

# MovieReview

## 软件需求规格说明

南京大学软件学院

infinity 团队

2017.03.11

0. 变更记录	4
1. 引言	5
1.1. 目的	5
1.2. 词汇表	5
1.3. 参考资料	5
2. 总体描述	5
2.1. 商品前景	5
2.1.1 背景和机遇	5
2.1.2 系统的边界和范围（系统用例图）	6
2.2 商品功能	6
2.3 用户特征	6
2.4 约束	7
2.5 假设和依赖	7
3. 详细需求描述	7
3.1 对外接口需求	7
3.1.1 用户界面	7
3.1.2 硬件接口	8
3.1.3 软件接口	8
3.1.4 通信接口	8
3.2 功能需求	8
3.2.1 查看电影评论信息	8
3.2.2 查看用户评论信息	8
3.3 非功能需求	9
3.3.1 安全性	9
3.3.2 可维护性	9
3.3.3 易用性	9
3.3.4 可靠性	9
3.3.5 业务规则	9

3.3.6 约束	9
3.4 数据需求	9
3.4.1 数据定义	9
3.4.2 默认数据	10
3.4.3 数据格式要求	10
3.5 其他需求	10

# 0. 变更记录

修改人员	日期	变更原因	版本号
宋奎熹	2017.03.11	最初草稿	v1.0

# 1. 引言

## 1.1. 目的

本文档描述了 MovieReivew (MR) 系统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将以此文档为依据。

除特殊说明外，本文档所包含的需求有高优先级需求需求。

本说明书的内容可能在项目实施过程中发生变更，但是必须由项目小组成员发出变更请求，小组讨论后最终决定，建立持续有效的版本控制。

## 1.2. 词汇表

词汇名称	词汇含义
MR	MovieReview 系统
_ui	表示某展示层
_bl	表示某逻辑层
_data	表示某数据层

## 1.3. 参考资料

1. IEEE std 1471-2000
2. 丁二玉,刘钦.计算与软件工程(卷二)[M]机械工业出版社,2012:134— 182
3. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Micheael Stal.Pattern-Oriented Software Architecture Volume 1: A system of Patterns [M]机械工业出版社,2003
4. Christopher Fox.软件工程设计导论：过程、原理与模式（UML2.0版）[M]清华大学出版社，2007

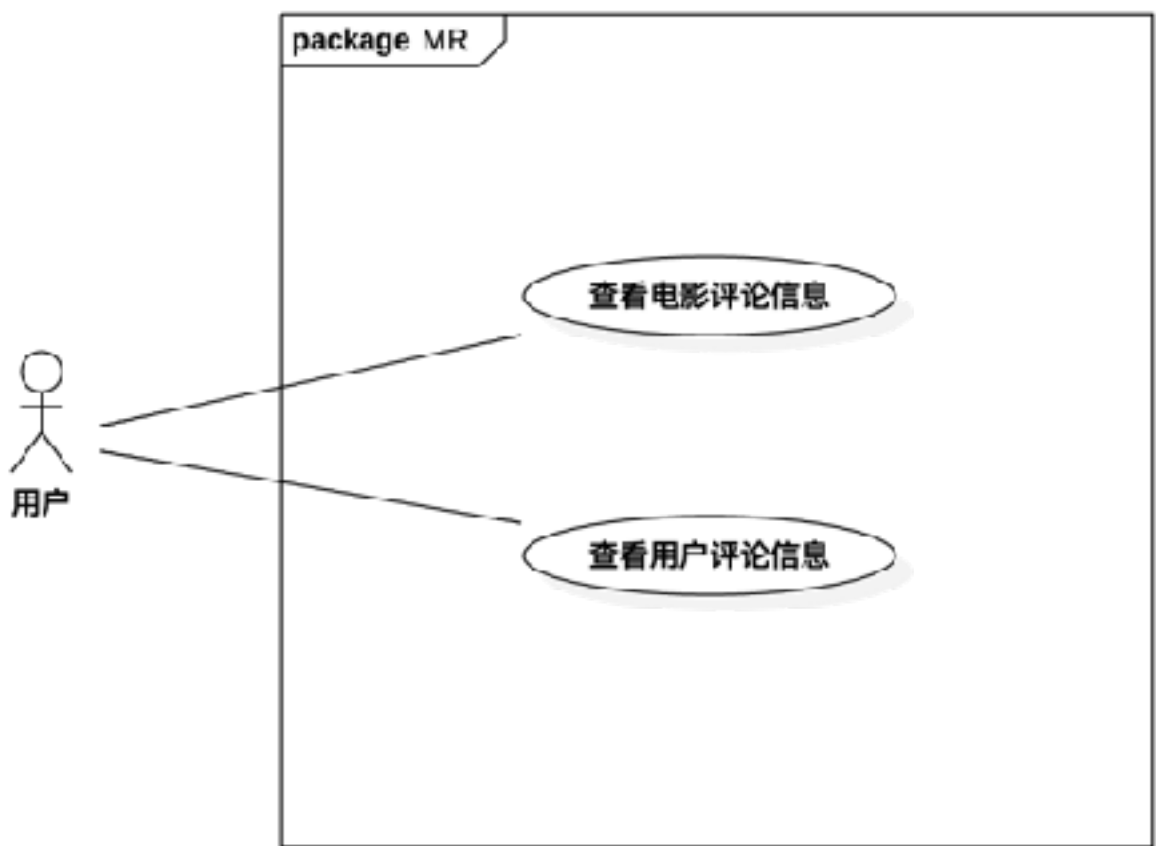
# 2. 总体描述

## 2.1. 商品前景

### 2.1.1 背景和机遇

在当今电影工业的蓬勃发展下，国内外每年都有大量电影产出。而这就导致一个问题：消费者和影迷们难以从浩如烟海的电影中挑选出精华所在，挑选出适合自己、符合自己口味的电影。于是 MR 系统应运而生，它在分析了 [amazon.com](https://www.amazon.com) 数百万条用户的电影评价之后，形成了一个庞大的数据库，从而帮助影迷迅速找到自己喜爱的、符合自己口味的电影，并排除掉那些口碑不良的作品。

