1. 引言

1.1 编写目的

1.2 项目风险

1.3 文档约定

1.4 预期读者和阅读建议

1.5 产品范围

1.6 参考文献

2. 综合描述

2.1 产品的状况

2.2 产品的功能

2.3 用户类和特性

2.4 运行环境

2.5 设计和实现上的限制

2.6 假设和约束(依赖)

3. 外部接口需求

3.1 用户界面

3.2 硬件接口

3.3 软件接口

3.4 通讯接口

4. 系统功能需求

4.1 说明和优先级

4.2 激励／响应序列

4.3 输入／输出数据

5. 其它非功能需求

5.1 性能需求

5.2 安全措施需求

5.3 安全性需求

5.4 软件质量属性

5.5 业务规则

5.6 用户文档

6. 词汇表

7. 数据定义

8. 分析模型

9. 待定问题列表

1. 引言

引言是对这份软件产品需求分析报告的概览，是为了帮助阅读者了解这份文档是如何编写的，并且应该如何阅读、理解和解释这份文档。

1.1 编写目的

说明这份软件产品需求分析报告是为哪个软件产品编写的，开发这个软件产品意义、作用、以及最终要达到的意图。通过这份软件产品需求分析报告详尽说明了该软件产品的需求规格，包括修正和(或)发行版本号，从而对该软件产品进行准确的定义。

如果这份软件产品需求分析报告只与整个系统的某一部分有关系，那么只定义软件产品需求分析报告中说明的那个部分或子系统。

1.2 项目风险

具体说明本软件开发项目的全部风险承担者，以及各自在本阶段所需要承担的主要风险，首要风险承担者包括：

●  任务提出者；

●  软件开发者；

●  产品使用者。

1.3 文档约定

描述编写文档时所采用的标准(如果有标准的话)，或者各种排版约定。排版约定应该包括：

●  正文风格；

●  提示方式；

●  重要符号；

也应该说明高层次需求是否可以被其所有细化的需求所继承，或者每个需求陈述是否都有其自己的优先级。

1.4 预期读者和阅读建议

列举本软件产品需求分析报告所针对的各种不同的预期读者，例如，可能包括：

●  用户；

●  开发人员；

●  项目经理；

●  营销人员；

●  测试人员；

●  文档编写入员。

并且描述了文档中，其余部分的内容及其组织结构，并且针对每一类读者提出最适合的文档阅读建议。

1.5 产品范围

说明该软件产品及其开发目的的简短描述，包括利益和目标。把软件产品开发与企业目标，或者业务策略相联系。

描述产品范围时需注意，可以参考项目视图和范围文档，但是不能将其内容复制到这里。

1.6 参考文献

列举编写软件产品需求分析报告时所用到的参考文献及资料，可能包括：

●  本项目的合同书；

●  上级机关有关本项目的批文；

●  本项目已经批准的计划任务书；

●  用户界面风格指导；

●  开发本项目时所要用到的标淮；

●  系统规格需求说明；

●  使用实例文档；

●  属于本项目的其它己发表文件；

●  本软件产品需求分析报告中所引用的文件、资料；

●  相关软件产品需求分析报告；

为了方便读者查阅，所有参考资料应该按一定顺序排列。如果可能，每份资料都应该给出：

●  标题名称；

●  作者或者合同签约者；

●  文件编号或者版本号；

●  发表日期或者签约日期；

●  出版单位或者资料来源。

2. 综合描述

这一部分概述了正在定义的软件产品的作用范围以及该软件产品所运行的环境、使用该软件产品的用户、对该软件产品己知的限制、有关该软件产品的假设和依赖。

2.1 产品的状况

描述了在软件产品需求分析报告中所定义的软件产品的背景和起源。说明了该软件产品是否属于下列情况：

●  是否是产品系列中的下一成员；

●  是否是成熟产品所改进的下一代产品；

●  是否是现有应用软件的替代品(升级产品)；

●  是否是一个新型的、自主型的产品。

如果该软件产品需求分析报告定义的软件系统是：

●  大系统的一个组成部分；

●  与其它系统和其它机构之间存在基本的相互关系。

那么必须说明软件产品需求分析报告定义的这部分软件是怎样与整个大系统相关联的，或者（同时）说明相互关系的存在形式，并且要定义出两者之间的全部接口。

2.2 产品的功能

因为将在需求分析报告的第4部分中详细描述软件产品的功能，所以在此只需要概略地总结。仅从业务层面陈述本软件产品所应具有的主要功能，在描述功能时应该针对每一项需求准确地描述其各项规格说明。如果存在引起误解的可能，在陈述本软件产品主要功能的作用领域时，也需要对应陈述本软件产品的非作用领域，以利读者理解本软件产品。

