封面

目錄

目錄

[圖目錄 III](#_Toc437985495)

[表目錄 IV](#_Toc437985496)

[**需求規格書的內容** 1](#_Toc437985497)

[壹、 簡介 2](#_Toc437985498)

[一、 系統目的 2](#_Toc437985499)

[二、 系統範圍 2](#_Toc437985500)

[三、 名詞定義、縮寫 2](#_Toc437985501)

[四、 參考 2](#_Toc437985502)

[五、 系統概觀 2](#_Toc437985503)

[貳、 系統整體性描述 2](#_Toc437985504)

[一、 產品角度 2](#_Toc437985505)

[二、 產品功能 2](#_Toc437985506)

[三、 系統使用者 2](#_Toc437985507)

[四、 系統限制 2](#_Toc437985508)

[五、 系統假設 2](#_Toc437985509)

[參、 需求詳述說明 3](#_Toc437985510)

[一、 外部介面需求 3](#_Toc437985511)

[A、 使用者介面 3](#_Toc437985512)

[B、 硬體介面 3](#_Toc437985513)

[C、 軟體介面 3](#_Toc437985514)

[D、 溝通介面 3](#_Toc437985515)

[二、 功能性需求 3](#_Toc437985516)

[A、 使用者一 （如：老師） 3](#_Toc437985517)

[1、 功能性需求 R101 3](#_Toc437985518)

[2、 ... 3](#_Toc437985519)

[3、 功能性需求 R109 3](#_Toc437985520)

[B、 使用者八（如：助教） 3](#_Toc437985521)

[1、 功能性需求 R801 3](#_Toc437985522)

[2、 ... 4](#_Toc437985523)

[3、 功能性需求 R809 4](#_Toc437985524)

[三、 效能需求 4](#_Toc437985525)

[四、 設計限制 4](#_Toc437985526)

[五、 軟體系統屬性（特性） 4](#_Toc437985527)

[六、 其他需求 4](#_Toc437985528)

[肆、 附錄 4](#_Toc437985529)

# 圖目錄

asdf

# 表目錄

aasdf

**需求規格書基本概念**

軟體需求規格書: Software Requirement Specification; SRS

使用者至上 -- 所有的系統功能都是因為要滿足使用者或是開發者的需求。

使用者的需求 = 系統要提供的功能

唯有清楚的描述需求，我們才有設計的依據。

需求分為兩種，**使用者需求**與**系統需求**。

[使用者需求](http://140.134.26.7/selabwiki/index.php?title=%E4%BD%BF%E7%94%A8%E8%80%85%E9%9C%80%E6%B1%82&action=edit) 是由使用者提出來的，比較抽象;

[系統需求](http://140.134.26.7/selabwiki/index.php?title=%E7%B3%BB%E7%B5%B1%E9%9C%80%E6%B1%82&action=edit) 是與開發端討論過的，是完整的、經過可性評估的、可作為後續設計的依據、使用者確認過的、加上系統觀點的。

SRS (System Requirement Specification）的內容主要以系統需求為主。

需求的描述多半用自然語言描述，但可以輔以模組語言，例如 UML 的 use case modeling, class diagram等。

SRS 與 SDS 要分開，SRS 交代系統的 WHAT (系統提供什麼功能？)，SDS 交代系統的 HOW (我們如何提供這些功能？）。

**需求規格書的內容**

需求規格書是定義你的系統有什麼功能，也包含你的系統分析

* 系統是給誰用的，用的目的是什麼？
* 整個系統使用的情境為何？整體性的流程為何？（可用 [UML activity diagram](http://140.134.26.7/selabwiki/index.php?title=UML_activity_diagram&action=edit) 來規劃）
* 使用者需求列表與說明
* 針對每一個需求（功能）給一個明確的 ID，並用文字、圖形、模組說明此需求的含意
* 多利用 ER, DFD, Use case, 流程圖等來說明你的需求功能
* 系統需求列表與說明 ([使用者需求與系統需求的差別](http://140.134.26.7/selabwiki/index.php?title=%E4%BD%BF%E7%94%A8%E8%80%85%E9%9C%80%E6%B1%82%E8%88%87%E7%B3%BB%E7%B5%B1%E9%9C%80%E6%B1%82%E7%9A%84%E5%B7%AE%E5%88%A5&action=edit))
* [需求追溯表](http://140.134.26.7/selabwiki/index.php?title=%E9%9C%80%E6%B1%82%E8%BF%BD%E6%BA%AF%E8%A1%A8&action=edit) 描述使用者需求與系統需求之間的關係
* 系統限制（法令、規範或設計上的限制，例如一定要用Java實做)

# 簡介

## 系統目的

## 系統範圍

## 名詞定義、縮寫

## 參考

## 系統概觀

# 系統整體性描述

## 產品角度

## 產品功能

## 系統使用者

## 系統限制

## 系統假設

# 需求詳述說明

## 外部介面需求

### 使用者介面

### 硬體介面

### 軟體介面

### 溝通介面

## 功能性需求

### 使用者一 （如：老師）

#### 功能性需求 R101

#### ...

#### 功能性需求 R109

### 使用者八（如：助教）

#### 功能性需求 R801

#### ...

#### 功能性需求 R809

## 效能需求

## 設計限制

## 軟體系統屬性（特性）

## 其他需求

# 附錄

軟體系統需求規格書 目錄

1. 企業需求

有如雲等級的 UseCase (目標不應該太多)

重點應該在於要滿足企業的哪些目標.

