

2014-11-14

反拖延系统需求规格说明

开发组

121250101 缪晓伟 121250006 查燚斐 121250128 宋程宇

121250220 郑轲阳

目录

- 1. 引言 1
 - 1.1 目的 1
 - 1.2 范围 1
 - 1.3 定义、首字母缩写和缩略语 1
 - 1.4 文档组织 2
 - 1.5 参考文献 2
- 2. 总体描述 2
 - 2.1 商品前景 2
 - 2.1.1 背景与机遇 2
 - 2.1.2 业务目标 2
 - 2.3 用户特性 3
 - 2.4 约束 3
 - 2.5 假设和依赖 3
 - 2.6 限制与排除 3
- 3. 详细需求描述 4
 - 3.1 对外接口需求 4
 - 3.1.1 用户界面 4
 - 3.2 功能需求 9
 - 3.3 非功能性需求 13
 - 3.3.1 安全性 13
 - 3.3.2 可维护性 13
 - 3.3.4 可靠性 14
 - 3.3.5 业务规则 14
 - 3.3.6 约束 14
 - 3.4 数据需求 14
 - 3.4.1 数据定义 14
 - 3.4.2 默认数据 14
 - 3.4.3 数据格式要求 15
 - 3.5 质量需求 15
 - 3.6 其他需求 15

3.6.1	安装需求	15
3.6.2	性能需求	15
附录	15

1. 引言

1.1 目的

本文档描述了“Ding”反拖延与效率应用的功能需求和非功能需求。开发小组的软件系统实现与验证工作都以此文档为依据。

除特殊说明外，本文档所包含的需求都是高优先级需求。

1.2 范围

“Ding”反拖延与效率应用是需求是客户面对当前人群中普遍存在的拖延症和做事效率不高的问题，为提高工作生活学习效率，聘请需求是我们需要个需求工作组开发的一个反拖延效率类应用。

1. 第一版范围

在软件的最初版本中，学生和白领要能够记录未来的日程安排；应用根据记录的日程安排提醒即将到来的 **deadline**；没有工作可做时，应用提供的建议从这个用户的历史日程中获取；工作、学习时段，不允许学生和白领主动打开设备上的娱乐应用；督促工作学习；工作学习一段时间后需要放松时，向用户给出提示。

2. 后续版本范围

在第一版范围中，SF-1 和 SF-2 已被完全实现。后续版本的范围定义如下表所示。

特性	版本 2	版本 3
SF-3	若开发进度能够满足，则将建议的来源扩大到其他学生和公司职员在同一时段的日程安排	建议的来源扩大到其他学生和公司职员历史上的和同一时段的日程安排
SF-4	在不允许用户主动打开娱乐应用的前提下，屏蔽这些应用的弹窗和消息推送	
SF-5	若开发进度能够满足，向用户展示此时此刻其他正在工作学习的学生、白领的日程安排	完全实现
SF-6	提供放松的形式从提示转变为提示得到用户认可后自动开启其它娱乐应用	
SF-7	不实现	大学生和白领可以查看开放了权限的其他用户的日程安排

1.3 定义、首字母缩写和缩略语

1.3.1 缩略语：

Ding = “Ding”反拖延与效率应用

DDL = deadline

1.3.2 定义：

拖延症：

拖延症是指自我调节失败，在能够预料后果有害的情况下，仍然把计划要做的事情往后推迟的一种行为。

放松：

在本文档中，“放松”被定义为能够消除因工作、学习而产生的疲劳

的一切行为，如听音乐、玩游戏、刷微博、刷人人等。

用户通常工作时间：

通常情况下，大学生的工作时间会安排在 8:00-22:00 这个时间段之内。对某些专业（如 CS、SE 等）的学生而言，工作时间可能被整体推后 4~6 个小时。对公司白领而言，典型的工作时间为朝九晚五，不含午休时间，但不排除一些白领需要晚上加班，但通常加班时间不会超过 23:00。

