**人海产品需求文档**

**(App部分)**

|  |  |
| --- | --- |
| 最新版本： | 0.1 |
| 最后更新时间： | 2013-08-12 |
| 文档链接： |  |

目录

图表目录 2

修订历史 3

1 概述 3

1.1 文档范围 3

1.2 面向读者 3

1.3 名词解释 3

1.4 术语说明 4

2 异常管理 4

3 网络管理 5

4 设备管理 5

5 服务器管理 6

6 视频聊天管理 6

7 配置管理 6

8 日志管理 6

9 备选需求 7

# 图表目录

图01. 人海App端概念理解图 4

# 修订历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2013-08-12 | 0.1 | 初稿 | Patrick |
|  | 0.2 | 根据初稿评审意见修改 | Patrick |
|  |  |  |  |

# 概述

## 文档范围

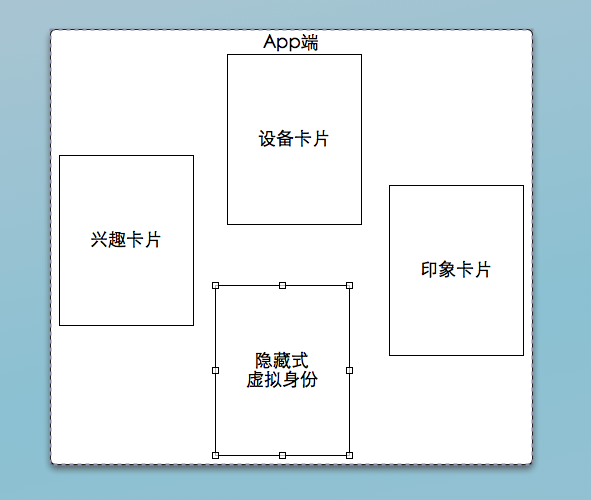
本文档旨在描述人海产品App部分的需求。

## 面向读者

本文档面向的主要读者：人海产品Server开发人员、App开发人员以及其他相关评审人员。

## 名词解释

* **用户**：App的使用者。
* **设备**：App所处设备的硬件实体。
* **设备卡片**：记录**设备**相关信息（包括设备唯一ID、类型、操作系统版本以及聊天相关数据等）的对象。
* **隐藏式虚拟身份：**用来隐性表示**用户**虚拟身份的对象，与**设备卡片**形成多对一的映射关系（目前App端只考虑一个隐藏式虚拟身份），与**用户**形成一对一的映射关系（也就是假定**设备**的**用户**是一个人）。
* **兴趣卡片：**记录**用户**兴趣话题列表的对象。
* **印象卡片：**记录**用户**在聊天后被对方给出的印象评价和印象关键字标签列表的对象。



1. 人海App端概念理解图

***🕮 说明：***

信令交互图以及App业务流程图参见需求目录下对应的文件，这里不再描述。

## 术语说明

|  |  |
| --- | --- |
| 缩略语 | 描述 |
| WebSocket | 基于HTTP协议实现的客户端服务器双向Socket通信的HTML5规范 |
| WebRTC | 基于HTTP协议实现的视频通话的HTML5规范 |

# 异常管理

1. WebRTC通信异常

当视频通话阶段WebRTC通信发生异常时，App需要在界面上向用户发送合适的提示信息，并在后台即时向Server发送通知，以保证业务流程的正常走向。

1. WebSocket通信异常

任何时候，如果App端的WebSocket通信异常，需要在界面上向用户发送合适的提示信息，并给出自动/手动重新连接的机会。

# 网络管理

1. WebRTC环境检测

当App启动后首先需要检测所处网络环境是否支持WebRTC，如果可以则继续业务流程，如果不行则需要中止业务流程并提示用户：

1. WebSocket连接建立

App需要和Server建立WebSocket连接，所有通信消息均基于此通道进行传递。

1. WebSocket连接维持

App需要通过定时发送心跳消息（WebSocket协议帧）来和Server保持WebSocket连接。

1. WebSocket连接关闭

App进入一些特定状态时，例如切换到后台并延续一定时间（10分钟），需要关闭WebSocket并释放相关资源。当App重新回到前台时，需要自动重连Server并完成接入部分的业务流程。

