

毕业设计（论文）开题报告

（含文献综述、外文翻译）



题 目 基于AngularJS的宿舍管理系统设计与开发

姓 名 周皙昊

学 号 31501105

专业班级 计算1501

所在学院 计算机与计算科学学院

指导教师（职称） 罗荣良（副教授）

二○一六 年 十二 月 三十一 日

毕业设计（论文）

开 题 报 告

（包括选题的意义、可行性分析、研究的内容、研究方法、

拟解决的关键问题、预期结果、研究进度计划等）

## 1. 选题的背景和意义

随着社会经济的整体发展，目前国民百姓的文化水平也在不断地提升，大学生的数量也是在逐年上升，而宿舍作为学生在学校生活的第二个家，良好的宿舍环境能更好地帮助学生们的日常生活。目前大部分的宿舍仍然是以人工管理的形式进行管理，一些琐碎的事务会有不便，而人们对于生产力的要求变得越来越高，需要更加高效便捷的方式来应对生活中所要面对的方方面面的问题。一个合理的宿舍管理可以有效帮助宿舍管理人员进行对学生的管理，也有助于学生们在宿舍中的事务处理，不会因消息的延迟传达造成各种不便。

### 1.1 选题的背景

在以往之中，很多宿舍的后勤部门仍然采用原始的宿舍管理方式，学生在宿舍的信息往往得不到有效及时的更新，而登记学生入住、更换宿舍、离校等等信息也都是以人工记录的方式进行统计，人工统计的方式麻烦费时并且效率低下，不足以满足实际的需求。学生在于宿舍的事务并不仅仅是居住，管理人员除了对人员进行管理之外，还有收取水电费、日常通知、卫生检查等等，学生需要与管理人员进行直接交涉处理任务，并不方便。

### 1.2 国内外研究现状

学生宿舍管理是大学管理之中很重要的一环，在某种程度对于学校的发展起着至关重要的作用,而随着计算机技术和互联网技术在各个领域的日益渗透，许多国内外高校都提出了实现数字化校园建设的发展规划。校园信息化的技术概念首次是有一位美国教授 Kenneth C.Green 提出的，并且申请了一份高校信息化管理的研究项目。之后的几年项目的实行得到了大多高校的认可，遍及高校众多领域包括科研教学，课题研究以及信息管理等。美国作为一个具有强大经济实力的国家，是世界上最早开展教育信息化，并且就目前来看也是教育信息化发展水平最高的国家，信息技术的发达带动了企业信息化建设的发展，也间接影响了美国高校教育信息化的进步，不仅有力推动了美国国内教育改革，也为其他国家和地区教育信息化的发展提供方向上的指引和经验上的借鉴。

相比较国内，CERNET 项目正式启动之后，不少高校深受影响开始进行数字化发展教育，建立网上信息教学网，部分大型高校逐渐开始构建各领域的信息化系统包括教学教务系统，办公自动化系统以及学生信息登陆系统等，信息化建设不断扩大，许多学校在科技研究，教育教学以及人事管理方面都实现了数字信息化，建立了各自的网络应用app系统。而且一些高校从国外引进信息技术在校园局域网的基础上开设了属于该校的宿舍管理系统，通过大型的网站平台为进行信息的共享以及更近，人性化的设计提高了学生对其不同标准的接纳程度。但是仍有很大一部分高校的宿舍管理由于资金的缺乏还采用传统的方式进行，特别是在一些水电费支付上仍使用现金支付，排队支付，这样的人工交付的形式在地方受限的情况下是一种人为地资源浪费，排队时间长，发生错误率高。

### 1.3 发展趋势

而在当今的信息社会之中，计算机在各行各业中的运用已经得到普及，自动化、信息化的管理越来越广泛应用于各个领域，更加系统的管理也会逐渐替代人力管理，合理的管理系统能在很大程度上帮助管理人员与学生。

如今不少国际高校的宿舍改革规模较大，系统也越来越成熟。其中北美欧洲等发达地区的高校信息化建设开设历史久远，和其他后发展的高校宿舍管理系统更进步的是通过紫外线感应人脸识别系统，通过头像扫描记录个人信息并且存入系统，这一个方便快捷的优势可以长期存在，但美国的部分地区的高校已经开始进行进一步的改革措施，最新采用技术十分精准的指纹识别进行学生信息的认证，这是一种生物特征识别方法，可以有效的避免学生身份信息的作假和系统的误区。学生寝室的钥匙必须放置于规定的储物柜内，学生需要进行两秒的指纹认证方能取出钥匙，并且宿舍的周围都装有多功能的监控设备实时掌控宿舍周围的环境。为了保障更安全的宿舍环境以及财产保护制度监控中心会根据实时情况进行分析和处理，一旦有异常现象出现，宿舍的严惩制度十分严格。因为可以随时提供自动洗衣房以及娱乐场所，美国的高校宿舍服务深受同学支持。

## 2．研究的基本内容

本系统开发的最终用户为宿舍管理人员与在宿舍的学生，研究的内容如何以网页端的形式来解决管理人员与学生在宿舍的日常事务处理，功能包括宿舍管理、人员管理、卫生管理、出入人员登记、通知等，尽可能的能在通过网页上处理学生在宿舍的所有事务。

### 2.1 基本框架

本系统是部署在windows上的java应用，将采用springboot + mysql + angularJs + jpa，以maven框架为库管理框架，在前端页面的框架上，结合bootstrap来实现页面的美化，并实现响应式网页的设计。

### 2.2 研究的重点和难点

（1）需求问题：因为是为更好地解决管理问题与提高学生们在宿舍中的体验设计的系统，充分了解两类用户的需求是十分重要的，所设计的功能需要符合用户的心理，否则便不利于系统部署后的使用。同时如何获取用户需求也是一个问题，比较传统的方式是问卷调查，将拟定的功能发布在问卷上获取调查对象的意见，缺点是暂无广泛传播问卷的方式，最终获取的样本容量可能并不足以代表大部分最终用户的意见。

（2）安全性问题：安全问题是网页端系统所必须要考虑的，本系统的用户是以账号区分的，所以需要考虑认证和授权问题，其次也要考虑session 与 cookie，两者可以有效提高用户体验，但一定程度上也降低了安全系数，需要谨慎使用，以避免不必要的麻烦

（3）页面设计：网页设计在网站建设中占据了额很重要的地位，良好的页面设计能够有效地提高用户的使用体验，同时还能够提高用户粘度。无论哪一个网站在建站的时候都会特别注意网页设计，再加上现在网页技术越来越先进，出现了越来越多精美的网页，丑陋的页面在其中无疑是“鹤立鸡群”。本系统是以可视化操作页面呈现在用户的面前，能够实现相应功能的相应前端接口必须符合人机关系，方便用户操作使用。