1.4 文档组织

本文档包括总体描述、详细需求描述两大内容，包含产品前景，产品功能，用户特性，约束；假设和依赖，对外接口需求，功能需求，性能需求，约束，质量属性，其他需求；附录。

采用标准多级序号样式。

1.5 参考文献

- 1) IEEE 标准。
- 2) 用例文档
- 3) 项目前景与范围文档
- 4) 面谈报告
- 5) 目标描述列表

2. 总体描述

2.1 商品前景

2.1.1 背景与机遇

目前，不少大学生和白领们正被拖延症困扰。一方面，**deadline** 近在眼前，学生和职员们的压力巨大；另一方面，他们却仍会在这种情况下毫无计划地度过一整天。除此以外，在他们最终下定决心开始工作时，拖延症又会导致工作效率低下，令他们不由自主地在工作过程中随手刷微博、刷人人，或是被手中的股票行情吸引，或是顺手给自己的某宝购物车添加新成员，或是做其它与工作学习无关的事，导致好不容易下的决心也没能产生什么成效。

市场上现有产品并未能对此问题提供有效的解决办法。

大量大学生和白领在使用 **Ding** 后，拖延症的症状有所改善，工作、学习的效率得到了提高。在每个 **deadline** 迫近前，应用给学生和白领们及时做出提示，好让他们有足够的时间来应对，缓解因猝不及防的 **deadline** 而产生的焦虑；当他们在某一天无所事事时，应用向他们提示，今天还有什么工作可做，驱使他们勤勉地度过每一天；在真正的工作学习过程中，应用给学生和白领们提供了有效的监督，不让他们因其它因素而分心，同时在工作学习一段时间后给出温馨的提示，建议他们稍事休息，从而保证了工作的效率，学生也不致过于疲劳。系统从勤奋、焦虑和疲劳三方面同时作用，有效解决了导致拖延症产生的三大诱因。

2.1.2 业务目标

业务目标：

BO-1: 在使用应用后的 1 个月内，拖延症的症状得到改善。

度量标准：用户的 **deadline** 与对应任务的实际完成时间。

计量方法：向用户调查最近 1 个月内所有任务的完成情况。

理想标准：所有任务在 **deadline** 到期前完成；一般标准：不超过一定数目的 **deadline** 未能达成，相应的数目取 2 个和最近 1 个月内任务总数 5% 中的较大值；最低标准：所有 **deadline** 的延期不超过规定时限的 15%。

BO-2: 在使用应用后的 1 周内，工作效率得到提高。

度量标准：用户每日除休息时间以外，工作、学习时间占剩余时间的百分比。
计量方法：向用户调查使用应用前后的详细日程安排，由应用监控既定的工作、学习时段内，真正用于工作学习的时间。

理想标准：真正用于工作的时间达到 90%；一般标准：真正用于工作的时间达到 75%；最低标准：真正用于工作的时间占比不低于使用应用前。

成功标准：

SC-1：在应用发布后的 1 个月后，使用这款应用的大学生和公司职员数量不少于 500。

SC-2：在使用本应用第一版后 1 个月内，用户在工作学习时间刷人人、微博、逛淘宝等的时间比使用前减少；使用本应用后续版本后 1 个月内，用户不再主动在工作学习时间进行上述活动。

2.2 商品功能

SF-1：记录学生和白领未来的日程安排。

SF-2：提醒学生和白领即将到来的 deadline。

SF-3：为无所事事的日程安排提供建议。

SF-4：监督学生和白领在工作、学习过程中注意力不被其它因素分散。

SF-5：督促学生勤勉学习，督促公司职员勤奋工作。

SF-6：在工作、学习中给学生和白领提供适当的放松。

SF-7：在用户允许的前提下向其他用户提供他（她）的工作日程安排。

2.3 用户特性

目标用户	每个终端上的该 APP 面向一名目标用户。包括大学生和公司白领。他们绝大多数为 90 后，早已习惯了数字生活给他们带来的巨大便利，普遍敢于并乐于接受新生事物。他们的生活离不开电子设备，因而能够熟练操作 PC、智能手机、平板电脑等各种设备。作为脑力劳动者，他们以学习和工作为主要任务，几乎每天都在进行，但苦于当下的学习效率低下，并受到拖延症的困扰。在学生和白领内部，随处可见以同学、同乡、同事关系维系着的若干个小团体。 他们对这款应用表现出强烈的兴趣，很想得知这个应用会以什么方式提高他们的工作效率。对他们来说，只要这款应用能提高他们哪怕是一丁点的工作效率，都给他们提供了莫大的帮助。这个应用可能会令其中的大学生改变原有的生活作息习惯，并在使用这个应用的初期令他们感到不适应。
------	--