1. WebSocket连接断开

App发生任何原因引起的WebSocket连接断开时，需要按照**WebSocket通信异常**（EX\_02）流程处理。

1. WebSocket重新连接

App发生WebSocket连接断开之后，应提供重连机制：建议方案为：先自动重连（重试N次，每次间隔M秒），重连失败后，提供手动重连界面（每次手动重连之间需要间隔X秒）。

1. 基于WebSocket的JSON消息通信

* JSON消息需要加密/解密（实现方案与Server一致）
* JSON消息需要压缩/解压缩（实现方案与Server一致）

1. 离线环境

* App需要从本地缓存中读取设备卡片数据并显示
* App需要从本地缓存中读取兴趣卡片数据并显示
* App需要从本地缓存中读取印象卡片数据并显示

# 设备管理

1. 设备卡片更新

* App每次连接时，需要向Server提交设备相关的不可变信息，主要是设备唯一ID和设备类型等。
* App每次连接时，需要向Server提交设备相关的可变信息，主要是操作系统版本，App版本等。

1. 设备卡片查询

App每次连接时，向Server要求查询设备相关信息，主要是服务状态和聊天数据相关。

1. 兴趣卡片查询

* App每次连接时，向Server要求查询兴趣卡片相关信息。
* App显示兴趣卡片界面时，向Server要求查询兴趣卡片相关信息。
* App显示兴趣卡片界面时，向Server要求查询当前/历史热门兴趣标签列表。

1. 兴趣卡片更新

用户修改了兴趣卡片后，向Server更新兴趣卡片相关信息。

* 修改兴趣卡片包括两种情况：新增、修改、删除某个兴趣标签；调整兴趣标签列表排序。
* 在线状态下，用户修改了兴趣卡片后，立即向Server更新兴趣卡片相关信息。
* 离线状态下，用户修改了兴趣卡片后，保存在缓存内，下一次上线后会向Server更新兴趣卡片信息。

1. 印象卡片查询

* App每次连接时，向Server查询自己的印象卡片相关信息。
* App在业务会话的双方确认状态时，向Server查询对方的印象卡片相关信息。

# 会话管理

1. 会话开始

App收到Server的业务会话请求后，会进入查看对方印象卡片的界面，在限定的时间内，由用户决定是否和对方进行视频聊天，如果拒绝，则由Server重新进行匹配调度；如果同意，则等待对方决定。这个阶段界面上应该显示对方的选择状态（查看中/已同意/已拒绝）。

1. 聊天开始

用户进入视频聊天的界面。

1. 聊天结束

用户在视频聊天界面可以手动结束，App会进入评价对方印象的界面，由用户根据聊天感受对对方进行评价。

1. 会话结束

用户完成对对方的印象评价之后，会话结束，用户可以选择继续等待Server匹配下一个聊天对象，也可以选择退出回到主界面。

# 服务器管理

1. 服务器状态查询

* App连接Server之后，可以定时向Server查询当前在线用户总数，并显示在界面上。
* App连接Server之后，可以定时向Server查询当前业务设备池的用户总数，并显示在界面上。

# 视频聊天管理

1. 视频聊天手动中止

App应允许用户手动中止视频聊天。

1. 视频聊结束后的业务流程

* 用户可选择继续等待匹配
* 用户可选择退出

1. 视频聊天界面布局自定义

App应允许用户根据自己习惯对视频聊天界面进行重排。

1. 印象卡片更新

App在业务会话的双方评价状态时，向Server更新对方的印象卡片相关信息。

# 配置管理

1. 推送通知开关

App应允许用户开启/关闭推送通知

1. 反馈信息

App应允许用户通过简单反馈界面快速提交意见和建议

1. 帮助信息

* App首次启动时应先展示帮助向导
* App应允许用户手动查看帮助向导

1. 3G环境下是否允许上线的开关
2. 关于

* App应提供开发者的相关信息

# 日志管理

1. 保存日志记录

App中所有业务操作都要有对应的日志记录，并保存在文本文件中。

1. 保存崩溃记录

App的崩溃需要被记录并提供后期分析查看的途径（可借助第三方解决方案）。

# 备选需求

1. App应在CM\_04中提供版本历史功能列表