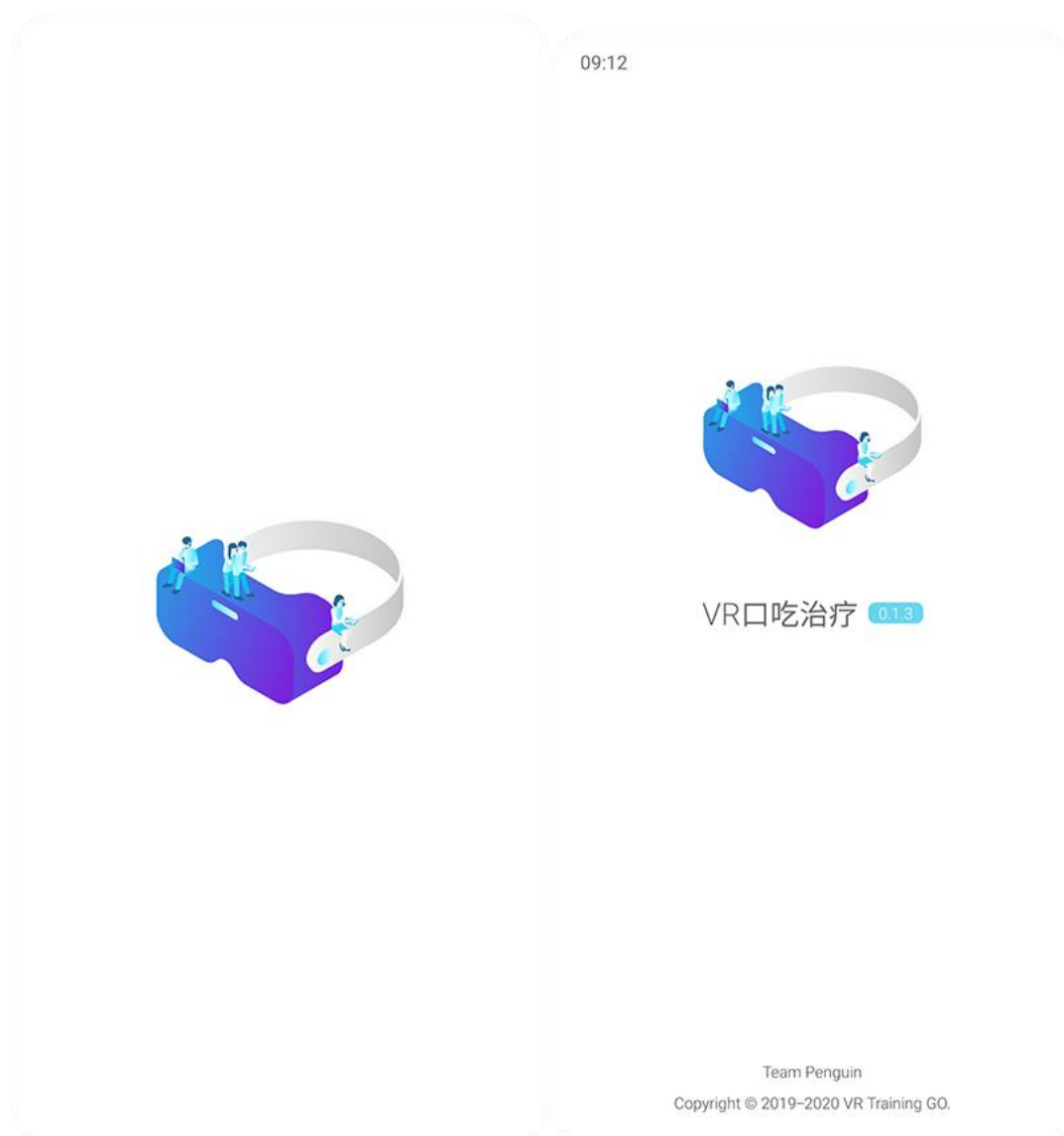
我们也在线下寻找了数十名同学进行软件使用体验，并将反馈的问题进行记录和改进。

序号	用户	角色	应用评价	改进意见
1	彭旭睿	2017 级计算机学院学生	界面、动效可以，都挺不错。	1. “口语卡片”页面，点击开始学习，希望能了解学习进度，不然不知道学到哪了。 2. “VR 实验室”的轮播图滑动时，希望有一些“阻尼”，一滑就滑得太多了。 3. VR 场景的文字提示一次给得太多，可以分阶段引导着提示。