2.4 约束

CON1：本产品 在 iOS、Andriod、Windows Phone 8、Windows 8 及以上版本运行

CON2：系统不使用 WEB 界面，而是图形界面。

CON3：项目使用瀑布模型进行开发。

CON4：在开发中，开发者要提交软件需求规格文档、设计描述文档和测试报告。

CON5：多平台开发要同步进行。

CON6：项目开发要在 4 个月内完成。

CON7：项目要由四人开发小组合作完成。

CON8：每一次项目开发会议时所有小组成员都必须参加。

2.5 假设和依赖

AS-1：使用应用的目标用户会在其使用的所有设备上安装此应用。

AS-2：目标用户在使用应用时如实记录了未来的日程安排。

AS-3：目标用户在实际生活中有意愿遵从应用的提醒来工作、学习。

DE-1：应用要能启动、中止一些特定的应用，如音乐播放器、微博、人人等。

2.6 限制与排除

LI-1：目标用户 在应用中记录的只是学习、工作的日程安排，休息时间的安排不在

其中。

UI-2: 应用只保证目标用户在工作、学习中，注意力不被其所拥有的设备吸引。

3. 详细需求描述

3.1 对外接口需求

3.1.1 用户界面

用户界面总体使用 flat ui 风格，并且尽量把相近功能放在同一界面内，尽量减少界面跳转和弹窗。

尽量使用滑动操作和动态提示，利用感官刺激和行为表现提高用户“完成DDL”的成就感。

字体和图标需在各种分辨率下能够清晰辨认。

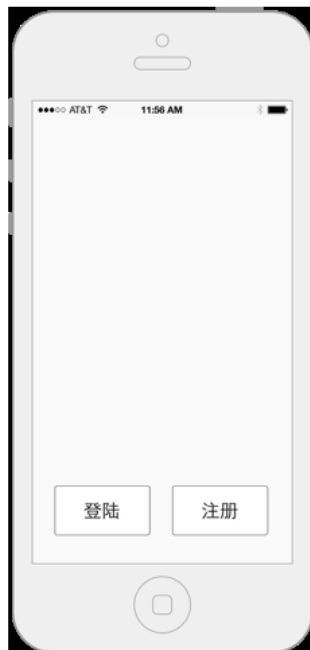
以下说明主要侧重移动应用用，重点在于表现移动应用用上主要的交互过程和主要应用用场景

由于各平台在设计语言言上的区别，原型上并没有覆盖全部平台，而而是选择 IOS 作为主要的媒介，其他平台在界面面区分和表现上基本与 IOS 平台一一一致，功能点上也完全一一一致

由于 Android 平台在应用用权限上的不同设计，可以达到完全屏蔽其他应用用的功能，但在 WindowsPhone 和 IOS 平台上，这点是无无法做到的，只能通过其他手手段来完成（如原型中所展示示的方方式）

对应 PC 和 MAC 端，设计思路有所上有所不同，但在第一一版原型中并没有加入 PC 和 MAC 的设计，会在后续工工作中补上。同时 PC 和 MAC 端由于无无法知晓用用户真正的意图和所浏览的内容，将主要以计时和提醒为主，不会像移动端一一进行行一一一定程度上的屏蔽

UI1: Starting 界面:



本 APP 的第一个界面，提供登陆和注册选项。点击登陆跳转至 UI2。点击注册跳转至 UI3

UI2: 注册界面:



新用户注册界面。点击下方的注册进行注册。左滑返回 UI1。注册成功进入 UI4

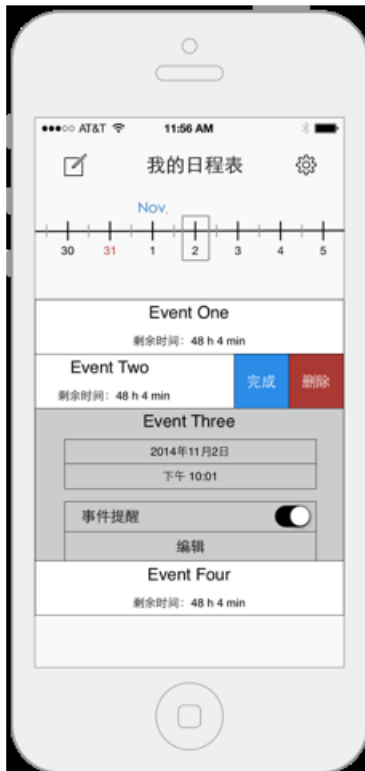
要求信息提示明确，让用户知晓需要输入的是什么信息。输入格式有误需有反馈。

UI3: 登陆界面



用户登录界面。登陆成功转入 UI4。左滑进入 UI1。登录失败停留在此屏。需要在登陆失败或成功时均有明确的反馈。登陆等待过程中有动画提示进度。

UI4: 主界面



主界面。一条 TIMELINE 标识当前 DDL 分布，下方显示 DDL 详情。可在同一屏便捷地进行编辑。点击左上角的添加按钮进入 UI5，点击右上角的设置按钮进入 UI6。点击时间线进入 UI8

UI5：添加新事件



添加新的日程安排。突出 DDL 的概念。添加成功或点击左上角返回按钮返回 UI4

UI6: 设置界面



设置界面，定义用户的工作时间。绑定用户的社交账号。控制是否同步惩罚消息到社交媒体。点击“我的账户”进入 UI7。点击注销，注销账号并进入 UI3。可返回 UI4。

UI7: 我的账号



查看账户信息，用户名，绑定的邮箱和学校/单位。可以在此界面修改密码。通过保存或返回可返回 UI4

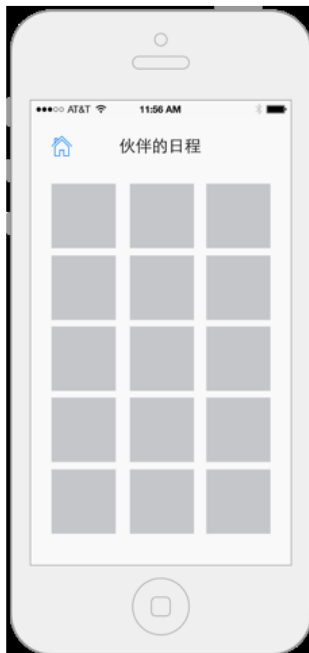
UI8: 走过的历史



“走过的历史”界面。可以回顾过去的 DDL 完成情况。可以以周和天两种视图来看。左滑返回 UI4

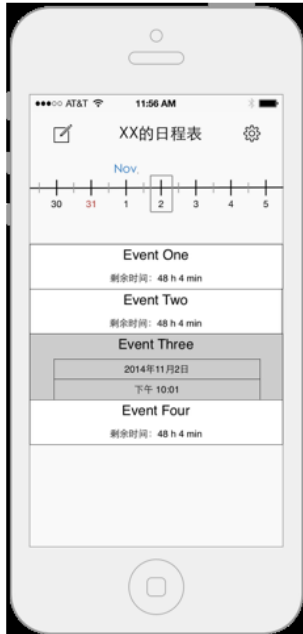
UI9: 伙伴的日程

UI9.1 伙伴的日程



查看自己的伙伴（好友）列表。好友以头像方式排列。

UI9.2 XXX 的日程表



呈现方式和主界面中我的日程表类似。

UI10 工作时警告和提醒



在用户设定的工作时间对用户进行提醒和警告。要求在达到警示效果的前提下不让用户感到反感。

3.2 功能需求

3.2.1: 记录日程安排

3.2.1.1 特性描述

用户输入日程的时间、地点、事件，选择是否提醒、是否开放其他用户的查看权限（版本三内容），系统记录新的日程安排

优先级：高

3.2.1.2 刺激/响应序列

- 刺激：用户选择安排日程任务
- 响应：系统显示添加日程界面
- 刺激：用户输入日程的时间、地点、事件，选择是否提醒、是否开放其他用户的查看权限（版本三内容）并确认输入
- 响应：系统验证时间，通过后记录新的日程安排