### 2.3 拟解决的关键问题

（1）在宿舍生活的学生在处理日常事务的时候需要与管理人员进行直接交涉，往往会因为个人的因素导致两者之间得不到及时的交流，不仅会给学生，也会给管理人员带来不便。学生处理部分事务的手续显得繁琐，比如一系列申请（在外住宿申请等），往往一次并不能够解决问题，需要和管理人员进行多次交流，而这过程就显得有些冗长。包括一些琐碎事务，如订水、缴纳水电费、申请维修等，如果能通过网上系统便能减少手续。

（2）传统宿舍管理人员的人工管理并不方便，因受时间、地点及人为等因素制约，流程难以高效流转，对一些陈旧数据的统计困难程度会随时间而逐渐上升。并且管理是需要有制度流程的，当管理人员进行更替时需要对制度流程重新学习，并且大部分宿舍的管理流程是通过经验以及口传而熟悉的，管理实际的运作流程并不如预料中流畅。

## 3．研究的方法及措施

（1）需求分析

需求分析是本系统设计与开发的一大关键点，因此也需要严格进行。初步打算用问卷法探求大学宿舍中学生们的主要需求，设计相关问卷以针对需求开发相应的系统功能。而因为管理人员相对于学生数量来说数量较少，因此采用访谈法，对宿舍的管理人员进行访谈，力求解决在实际管理过程中所遇到的种种不便。

（2）系统设计与开发

针对在需求分析阶段获取的用户需求设计相应的功能，模拟实际情况下宿舍各事务的运作来设计数据库的结构以及设计模块的控制流程，为了后期对系统的优化与修改，需要将系统分解为若干子系统，尽量减少各模块功能之间的耦合程度。在完成设计之后便是对系统的coding过程。

（3）软件测试

在完成系统之后需要进行对系统的测试，最主要的测试过程是功能测试与需求的回归测试。在测试过程中需要对学生进行抽样测试使用本系统，一是检测系统是否达到了预期的功能，二也是得知从用户角度看来系统是否还有不足与改进以及有没有进一步的需求。之后根据反馈的结果对系统进行改进。

## 4．预期研究成果

实现管理人员对于宿舍内的人员、资源管理，可以对学生们提交的申请进行审批。学生们可以在系统上查看个人的信息，提交各项申请，拥有和管理人员直接交涉的通道。除了以上基本功能之外，还预期实现以下几种功能：

1. 数据的录入与生成

系统初建成时的导入数据过程如果采用一条一条手工录入的方法的话，效率与工作量必然是非常地低下和庞大的，因此需要拥有从已有数据如Excel表中自动导入数据。同理为了方便数据的迁移，生成文档的功能也是需要有的。

1. 定时任务的实现

一些宿舍中的事务是有时限性的，例如报名等，管理人员应当拥有定时开启和关闭功能的相应接口。

## 5．研究工作进度计划

1、第七学期 1 - 8 周 文献的收集与整理

2、第七学期 9 - 12 周 文献综述和外文翻译

3、第七学期 12 - 15 周 开题报告

4、第七学期 16、17 周 和第八学期 1 - 9周 设计、开发

5、第八学期 10 - 12 周 撰写毕业设计论文

6、第八学期 13 -14 周 论文的答辩与修改

毕业设计（论文）

文 献 综 述

（包括国内外现状、研究方向、进展情况、存在问题、参考依据等）

基于AngularJS的宿舍管理系统设计与开发

## 1. 国内外研究现状

学生宿舍管理是大学管理之中很重要的一环，在某种程度对于学校的发展起着至关重要的作用。而随着计算机技术和互联网技术在各个领域的日益渗透，许多国内外高校都提出了实现数字化校园建设的发展规划[1]。校园信息化的技术概念首次是有一位美国教授 Kenneth C.Green 提出的，并且申请了一份高校信息化管理的研究项目。之后的几年项目的实行得到了大多高校的认可，遍及高校众多领域包括科研教学，课题研究以及信息管理等[2]。美国作为一个具有强大经济实力的国家，是世界上最早开展教育信息化，并且就目前来看也是教育信息化发展水平最高的国家，信息技术的发达带动了企业信息化建设的发展，也间接影响了美国高校教育信息化的进步，不仅有力推动了美国国内教育改革，也为其他国家和地区教育信息化的发展提供方向上的指引和经验上的借鉴[3]。

相比较国内，CERNET 项目正式启动之后，不少高校深受影响开始进行数字化发展教育，建立网上信息教学网，部分大型高校逐渐开始构建各领域的信息化系统包括教学教务系统，办公自动化系统以及学生信息登陆系统等，信息化建设不断扩大，许多学校在科技研究，教育教学以及人事管理方面都实现了数字信息化，建立了各自的网络应用app系统。而且一些高校从国外引进信息技术在校园局域网的基础上开设了属于该校的宿舍管理系统，通过大型的网站平台为进行信息的共享以及更近，人性化的设计提高了学生对其不同标准的接纳程度。但是仍有很大一部分高校的宿舍管理由于资金的缺乏还采用传统的方式进行，特别是在一些水电费支付上仍使用现金支付，排队支付，这样的人工交付的形式在地方受限的情况下是一种人为地资源浪费，排队时间长，发生错误率高。最后一方面，是系统的多样性较难控制[4]。由于学校的校风不同，信息管理制度下达的规定标准不同，因此开设的宿舍管理系统各有千秋，产生了不同版本的单机版系统引用。 为了方便学生进出寝室时接受系统的分配安排，寝室资源的合理安排芬芳，各大高校的寝室管理部门应理性地，规范地开发各自的高校寝室管理系统[5]。

单从举例以宿舍分配这一块功能来说，传统手动分配宿舍的方法需要浪费大量的人力、物力对学生和宿舍之间的信息进行分类、汇总、安排，不仅效率低、易出错，也不能实时了解宿舍的入住情况，难以满足目前高校宿舍分配和管理的要求。因此在高校校园信息化建设的背景下，在高校新生的入住系统中增加宿舍安排模块，不仅可以有效提高入住的效率与宿舍分配的质量还可以对高校校园信息化建设起到推动作用[6]。

## 2．研究方向

J2EE 技术和平台出现以后，因为其安全性、稳定性及可移植性等优点被广泛应用于诸多的大型商务系统 [7] ，许多的管理系统也是基于这个平台。Spring boot 是由Pivotal团队提供的全新框架，在 Spring Boot 中集成的基础框架，是在开发中经常需要使用的框架，如内嵌容器（Tomcat、Jettty、Undertow）、日志框架、JMS框架、持久化框架、流行的NOSQL数据库（Cassandra、MongoDB）、缓存框架等。在传统的Spring 框架中开发，用户需要自行编写XML文件，并在 Maven、Gradle 中加入相应的依赖包，在响应的代码中添加功能代码，才能使用。同时对于引入多个依赖包时，包之间的版本调整也是个困难的问题，经常需要对依赖包的版本冲突进行处理。调整依赖包的版本以及冲突问题是一个费时费力、重复、繁琐的工作。如果使用 Spring Boot 开发，仅需在 Maven、Gradle 配置文件中加上少量的配置，即可在代码中使用所需的框架，让原本的配置简化到几乎是零代码、零 XML 配置，同时依赖包的版本问题也由 Spring Boot 轻松解决[8]。