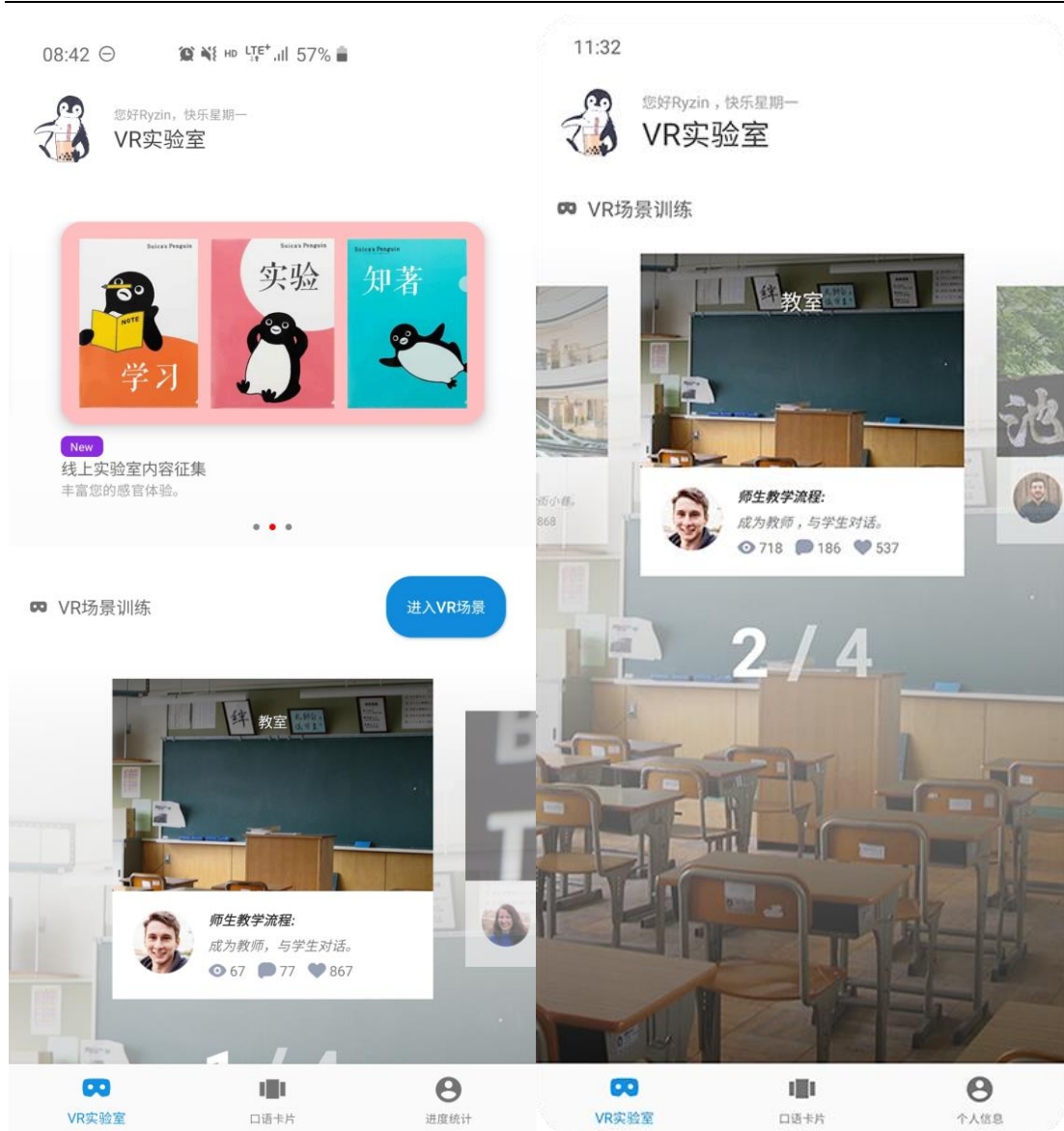
2	陈栩明	2017 级计算机学院学生	很好。	1. “进度统计”图表的文字重叠了，可以分开一点。
3	郭永聪	2017 级计算机学院学生	很好玩，没什么问题。	1. 在 VR 场景中，提示较少，一开始不知道要盯着老师才能到下一步。
4	朱炜	2017 级计算机学院学生	太完美以致于难以给出评价。	1. 如果“进度统计”页面的数据能改成动态的就好了。
5	韦靖鸿	2017 级计算机学院学生	挺完美的	1. 场景有点少。
6	司徒俊鸿	2017 级计算机学院学生	动画流畅，页面衔接做得很好，功能完善，很充实，作为学生作品似乎已经没有改进的地方了。	1. 某些地方颜色有些鲜艳。
7	何泽霖	2017 级计算机学院学生	这个软件感觉怎么能广泛传播，可能只会在特定团体中使用。	1. “VR 实验室”页面下方太空。 2. “口语卡片”功能中的卡片也太空。 3. 卡片分数可以大一些。这样用户可能更加兴奋。 4. 初次使用最好有指引。
8	孔思哲	2017 级计算机学院学生	效果很酷炫。	1. 希望未来实现登录功能、进度记录。 2. 分享功能分享的海报可以有二维码。
9	朱建忠	2017 级计算机学院学生	感觉动画和各种切换和过渡都很舒适，我都想不到可以这样做。这是	1. 口语卡片切换动画，可以更加平滑，现在太快了。