3.2.1.3 相关功能需求

Ding.Event	系统允许用户进行日程安排
Ding.Event.add	用户添加日程安排
Ding.Event.modify	用户修改日程安排
Ding.Event.cancel	用户添加或修改日程安排过程中取消，系统要求用户确认
Ding.Event.check	用户完成输入后，系统查看日程安排是否成功设置，并返回结果
Ding.Event.end	用户有效输入并确认后，系统执行保存并更新数据的任务
Ding.Event.quit	系统允许用户退出当前界面
Ding.Event.update	系统向服务器发送更新数据的请求
Ding.Event.show	系统显示日程安排

3.2.2: 提醒 deadline

3.2.2.1 特性描述

- 用户输入 deadline 的时间和具体任务，系统记录新的 deadline
- 优先级：高

3.2.2.2 刺激/响应序列

- 刺激：用户选择记录 deadline 任务
- 响应：系统显示 deadline 修改界面
- 刺激：用户输入 deadline 的最后截止时间和具体任务并确认输入
- 响应：系统验证时间，通过后记录新的 deadline

3.2.2.3 相关功能需求

Ding.Deadline	系统允许用户设置 Deadline
Ding.Deadline.add	用户添加 Deadline
Ding.Deadline.modify	用户修改 Deadline
Ding.Deadline.cancel	用户添加或修改 Deadline 过程中取消，系统要求用户确认
Ding.Deadline.check	用户完成输入后，系统查看 Deadline 是否成功设置，并返回结果
Ding.Deadline.end	用户有效输入并确认后，系统执行保存并更新数据的任务
Ding.Deadline.quit	系统允许用户退出当前界面
Ding.Deadline.update	系统向服务器发送更新数据的请求
Ding.Deadline.show	系统显示 Deadline 安排

3.2.3: 强制启动学习/工作模式

3.2.3.1 特性描述

- 系统推送建议开始学习/工作，用户确认后学习/工作模式即时强制开启

优先级：高

3.2.3.2 刺激/响应序列

刺激：用户已经登录，系统中存有用户要求提醒的 deadline 安排，在 deadline 即将到来之前三天内无任何已设置的学习/工作时间段。

响应：系统推送建议开始学习/工作

刺激：用户选择确认

响应：系统强制开启学习/工作模式

3.2.3.3 相关功能需求

Ding.block	系统强制启动学习/工作模式的功能
Ding.block.check	系统检查是否达到提醒条件
Ding.block.start	系统强制启动学习/工作模式
Ding.block.stop	系统停止学习/工作模式
Ding.block.cancel	用户取消开启强制模式

3.2.4：记录 deadline 完成情况

3.2.4.1 特性描述

系统提醒 deadline 已经到达，用户选择是否完成

优先级：高

3.2.4.2 刺激/响应序列

刺激：到达 deadline 最后期限

响应：系统推送消息，提醒用户 deadline 已经到达，请求用户选择任务是否完成

刺激：用户选择是否完成

响应：系统记录 deadline 任务完成情况

3.2.4.3 相关功能需求

Ding.ifEnd	系统记录 deadline 完成情况
Ding.ifEnd.check	判断是否达到 deadline
Ding.ifEnd.ask	系统询问用户任务完成情况
Ding.ifEnd.update	系统更新任务完成情况

3.2.5：查看 deadline 完成情况统计

3.2.5.1 特性描述

用户选择查看 deadline 完成情况统计，系统给出展示。目标是方便了解此前 deadline 完成情况及使用该系统之后效率的变化趋势

优先级：低

3.2.5.2 刺激/响应序列

刺激：用户选择查看 deadline 完成情况统计任务

响应：系统读取过往 deadline 完成情况数据并用表格和图表整理后显示

3.2.5.3 相关功能需求

Ding.count	系统具备 DDL 完成情况统计功能
Ding.count.show	展示 DDL 完成情况统计
Ding.count.clear	清除 DDL 完成情况统计