为了很好地组织产品功能，使每个读者都容易理解，可以采用列表的方法给出。也可以采用图形方式，将主要的需求分组以及它们之间的联系使用数据流程图的顶层图或类图进行表示，这种表示方法是很有用的。

参考用户当前管理组织构架，了解各个机构的主要职能，将有助于陈述软件产品的主要功能。

2.3 用户类和特性

确定有可能使用该软件产品的不同用户类，并且描述它们相关的特征。往往有一些软件需求，只与特定的用户类有关。描述时，应该将该软件产品的重要用户类与非重要用户类区分开。

用户不一定是软件产品的直接使用者，通过报表、应用程序接口、系统硬件接口得到软件产品的数据和服务的人、或者机构也有他们的需求。所以，应该将这些外部需求视为通过报表、应用程序接口、系统硬件接口附加给软件产品的附加用户类。

2.4 运行环境

描述了本软件的运行环境，一般包括：

●  硬件平台；

●  操作系统和版本；

●  支撑环境(例如：数据库等)和版本；

●  其它与该软件有关的软件组件；

●  与该软件共存的应用程序。

2.5 设计和实现上的限制

确定影响开发人员自由选择的问题，并且说明这些问题为什么成为一种限制。可能的限制包括下列内容：

●  必须使用的特定技术、工具、编程语言和数据库；

●  避免使用的特定技术、工具、编程语言和数据库；

●  要求遵循的开发规范和标准

例如，如果由客户的公司或者第三方公司负责软件维护，就必须定义转包者所使用的设计符号表示和编码标准；

●  企业策略的限制；

●  政府法规的限制；

●  工业标准的限制；

●  硬件的限制

例如，定时需求或存储器限制；

●  数据转换格式标淮的限制。

2.6 假设和约束(依赖)

列举出对软件产品需求分析报告中，影响需求陈述的假设因素(与己知因素相对立)。如果这些假设因素不正确、不一致或者被修改，就会使软件产品开发项目受到影响。这些假设的因素可能包括：

●  计划使用的商业组件，或者其它软件中的某个部件；

●  假定产品中某个用户界面将符合一个特殊的设计约定；

●  有关本软件用户的若干假定(例如：假定用户会熟练使用SQL语言。)；

●  有关本软件开发工作的若干假定(例如：用户承诺的优惠、方便、上级部门给予的特殊政策和支持等。)；

●  有关本软件运行环境的一些问题；

此外，确定本软件开发项目对外部约束因素所存在的依赖。有关的约束可能包括：

●  工期约束；

●  经费约束；

●  人员约束；

●  设备约束；

●  地理位置约束；

●  其它有关项目约束；

3. 外部接口需求

通过本节描述可以确定，保证软件产品能和外部组件正确连接的需求。关联图仅能表示高层抽象的外部接口，必须对接口数据和外部组件进行详细描述，并且写入数据定义中。如果产品的不同部分有不同的外部接口，那么应该把这些外部接口的全部详细需求并入到这一部分实例中。

注意：必须将附加用户类的特征与外部接口需求加以区分，附加用户类的特征描述的是通过接口取得软件产品的数据和服务的人的需求；而外部接口需求描述的是接口本身的需求。

3.1 用户界面

陈述需要使用在用户界面上的软件组件，描述每一个用户界面的逻辑特征。必须注意，这里需要描述的是用户界面的逻辑特征，而不是用户界面。以下是可能包括的一些特征：

●  将要采用的图形用户界面(GUl)标准或者产品系列的风格；

●  有关屏幕布局或者解决方案的限制；

●  将要使用在每一个屏幕(图形用户界面)上的软件组件，可能包括：

n  选单；

n  标准按钮；

n  导航链接；

n  各种功能组件；

n  消息栏；

●  快捷键；

●  各种显示格式的规定，可能包括：

n  不同情况下文字的对齐方式；

n  不同情况下数字的表现格式与对齐方式

n  日期的表现方法与格式；

n  计时方法与时间格式；

n  等等。

●  错误信息显示标准；

对于用户界面的细节，例如：一个特定对话框的布局，应该写入具体的用户界面设计说明中，而不能写入软件需求规格说明中。

如果采用现成的、合适的用户界面设计规范(标准)，或者另文描述，可以在这里直接说明，并且将其加入参考文献。

3.2 硬件接口

描述待开发的软件产品与系统硬件接口的特征，若有多个硬件接口，则必须全都描述。接口特征的描述内容可能包括：

●  支持的硬件类型；

●  软、硬件之间交流的数据；

●  控制信息的性质；

●  使用的通讯协议；

3.3 软件接口

描述该软件产品与其它外部组件的连接，这些外部组件必须明确它们的名称和版本号以资识别，可能的外部组件包括：

●  操作系统；

●  数据库；

●  工具；

●  函数库；

●  集成的商业组件

说明：这里所说的“集成的商业组件”，是指与系统集成的商业组件，而不是与软件产品集成的商业组件。例如：中间件、消息服务，等等。

描述并且明确软件产品与软件组件之间交换数据或者消息的目的。描述所需要的服务，以及与内部组件通讯的性质。确定软件产品将与组件之间共享的数据。如果必须使用一种特殊的方法来实现数据共享机制，例如：在多用户系统中的一个全局数据区，那么就必须把它定义为一种实现上的限制。