			这个应用 ui 设计的特色。	
10	陈子扬	2017 级计算机学院学生	UI 很符合安卓设计理念，就是扁平化。很期待后面的功能	1. VR 场景提示字体比较难看 2. 卡片音标，可以换成真的声调，而不是数字。 3. 识别类型不知道有什么用。

（四）软件完成情况基本展示

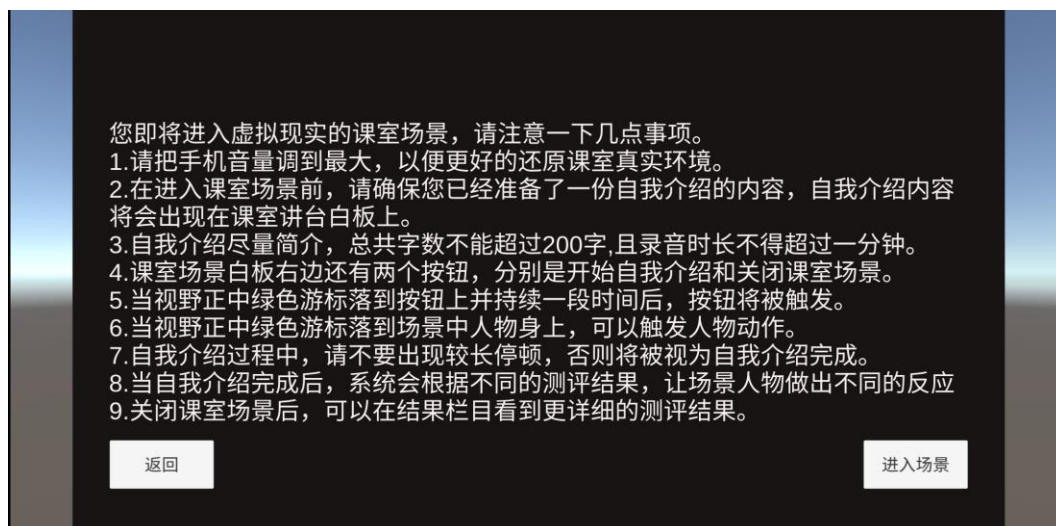


软件启动页