3.2.6：绑定社交网络

3.2.6.1 特性描述

用户选择需要绑定的社交网络，输入相关信息并同意授权，目标是让系统快速取得相关社交网络的授权

优先级：中

3.2.6.2 刺激/响应序列

- 刺激：用户选择绑定社交网络任务
- 响应：系统显示各社交网络图标允许用户选择
- 刺激：用户选择相应社交网络图标
- 响应：系统允许用户输入用户名和密码
- 刺激：用户输入用户名和密码并确认
- 响应：系统验证用户名和密码，通过后请求用户授权
- 刺激：用户选择同意授权
- 响应：系统取得该社交网络授权，提示绑定成功

3.2.6.3 相关功能需求

Ding.sns	系统具备社交网络相关功能
Ding.sns.add	添加绑定
Ding.sns.cancel	用户取消添加或修改绑定
Ding.sns.modify	用户修改绑定
Ding.sns.show	获取当前社交网络绑定信息
Ding.sns.update	更新社交网络绑定信息

3.2.7：设定学习/工作时间段

3.2.7.1 特性描述

- 用户输入学习/工作起止时间，选择重复频率，目标是准确快速地设定学习/工作时间段，以便在这个时间段内保证高效率的学习与工作
- 优先级：中

3.2.7.2 刺激/响应序列

- 刺激：用户选择设定学习/工作时间段任务
- 响应：系统进入设定界面
- 刺激：用户输入学习/工作起止时间，选择重复频率，选择确认
- 响应：系统验证时间，通过后保存学习/工作时间段

3.2.7.3 相关功能需求

Ding.worktime	系统允许用户设置学习/工作时间段
Ding.worktime.add	用户添加 worktime
Ding.worktime.modify	用户修改 worktime
Ding.worktime.cancel	用户添加或修改 worktime 过程中取消，系统要求用户确认
Ding.worktime.check	用户完成输入后，系统查看 worktime 是否成功设置，并返回结果
Ding.worktime.end	用户有效输入并确认后，系统执行保存并更新数据的任务
Ding.worktime.quit	系统允许用户退出当前界面
Ding.worktime.update	系统向服务器发送更新数据的请求
Ding.worktime.show	系统显示 worktime 安排

3.2.8：进入学习/工作时间段

3.2.8.1 特性描述

- 在设定的学习/工作时间段内，拒绝用户主动打开娱乐应用；在一段时间后提醒用户休息，允许用户打开娱乐应用运行一段时间或者得到确认后自动打开一个娱乐应用并运行一段时间（版本二内容）

优先级：高

3.2.8.2 刺激/响应序列

刺激：时间到达学习/工作时间段

响应：系统提醒并显示“请开始学习/工作”

刺激：用户选择确认

响应：系统显示“正在学习/工作”字样，屏蔽娱乐应用的弹窗和消息推送

3.2.8.3 相关功能需求

Ding.pushToWork	系统关于进入学习/工作时间段的功能
Ding.pushToWork.check	系统检查是否达到提醒条件
Ding.pushToWork.start	系统进入学习/工作时间段
Ding.pushToWork.stop	系统退出学习/工作时间段
Ding.pushToWork.cancel	用户取消进入学习/工作时间段

3.2.9：查看他人日程

3.2.9.1 特性描述

用户选择查看其他用户日程，系统查找其他用户开放了权限的日程并显示。目标是查看其他用户开放了权限的日程安排，以便在安排自己的日程时获得参考

优先级：低

3.2.9.2 刺激/响应序列

刺激：用户选择查看他人日程的任务

响应：系统读取其他用户开放了权限的日程并显示

3.2.9.3 相关功能需求

Ding.partner	系统关于伙伴的功能
Ding.partner.showSchedule	查看伙伴的日程
Ding.partner.showSchedule.check	判断伙伴是否开放了日程查看