Spring Boot 对目前的的主流构建工具 Maven、Gradle 都提供了良好的支持，对其他构建工具也提供了支持，如Ant，但是相对 Maven、Gradle 来说，支持的力度相对要少些，所以需要尽量使用 Maven、Gradle 来构建 Spring Boot 程序。

为了能够花费更多的时间在功能开发，而花费更少的时间来与底层技术（如HTML、CSS、JavaScript）搏斗[9]，我们需要一门技术来处理“设计”和“实现”这两者之间的平衡。AngularJs是一个开发动态Web应用的框架。它让我们可以使用HTML作为模板语言并且可以通过扩展的HTML语法来使应用组件更加清晰和简洁。它的创新之处在于，通过数据绑定和依赖注入减少了大量代码，而这些都在浏览器端通过JavaScript实现，能够和任何服务器端的技术完美结合。通常，我们通过两种手段来解决动态应用和静态文档之间不匹配的问题：

1、类库：一些在开发WEB应用时非常有用的函数的集合。代码起主导作用，并且决定何时调用类库的方法。例如：jQuery等。

2、框架：一种WEB应用的特殊实现，代码只需要填充一些具体信息。框架起主导作用，并且决定何时调用你的代码。例如：knockout, ember等。

而Angular另辟蹊径，它尝试去扩展HTML的结构来弥合以文档为中心的HTML与实际Web应用所需要的HTML之间的鸿沟。Angular通过指令（directive）扩展HTML的语法 ，可以通过 {{}} 进行数据绑定，使用DOM控制结构来进行迭代或隐藏DOM片段等等。

## 3．进展情况

如今不少国际高校的宿舍改革规模较大，系统也越来越成熟。其中北美欧洲等发达地区的高校信息化建设开设历史久远，和其他后发展的高校宿舍管理系统更进步的是通过紫外线感应人脸识别系统，通过头像扫描记录个人信息并且存入系统，这一个方便快捷的优势可以长期存在，但美国的部分地区的高校已经开始进行进一步的改革措施，最新采用技术十分精准的指纹识别进行学生信息的认证，这是一种生物特征识别方法，可以有效的避免学生身份信息的作假和系统的误区。学生寝室的钥匙必须放置于规定的储物柜内，学生需要进行两秒的指纹认证方能取出钥匙，并且宿舍的周围都装有多功能的监控设备实时掌控宿舍周围的环境。为了保障更安全的宿舍环境以及财产保护制度监控中心会根据实时情况进行分析和处理，一旦有异常现象出现，宿舍的严惩制度十分严格。因为可以随时提供自动洗衣房以及娱乐场所，美国的高校宿舍服务深受同学支持。

## 4．存在问题

1. 高校宿舍管理权和学生权利平衡

随着法治的发展，“特殊权利关系理论”、法律的“保留原则” 在高校的管理实践中空间越来越少，相反地，权利意识的空前高涨，导致对特殊人群（包括弱势群体）的保护力度越来越大，学生宿舍虽然具有“公”的性质，但不是完全封闭的，并不代表绝对排除“私”的性质，只是不凸显而已[10]。因此，在设计系统时管理人员和学生两者拥有权限的平衡就必须要纳入考虑之中，对于管理人员用户来说，能够操作学生的用户肯定是不可取的，要尽量把两者用户放在同一层面上。

2.系统的功能设计

为了满足大多数用户的需求，设计本系统需要本着实用、先进、开放、可靠、可扩展的设计原则，围绕着学生从入住到退宿的每个环节，将宿舍资源管理、学生入住管理、宿舍日常管理、学生退宿管理等模块有机地集成[11]。

3.网页开发上的安全问题

计算机的安全问题似乎每天都出现。黑客、病毒、蠕虫、木马、拒绝服务攻击都称为常见的术语了，每个计算机用户都遇到过这些麻烦的事情,在受到攻击后,系统遭到破坏,丢失了重要数据，因此安全问题也是必须要注意的[12]。目前值得关注的几点有：

A.系统是否有超时限制。即当用户在登陆情况下长时间没有操作的情况下，是否需要重新进行登录操作才能够继续浏览网页。

B.SQL注入。SQL注入攻击的存在的主要是因为对用户输入的原始数据没有进行严格的验证，使得攻击者有机会利用用户输入将SQL命令或者其他非法的数据经Web系统传递到数据库，获取数据库中的数据或者进行恶意删除修改[13]。

C.跨站脚本攻击。也成为XSS，简单来说就是网站漏洞从用户那里恶意盗取信息，现如今，黑客的技术越来越强大，他们为了获取想要的机密信息，会采取很多非法手段，跨站脚本攻击就是常用手段之一，他们善于抓住网站中存在的网站漏洞和缺陷，设置有陷阱的链接，例如被插入恶意代码的链接，用户在点击链接之后，信息就会被盗取[14]。

参考文献（含开题报告和文献综述）

[1] 孙金娟. 我国高校数字校园建设解决方案[D].上海：华东师范大学，2004.

[2] 孟世和. 重庆宽仁医院医疗设备管理信息系统的设计与实现[D]. 北京：电子科技大学，2013.

[3] 李理、李哲. 美国高校数字校园的发展及推动因素研究[J].中国教育信息化（高教职教），2009（19）：16 - 19.

[4] 王明德. 中国家电工业现状分析[J]. 中国经济信息，2001，（7）：33 - 41.

[5] 万德生. 高校宿舍管理系统的设计与实现[D].吉林大学,2016.

[6] 赵相彬. 数字化迎新系统宿舍分配模块的设计与实现[J].电脑知识与技术,2016,12(25):92 - 94.

[7] 赵志成. 基于J2ee协同办公管理系统的设计与实现[J].哈尔滨师范大学自然科学学报，2015，31(1):85-87

[8] 王永和，张劲松，邓安明，周智勋.Spring Boot研究和应用 [J].信息通信,2016(10):91-94.

[9] Balasubramanee, V., Wimalasena, C.,Singh, R.,Pierce, M.. Twitter bootstrap and AngularJS: Frontend frameworks to expedite science gateway development [P]. 2013.

[10] 廖善光. 高校宿舍管理权和学生权利的博弈与平衡[J].教育理论与实践,2015,35(33):15 - 17.

[11] 孙赟,孔钦. 基于Spring+Extjs的高校住宿管理系统的设计与实现[J].电子技术与软件工程,2016(24):63 - 65.