“VR 实验室” 页面

VR 场景训练中, 左右滑动可以选择不同的场景(但只完成了前两个场景), 然后点击“进入 VR 场景”按钮可以跳转到 Unity 项目。

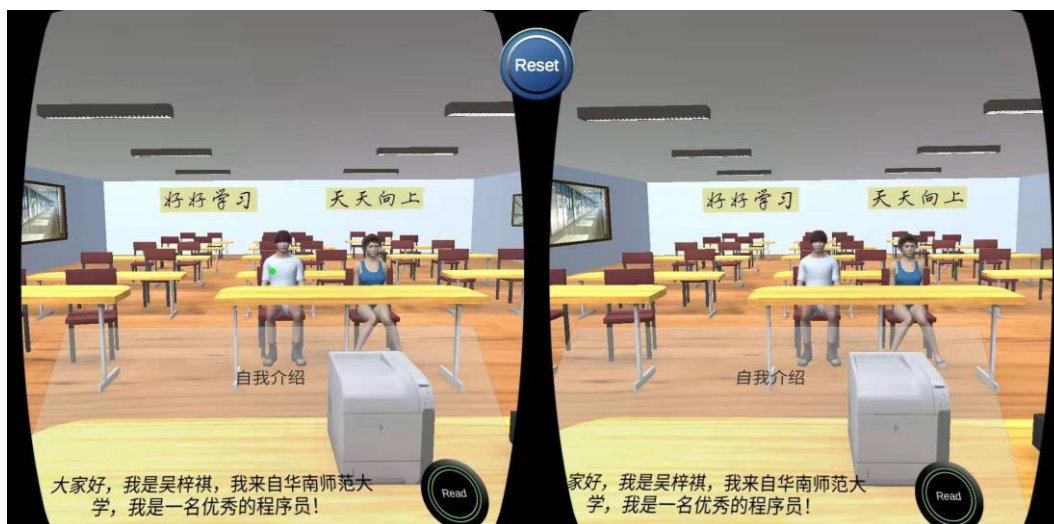


VR 场景注意事项提示

进入 VR 场景后，会有使用提示。在教室场景中，将游标对准教师的眼睛后，教师会邀请你上台演讲。点击“Reset”按钮可以重置视角。



