



第十四届全国大学生软件创新大赛

文档编号：SWC2021-T20210347-领航者



护航者

DriverEscort

项目测试文档

Version: 1.0.0



领航者

2020-11-26

All Rights Reserved

目录

1	测试计划	1
1.1	测试策略与目标	1
1.2	测试范围	2
1.3	测试环境	3
2	单元测试	5
2.1	视频录制模块	5
2.1.1	测试用例与结果分析	5
2.1.2	测试结果综合分析及建议	5
2.1.3	测试经验总结	5
2.2	视频转发模块	6
2.2.1	测试用例与结果分析	6
2.2.2	测试结果综合分析及建议	6
2.2.3	测试经验总结	6
2.3	服务端模型模块	7
2.3.1	测试用例与结果分析	7
2.3.2	测试结果综合分析及建议	7
2.3.3	测试经验总结	7
3	功能测试	8
3.1	路怒检测功能	8
3.1.1	测试用例与结果分析	8
3.1.2	测试结果综合分析及建议	8
3.1.3	测试经验总结	8
3.2	疲劳驾驶检测功能	9
3.2.1	测试用例与结果分析	9
3.2.2	测试结果综合分析及建议	9
3.2.3	测试经验总结	9
3.3	历史分析功能	10
3.3.1	测试用例与结果分析	10
3.3.2	测试结果综合分析及建议	10
3.3.3	测试经验总结	10
3.4	导航功能	11
3.4.1	测试用例与结果分析	11
3.4.2	测试结果综合分析及建议	11
3.4.3	测试经验总结	11
3.5	音乐播放功能	12
3.5.1	测试用例与结果分析	12
3.5.2	测试结果综合分析及建议	12

3.5.3	测试经验总结.....	12
3.6	新闻功能.....	13
3.6.1	测试用例与结果分析.....	13
3.6.2	测试结果综合分析及建议.....	13
3.6.3	测试经验总结.....	13
4	系统测试.....	14
4.1	模型性能测试.....	14
4.1.1	测试用例与结果分析.....	14
4.1.2	测试结果综合分析及建议.....	14
4.1.3	测试经验总结.....	14

文档修订历史

序号	修订原因	版本号	作者	修订日期	备注
1	创建	V1.0.0	队员 A	2020-11-26	
2	完善文档	V1.0.1	队员 B	2020-11-30	
3	细化测试策略与目标	V1.0.2	队员 C	2020-12-10	
4	全方位细节修改	V1.0.3	队员 D	2020-12-30	
5	汇总	V1.0.4	队员 A	2021-1-8	初赛文档
6	完成单元测试	V1.1.0	队员 B	2021-2-12	
7	添加路怒和疲劳驾驶检测测试	V1.2.0	队员 A	2021-2-15	
8	添加历史分析功能	V1.3.0	队员 A	2021-3-1	
9	添加导航功能	V1.4.0	队员 B	2021-3-12	
10	添加音乐功能和新闻功能	V1.4.1	队员 C	2021-3-20	
11	完善细节和排版	V1.4.2	队员 A	2021-4-1	
12	整合	V1.5.0	队员 A	2021-4-9	复赛文档

1 测试计划

1.1 测试策略与目标

测试策略：

1. 分析需求
2. 评估风险
3. 定义测试范围
4. 确定测试方法
5. 尽早并且不断地测试，测试时间要充裕
6. 考虑到合法的输入和不合法的输入，即让程序拥有健壮的鲁棒性
7. 对错误测试结果有确认过程并反馈修改
8. 进行版本控制，制定变更测试文档的流程

测试分层	功能	测试要点	备注
前端	路怒检测	能够检测出用户是否产生路怒，并给出有效的提示。	
	疲劳检测	能够检测出用户是否产生疲劳，并进行有效的提示和最终目的地的引导。	
	历史分析及记录	能够记录用户的过往情绪状态并进行分析呈现	
	导航	能够对用户在开启 GPS 的情况下进行导航。	
	音乐播放	能够在线搜索音乐并进行播放	
	新闻	能够阅览当日最新要闻	
	界面交互	界面优美简洁且易于交互	
后端	路怒检测 api	能够检测出用户是否产生路怒，并将结果正确返回给前端。	
	疲劳检测 api	能够检测出用户是否产生疲劳，并将结果正确返回给前端。	
	历史分析及记录 api	能够记录用户的过往情绪状态存入数据库，并将数据返回给前端	

	导航 api	成功调用百度地图 api	
	音乐播放 api	能够将音乐的 URL 和头图、歌手信息、歌曲信息等返回。	
	新闻 api	成功调用聚合数据 api	
模型部分	文本情绪识别	成功且迅速输出识别结果	
	路怒检测	成功且迅速输出识别结果	
	疲劳驾驶检测	成功且迅速输出识别结果	
Raspberry 端	视频录制	视频录制是清晰且编码压缩的	
	视频转发	转发正常且速度是迅速的	

表 1.1.1 测试分层

目标：

1. 软件无中型或大型 Bug，软件运行稳定
2. 由于环境影响等客观原因，疲劳驾驶和路怒驾驶预期期望达到 85% 以上的正确率，并且具有一定的抗干扰能力
3. 数据传输应当是快速且稳定的
4. 云端处理数据足够地快速且具备一定的并发能力
5. 软件的界面美观得体，是交互友好型的

1.2 测试范围

测试广度：

- 功能测试：检测软件的各项功能是否正常运行
- 可用性测试：软件的使用是否合理易用，呈现是否直观
- 性能测试：对软件的响应速度，事务处理速度等进行评测和评估
- 负载测试：测试云服务器的负载能力
- 安全性和访问控制测试：即非法访问云服务器时，能否做出有效地拦截和屏蔽
- 客户端兼容测试：对于不同尺寸，不同安卓版本的设备进行测试

测试深度：

在广度上我们验证了各个层级各个方面的正确性，而在深度上我们还要加大测试力度，尽量细致地给出具体的性能量化指标，大量轮次地运行某一功能给出数学呈现和运行规律，并深入排查所写代码的潜在安全隐患。

系统前期开发过程中的测试内容主要集中在单元测试部分，当团队完成系统内最小的模块单元时应当对其立即进行单元测试，如前后端接口交互是否正常，

数据收发是否正常，界面呈现是否存在相应的问题，并力求细节生动。

在集成测试阶段，需要重点检验各个系统模块的接口是否卯榫相扣，是否能够正常进行数据通信，检验接口的正确性之后仍需测试组装后的整体功能是否符合预期的表现，排除模块与模块之间的功能相互影响，例如 Android 端同服务器端交互是否影响 Raspberry 同服务器的交互。一旦发现问题要及时想出可靠的办法进行解决。从而避免因局部问题的累加，导致最后的问题被放大化。

系统测试过程中，需要对已经集成好的单元进行彻底的测试，以验证系统的正确性和性能是否能满足整个项目的需求。在此测试阶段应当着重解决各个组成模块代码是否符合开发规范、再次验证接口正确性、分析整体功能是否有误、界面设计是否生动移用、性能和正确性是否达到预期需求这些问题。在保证各项指标符合预期之后，还要对代码进行最终的漏洞排查以及规范化。

在部署运行测试阶段，要查看系统在真实环境中的运行效果，表现是否符合预期，若不符合预期，则应找出导致差异化的原因。此外还要对系统进行最终的高负载鲁棒性测试、可用性测试、易用性测试等。在条件允许时，对用户进行问题收集以及满意度评价，并进行最终的整合修改。

1.3 测试环境

服务端环境:

硬件环境	处理器: 48 Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
	内存: 128GB 2933MHz (32GB*4)
	显卡: NVIDIA GeForce RTX 2080TI
	硬盘: 7.9 TB
软件环境	Ubuntu 16.04.7 LTS

表 1.3.1 服务端环境

移动端环境:

硬件环境	手机型号	OPPO Reno 5 Pro+	小米 11
	内存	8G	8G
	处理器	高通骁龙 865	高通骁龙 888
	ROM 空间	128G	256G
	屏幕尺寸及分辨率	6.55 英寸 2400*1080 像素	6.81 英寸 3200*1440 像素
软件环境	系统版本	Android 11 Color OS 11.1	Android 11 MIUI 12.5

表 1.3.2 移动端环境

树莓派环境:

硬件环境	处理器: BCM2711B0 64 位四核处理器 (ARM 架构)
	内存: 4GB
	硬盘: 16GB
软件环境	Linux version 5.4.83-v7l+ (dom@buildbot) (gcc version 8.4.0 (Ubuntu/Linaro 8.4.0-3ubuntu1)) #1379 SMP Mon Dec 14 13:11:54 GMT 2020

表 1.3.3 Raspberry 环境

网络环境:

中国移动/联通/ 4G + 5G



图 1.3.1 网络结构

2 单元测试

2.1 视频录制模块

2.1.1 测试用例与结果分析

单元测试用例：

用例编号	001			
测试单元描述	摄像头画面获取最佳参数成像			
用例目的	让摄像头根据光照等环境因素摄制最佳成像			
前提条件	Rispberry 4 供电正常，捕获到正确的人脸图像			
特殊的规程说明	无			
用例间的依赖关系	摄像头录制数据依赖 Raspberry 供电正常，用户摆放位置正确			
具体步骤	输入	期望输出	实际输出	备注
1	摄像头录制的湖面	自适应的画面参数调优	调整优化后的最佳范围内参数	无
2	调整后的最佳范围内参数	调整后的最佳范围内图像帧	调整后的最佳视频成像	无

测试结果分析：一般情况下画面质量有明显的提升，但在逆光等不利环境因素下提升效果并不明显。下一步工作任务可着重针对逆光的情况进行优化。

2.1.2 测试结果综合分析及建议

- 测试结果分析：一般情况下画面质量有明显的提升，但在逆光等不利环境因素下提升效果并不明显。下一步工作任务可着重针对逆光的情况进行优化。
- 建议：使用本产品时应尽量避免逆光等不利环境条件；若不得已在逆光环境下工作，则需要适当地对摄像头角度进行调整。

2.1.3 测试经验总结

测试的时候应当首要保证功能正常，继而再进行各种特殊情况下的极限鲁棒性测试和高负荷运转下的极限性能测试。

2.2 视频转发模块

2.2.1 测试用例与结果分析

单元测试用例：

用例编号	002			
测试单元描述	将压缩后的.264 视频传输服务器端			
用例目的	将编码的视频流以高速度、低延迟、低带宽传送给服务端			
前提条件	Raspberry 端生成了正确格式的.264 视频			
特殊的规程说明	无			
用例间的依赖关系	视频转发模块依赖 Raspberry 产生了正确格式的视频			
具体步骤	输入	期望输出	实际输出	备注
1	原始视频	编码压缩后的视频	编码压缩后的视频	无
2	编码压缩后的视频	解码后的原始视频	解码后的原始视频	无

测试结果分析：一般情况下视频体积可压缩 50%左右，视频发送速率提升了 40%，且解码后的视频无成像损失，极大节省了带宽和传送时间。

2.2.2 测试结果综合分析及建议

- 测试结果分析：一般情况下视频体积可压缩 50%左右，视频发送速率提升了 40%，且解码后的视频无成像损失，极大节省了带宽和传送时间。
- 建议：使用时保证网络畅通，且面部正对摄像头。

2.2.3 测试经验总结

测试时应保证正确的输入，比如画面中是要保证人脸存在的。其次通过测试发现，如果没有给出正确的输入，那么服务端的代码会产生错误。由此，我们联想到一方面可以将摄像头的方向调整以保证正确的人像输入，另一方面可以增加模型的鲁棒性，比如检测到画面中不含人脸，则给出前端相应的提示，提醒用户调整摄像头位置。

2.3 服务端模型模块

2.3.1 测试用例与结果分析

单元测试用例：

用例编号	003			
测试单元描述	将音频转化后的文本，视频作为输入，服务端给出相应的情绪识别结果			
用例目的	将输入转化为相应的情绪识别结果			
前提条件	输入格式正确			
特殊的规程说明	无			
用例间的依赖关系	服务端代码依赖正确的输入			
具体步骤	输入	期望输出	实际输出	备注
1	原始视频	情绪识别结果	情绪识别结果	无

测试结果分析：语音识别速度快，可以在 1-2s 内跑出结果，判断准确率可达 90%；路怒视频模型和疲劳检测视频模型正确率较高，可达 85%左右。

2.3.2 测试结果综合分析及建议

- 综合结果分析：语音识别速度快，可以在 1-2s 内跑出结果，判断准确率可达 90%；路怒视频模型和疲劳检测视频模型正确率较高，可达 85%左右。
- 建议：视频输入时间应当较短，但发送频率要高。这样既能保证模型出结果返回速度快，又能保证不遗漏捕捉用户每时每刻的驾驶状态。

2.3.3 测试经验总结

测试时应保证正确的输入，比如画面中是要保证人脸存在的。其次通过测试发现，如果没有给出正确的输入，那么服务端的代码会产生错误。由此，我们联想到一方面可以将摄像头的方向调整以保证正确的人像输入，另一方面可以增加模型的鲁棒性，比如检测到画面中不含人脸，则给出前端相应的提示，提醒用户调整摄像头位置。另一方面，语音转化为文本比较简单，模型的鲁棒性非常强。

3 功能测试

3.1 路怒检测功能

3.1.1 测试用例与结果分析

测试用例：

用例编号	001			
功能描述	检测司机是否产生路怒，并进行友好的提醒			
用例目的	检测司机是否产生路怒并进行提醒			
前提条件	服务端正确返回结果			
特殊的规程说明	无			
用例间的依赖关系	路怒检测功能依赖服务端正确返回结果			
具体步骤	输入	期望结果	实际结果	备注
1	语音和视频	服务端产生相应结果	服务端产生相应结果	
2	服务端的结果	前端接收结果	前端接收结果	
3	前端接受的结果	根据结果产生相应动作	根据结果产生相应动作	...

测试结果分析：功能正常，判断正确率在 85%-90%之间，延迟在 5s 以内。

3.1.2 测试结果综合分析及建议

- 测试结果分析：功能正常，判断正确率在 85%-90%之间，延迟在 5s 以内。
- 建议：用户正常使用即可。

3.1.3 测试经验总结

功能上出问题，首先要由表及里、自顶向下分析原因，表象出现了问题，则要深入分析代码上的错误或者漏洞。涉及到 Debug 的问题，则要善于查看报错信息，以及利用 Log 日志和 Debug 工具，抽丝剥茧式地逐层分析，精确定位，找到问题代码。要归纳总结究竟犯了什么错误，例如：如果是空指针异常，则可以提前加个 if 判断指针为空的情况，或者利用 try/catch 捕获问题代码。要切实考虑清楚是思维的不严谨还是设计模式的使用不当。思维的不严谨可以通过多写代

码来提升；设计模式如果使用不当，则要多从项目架构方面深入考虑，优化设计出最良好的设计模式。

3.2 疲劳驾驶检测功能

3.2.1 测试用例与结果分析

测试用例：

用例编号	002			
功能描述	检测司机是否产生疲劳驾驶，并进行友好的提醒和引导休息			
用例目的	检测司机是否产生疲劳驾驶并进行提醒和休息引导			
前提条件	服务端正确返回结果			
特殊的规程说明	无			
用例间的依赖关系	疲劳驾驶检测功能依赖服务端正确返回结果			
具体步骤	输入	期望结果	实际结果	备注
1	视频	服务端产生相应结果	服务端产生相应结果	
2	服务端的结 果	前端接收结果	前端接收结果	
3	前端接受的 结果	根据结果产生相 应动作	根据结果产生 相应动作	...

测试结果分析：功能正常，判断正确率在 85%左右，延迟在 5s 以内。

3.2.2 测试结果综合分析及建议

- 测试结果分析：功能正常，判断正确率在 85%左右，延迟在 5s 以内。
- 建议：用户正常使用即可。

3.2.3 测试经验总结

功能上出问题，首先要由表及里、自顶向下分析原因，表象出现了问题，则要深入分析代码上的错误或者漏洞。涉及到 debug 的问题，则要善于查看报错信息，以及利用 log 日志和 Debug 工具，抽丝剥茧式地逐层分析，精确定位，找到问题代码。要归纳总结究竟犯了什么错误，例如：如果是空指针异常，则可以提前加个 if 判断指针为空的情况，或者利用 try/catch 捕获问题代码。要切实考虑清楚是思维的不严谨还是设计模式的使用不当。思维的不严谨可以通过多写代码来提升；设计模式如果使用不当，则要多从项目架构方面深入考虑，优化设计出最

良好的设计模式。

3.3 历史分析功能

3.3.1 测试用例与结果分析

测试用例：

用例编号	003			
功能描述	对用户过往驾驶情绪和状态进行分析			
用例目的	对用户过往驾驶情绪和状态进行记录和分析			
前提条件	服务端存储相应数据			
特殊的规程说明	无			
用例间的依赖关系	依赖于服务端存储相应数据			
具体步骤	输入	期望结果	实际结果	备注
1	服务端产生的结果	结果存入数据库并在前端进行呈现	结果存入数据库并在前端进行呈现	

测试结果分析：功能正常，呈现形式形象生动。

3.3.2 测试结果综合分析及建议

- 测试结果分析：功能正常，呈现形式形象生动。
- 建议：用户正常使用即可。

3.3.3 测试经验总结

- 应当注意 Hello-charts 折线图日期的正确性和饼状图的区域划分问题，否则会导致数据呈现错误。
- Hello-charts 所有图应考虑用户所选中选项时数据库无数据的情况，要有默认数据（100%正常），避免程序崩溃
- 多下拉菜单应测试下拉菜单之间的相互触发关系，以及最小菜单选中后对图标的刷新情况是否正确。

3.4 导航功能

3.4.1 测试用例与结果分析

测试用例:

用例编号	004			
功能描述	为司机驾驶指引道路, 并配合疲劳驾驶功能指引最近的休息地点			
用例目的	为司机驾驶指引道路, 并配合疲劳驾驶功能指引最近的休息地点			
前提条件	开启 GPS 功能			
特殊的规程说明	无			
用例间的依赖关系	依赖手机内置 GPS			
具体步骤	输入	期望结果	实际结果	备注
1	目的地	产生目的地指引	产生目的地指引	

测试结果分析: 导航功能正常, 细节完备。

3.4.2 测试结果综合分析及建议

- 测试结果分析: 导航功能正常, 细节完备。
- 建议: 用户正常使用即可。

3.4.3 测试经验总结

出现 bug 时, 首先运行编译器的调试功能, 逐行运行代码, 查看里面的值是不是预想的值, 代码的运行顺序是不是正确的, 若不是, 则对问题代码进行分析、更改, 直到在调试中代码运行正确, 调试值正确为止。比如地图中涉及到地点加载的问题, 如果给一个匹配库中不存在的地点的参数, 则会引发报错。所以要严格注意这类问题。

3.5 音乐播放功能

3.5.1 测试用例与结果分析

测试用例:

用例编号	005			
功能描述	音乐点播和播放功能			
用例目的	为司机进行音乐点播和音乐播放			
前提条件	开启网络			
特殊的规程说明	无			
用例间的依赖关系	依赖网络顺畅和数据库音乐的匹配			
具体步骤	输入	期望结果	实际结果	备注
1	歌手名称/歌曲名称	返回歌曲的 url 和图片名称等	返回歌曲的 url 和图片名称等	

测试结果分析: 功能正常, 细节比较完备生动。

3.5.2 测试结果综合分析及建议

- 测试结果分析: 功能正常, 细节比较完备生动。
- 建议: 用户正常使用即可。

3.5.3 测试经验总结

- 由于播放器播放音乐需要不依赖 app 界面依然正常工作, 因此需要测试息屏, 回到桌面, 进去其他 app 音乐播放的正确性。
- 播放器实现时, 开始暂停播放都设计对象的释放和创建等, 因此重复点击暂停开始, 下一首, 上一首等也是关键测试内容。
- 应当边界测试, 当一首音乐播放结束, 自动切换是否成功。当正在播放第一首音乐, 点击上一首是否能正常工作, 当播放最后一首音乐, 点击下一首是否能正常工作。
- 测试当前播放歌曲记录情况, 播放时间点记录情况。保证下次打开音乐功能时恢复到上次工作的情况。

3.6 新闻功能

3.6.1 测试用例与结果分析

测试用例：

用例编号	006			
功能描述	每日要闻查看			
用例目的	为司机提供每日新闻和司机相关新闻			
前提条件	开启网络			
特殊的规程说明	无			
用例间的依赖关系	依赖于网络畅通			
具体步骤	输入	期望结果	实际结果	备注
1	点击感兴趣的新闻条目	返回相应的新闻内容	返回相应的新闻内容	

测试结果分析：功能正常，细节比较完备生动。

3.6.2 测试结果综合分析及建议

- 测试结果分析：功能正常，细节比较完备生动。
- 建议：用户正常使用即可。

3.6.3 测试经验总结

要注意接口规则的正确使用，否则无法从相应的 api 拿到正确的数据。

4 系统测试

4.1 模型性能测试

4.1.1 测试用例与结果分析

预期性能指标测试用例：

用例编号	001			
性能描述				
用例目的				
前提条件				
特殊的规程说明				
用例间的依赖关系				
具体步骤	输入/动作	期望的性能 (平均值)	实际的性能 (平均值)	备注
1				
2				
...

测试结果分析：

4.1.2 测试结果综合分析及建议

4.1.3 测试经验总结