[12] 冯学军. 基于SSH框架的Web网站设计与实现[D].长春理工大学,2010.

[13] 胡敏. Web系统下提高MySQL数据库安全性的研究与实现 [D].北京邮电大学, 2015.

[14] 石慧，董俊磊.Web网页服务安全防护对策分析 [J].网络安全技术与应用，2014 (11):98+100.

毕业设计（论文）

译文及原稿

译文题目 基于ANGLARJS框架的Web应用模型驱动开发

原稿题目 A Model-Driven Development of Web Applications Using AngularJS Framework

原稿出处 [Chansuwath W , Senivongse T . [IEEE 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS) - Okayama, Japan (2016.6.26-2016.6.29)] 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS) - A model-driven development of web applications using AngularJS framework[J]. 2016:1-6](https://ieeexplore.ieee.org/document/7550838)

基于AngularJs框架的Web应用模型驱动开发

**摘要**

**AngularJS是现代单页Web应用程序开发的广泛使用的框架之一，其设计目的是支持应用程序中的动态视图。为了进一步帮助AngularJS开发人员，本研究提出了模型驱动开发的概念如何应用于基于AngularJS的开发。我们提出了一个Unjaljs的UML配置文件，用于构建一个角度Web的模型。我们提出了一个Angular的UML配置文件来构建AngularJS web应用，还有将模型转换为代码模板的一组转换，然后开发人员可以填写模板来完成一个完整的可操作的Web应用程序。此外，开发了一种变换工具以帮助构造代码模板。使用案例研究应用程序，从转换率的角度进行评估表明，自动生成的代码覆盖了案例研究的完整代码的87%，这意味着它可以大大缩短开发时间。**

**关键字：模型驱动开发 ；AngularJS；UML配置文件；web应用**

# 1.介绍

软件公司需要尽快交付软件产品以保持竞争力，并且必须找到方法减少上市时间，同时为客户生产高质量的软件。对于这种需求，一种响应是模型驱动开发的概念，它使用软件模型来驱动软件开发。OMG模型驱动体系结构（MDA）[1]通过将一个软件模型转换为另一个模型来促进模型驱动的开发。软件模型可以处于不同的级别，例如，独立于平台的模型（PIM）级别（它是表示业务功能的软件模型）、特定于平台的模型（PSM）级别（它是专注于在特定平台上实现的业务功能的软件模型）和代码级别。一个级别的模型可以基于一组转换从另一个级别的模型自动生成。这种机制不仅有助于减少开发时间，而且降低了在开发过程中引入bug的风险，并提高了应用程序的可移植性。

模型驱动的开发概念已经被应用到各种软件的开发中，包括web应用程序。随着时间的流逝，web应用程序技术已经显著改进，使得web应用程序现在更像桌面应用程序，例如在性能方面。一种在响应时间方面改进web应用程序性能的技术是通过将网站创建为单个页面应用程序[2]。在单个页面应用程序中，所有代码都是通过动态加载到页面的年龄或资源的单个加载来检索的，而不必不断地重新加载页面或将控制转移到其他页面。将Web站点作为单页应用程序是谷歌的AngularJS〔3〕〔4〕。由于它的流行性，我们看到将模型驱动的开发概念应用于基于AngularJS的Web应用程序将极大地有益于Web应用程序的开发。

在本研究中，Web应用程序的开发将由PSM级的模型即AngularJS平台的模型驱动。我们使用UML概要文件机制[5]来扩展UML来建模基于AngularJS的应用程序。Web开发人员可以使用这个用于AngularJS的UML概要文件来构建Web应用程序的设计模型。基于所提出的转换模板，开发人员可以填写从模型生成的代码模板，以便构建完整的应用程序。我们还使用大型软件公司开发的金融应用程序开发了一个转换工具并报告了对该方法的评估。

第二节提出了一些背景和相关工作。第三节描述了AngularJS的UML概要文件以及用于代码生成的转换，它们被应用到第四节的案例研究中。

# 2. 背景及相关工作

## A. 模型驱动开发和UML概要

如前所述，模型驱动的开发概念，如MDA[1]，可以用于从模型生成软件代码。开发团队可以通过首先在功能PIM级别构建软件模型（例如，在UML中）来响应业务需求。通过使用识别转换规则的转换工具，可以将该模型转换为较低级别的模型，即PSM级别和代码级别。这些规则定义了用于在两个连续级别上构建模型的元模型之间的映射。在本研究中，UML概要文件[5]用于扩展UML元模型，以创建用于构建PSM级别模型的特定于平台的元模型。UML配置文件允许定义原型（由<<>>表示的特定元类）、标记值（如特定978-1-5090-0806-3/16/$31.00版权2016IEEE ICIS 2016、6月26-29、2016年冈山市、日本元属性）和约束，它们一起用于在规范上对软件建模于特定平台，即AngularJS平台。

## B. AngularJs

AngularJS [3][4] 是用于创建前端单页面web应用程序的流行JavaScript框架。它被设计成支持动态视图，使得浏览页面像在本地应用程序中那样顺畅。重要特征是：

1. 模型-视图-控制器：这种体系结构将应用程序分为三层：(1) 视图对应浏览器上显示的用户界面， (2) 模型对应视图上显示给用户的数据， (3) 控制器对应控制视图上显示数据的逻辑。
2. 模板：这是一组HTML，它被转换为文档对象模型（DOM），用于显示用户界面。
3. 双向数据绑定：这是一种视图根据模型的改变而改变的机制。
4. 依赖注入：这是一种机制，允许AngularJS在处理任何事情之前完全加载所有服务。

5）指令：这是一组由开发人员编程的模板。

## C. 相关工作

现在存在用于Web应用程序的基于UML的建模和模型驱动开发的几种方法。例如，黄等人（6）提出了一种用于在PSM级建模Web应用程序的UML概要文件，并生成了Servlet、JSP和Java代码模板。Hsu [7] 给出了用于web 2.0mashup的UML概要文件，该概要文件用于使用map和web提要服务的Web mashup的模型驱动开发。Kaewkao和Senivongse[8] 基于建议的UML配置文件生成他们的Google App Engine应用程序。其他研究人员，如Mubin和Jantan[9]只给出了一个UML概要文件，用于对Web应用程序的概念、导航和用户界面特征建模，而Ka.a等人。[10] 呈现一个以组件为中心的UML配置文件，用于对使用ASP、JSP、PHP、Servlet和JavaBeans技术的web应用程序建模。与这些研究类似，我们通过基于UML概要文件的建模来驱动Web应用程序的开发，但是是以AngularJS平台为目标的。