VR 教室场景

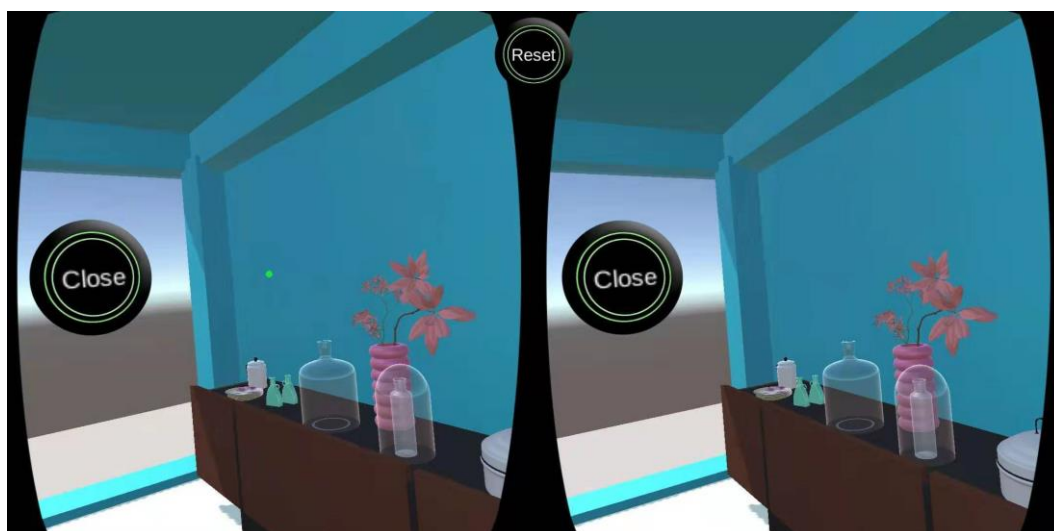


VR 教室场景

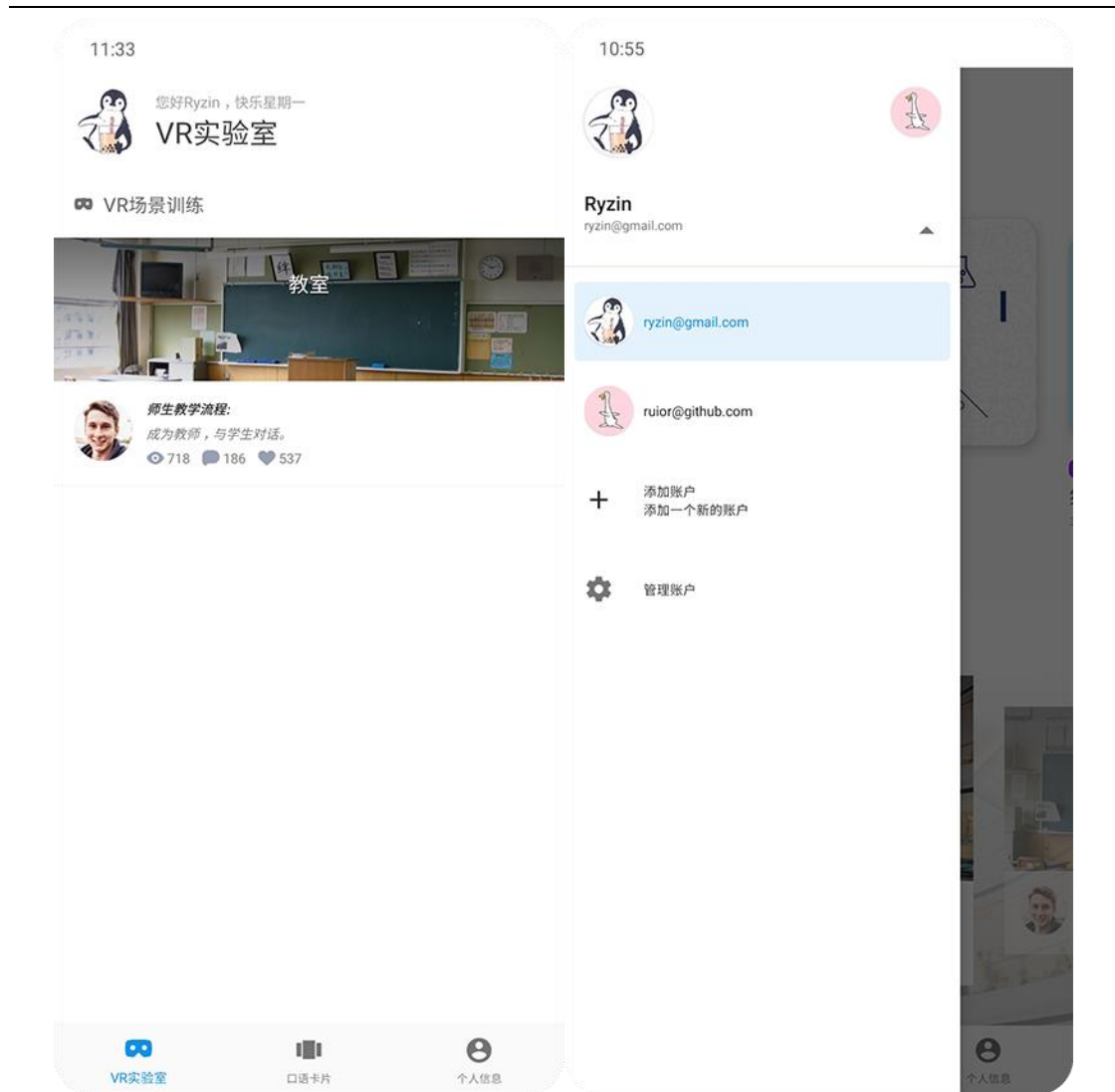
上台后，将游标滞留在讲台上的“Read”按钮上，会出现加载圈，表示即将开始演讲。
当想结束时，可以再次停留在“Read”按钮上。