2.1.2 系统的边界和范围（系统用例图）



2.2 商品功能

- SF1: 为用户提供一部电影的平均评分，以供平时观影时参考。
- SF2: 展示一部电影的所有用户评论，以供浏览和平时观影时参考。
- SF3: 对一个用户发出的评论进行展示，以便浏览。

2.3 用户特征

用户	特征
----	----

用户	<p>根据现实背景考虑，本系统的客户数量应该在10万人以上。他们可能是一般的电影消费者，也可能是资深影迷。他们希望能够在短时间内快速、精准地找到符合自己的口味和要求的电影。</p> <p>客户中可能有一部分人群接触计算机软件较少，他们往往因为系统过于复杂就不愿意尝试，希望新的系统可以简单上手，实现高效寻找。</p>
网站管理人员	<p>酒店管理系统有1个网站管理人员。主要负责对用户账户的管理，以及电影数据的增加删除。网站管理人员的计算机操作技能很好。</p>

## 2.4 约束

CON1：采用 Java 语言开发，后期可能采用 HTML/CSS/JS/Objective-C 等语言开发。

CON2：系统使用 PC 端图形界面，后期可能采用 Web 端/移动端界面。

CON3：项目要使用持续集成方法进行开发。

CON4：项目采用分层模式开发。

CON5：项目后期可能使用数据库系统。

CON6：项目需要为自动化测试预留接口。

CON7：用户远程使用系统。

## 2.5 假设和依赖

AE1：用户的网络状态良好，不会出现中断。

AE2：数据库所在硬盘不会崩溃。

# 3. 详细需求描述

## 3.1 对外接口需求

### 3.1.1 用户界面

界面风格：本系统采取图形化用户界面，界面风格简洁，操作方便，支持用户用鼠标和键盘进行操作。

界面布局：界面布局整齐合理，不会过于复杂。

### 3.1.2 硬件接口

无

### 3.1.3 软件接口

系统可以运行在搭载了 Windows7/8/10 或 macOS 系统，且可以与互联网连接的电脑上。

### 3.1.4 通信接口

无

## 3.2 功能需求

### 3.2.1 查看电影评论信息

#### 3.2.1.1 特性描述

按照电影 ID，搜索并查看 [amazon.com](https://amazon.com) 上对该电影的所有用户评论。

#### 3.2.1.2 刺激/响应序列

刺激：用户输入电影 ID

响应：系统展示与该电影 ID 对应的电影信息，和所有用户评论。

#### 3.2.1.3 相关功能需求

查看电影评论信息	
编号	需求描述
MovieReviews.show	系统展示所有与该电影 ID 对应的用户评论

### 3.2.2 查看用户评论信息

#### 3.2.2.1 特性描述

按照用户 ID，搜索并查看 [amazon.com](https://amazon.com) 上该用户所发出的所有评论。

#### 3.2.2.2 刺激/响应序列

刺激：用户输入用户 ID

响应：系统展示与该用户 ID 对应的用户信息，和所有发出的评论。

#### 3.2.2.3 相关功能需求



查看用户评论信息	
编号	需求描述
UserReviews.show	系统展示所有与该用户 ID 对应的评论

### 3.3 非功能需求

#### 3.3.1 安全性

Safety1: 数据从 API 获得后立即存在本地, 不进行外界通信, 保证数据安全。

#### 3.3.2 可维护性

Modifiability1: 如果要增加对新的电影评论网站的数据的接入, 要能够在 1 个人月内完成。

Modifiability2: 若要改用数据库, 可以在 1 个人月时间内完成。

#### 3.3.3 易用性

Usability1: 不需要用户使用手册或进行系统使用培训, 用户也能够使用本系统所有功能。

#### 3.3.4 可靠性

Reliability1: 当 10000 以内的人同时在线时, 系统不会崩溃。

Reliability2: 系统崩溃后, 已更新保存的数据不会丢失。

Reliability3: 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

Reliability3.1: 客户端应该检测到故障, 并尝试重新连接服务器 5 次, 每次 10 秒。

Reliability3.1.1: 如果重新连接不成功, 客户端报警。

Reliability4: 采用分层模式, 接口稳定。

#### 3.3.5 业务规则

BR1: 电影的评分为1~5的一个整数, 在展示时采取四舍五入制。

#### 3.3.6 约束

IC1: 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: 数据要每隔 1 周备份一次。

### 3.4.2 默认数据

无

### 3.4.3 数据格式要求

Format1: 电影 ID 为 10 位数字+英文字符编码。

Format2: 用户 ID 为 14 位数字+英文字符编码。

Format3: 评分采用小数点后 1 位小数。

Format4: 评论时间采用 Unix 时间戳。

## 3.5 其他需求

该项目必须在两周内完成设计、编码、测试、部署。考虑到服务器带宽有限，系统显示、查询、排序须在 5s 内响应。