# 3. 基于AngularJS的模型驱动开发

本节概述我们的方法，AngularJS的UML概要文件，模型到代码的转换。

## A.概述

图1说明了该方法的概述。给定软件需求，系统分析人员使用UML建模工具基于AngularJS的UML概要文件为应用程序构建模型（步骤1）。模型导出为XMI格式[11]（步骤2），然后用作M2C（模型到代码）代码转换器的输入，M2C是识别转换模板的转换工具（步骤3）。转换结果是盎格鲁JS源代码模板（步骤4）。给定代码模板，开发人员将进一步填充业务逻辑以生成完整的Web应用程序。

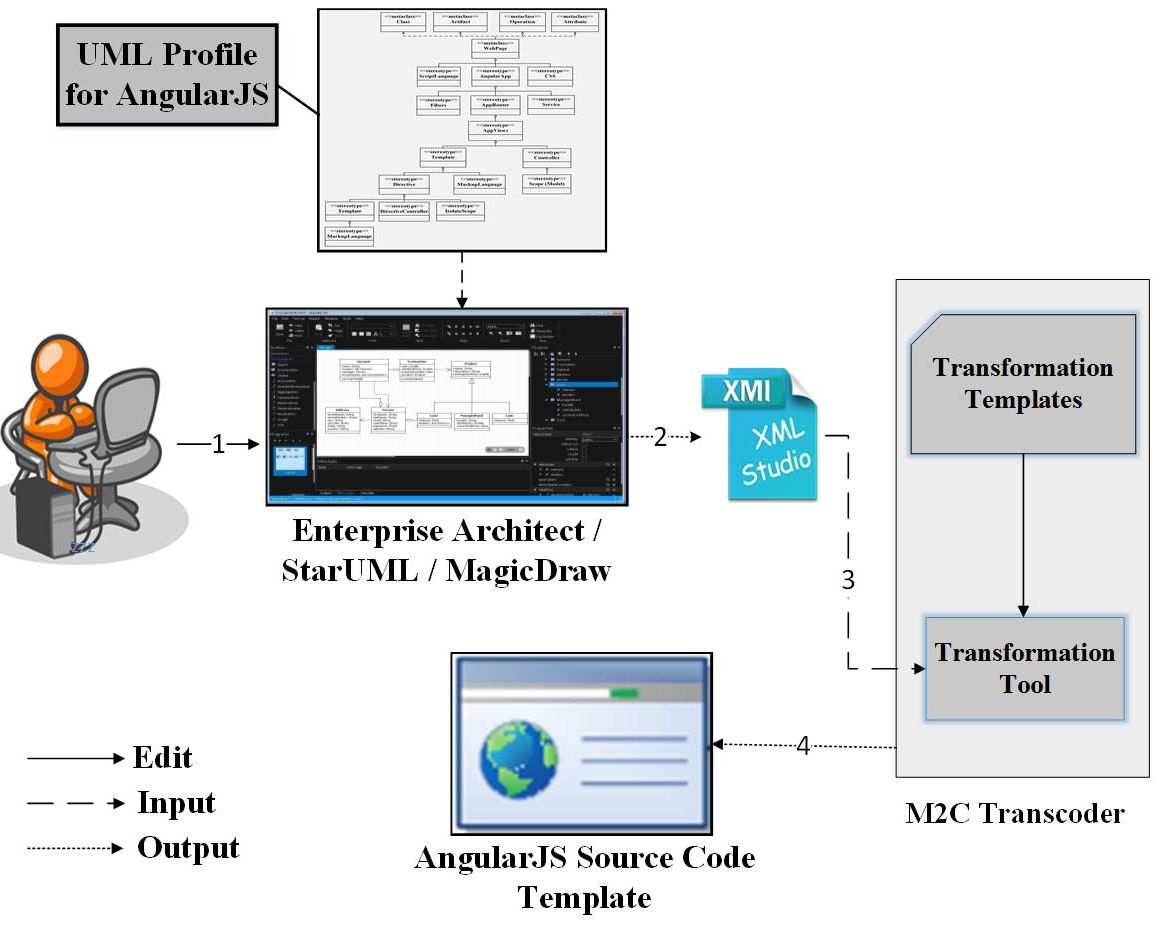


图1 方法概述

## B. AngularJS的UML概要文件

图2中描述了AngularJS的UML概要文件，表I列出了AngularJS域中所有原型的定义。

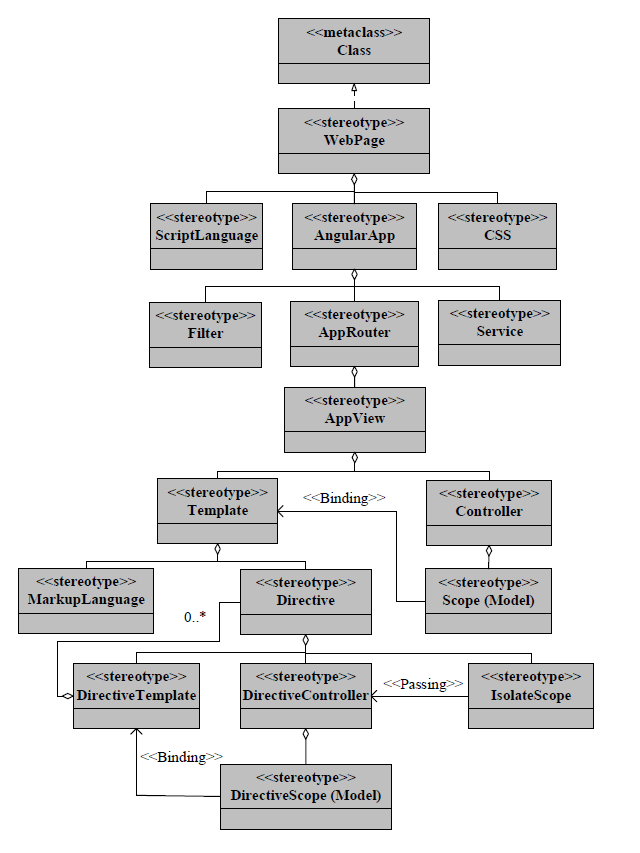


图2 AngularJS的UML概要文件综述

表1 AngularJs原型

|  |  |  |
| --- | --- | --- |
| 原型 | UML元模型 | 描述 |
| AngularApp | Class | 定义整个AngularJs应用 |
| AppRouter | Class | 定义要导航的视图 |
| AppView | Class | 定义单个网页的整个用户界面 |
| Binding | Association | 表示范围和模板之间的关系。如果Scope被更改，它将立即影响在Web浏览器上显示Scope值的模板。 |
| Controller | Artifact | 处理所有业务逻辑的函数 |
| CSS | Artifact | Web应用中的CSS标签 |
| Directive | Artifact | AngularJS指令 |
| DirectiveTemplate | Artifact | 指令的部分HTML |
| DirectiveController | Artifact | 来处理指令内的所有业务逻辑函数 |
| DirectiveScope | Artifact | 指令使用模型 |
| Filter | Artifact | 格式ANGLARJS应用程序的任何值的函数 |
| IsolateScope | Class | 用在指令里的模型 |
| MarkupLanguage | Association | 在客户端渲染HTML标签 |
| Passing | Class | IsolateScope与DirectiveController的联系。IsolateScope将把从Directive设置的值传递给DirectiveController，以处理任何业务逻辑。 |
| Scope | Artifact | 用于模板的模型 |
| ScriptLanguage | Class | 在Web应用程序中使用的JavaScript文件 |
| Service | Artifact | 在AngularJS应用程序中处理特定业务逻辑的共享函数 |
| Template | Artifact | AngularJS视图的用户界面 |
| WebPage | Class | 从Web服务器发送的网页 |