2. 使用者需求

以使用者的需要, 想要做的事情, 希望解決的問題, 希望達到的目標來做為章節。

我認為應該有一份使用者需求 "清單"，再依照每一條需求詳加說明.

使用者需求發掘重點...

1. 使用者在執行公司政策時遇到的問題，問題的解決方案為何？

2. 使用者因應公司政策而提出新的流程或解決方案，要求系統必須配合其流程增加(修改)功能特性(Feature)。

\*) 使用者包含 CIM 本身

聚焦

在瞭解問題時請不斷的詢問自己以下的問題,

1. 使用者在現況遇到怎樣的問題 ?

2. 這些問題對使用者造成什麼影響 ?

3. 使用者認為可能的原因是什麼 ?

4. 使用者希望如何改善 ?

5. 使用者認為這樣的改善可以增進哪方面的效率 ?

6. 誰可以執行這樣的作業? 是否需要誰的允許才可執行?

『假設』是我們信以為真的事情。

除非你知道顧客的假設是什麼，否則不可能予以驗證。當你要證實顧客的需求時，你得先確定他的假設是正確的。

2.1 現狀與未來

2.2 需求清單

2.3 URxx\_<希望解決的問題>

1. Input: 找出問題

＊使用者在現況遇到怎樣的問題 ?

＊這些問題對使用者造成什麼影響 ?

＊使用者認為可能的原因是什麼 ?

2. Process:

＊分析上述問題的真正原因為何?

＊利用 5W2H 層層深入, 就像在撥洋蔥一樣.

＊What : 問題是什麼？

＊Why : 為什麼會有這個問題？ 為什麼覺得這是個問題？

＊Where : 問題會在哪邊發生？

＊Who : 誰覺得這是個問題？這個問題和誰相關？誰對問題的解決方案有影響力？

＊When：問題何時會發生？問題何時要被解決？

＊How : 問題如何發生？如何證實可以解決這個問題？

＊How much : 有多少效益？要花多少成本？完成要多久？ROI ?

3. Output: 寫出真正問題的解決方案

4. 與使用者談論如何驗證此方案能解決他們的問題

5. Validation:

＊整體思考方式需從問題導向至解決方案

＊必需要能驗證方式

＊使用者需求必須能達成解決方案

＊不可出現模糊的形容 (如: 變好，改善, 比例, 變快...)

＊規格說明文法必須正確。在一句話中至少要有主詞+ 動詞+ 受詞。

＊需求必須給予編號

2.4 URxx\_<想要做的事情>

3. 系統需求編號方式

4. 系統規格書

4.1 序言

4.1.1 目的

4.1.2 範圍

4.1.3 系統概述

4.1.3.1 子系統 A

4.1.3.2 子系統 B

4.1.4 術語詞彙

4.1.4.1 系統術語

4.1.4.2 企業術語

4.2 系統功能描述

4.2.1 概觀

4.2.1.1 功能介紹

4.2.1.2 環境圖

4.2.1.3 使用者特徵

4.2.1.4 設計與實作限制

4.2.1.5 假設與相依

4.2.1.6 操作概念與腳本

4.2.1.7 介面需求

4.2.1.7.1 系統界面

4.2.1.7.2 用戶界面

4.2.1.7.1.1 UI

4.2.1.7.3 硬體界面

4.2.1.7.4 軟體界面

4.2.1.7.4.1 API

4.2.1.7.4.2 與外部系統之間

4.2.1.7.5 通信界面

4.2.1.7.6 記憶體限制

4.2.1.7.7 場所適應需求

4.2.2 業務/系統實體

4.2.3 資料模型

4.2.3.1 概念資料模型

4.2.4 功能或作業需求

4.2.4.1 <模組>

4.2.4.1.1 功能描述

4.1.4.1.2 功能需求清單

4.1.4.1.3 使用案例圖

4.1.4.1.4 操作概念與腳本

4.1.4.1.5 使用案例組

4.1.4.1.5.1 SUCxxx: XXXXX

4.1.4.1.5.2 UCxxx: XXXXX

4.2.5 非功能性需求

4.2.6 設計與實做限制

4.3 <子系統功能描述>

4.4 需求分配

4.5 未解決的議題