将游标对准学生，学生可以发出声音。

若想退出，可以将游标停留在“close”按钮上。

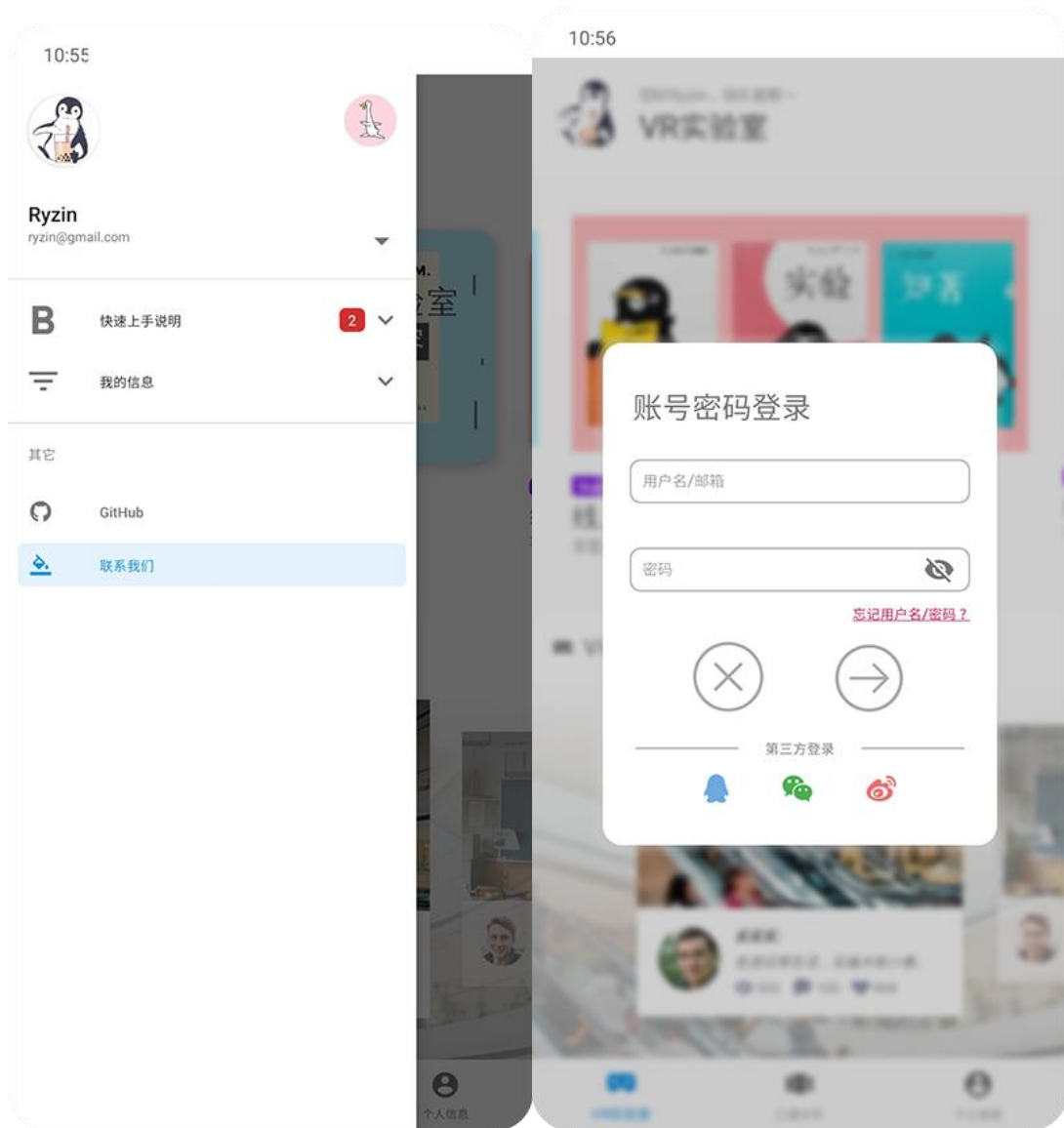


VR 卧室场景

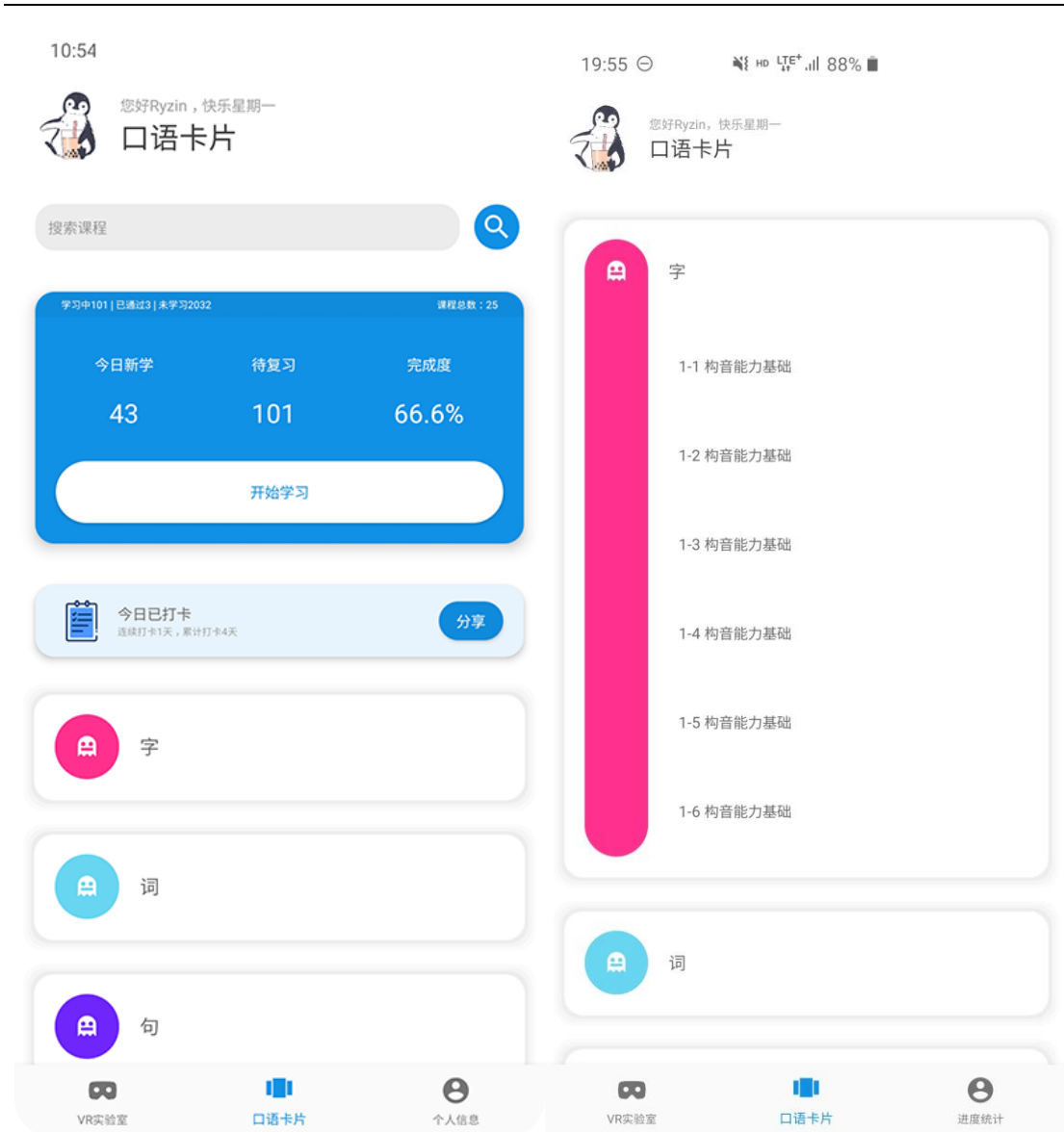


VR 场景卡片伸缩和侧边导航栏

在侧边导航栏，点击添加账户，会弹出登录对话框。

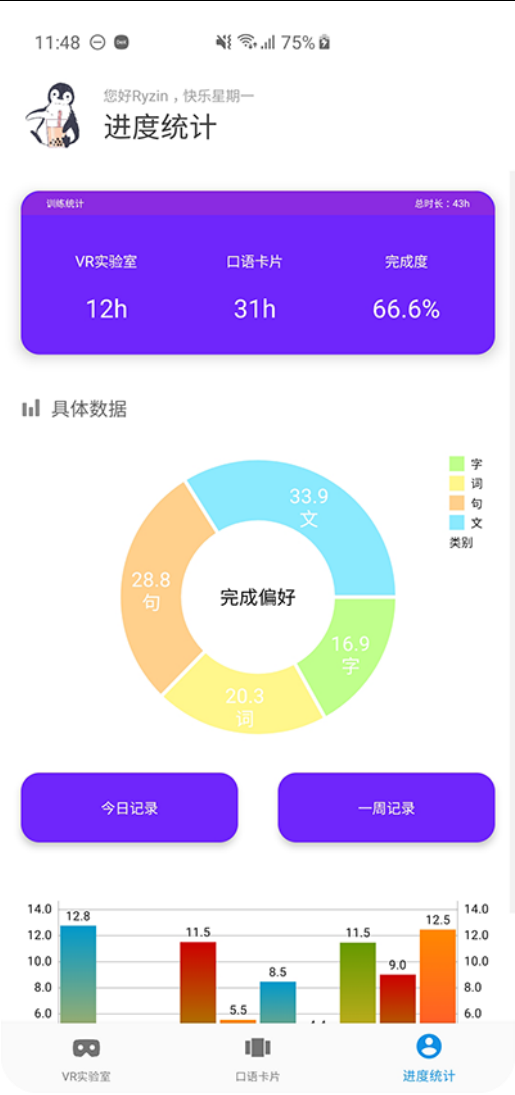


侧边导航栏和登录窗口



“口语卡片”页面

在“口语卡片”页面，可以选择不同的课程进行学习。点击分享按钮，可以分享应用海报到其他应用。



“进度统计”页面



“口语卡片”功能展示

进入相应课程后, 点击“麦克风”按钮, 用户就可以开始朗读卡片上的文字。再点击一次“麦克风”就结束录制。届时如果网络状况良好, 即可获得得分。