3.4 通讯接口

描述与软件产品所使用的通讯功能相关的需求，包括：

●  电子邮件；

●  WEB浏览器；

●  网络通讯标准或者协议；

●  数据交互用电子表格；

必须定义相关的：

●  消息格式；

●  通讯安全或加密问题；

●  数据传输速率；

●  同步和异步通讯机制；

4. 系统功能需求

需要进行详细的需求记录，详细列出与该系统功能相关的详细功能需求，并且，唯一地标识每一项需求。这是必须提交给用户的软件功能，使得用户可以使用所提供的功能执行服务或者使用所指定的使用实例执行任务。描述软件产品如何响应己知的出错条件、非法输入、非法动作。

如果每一项功能需求都能用一项，也只需要用一项测试用例就能进行验证，那么就可以认为功能需求已经适当地进行描述了。如果某项功能需求找不到合适的测试用例，或者必须使用多项测试用例才能验证，那么该项功能需求的描述必然存在某些问题。

功能需求是根据系统功能，即软件产品所提供的主要服务来组织的。可以通过使用实例、运行模式、用户类、对象类或者功能等级来组织这部分内容，也可以便用这些元素的组合。总而言之，必须选择一种是读者容易理解预期产品的组织方案。

用简短的语句说明功能的名称，例如：“4.1系统参数管理”。按照服务组织的顺序，逐条阐述系统功能。无论说明的是何种功能，都应该针对该系统功能重复叙述4.1~ 4.3这三个部分。

可以通过各种方式来组织这一部分内容，例如采用：使用实例、运行模式、用户类、对象类、功能等级等，也可以采用它们的组合。其最终目的是，让读者容易理解即将开发的软件产品。一般来说，每个使用实例都对应一个系统功能，因而按照使用实例来组织内容比较容易让用户理解。

对应一些被共享的独立使用实例，可以定义一些公用系统功能。

必须特别注意的是，在2.2节“产品的功能”中描述的全部需求，以及它们的规格说明；必须在某个系统功能描述中有所反映，而且不应重复。

4.1 说明和优先级

对该系统功能进行简短的说明，并且指出该系统功能的优先级是：高、中、还是低。需要的话，还可以包括对特定优先级部分的评价，例如：利益、损失、费用和风险，其相对优先等级可以从1(低)到9(高)。

4.2 激励／响应序列

列出输入激励(用户动作、来自外部设备的信号或者其它触发)并且定义针对这——功能行为的系统响应序列，这些序列将与使用实例中相关的对话元素相对应。

描述激励／响应序列时，不仅需要描述基本过程，而且应该描述可选(扩充)过程，包括例外(引起任务不能顺序完成的情况称为例外)。疏忽了可选过程，有可能影响软件产品的功能；如果遗漏例外过程，则有可能会引发系统崩溃。

如果采用流程图来描述激励／响应序列，比较容易让用户理解。

4.3 输入／输出数据

列出输入数据(用户输入、来自外部接口的输入或者其它输入)并且定义针对这些输入数据的处理(计算)方法，以及相应地输出数据，描述对应区别：输入数据和输出数据。

当有大量数据需要描述时，也可以分类描述数据，并且注明各项数据的输入、输出属性。

对于每一项数据，均需要描述：

●  数据名称；

●  实际含义；

●  数据类型；

●  数据格式；

●  数据约束；

对于复杂的处理方法，仅仅给出算法原理是不够的，必须描述详细的计算过程，并且列出每一步具体使用的实际算式；如果计算过程中涉及查表、判断、迭代等处理方法，应该给出处理依据和相关数据。如果计算方法很简单，也可以将其从略，不加描述。

5. 其它非功能需求

在这里列举出所有非功能需求，主要包括可靠性、安全性、可维护性、可扩展性、可测试性等。

5.1 性能需求

阐述不同应用领域对软件产品性能的需求，并且说明提出需求的原理或者依据，以帮助开发人员做出合理的设计选择。尽可能详细地描述性能需求，如果需要，可以针对每个功能需求或者特征分别陈述其性能需求。在这里确定：

●  相互合作的用户数量；

●  系统支持的并发操作数量；

●  响应时间；

●  与实时系统的时间关系：

●  容量需求

n  存储器；

n  磁盘空间；

n  数据库中表的最大行数。

5.2 安全措施需求

详尽陈述与软件产品使用过程中可能发生的损失、破坏、危害相关的需求。定义必须采取的安全保护或动作，以及必须预防的潜在危险动作。明确软件产品必须遵从的安全标准、策略、或规则。