在这个配置文件中，AngularApp原型表示一个Angular Web应用程序。通过使用AppRouter，它可以在客户端的AngularJS Web应用程序中加载用户希望看到的新视图。AppRouter将控制如何重定向或导航到其他视图。视图是由AppView原型建模的。如果用户想要导航或重定向到新视图，AngularJS将在AngularJS web应用程序中加载该视图，而不需要向服务器请求在整个Web浏览器上呈现整个HTML。该机制将提供类似桌面应用的平滑响应用户体验。AppView看起来像一个普通的HTML标记，但是它可以被划分为称为Directives的较小组件，每个组件用于特定的目的。

指令使用模型视图控制器体系结构的概念，包括三个部分。第一个是DirectiveTemplate（View），它是要向用户显示的HTML。第二个是DirectiveScope(Model)，它是要绑定到DirectiveTemplate(View)的模型，也就是说，如果模型被更改，则视图被相应地更改。最后一个是DirectiveController（Controller），它是AngularJS的控制器，用于执行任何业务逻辑并将值设置回DirectiveScope以便在屏幕上显示。IsolateScope是一个属性，其值是从指令外部分配的。IsolateScope值将传递给DirectiveController，用于处理某些业务逻辑。

AngularJS框架的一个重要部分涉及如何在呈现到屏幕之前更改数据的格式。这称为Filter，AngularJS配置文件为此提供了Filter原型。

## C. 模型到代码变换

表I定义了许多来将AngularJS元模型映射到代码的转换。参见在下一节的图6和7。为了简单起见，我们以AngularJS代码模板的形式定义这些转换，在应用程序的模型中，基于AngularJS的UML概要文件为相应的标记值（即原型属性的值）设置占位符。当我们将这些模板应用于一个案例研究时，这些模板将再次得到解决。

# 4.案例学习

我们演示了如何将AngularJS的UML概要文件和转换应用于大型软件公司的一个案例，该公司为金融领域开发Web应用程序。案例研究是一个显示股票行情表的Web应用程序。交易员或经纪人可以监控价格变动，并比较表中不同股票的价格。图3显示了该Web应用程序的屏幕。本节的其余部分将讨论导致此屏幕的基于AngularJS的模型驱动开发。

首先，必须基于AngularJS的UML概要文件中的元模型元素来设计应用程序的PSM级模型。图4显示了整个模型。模型的指令部分（左下）在图5中进一步详细说明。我们描述了一些元素（那些由一个数字注释）如下。

这个单页应用程序被称为“应用程序”（图4中的1）。“App”是AngularJS框架在这个Web应用程序中使用的对象。在创建一个空的应用程序之后，我们必须创建一个视图来显示应用程序内部。我们创建一个名为quotePrice的视图，显示在stateName（图4中的2）中，用于显示股票符号和价格。AngularJS将自动向stateUrl（图4中的3）添加前缀#以导航到视图，即URL变成#/QuotePrice。