获得得分后, 点击中间的“朗读历史”按钮, 即可查看分析结果, 折线图表示每个字的得分。柱状图表示每个字的朗读情况。如“错”字, 将会分解为2个音素“c”和“uo”; 又如“天”字, 也将分解为“t”和“ian”两个音素。每个音素的朗读时长将在栈式柱状图展示。

点击“喇叭”按钮, 可以聆听自己刚才的朗读情况。

当用户重读, 回读, 漏读时, 可以直观地从图表中发现。图表也提供了拉伸和放大地功能, 用户可以仔细查看图表。



卡片左右滑动功能

卡片右滑时，将会从栈中删除该卡片；左滑时将在栈底重新加入该卡片。当用户不想滑动时，也可以点击下方 FloatingActionButton。中间的“撤销”按钮，可以将之前删除的卡片重新加入栈顶。

七、项目分工

（一）项目分工

吴梓祺：Unity 3D 开发，负责场景建模，人物建模，人物动画等等。“VR 实验室”功能的实现。科大讯飞 SDK 的封装。云数据库的配置。文档撰写。

罗佳海：Android 开发，负责 UI 设计，动效的实现。“口语卡片”功能、“进度统计”、

侧边导航栏，轮播图等实现。功能质量审核。文档排版。

谢礼东：Unity 项目和 Android 项目的合并。文档撰写。负责统筹具体赛事。软件测试。

李佳蔚：组织团队分工，负责统筹具体赛事。Unity 项目场景渲染。实现语音合成模块。文档撰写。

参考资料

- [1] 刘旭刚, 徐杏元, 彭聘龄, 林岚. 口吃复发的研究进展述评 [J]. 中国特殊教育, 2005(04):87-92.
- [2] 本产品是由 **2019 年大学生创新创业训练计划**项目《基于人工智能与 VR 的口吃治疗训练相关技术研究与应用》孵化的项目。
- [3] Github 开源项目 yuyakaido/CardStackView, <https://github.com/yuyakaido/CardStackView>
- [4] Github 开源项目 mikepenz/MaterialDrawer, <https://github.com/mikepenz/MaterialDrawer>