5.3 安全性需求

详尽陈述与系统安全性、完整性问题相关的需求，或者与个人隐私问题相关的需求。这些问题将会影响到软件产品的使用，和软件产品所创建或者使用的数据的保护。定义用户身份认证，或备授权需求。明确软件产品必须满足的安全性或者保密性策略。也可以通过称为完整性的质量属性来阐述这些需求。一个典型的软件系统安全需求范例如下：“每个用户在第一次登录后，必须更改他的系统预置登录密码，系统预置的登录密码不能重用。”

5.4 软件质量属性

详尽陈述对客户和开发人员至关重要的在软件产品其它方面表现出来的质量功能。这些功能必须是确定的、定量的、在需要时是可以验证的。至少也应该指明不同属性的相对侧重点，例如：易用性优于易学性，或者可移植性优于有效性。

5.5 业务规则

列举出有关软件产品的所有操作规则，例如：那些人在特定环境下可以进行何种操作。这些本身不是功能需求，但是他们可以暗示某些功能需求执行这些规则。一个业务规则的范例如下：“进行达到或者超过10，000，00元人民币的储蓄业务时，必须通过附加的管理员认证。”

列举业务规则时，可以根据规则的数量，选取合适的编目方式。

5.6 用户文档

列举出将与软件产品一同交付的用户文档，并且明确所有己知用户文档的交付格式或标准，例如：

●  安装指南

纸质文档，16开本；

●  用户手册

纸质文档，16开本；

●  在线帮助

●  电子文档，与软件产品一同分发、配置；

●  使用教程电子文档，与软件产品一同分发、配置。

6. 词汇表

列出本文件中用到的专业术语的定义，以及有关缩写的定义(如有可能，列出相关的外文原词)。为了便于非软件专业或者非计算机专业人士阅读软件产品需求分析报告，要求使用非软件专业或者非计算机专业的术语描述软件需求。所以这里所指的专业术语，是指业务层面上的专业术语，而不是软件专业或者计算机专业的术语。但是，对于无法回避的软件专业或者计算机专业术语，也应该列入词汇表并且加以准确定义。

7. 数据定义

数据定义是一个定义了应用程序中使用的所有数据元素和结构的共享文档，其中对每个数据元素和结构都准确描述：含义、类型、数据大小、格式、计量单位、精度以及取值范围。数据定义的维护独立于软件需求规格说明，并且在软件产品开发和维护的任何阶段，均向风险承担者开放。

如果为软件开发项目创建一个独立的数据定义，而不是为每一项特性描述有关的数据项，有利于避免冗余和不一致性。但是却不利于多人协同编写需求分析报告，容易遗漏数据，也不方便阅读。因此还是建议为每个特性描述有关的数据项，汇总数据项创建数据定义，再根据数据定义复核全部数据，使得它们的名称和含义完全一致。必须注意的是，为了避免二义性，在汇总数据项时应该根据数据项所代表的实际意义汇总，而不是根据数据项的名称汇总。

在数据定义中，每个数据项除了有一个中文名称外，还应该为它取一个简短的英文名称，该英文名称应该符合命名规范，因为在软件开发时将沿用该英文名称。可以使用等号表示数据项，名称写在左边，定义写在右边。常见数据项的描述方式如下：

●  原数据元素

一个原数据元素是不可分解的，可以将一个数量值赋给它。定义原数据元素必须确定其

含义、类型、数据大小、格式、计量单位、精度以及取值范围。采用以星号为界的一行

注释文本，描述原数据元素的定义。

●  选择项

选择项是一种只可以取有限离散值的特殊原数据元素，描述时一一枚举这些值，并用方

括号括起来写在原数据元素的定义前。在两项离散值之间，使用管道符分隔。

●  组合项

组合项是一个数据结构或者记录，其中包含了多个数据项。这些数据项可以是原数据元

素，也可以是组合数据项，各数据项之间用加号连接。其中每个数据项都必须是数据定

义中定义过的，结构中也可以包括其它结构，但是绝对不允许递归。如果数据结构中有

可选项，使用圆括号把该项括起来。

●  重复项

重复项是组合项的一种特例，其中有一项将有多个实例出现在数据结构中，使用花括号

把该项括起来。如果知道该项可能允许的范围，就按“最小值：最大值”的形式写在花

括号前。

8. 分析模型

这是一个可选部分，包括或涉及到相关的分析模型，例如：

●  数据流程图；

●  类图；

●  状态转换图；

●  实体-关系图。

9. 待定问题列表

编辑一张在软件产品需求分析报告中待确定问题时的列表，把每一个表项都编上号，以便跟踪调查。