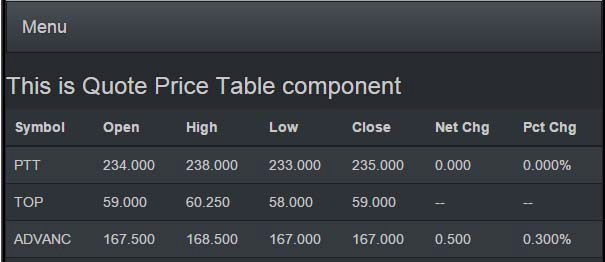


图3 案例研究应用的屏幕

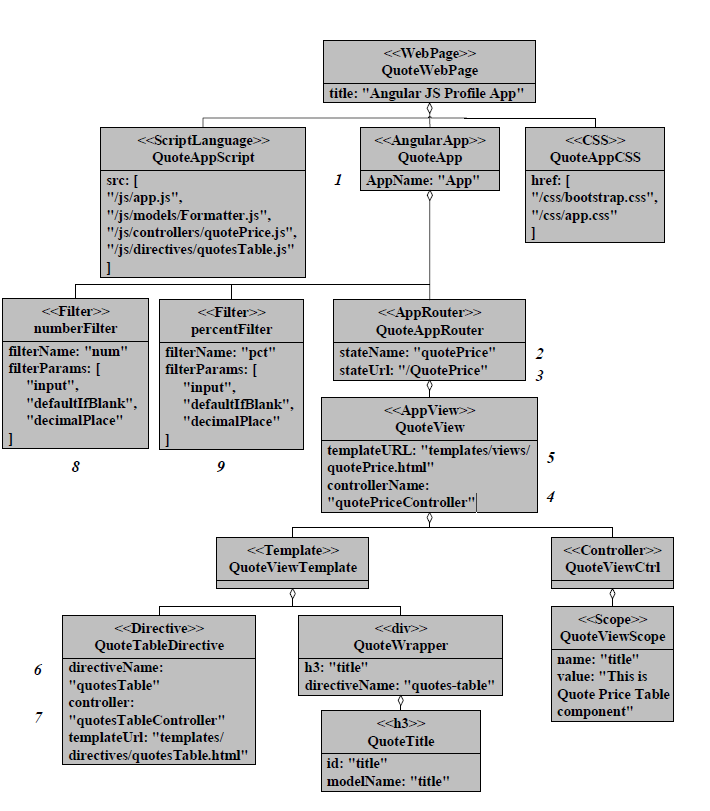


图4. 基于UML Profile的AngularJS案例研究设计模型

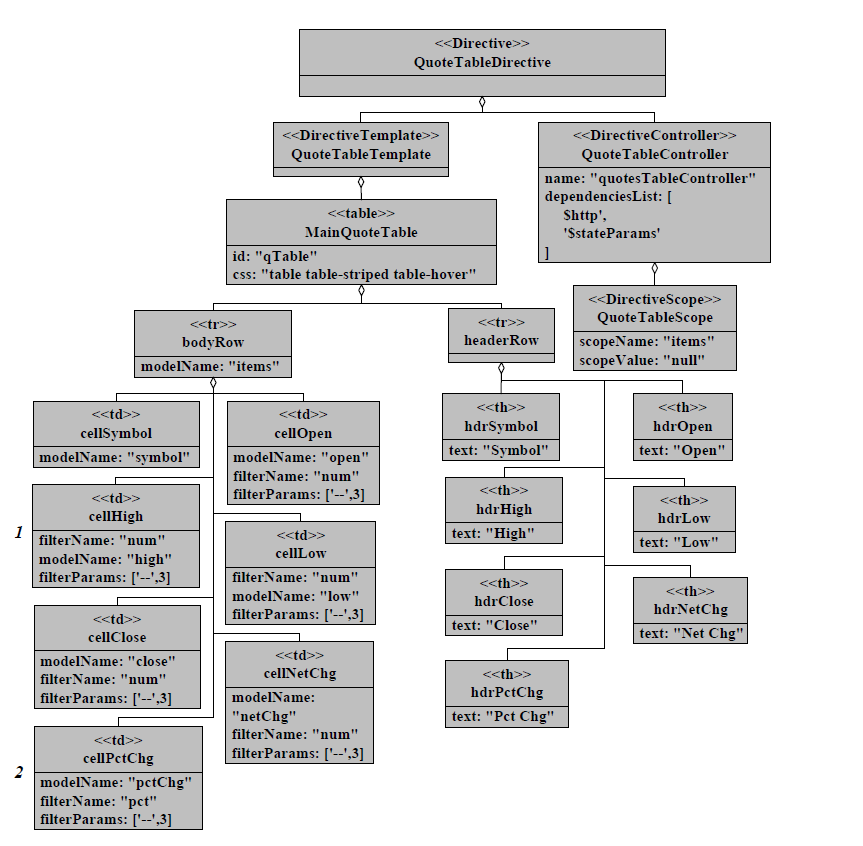


图5. 案例研究的设计模式指导部分

由于服务器必须托管所有与AngularJS相关的文件（即AngularJS库、HTML、CSS、JavaScript）和响应应用程序的Web浏览器请求，服务器将生成前缀URL作为http://localhost:9999/App/QuoteApp。因此，完整的URL是http://localhost:9999 /APP/QuoteAPP#/QuotePrice。

该视图的控制器是quotePriceController，如controllerName（图4中的4）所示，它执行业务逻辑，并将quotePrice.html（图4中的5）作为quotePrice视图的视图或用户界面。此外，quotePrice视图将添加一个名为quotesTable（图4中的6）的指令，用于在表中显示数据。这个指令的控制器是quotesTableController（图4中的7），这个指令的视图或用户界面是quotesTable.html。这个指令创建整个表，包括用于显示股票符号和价格的表头（例如symbol、open、high、low等）。在quotesTable指令中，我们在屏幕上显示之前，添加AngularJS过滤器（图5中的1、2）来格式化每个单元格的值。我们还创建了一个名为num（图4中的8）的过滤器，将数字值格式化为3个十进制数字。另一个过滤器称为pct（图4中的9），它将数字值格式化为三位小数，并在格式化后的值上附加百分号。在图3的表中，num Filter与Open、.、.、Close和Net Chg列一起使用，而pct Filter与Pct Chg列一起使用。

转换工具接收图中的设计模型。4和5（以XML格式）作为转换的输入。该工具识别图中的转换模板。6和7， 每个模板的关键组件是占位符“<<stereotype>>”属性，该属性将被模型元素的stereotype属性的标记值替换。

图中最右边的一列。6和7显示了用于案例研究的生成的Angular源代码模板。必须将更多的业务逻辑添加到代码中，以完成完整的股票报价应用程序。

# 5.评估

在评估中，我们根据转换率[6]来确定方法的性能，即与开发人员填充了应用程序的缺失业务逻辑之后的完整代码相比，可以自动生成的代码的百分比。尽管转换速率可以根据所添加逻辑的大小和复杂度而变化，但是它可以给出在特定应用程序的实现中可以节省多少努力的概念。我们根据代码行数（LOC）来测量代码大小。为案例研究应用生成的代码的LOC为95。我们有六个开发人员，每个都具有AngularJS经验，他们向生成的代码模板添加必要的代码，以生成完整的应用程序。结果见表II。生成的代码分别覆盖了较多和较少经验的组的84-90%的完整代码，即平均大约87%。

表2 评估结果

|  |  |  |  |
| --- | --- | --- | --- |
| 经验 | 平均手动添加LOC | 平均总LOC | 手动添加LOC的% |
| 3年及以上 | 11 | 105 | 10% |
| 少于3年 | 18 | 113 | 16% |

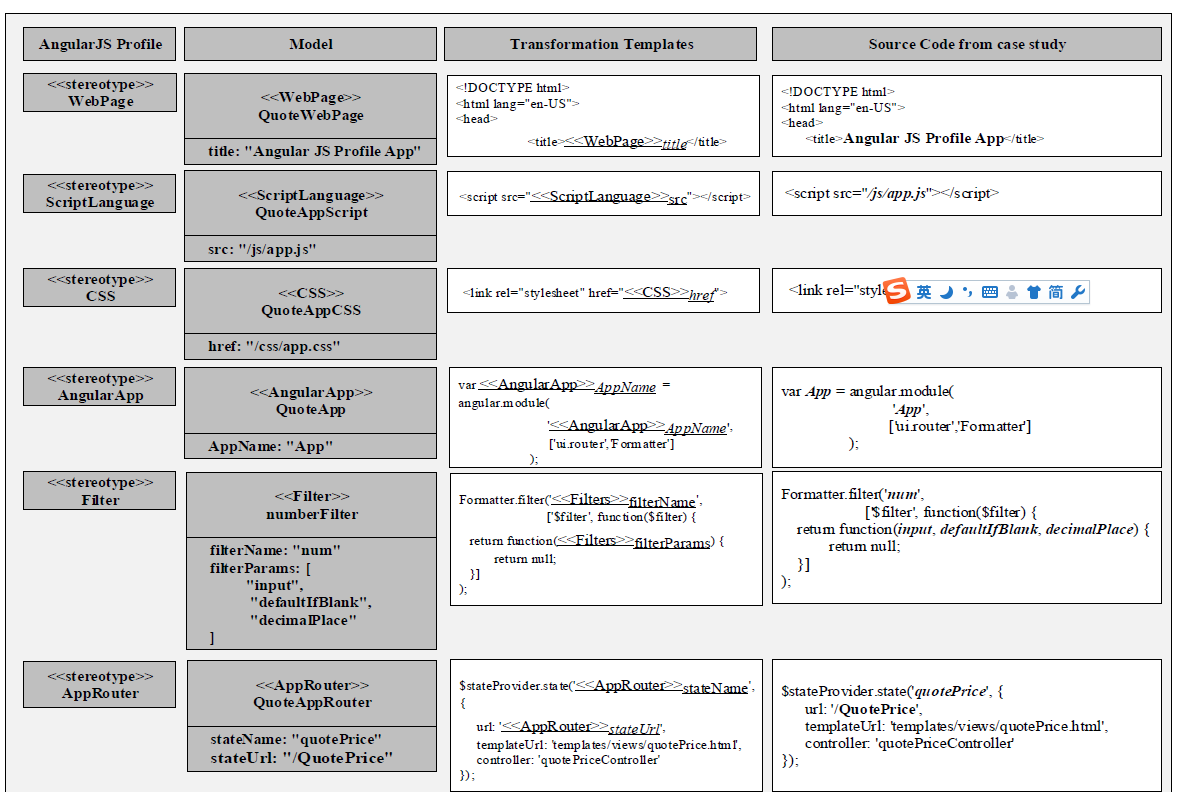


图6 转换模板和生成案例的源代码示例（第1部分）

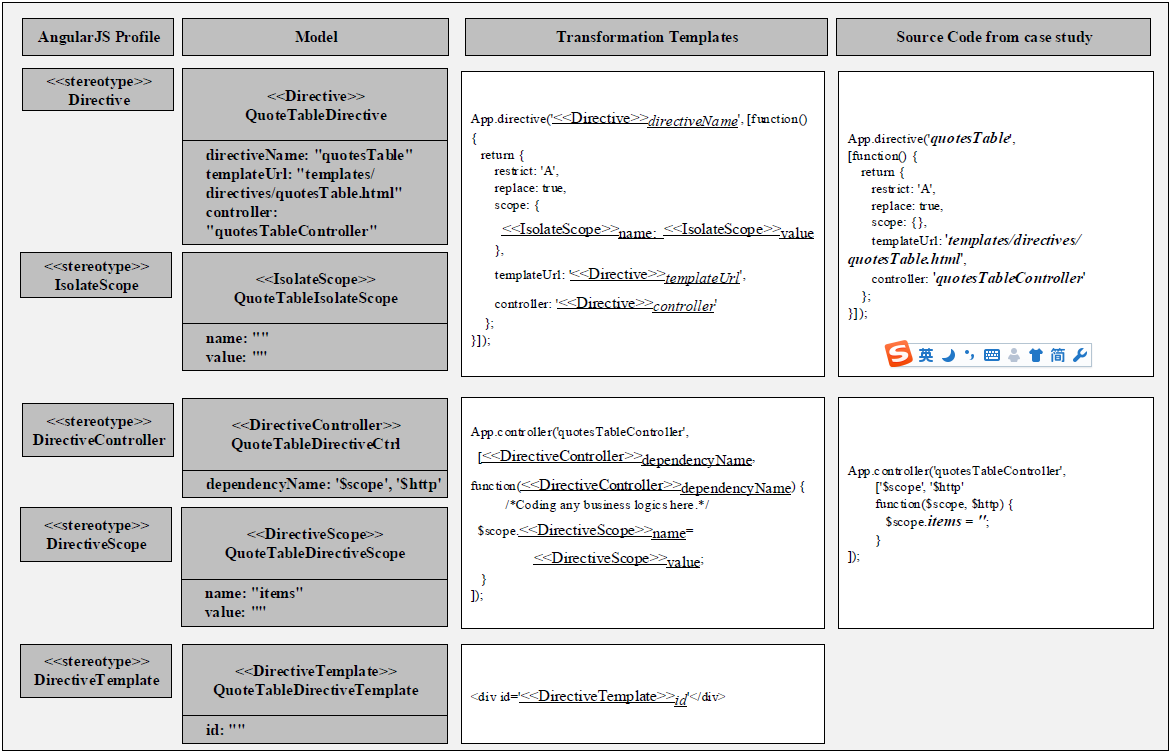


图7 转换模板和生成案例的源代码示例（第2部分）

# 6.总结

本文支持模型驱动的开发概念，提出了AngularJS的UML概要文件以及相应的转换模板，分别用于为AngularJS应用程序建模和生成代码。使用简单的案例研究应用程序进行的评估报告说，该方法能够将必须手动合并到应用程序中的代码行数平均减少到大约13％（即自动生成的87％）。尽管建模活动仍然需要花费时间，但我们获得模型作为有价值的文档，可以帮助减少整个Web应用程序的编码时间。一般来说，这也可以降低开发过程中引入错误的风险。对于将来的工作，计划对不同大小和复杂性的应用程序进行更广泛的评估。所提议的UML概要文件的扩展将支持特定领域中的应用程序的建模。例如，金融领域的AngularJS的UML概要文件应该包括与现有金融元模型相对应的元模型。

# 参考资料

[1] OMG, MDA - The Architecture of Choice for a Changing World [Online]. Available: http://www.omg.org/mda/. Last Accessed: 30 Mar2016.

[2] M. Mikowski and J. Powell, Single Page Web Applications: JavaScript End-to-End”, Manning Publications Co., 2013.

[3] Google, AngularJS. [Online]. Available: https://angularjs.org/. Last Accessed: 30 Mar 2016.

[4] N. Jain, P. Mangal, and D. Mehta, “AngularJS: A modern MVC framework in JavaScript,” J. Global Research in Computer Science, vol. 5, no. 12, 2015, pp. 17-23.

[5] OMG, UML Profile. [Online]. Available: http://www.omg.org/mda/specs.htm. Last Accessed: 30 Mar 2016.

[6] Y.-C. Huang, C.-P. Chu, Z.-A. Lin, and M. Matuschek, “Transformation from web PSM to code,” Proc. Int. Conf. Distributed Multimedia Systems (DMS), 2009.

[7] I-C. Hsu, “Visual modeling for web 2.0 applications using model driven architecture approach,” Simulation Modelling Practice and Theory, vol. 31, Feb 2013, pp. 63-76.

[8] S. Kaewkao and T. Senivongse, “A model-driven development of webbased applications on Google App Engine platform,” Proc. 10th National Conf. Computing and Information Technology (NCCIT 2014), Bangkok, Thailand, 8-9 May 2014, pp. 140-145 (in Thai). [9] S. A. Mubin and A. H. Jantan, “A UML 2.0 profile web design framework for modeling complex web application,” Proc. 2014 Int. Conf. Information Technology and Multimedia (ICIMU), 2014, pp. 324-329.

[10] M. Kataria, R. Yadav, and A. Khunteta, “A component-centric UML based approach for modeling the architecture of web applications,” Int. J. Recent Research and Review, vol. V, Mar 2013, pp. 22-27.

[11] OMG, Documents Associated With XML Metadata Interchange (XMI), Version 2.4.2 [Online]. Available:http://www.omg.org/spec/XMI/2.4.2/. Last Accessed: 30 Mar 2016.