3.3 非功能性需求

3.3.1 安全性

Safety1：系统应该只允许经过验证和授权的用户访问。

Safety2：系统应该按照用户身份验证用户的访问权限。

用户的身份授权参见功能需求 3.2.XX；

其他用户没有访问权限。

Safety3：系统使用加密方式存储用户密码。

Safety4：进行编辑操作时，超过 5min 没有操作则自动放弃修改并返回上一层界面。

Safety5：登陆系统后，超过 10min 没有操作则自动注销登陆并清除登录信息。

Safety6：同一账号连续尝试登录 5 次密码皆错误，30min 后才能再次尝试。
1 天内累计登录 10 次密码皆错误，锁定账号 1 天。

3.3.2 可维护性

Modifiability1：如果系统新增某用户的功能，要能在 0.25 个人月内完成

Modifiability2：如果系统新增某项目的新属性，系统要能在 4 人 1 天内完成。

3.3.3 易用性

Usability1：多平台的应用保持设计语言的统一

Usability2：各类信息提供格式化导出功能

Usability3：用户新注册使用时有使用导引

Usability4: 用户确认输入后，应用要在 0.3 秒内完成信息验证并对添加结果给出提示

Usability5: 系统提示时使用的振动、铃声、弹窗等手段的强度要足够引起用户注意

3.3.4 可靠性

Reliability1: 在客户端与服务器通信时，如果网络故障，系统不能出现故障。

Reliability1.1: 客户端应该检测到故障，并尝试重新连接网络 3 次，每次 10 秒。

Reliability1.1.1: 重新连接后，客户端应该继续之前的工作。

Reliability1.1.2: 如果重新连接不成功，客户端应该等待 5min 后再次尝试重新连接。

Reliability1.1.2.1: 重新连接后，客户端应该继续之前的工作。

Reliability1.1.2.2: 如果重新连接依然不成功，客户端报警。

Reliability2: 在所有需要数据输入的场景进行数据有效性保护。

Reliability3: 服务器端能同时处理多个客户端用户服务请求。

3.3.5 业务规则

BR1: 时间验证规则：新的日程安排的时间必须是将来的时间

BR2: 时间验证规则：新的 deadline 最后截止的时间必须是将来的时间

BR3: Deadline 即将到来前的多个时间点分别指 deadline 前的两周、一周、三天、一天、一小时

BR4: 登陆/注册验证时间不超过 2s

3.3.6 约束

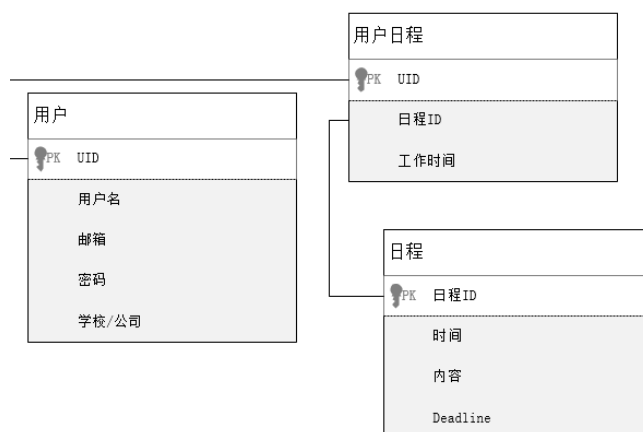
IC1: 在开发过程中缺少具体的用户日程信息，需要自己虚构一些。

IC2: 系统要在网络上分布为一个服务器和多个客户端。

3.4 数据需求

3.4.1 数据定义

DR1: 系统需要存储的数据实体及其关系：



DR2: 所有数据的有效期均为无限长，以保证历史数据的可追溯性

3.4.2 默认数据

默认格式用于以下两种情况：

- 系统中新增加数据时。
- 编辑数据时不小心将相关内容清空时。

没有默认格式的数据缺省条件下为空，需要选择或输入

3.4.3 数据格式要求

Format1: 日期的格式为 yyyy/mm/dd。

Format2: 用户名要求为 6-20 位不含特殊符号的英语、汉字、数字组合

Format3: 邮箱格式为 xxx@xxx.xxx

3.5 质量需求

Quality1: 单元测试时间少于评审时间总和的 1/2

3.6 其他需求

3.6.1 安装需求

Install1: 安装此系统的 PC、MAC 客户端需要 50M 以内的磁盘空间。

Install1: 安装此系统的移动客户端需要 10M 以内的磁盘空间。

3.6.2 性能需求

Performance1: 能够在各主流平台中低端及以上产品流畅运行。

Performance2: 各项查询、保存、反馈总耗时不超过 0.3s。

附录

1, 系统概念类图:

