

中图分类号：

密级：公 开

学科分类号：

论文编号：212206036

山东财经大学

硕士学位论文

（专业学位）

F 公司敏捷开发软件项目进度风险管理研究

作者姓名：陆文倩

学科专业：工程管理（MEM）硕士

指导教师：赵爱武（教授）

合作导师：刘春

培养学院：管理科学与工程学院

二〇二三年三月四日

Research on Schedule Risk Management of Software Project Based on Agile Development in Company F

A Dissertation Submitted for the Degree of Master

Candidate: Lu Wenqian

Supervisor: Zhao Aiwu

School of Management Science and Engineering
Shandong University of Finance and Economics

中图分类号:

密级: 公 开

学科分类号:

论文编号: 212206036

硕士学位论文

F 公司敏捷开发软件项目进度风险管理研究

作者姓名: 陆文倩

申请学位级别: 管理学硕士

指导教师姓名: 赵爱武

职称: 教授

学科专业: 工程管理 (MEM)

研究方向: 大数据工程与管理方向

学习时间: 自 2021 年 9 月 1 日起至 2024 年 6 月 30 日止

学位授予单位: 山东财经大学

学位授予日期: 2024 年 6 月

摘要

伴随软件行业和信息技术的飞速发展,软件产品尽早投放市场对于企业而言意味着占据市场份额、降低成本、提高客户忠诚度,在市场上获得更多的竞争优势,实现可持续发展。而传统开发方式已经无法满足市场要求,敏捷开发作为一种灵活软件开发方法,已经在软件开发行业中得到广泛的应用。然而,由于项目进程的随机性,敏捷开发也面临着一定的进度风险。F 公司作为软件行业的小型企业,有着立足市场、稳步发展的长远目标,按时交付是维护企业竞争力的重要指标之一。

本文梳理 F 公司的企业和项目情况,并列举当前进度管理过程应用的管理方法。进而从需求、沟通、人员方面总结当前进度管理中的存在问题。利用文献分析法、访谈法和头脑风暴法收集到 F 公司敏捷开发项目的 25 项风险因素。在此基础上结合风险矩阵法对 25 项风险因素进行评价进一步收缩范围为贝叶斯网络建模准备。借助贝叶斯网络分析工具对现有研究内容建模,并计算出进度延迟最大致因链为:前期规划不合理导致的资源配置不到位;项目估算不准确导致的资源配置不到位;以及开发人员技术能力不足三条链路。并结合 F 公司实际情况,从项目管理、团队管理、客户管理三方面针对性管理建议。

综上所述,本文通过定性和定量分析的科学评估和论证,将有助于 F 公司敏捷开发软件项目更好的进行进度管理,降低进度风险发生的可能,为实践中的敏捷开发项目提供有益的参考和指导。

关键词: 敏捷开发; 进度管理; 风险管理; 贝叶斯网络

Abstract

With the rapid development of the software industry and information technology, the early launch of software products means that enterprises occupy market share, reduce costs, improve customer loyalty, gain more competitive advantages in the market, and achieve sustainable development. As a flexible software development method, agile development has been widely used in the software development industry. However, due to the randomness of the project process, agile development also faces certain schedule risks. As a small enterprise in the software industry, Company F has the long-term goal of market-based and steady development, and timely delivery is one of the important indicators to maintain the competitiveness of enterprises.

This thesis reviews the enterprise and project situation of Company F, and lists the management methods applied in the current progress management process. Then summarize the existing problems in the current progress management from the aspects of demand, communication and personnel. Through literature analysis, questionnaire survey and brainstorming, 25 risk factors of agile development project of Company F were collected. On this basis, the risk matrix method was combined to evaluate 25 risk factors and further shrink the range to prepare for Bayesian network modeling. The Bayesian network analysis tool is utilized for modeling the existing research content, and the most prevalent causal chain of schedule delay can be calculated as follows: inadequate resource allocation resulting from unreasonable early planning; insufficient resource allocation caused by inaccurate project estimation; and suboptimal technical proficiency of developers within three links. And the developer's technical ability is insufficient for three links. Combined with the actual situation of Company F, targeted management recommendations were provided focusing on project management, team management, and customer management.

In summary, Through the scientific evaluation and demonstration of qualitative and quantitative analysis, it is helpful for Company F to better manage the progress of software projects adopting agile development mode, reduce the possibility of schedule risk, and

provide useful reference and guidance for the practice of agile development.

Key words: Agile Developmen; Schedule Management; Risk Management; Bayesian Network

目 录

第 1 章 绪论.....	1
1.1 研究背景与意义.....	1
1.1.1 研究背景.....	1
1.1.2 研究意义.....	2
1.2 国内外研究现状.....	2
1.2.1 项目进度管理研究现状.....	2
1.2.2 敏捷开发项目的进度管理研究现状.....	4
1.2.3 贝叶斯网络及其在项目进度风险管理中的应用研究	5
1.2.4 研究现状述评.....	6
1.3 研究内容与研究方法.....	7
1.3.1 研究内容.....	7
1.3.2 研究方法.....	7
1.4 研究思路与论文结构.....	8
1.5 主要创新点.....	9
第 2 章 相关概念及理论基础	10
2.1 软件项目敏捷开发概述.....	10
2.1.1 敏捷开发起源与定义.....	10
2.1.2 敏捷开发的特点.....	10
2.1.3 敏捷开发过程概述.....	12
2.2 项目的进度管理.....	13
2.2.1 项目进度管理内容.....	13
2.2.2 项目进度管理方法.....	13
2.3 风险识别与评估.....	14
2.3.1 风险识别.....	14
2.3.2 风险评估.....	15
2.4 贝叶斯网络基础.....	16
2.4.1 贝叶斯网络概念.....	16
2.4.2 贝叶斯网络原理.....	17
2.4.3 贝叶斯网络建模.....	18
2.4.4 贝叶斯网络应用.....	18
2.5 本章小结.....	19
第 3 章 F 公司敏捷开发软件项目管理现状及存在问题	20
3.1 软件项目概况.....	20
3.1.1 F 公司概况.....	20

3.1.2 F 公司项目总体概况	21
3.1.3 F 公司项目进度管理现状	22
3.2 软件项目进度管理存在问题及成因分析	22
3.2.1 过多的需求蔓延	22
3.2.2 沟通问题	24
3.2.3 人员问题	24
3.3 本章小结	25
第 4 章 F 公司敏捷开发软件项目的进度风险识别	26
4.1 风险初识别	26
4.1.1 进度风险因子识别过程	26
4.1.2 进度风险因子识别结果	27
4.2 进度风险因素确定与分析	31
4.2.1 进度风险因子评估标准的制定	31
4.2.2 进度风险因子评估分析	32
4.3 本章小结	37
第 5 章 F 公司敏捷开发项目进度风险推理与管理建议	38
5.1 贝叶斯网络风险评价模型构建	38
5.1.1 贝叶斯网络结构	38
5.1.2 参数学习	42
5.2 敏捷开发贝叶斯网络推理	44
5.2.1 逆向推理	44
5.2.2 最大致因链分析	46
5.3 进度风险管理建议	48
5.3.1 项目管理	48
5.3.2 团队管理	49
5.3.3 客户管理	50
5.4 本章小结	51
第 6 章 总结与展望	52
6.1 总结	52
6.2 展望	52
参考文献	54
附录 A 关于敏捷开发项目进度影响因素的问卷调查	57

第 1 章 绪论

1.1 研究背景与意义

1.1.1 研究背景

随着数字化转型进程的不断推进,软件项目已经成为推动经济社会发展的新动能。根据工业和信息化部发布的《2023 年软件业经济运行情况》可见,2023 年全国软件和信息技术服务业规模以上企业超 3.8 万家,行业竞争激烈;累计完成软件业务收入 123258 亿元,同比增长 13.4%,软件项目作为兼具复杂性、可变性、风险性、创新性、团队合作和生命周期等特点的软件开发活动,为软件行业提供了高速增长的业务收入和持续稳定的盈利能力。在工信部《“十四五”软件和信息技术服务业发展规划》中提到加快推动数字产业化,培育壮大人工智能、大数据、区块链、云计算、网络安全等新兴数字产业,提升通信设备、核心电子元器件、关键软件等产业水平。因此,面对软件行业激烈的竞争环境和政府政策利好,企业需要通过创新产品和服务、保证产品质量、控制研发成本、尽快推出市场等方式来提升客户满足度,占据市场份额,增强自身的竞争力。

软件项目的发展历程中,需求变更、资源分配、技术更新、时间管理的问题屡见不鲜。在软件开发过程中,客户的需求时常发生变化,导致项目范围的扩大,进度的延迟,以及成本的增加。软件项目涉及人力、硬件、软件、资金等多方面资源,分配不当将极大影响项目的进度和质量。使用新技术或实现复杂功能时面临技术挑战,需要不断更新和升级,以保持其竞争力和相关性。时间管理不当,诸如进度安排不合理、紧急任务的出现等,可能直接导致项目延期。因此,不断有新的开发方法和技术涌现。敏捷开发应对软件项目时具有快速交付、灵活适应变化、团队协作、持续改进和风险控制等优势,能够更好地满足用户需求和提高项目成功率。

进度是软件开发项目的重要组成,并直接关系到项目的成本、质量和客户满意度,是影响项目成败的关键因素之一。F 公司作为小型企业,为了更加灵活和敏捷的组织方式来适应市场变化,由初创阶段的瀑布模型调整为敏捷开发方式。尽管开发方式转变提升了团队效率,改善了进度延迟的问题,实际仍存在进度风险。进度风险可能发生于产品规划阶段、迭代开发阶段、测试阶段、集成和部署阶段中的任一或多个阶段,需要有效的进度管理规避。初创阶段的 F 公司管理方式方法主要参考其他成功企业,

套用部分管理模式或规章。随着企业规模和业务版图的扩大,项目延期问题频发,尽管采取了一定的管理,但管理模式的不适应性凸显。为了降低项目延期的发生,匹配企业持续发展,亟需适应企业自身特点的进度管理方式,改善项目延期现状,扭转引发的不良后果。

本文聚焦 F 公司敏捷开发软件项目,研究可能导致项目延期的风险因素。以贝叶斯网络构建模型,通过真实案例数据推演进度延期的可能性,借助进度管理的理论与方法对可能发生延期交付的风险提出调整策略。

1.1.2 研究意义

进度是项目的重要组成部分并且意义重大,它直接关系到项目的成功与否。项目中有有效的进度管理不仅确保软件项目按时交付,并且提高项目的质量、效率,降低延迟风险,增强团队凝聚力。当前很多学者关注到软件项目的进度管理,但较少聚焦到敏捷开发这一迭代、增量的软件开发方法在进度管理上的风险可能。其次,敏捷开发虽然使用占比逐年升高,但多数团队更多关注的是如何最佳实践,对于进度管理单方面关注度不高。实际敏捷开发过程同样可以划分生命周期,不同敏捷开发团队易发生风险环节不同。故项目延迟风险研究,助于深化对项目进度管理理论的理解,提高项目进度管理的实践水平。通过研究进度风险的发生规律和影响因素,可以完善进度风险管理的方法和策略,为实际项目提供理论指导。本文针对敏捷开发软件项目的延迟风险影响因素进行研究分析,引入贝叶斯网络构建模型,推算进度风险发生的可能性,具有较强的理论意义,并且对于善敏捷开发软件项目延迟有实际应用价值,为项目管理人员提供参考和借鉴。

1.2 国内外研究现状

1.2.1 项目进度管理研究现状

1970 年代至 1990 年代是软件工程的基础理论和实践的爆发期,当时按照物理工程的建设思路、管理理论与方法,由此产生了瀑布模型。在整个软件项目过程中,各个阶段受到人的影响极大,需求阶段客户本身就很难将需求表达准确、全面,而开发人员对于内容上的理解也总是存在差异。所以瀑布模型对客户需求的的要求极高,无法接纳需求的频繁变更^[1]。到 90 年代初软件项目管理方法遇到了前所未有的挑战,这种危机经常被称为“软件危机”^[2],表现为软件系统交付的严重滞后。对于项目的进

度管理,在美国最早可追溯至第一次世界大战,美国人亨利·劳伦斯·甘特首先提出将项目的活动进行排序并与时间相关联的方法——甘特图,以此来制定生产计划的进度图,甘特图可以直观的展示出计划进度与实际进展的对比,现如今甘特图在简单的中小型项目中仍然是简单实用的管理方法之一^[3]。

因为进度管理贯穿项目整个生命周期,很多学者从一个或几个环节上优化管理方法,实现进度管理的更有效实施。比如范俊辉(2018)从进度计划编制问题入手,总结控制项目进度的3个关键点:进度、成本、质量,通过管理措施提升进度控制^[3]。ZHANG Yu等学者(2020)为了解决里程碑延迟、人员效率低下、资源投入、外部环境变化等问题,基于关键链方法结合进度管理的相关理论、方法和技术,提出了一种新的项目进度管理模型,主要实现将不确定性纳入工期估算,将各工序的安全时间纳入统一管理,并验证改进后的关键链方法在实现缩短项目研发周期,提高员工工作效率上确有成果^[4]。王坤琦(2021)将进度时间预测算法、时间序列算法、项目进度管理软件PROJECT ONE结合,实现进度计划的调整,提高开发效率,优化软件功能^[5]。王一然(2021)在项目不同阶段引入层次分析法、Scrumbanfall模型管理方法、缓冲区监控、燃尽图监控和神经网络等扭转延期交付的局面^[6]。也有部分学者是在传统的进度管理方法上进行研究升级,以达到优化进度管理的目的。陈祥(2021)以管理信息系统建设为切入点,分析常用的方法包括甘特图、关键路径法、项目计划审评技术及计划制订,提出了相应的措施以提高进度管理的效果^[7]。刘心语(2022)利用wbs工作分解结构结合风险管理理论详细规划项目工时,提高工时估算的精准度;设立里程碑节点、运用挣值法跟踪项目开发过程中的进度、人员分配情况;完善沟通形式、培养员工积极沟通意识、在当地设立接口团队,畅通异地沟通做补充,形成一套有效的进度管理方案^[8]。另外,还有部分学者以更先进的理论或建模技术辅助项目进度管理,如周榕(2019)针对A公司F仿真软件定制开发项目采用因果图法分析项目过程出现的进度延误现象。通过优化项目计划、增加范围定义的准确度,加强团队建设等手段达到提高项目进度管理的目的^[2]。邵俊,倪枫(2022)等学者以TPM软件为研究对象,在软件开发进度管理模型基础上,同时结合熵权法针对进度管理风险进行量化分析,不仅解决了软件开发进度管理中流程不明、管理混乱的问题,而且大大提高了软件开发项目进度管理量化指标的合理性,可为实际生产经营中的软件开发项目进度管理提供参考^[9]。Acebes Fernando, Poza David(2022)等学者提出挣时管理(EDM)替代挣值管理(EVM),以应对活动成本不能很好估算项目持续时间的场景。

也有学者在研究中引入敏捷开发改善进度问题。刘华清（2013）分析敏捷开发特征，结合实际项目管理，研究敏捷开发进度管理中缩短项目工期、进度信息的获取与核实、进度信息的展现、传播及其激励作用^[10]。武红兰（2019）和陈建（2020）都是因为项目进度失控问题，为改善现状引入敏捷开发相关理论，为配合公司进度管理提出“瀑布+敏捷”模型，依据流程实践敏捷开发方法，并对实施效果进行评价与分析，推进软件项目进度管理的规范化和制度化^{[11][12]}。宋戈（2021）研究则是分析项目在组织、需求管理和项目执行上存在问题，并由此展开敏捷化调整。并通过 A 软件项目验证了敏捷模型，改进了项目的组织、需求管理以及项目执行的方法，切实的提升了软件开发效率和产品质量^[13]。

1.2.2 敏捷开发项目的进度管理研究现状

21 世纪，应对快速变化的需求，缩短交付周期，敏捷开发应运而生。赵剑东、林建（2007）两位学者注意到中小型软件项目面临开发时间紧迫、人手不足、需求不断变化的困难，而传统重量级的软件开发方法无法应对这样的挑战。打开思路引入敏捷方法解决轻量级项目管理和开发所碰到的问题。通过分析敏捷方法的主要目标、观点和原则，并结合一个实际的管理信息系统项目开发，从项目计划、项目文档、重构的改进和项目维护的 4 个方面探讨了敏捷方法的实践应用。并证明，采用敏捷方法的观点和原则进行必要的改进，能取得项目开发的成功^[14]。许孟杰、徐昶（2022）两位学者在 Java 编程技术基础和 HTML 网页设计两门课程教学中使用项目驱动的敏捷开发模型，以学生为中心，利用信息技术的优势，改变传统的教学方法，要求学生主动学习、解决问题。实际取得了较好的教学效果，对学生和教师均有提升^[15]。王晓龙（2023）基于敏捷开发设计并实现了一套持续交付系统，利用敏捷思想消除流程障碍，提高团队协作，结合持续交付实践，达成软件开发的持续性。流程以需求为驱动，对标软件开发周期囊括了需求管理、代码管理、持续集成、自动化部署、集成测试、结果反馈和数据维护几个阶段。通用性强且易于移植和实现^[16]。尽管敏捷开发的引入一定程度改善了原有的进度问题，但仍然存在诸多风险。过往学者也注意到敏捷开发可能存在的进度风险，并加以研究。Sultan Alyahya(2013)等学者研究了一种基于计算机的整体方法，以提高对分布式敏捷项目中开发进度的认识。该方法分析了技术活动（如源代码版本控制、单元测试、验收测试、集成和发布）如何影响开发进度，并提供了自动机制，帮助协调分布式团队成员的进度变化。整体方法的使用可以为团队成员提

供更好的进度意识,进而达成改善进度问题的目的^[17]。吴风景(2019)就保持进度计划的有效性和快速响应项目的调整展开研究。通过 Scrum 敏捷开发在项目中的实践过程入手,实践包含 Scrum 敏捷开发的实践流程、运用计划扑克法进行工时估算的过程、采用根方差法制定项目进度缓冲区、通过每日站会与迭代评审会议跟踪项目进度与检查结果等。分析结果显示 Scrum 敏捷开发在初创型企业软件项目中的应用能很好的提高项目进度、有效控制项目进度滞后、延期的问题^[18]。张立业(2019)在软件公司风险管理的大视角下,选择 Barry Boehm 风险管理模型,并与敏捷迭代周期相融合,建立起了适合敏捷软件开发的进度风险管理体系^[19]。Shrivastava Suprika Vasudeva、Rathod Urvash(2019)针对软件公司正在使用的分布式敏捷开发(DAD)提出一种目标驱动的方法来管理 DAD 项目中的风险。这种风险管理方法将使项目经理能够识别与要实现的目标相关的最重要的风险,并首先集中精力管理这些风险。影响特定项目目标的顶级风险因素的认识将有助于项目经理以一种能够实现预期项目目标的方式控制风险^[20]。Abhishek Srivastava、Deepti Mehrotra(2020)为了使软件开发过程富有成效,分析敏捷开发成功的关键因素,采用层次分析法(ANP)确定影响软件行业质量的最佳敏捷成功因素的方法,优化敏捷开发实践路径^[21]。Baham Corey、Hirschheim Rudy(2021)认为尽管敏捷软件开发已经历经 20 多年的理论研究,但敏捷软件开发的理论核心尚未被确定。所以研究提供了 ASD (agile software development) 研究的理论核心,阐明了敏捷性的本质和次要,并提供了裁剪核心含义的理解方法^[23]。

1.2.3 贝叶斯网络及其在项目进度风险管理中的应用研究

传统项目进度管理方法包括甘特图法、里程碑法、关键路径法、里程碑计划法等,上述方法多是通过编制计划,实际数据与计划数据比较来识别进度是否发生延期。而在项目推进过程中,当偏差出现往往已经发生事实延期,所以,相比传统的进度管理方法,预测可以在问题发生前将进度风险识别出来显然是更优解。

贝叶斯网络作为能够推理预测未知结果、帮助决策的工具,在风险预测、结果模拟研究中已经有很多应用。在使用贝叶斯网络对风险问题建模应用中,多需要其他理论、技术辅助。李义杰,蒋靖(2011)在 PERT 图的基础上构造贝叶斯网络模型,由专家判断和工程经验确定网络中的概率参数,提出了基于贝叶斯网络的软件项目进度管理模型,贝叶斯网络能够很好地表示变量的随机不确定性和相关性,为弥补当时评估方法不足提供了研究参考^[23]。严莉敏(2018)为应对风险动态变化的预测和量化控

制这一难题,提出了基于贝叶斯网络的风险管理模型。项目经理运用该模型可以达到对动态变化的风险最大程度管控的目的^[24]。胡姗(2019)以 ERP 项目作为切入点,结合风险识别方法和专家指导法建立风险管理指标体系,为后续风险管理模型构建风险识别子模型提供参考^[25]。尤建新,秦云(2019)两位学者在工程项目工作分解(WBS)和贝叶斯网络技术的基础上,建立了关键链项目管理贝叶斯网络模型(CCPMBN),通过蒙特卡洛仿真项目施工完成时间,证明 CCPMBN 模型为控制施工建设进度与实现提供了重要参考^[26]。杜元伟,石方园,杨娜(2015)提出了基于证据理论/层次分析法(DS/AHP)能够从专家推断信息中提取最优条件概率的方法,改善了依据专家知识推断贝叶斯网络中条件概率表(CPT)时存在完备性和精确性不足、整体集成结果缺乏科学性的问题,以此构建的贝叶斯网络与传统方法对比分析表明所提方法的科学有效性和应用可行性^[27]。

贝叶斯网络因为其建模不确定性、灵活的推理、解释性、数据驱动的学习等优势,被广泛的用于基建工程项目。Chen Zhaorong, Deng Xiaoji(2022)以建筑工程为研究对象,采用动态贝叶斯网络方法定量研究影响未完工建筑施工进度的主要风险源,对已开工 8 年以上的高层项目进行进度风险识别定量分析。通过逆向推理,确定了多高层旧建筑改造进度延迟的最大可能因素和最可能的原因链^[28]。Sung Hongsuk 等学者(2022)站在以项目为基础的运营公司立场,因为通常预估项目完成时间使用的“PERT/CPM 技术建模+蒙特卡罗模拟方法风险评估”方式不适用于运营公司,因为他们通常不具备上述技术需要的累计数据,而且不能反映项目执行过程中通常会满足的各种进度约束,因此,研究提出了一种基于贝叶斯网络的方法,通过识别和执行项目风险及其应对计划来估计项目进度风险,这些风险可能发生在储罐工程和建筑项目环境中。首先,将包含项目风险及其应对计划的进度网络转化为贝叶斯网络。其次,对综合贝叶斯网络进行了分析,提出了用仿真方法估算项目进度风险的方法。最后,将该方法应用于储罐建设项目,验证可行性^[29]。

1.2.4 研究现状述评

综上所述,尽管关于软件项目进度管理的研究和关注度一直很高,但由于其复杂性和关联性,仍然在理论方法和实践技术上有改善空间。从甘特图、关键路径法等典型进度管理手段结合模型、计算机技术优化。敏捷开发方法诞生背景是为是基于对传统瀑布式开发模式的反思和改进,尽管一定程度上解决部分问题,但软件项目管理同

样具有诸多风险，其中包括进度风险。对于敏捷开发软件项目的进度研究也按照生命周期原则看待，筛选进度风险因素可能出现的环节。因此，本文会全面考虑影响进度的各种风险因素。

1.3 研究内容与研究方法

1.3.1 研究内容

首先，搜寻项目进度管理、敏捷开发实践、风险管理、贝叶斯网络相关的研究文献，分析并总结现有研究的不足，确定论文的研究问题，总结其研究意义。

其次，根据敏捷开发生命周期不同阶段和 F 公司实际项目状况设计访谈内容，向参与敏捷开发的项目团队面对面交流，根据汇总访谈结果结合头脑风暴确定进度风险因素。在此基础上设计调查问卷收集进度风险发生频度和影响程度的数据，得到风险矩阵。

再次，采用贝叶斯网络分析软件 GeNIe 2.0 构建模型，组织 F 公司管理、技术专家对建模结构提供支持。确定 F 公司敏捷开发项目进度延迟的最大致因链，并提出针对性改善策略。

1.3.2 研究方法

(1) 文献分析法。文献研究法的前提是大量收集相关文献，并对收集的文献进行分类、分析、整理和总结，找到参考或辅助自己研究的过程方法。本文第四章通过查找历史文献，对敏捷开发和进度管理进行学习分析，参考利用历史文献研究的软件项目进度风险因子进一步拓展自己的研究思路。

(2) 访谈法。又称晤谈法，以口头形式进行研究性交谈，根据被询问者的答复搜集客观的、不带偏见的事实材料，以准确地说明样本所要代表的总体的一种方式。尤其是在研究比较复杂的问题时需要向不同类型的人了解不同类型的材料。

(3) 问卷调查法。是一种通过设计问卷，以书面形式收集信息的方法，主要用于调查和收集市场调查、社会调查、文化研究等方面的数据。本文通过 1 套设计问卷，通过针对性的发放，收集了 F 公司内开发团队成员对于导致敏捷开发软件项目进度风险发生下的影响程度。

(4) 头脑风暴法。是一种激发创造性思维的方法，于 1939 年由美国创造学家 A.F.奥斯本首次提出，1953 年正式发表的。头脑风暴法分为两类：直接头脑风暴法依

靠专家群体决策尽可能激发创造性，产生尽可能多的设想；质疑头脑风暴法则是对前者提出的设想、方案逐一质疑，分析其现实可行性的方法。

（5）模拟和仿真分析方法。模拟与仿真分析是通过使用数学模型和计算机算法来预测和模拟现实世界中的各种现象和系统行为。本文采用贝叶斯网络原理构建敏捷开发软件项目进度风险模型，进行项目进度的预测。

1.4 研究思路与论文结构

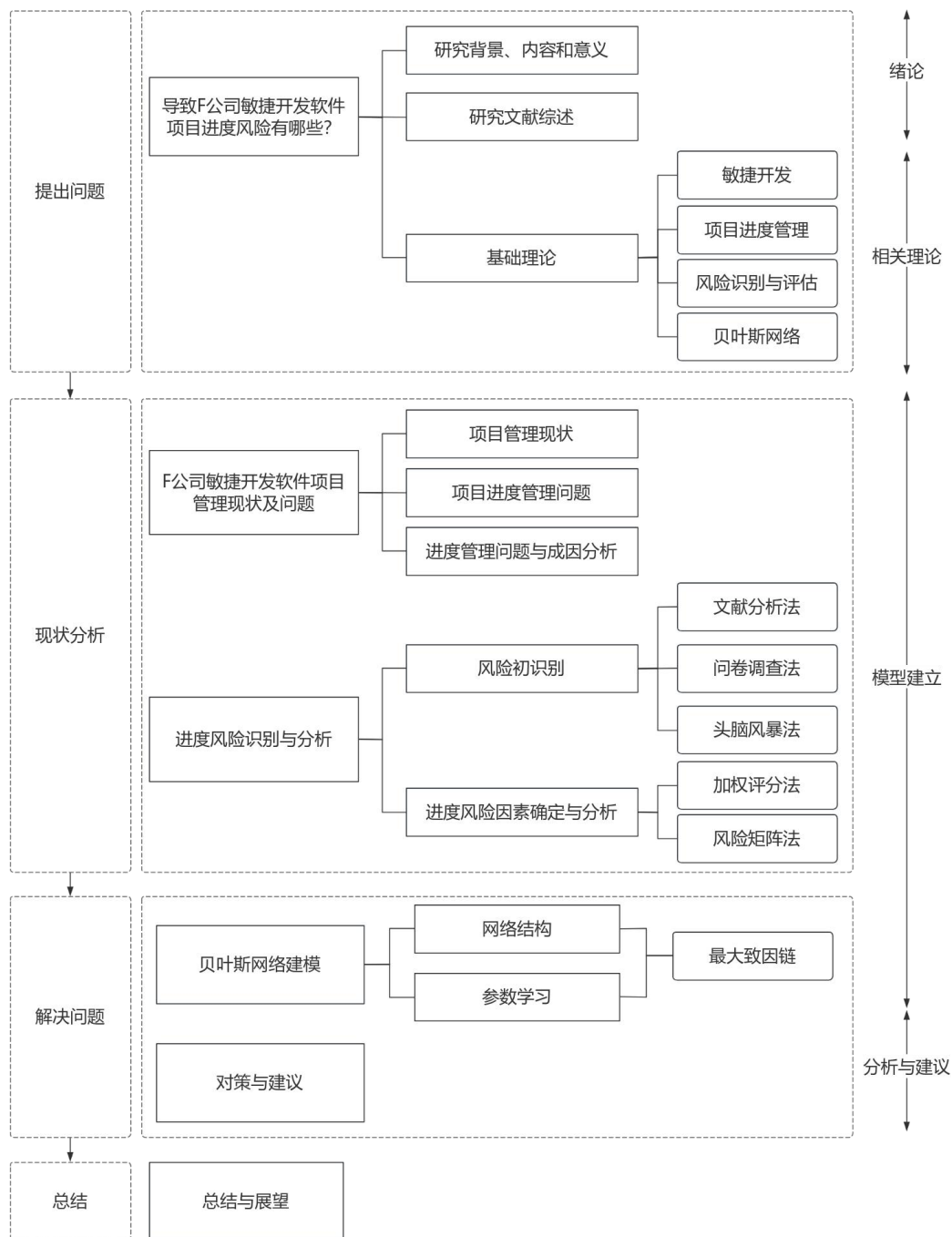


图 1-1 研究技术路线图

本文主要分为六个章节，第一章是绪论，主要阐述了研究背景与意义，收集整理国内外研究现状，应用到的研究方法和研究思路等内容；第二章为相关概念与理论基础，重点对四个方面进行了研究，即敏捷开发、进度管理理论、风险识别与评估以及贝叶斯网络基础理论；第三章整理 F 公司项目及进度管理现状，并分析了现有进度延迟问题的部分原因；第四章通过多种研究方法识别出 F 公司敏捷开发项目的进度风险因素，通过定性和定量的研究方法对风险因素进行过滤整理得到最终风险清单，作为后续建模的输入；第五章采用模型构建的方法借助 GeNIe2.0 软件对 F 公司敏捷开发项目进度延迟的相关因素和数据进行分析，得出最大致因链，并再次基础上从项目、团队、客户三个管理维度提出相关改善建议；第六章为归纳总结和对未来的展望。

1.5 主要创新点

传统的软件开发项目存在延期交付的问题，很多开发团队引入敏捷开发用以解决延期交付问题。实际上，敏捷开发方式同样存在进度延迟风险，本文研究以 F 公司的敏捷开发软件项目为入口，围绕其团队组成特点和现存问题汇总，借助文献研究、访谈、风险矩阵法，得出导致 F 公司敏捷开发软件项目的重要风险因素。进一步采用数学建模的方式，针对敏捷开发项目进度风险的不确定性和复杂性，引入贝叶斯网络，利用其推理能力找出 F 公司敏捷开发项目的进度延迟的最大致因链。对影响进度的敏感和关键风险因素深入挖掘分析并提出管理建议，对未来项目的进度风险管理有实际参考意义，是模型推理的创新应用，对进度风险管理实际应用层面有一定的创新，对于其他同行企业有一定借鉴意义。

第 2 章 相关概念及理论基础

2.1 软件项目敏捷开发概述

2.1.1 敏捷开发起源与定义

基于瀑布模型开发的软件项目往往存在延期交付的问题,不能适应软件需求的不确定性和多变性。2001 年,由 Martin Fowler、Jim Highsmith 等 17 位软件开发专家起草的敏捷宣言发表,敏捷联盟成立,敏捷开发作为一种新的方法正式诞生。

敏捷开发(agile development)是一种以人为核心、迭代、循序渐进的开发方法。在敏捷开发中,软件项目的构建被切分成多个子项目,各个子项目的成果都经过测试,具备集成和可运行的特征。也就是将一个大项目分为多个相互联系,但也可独立运行的小项目,并分别完成,在此过程中软件一直处于可使用状态^{[10][30]}。

2.1.2 敏捷开发的特点

在相当长的时间内,瀑布模型作为一种主流的软件开发模型,其管理思路总结于工程类的大型项目经验,一般项目周期长、需求相对固定。瀑布模型通过将软件生存周期的各项活动划分成按照固定顺序而连接的若干阶段工作,遵循从需求分析到设计,再到实现的线性过程,如图 2-1 所示,用户要在项目接近完成时才能看到产品。这意味着项目开发每一步都会受到前一步的约束,无法并行工作,降低了开发效率。另一方面,由于用户看到产品的时点相对滞后,导致反馈周期长,增加了误解和错误的风险。瀑布模型对于文档的依赖较高,每个阶段都要求明确的文档成果输出并强调其重要性,为此项目不得不投入大量时间和精力来编写文档。不仅延迟了开发进程,而且往往导致文档与实际代码之间存在差异。一旦相对后期的环节发现问题或存在突发情况,纠正过程将造成大量成本投入。需求分析阶段是固定且先行的,一旦进入开发阶段,需求变更就变得非常困难。然而,软件项目通常需要应对不断变化的市场需求和用户反馈,瀑布模型无法适应这种变化。

敏捷开发的出现对于以上问题提供了一个新的解决方式,通过把一个大项目分为多个相互联系,可独立运行的小项目,过程中软件一直处于可使用状态^[31]。与瀑布模型线性流程不同,敏捷开发轻文档以需求为驱动,在迭代期中持续编程-测试-发布的小循环,如图 2-2 所示,并在循环中逐步完善产品功能。敏捷宣言提出 12 条原则,

其中最高优先级原则即通过尽早和持续交付有高价值的软件，满足客户，一定程度解决了瀑布式开发在后期才能允许用户看到产品的弊端，同时也能够在迭代中消除误解和错误的风险。敏捷开发意识到需求是会随着时间变化的，因此非常注重对需求的灵活性和适应性。敏捷团队能够及时调整并响应需求变更，通过频繁的迭代和持续交付来满足客户的新需求。但变更一般不会对正在进行的 Sprint 进行调整，而是将变更内容安排在后续的 Sprint 中。另外，敏捷开发强调团队成员通过日常的沟通 and 协作，能够更好地理解 and 解决问题。高效的面对面沟通，团队成员拥有多样的技能和角色，能够相互补充和支持，提高工作效率和质量。敏捷开发通过频繁的迭代和持续交付快速获得用户反馈，团队根据反馈及时调整开发方向和优化产品，实现持续改进和适应市场需求的能力。敏捷开发的特点还包括透明和可见性，通过可视化的看板、日常的站会、进度报告等方式，使整个团队对项目的进展、问题和需求有清晰的了解，促进沟通和决策的透明性。

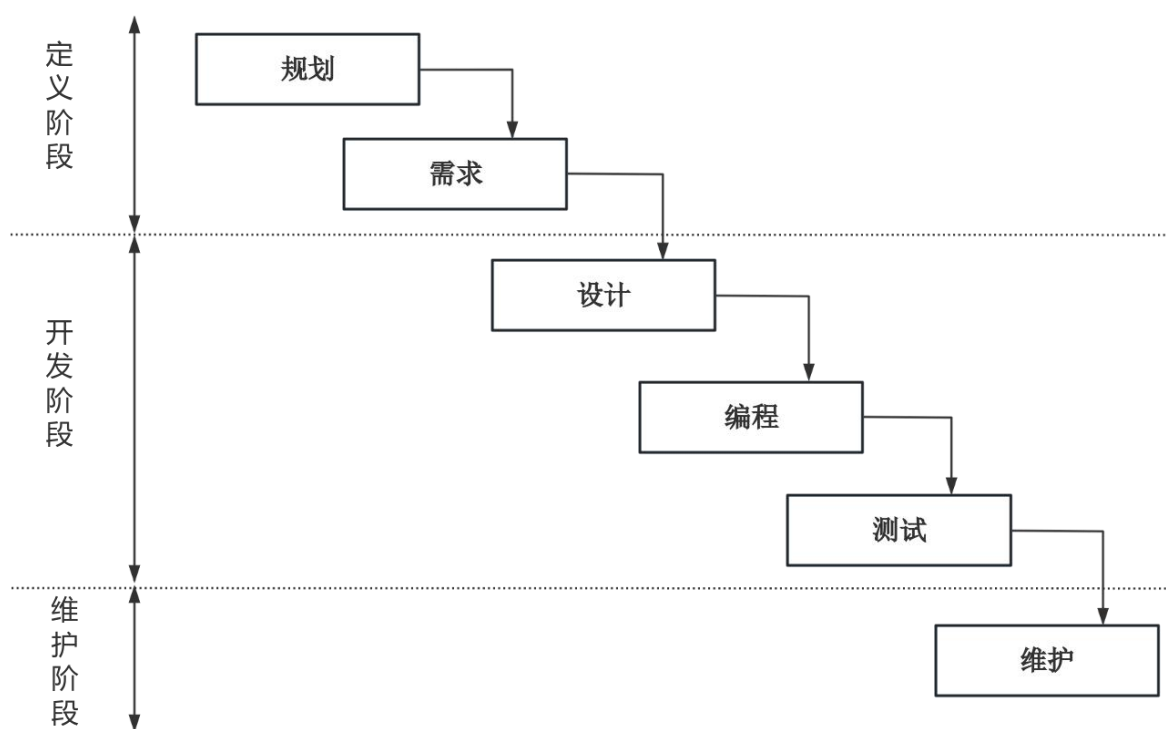


图 2-1 瀑布模型流程示意图

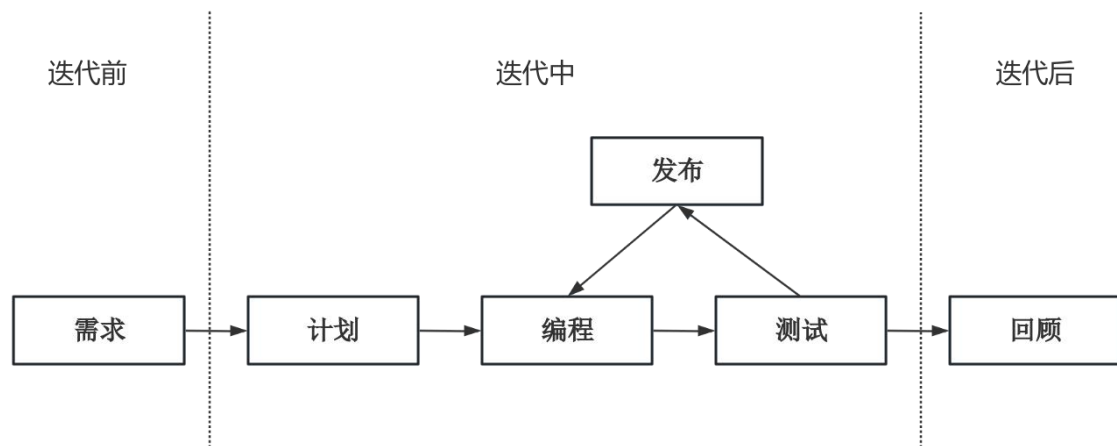


图 2-2 敏捷开发流程示意图

对比两种开发方式可见，在瀑布模型流程各阶段产出中，文档作为重要成果物，而敏捷开发弱化了文档内容。需求作为软件项目的重要环节，通常在项目启动阶段就要求输出成果，瀑布模型通常采用收集-分析-制定文档-评审的方式形成最终需求成果文件，而敏捷开发则是通过用户故事列表来收集形成最终产品需求。

2.1.3 敏捷开发过程概述

敏捷开发是一种以迭代、增量和协作为基础的软件开发方法。它强调灵活性、快速交付和持续改进，以满足客户需求并应对快速变化的市场环境。下面是敏捷开发过程的概述：

（1）需求规划与管理

在敏捷开发中，需求是持续变化的，因此需要进行规划和管理。在这个阶段，团队与客户和利益相关方合作，明确需求、优先级和期望，创建产品特性列表和用户故事。

（2）迭代规划

在迭代规划阶段，团队根据需求和可用资源，确定每个迭代的目标和范围。这包括评估故事点或规模，确定迭代的持续时间，并安排团队成员的工作。

（3）迭代开发

在每个迭代中，团队进行实际的软件开发工作。团队成员协作开发代码，使用自动化测试和持续集成来确保代码的质量。在迭代期间，团队与客户和利益相关方保持紧密的沟通，并及时获得反馈。

（4）迭代评审和回顾

在每个迭代结束时，团队进行评审和回顾。评审会议上，团队展示完成的功能和产品，接受客户和利益相关方的反馈。回顾会议上，团队对迭代的过程和工作进行总结，探讨改进的机会。

（5）迭代交付

在每个迭代结束时，团队交付可用的软件版本。这意味着每个迭代都会产生一个具有功能的、可测试的部分产品。这种持续的交付周期可以使客户和利益相关方尽早看到和体验产品。

（6）持续改进

敏捷开发强调不断改进的文化。团队通过回顾会议和团队讨论，评估过程中的问题和障碍，并制定改进计划。改进可以涉及流程、工具、沟通或技术方面的各个层面。

通过循环迭代的过程，敏捷开发团队能够灵活地响应需求变化，快速交付高质量的软件，并与客户和利益相关方保持紧密的合作和沟通。这种敏捷的开发方式鼓励适应变化、持续学习和优化，以实现项目或产品的最佳业务价值。

2.2 项目的进度管理

2.2.1 项目进度管理内容

项目管理的兴起与发展经历两个显著时期，首先是为匹配二十世纪五十年代大中型规模项目的实施需求，其次是乘风信息技术与知识经济的发展。前者促成项目管理成为一门学科，后者协助项目管理应用落地，提供了对应的实践方法与工具。项目进度管理作为项目管理的重要组成部分起着至关重要的作用，其专注于确保项目按时完成，侧重于对项目工作的合理安排、规划和监控，并及时发现和解决可能引发项目延迟的各种问题。因此，项目进度管理是一种将时间、成本、资源、计划与调整多维度融汇的管理方法。这种管理方式不仅有助于提高项目的效率和质量，还能增强团队的凝聚力和协作能力，从而为项目的成功提供有力保障。

2.2.2 项目进度管理方法

项目进度管理的方法大多侧重于计划编制，通过有效的工作分解与规划实现项目的有序进行。其中项目进度常用的计划技术如下：

（1）甘特图（Gantt chart）是以横线来表示每项活动的起止时间。这种灵活且易于理解的工具，广泛应用于各种项目管理环境中，是项目计划、进度控制和资源管理

的重要辅助工具。甘特图清晰地显示了项目的时间轴，用箭头或线段表示各个任务之间的依赖关系，帮助团队成员了解任务的周期和工期，以及任务之间的先后关系，以便进行优化和调整。但各项活动之间的关系却没有充分表示出来，同时也没有指出影响项目寿命周期的关键所在，因此，对于复杂的项目来说，甘特图就显得不合适^[32]。

(2) 关键路径法 (Critical Path Method, 简称 CPM) 是一种用于确定项目中最长持续时间的任务路径，以及整个项目的关键任务的项目管理技术。关键路径法通过分析任务之间的依赖关系和所需时间来帮助项目管理者规划、安排和控制项目进程。具体步骤包括任务识别-任务排序-估算任务持续时间-确定关键路径-管理与调整，其优势在于能够帮助项目管理者清晰地了解项目进程和任务之间的依赖关系，提高项目计划的准确性和可控性。它还可以帮助识别项目中的延误风险，并有助于资源分配和优化项目进度^[32]。

(3) 计划评审技术，主要用于确定项目活动的时间安排和关键路径。它通过网络图的形式来表示项目的活动顺序和时间依赖关系，并通过概率分析来考虑项目活动时间的不确定性，从而帮助项目管理人员更好地制定项目计划和控制项目进度。PERT 技术中的三个重要概念是活动、事件和路线。活动表示项目中的具体任务，事件表示活动结束的点，路线表示项目中一系列连续的活动。通过网络图和概率分析来描述项目的计划和进度，并提供了一个框架来考虑项目活动时间的不确定性。

2.3 风险识别与评估

2.3.1 风险识别

风险识别是指通过系统性的方法和过程，识别潜在的风险因素和事件，以便在其对目标实现产生负面影响之前对其进行预测、评估和管理的过程。风险识别常见方法如下：

(1) 文献研究法。文献研究法是一种系统性的方法，用于收集、评估和综合已有的相关文献和资料，以支持学术研究、论文撰写和决策制定。是一种基于已有知识和研究成果的学术研究方法。文献法的提出课题或假设是指依据现有的理论、事实和需要，对有关文献进行分析整理或重新归类研究的构思。根据本课题的研究目的，在大量收集文献的基础上，对文献进行分析、归纳和整理，进而在文献中找到自己研究的相关类似内容，并依此为参照，为自己的研究提供理论基础和研究背景。其优点在

于能够基于已有的研究成果，进行系统性的综合和分析，从而揭示研究问题的现状、潜在机遇和未解决的问题。通过文献研究，研究者可以避免重复研究、了解最新的研究动态和学术观点。

(2) 调查法。调查法是一种通过直接或间接的方式收集数据和信息的研究方法，被广泛应用于了解公众意见、市场趋势、消费者行为等方面，目的是为了获取有关特定主题的第一手或二手数据，以便进行分析和得出结论。调查法可以分为多种类型，包括问卷调查、访谈、观察法、案例研究和试验法。本文为更全面地获取影响敏捷开发项目进度的风险因素，访谈了多位在敏捷开发中担当管理、开发的经验人士，他们参与敏捷开发的角色不同，责任与思考维度不同，提供的影响因素丰富了访谈内容。同时，从风险因子收集和风险因子影响程度两个维度设计调查问卷，可详见附录 A，并通过有针对性的渠道发放问卷。

(3) 头脑风暴法。该方法是一种集思广益的创新技术，旨在通过集体讨论和自由发散的方式获取各种创意和解决问题的思路。这一方法最早由亚历克斯·奥斯伯恩于 20 世纪 50 年代引入。该方法是组织召集特定人群以会议讨论的形式开展，参与者被鼓励放开思维，任意发散想法，不加限制地提出各种各样的观点和解决方案，避免初始阶段的批判和评价。一般会议安排一位主持人（或组织者）负责引导头脑风暴的进行，激发创意，并确保参与者的平等机会。在头脑风暴会议结束后，主持人整理和归纳所有的想法和解决方案，并进一步进行评估和筛选。这种技术不仅可以用于解决问题和制定新策略，还可以促进团队合作和交流，增强团队的创新能力。

2.3.2 风险评估

风险评估是指通过系统性的方法和过程，对潜在风险进行全面分析、评估和量化的过程。它的目的是识别和理解风险，以便为风险管理和决策制定提供有关风险概率、影响和优先级的信息^[33]。

(1) 风险矩阵法(简称 LS)，一种常用的风险评估和管理方法，也被称为风险矩阵分析、风险评价矩阵等。它通过将风险的可能性与影响程度结合在一个矩阵中，帮助识别和评估各种潜在风险，并制定相应的风险管理略。在风险矩阵法中，通常使用一个二维矩阵来表示风险的可能性和影响程度，将不同风险按照可能性和影响程度进行分类，并将其映射到不同的风险级别或优先级。常见的风险可能性和影响程度的划分可以使用数字标度（如 1-5、1-10）或是描述性词语（如低、中、高）。风险矩阵

分析法 $R = L \times S$ ，其中 R 是危险性(也称风险度)，事故发生的可能性与事件后果的结合， L 是事故发生的可能性； S 是事故后果严重性； R 值越大，说明该系统危险性大、风险大。

(2) 风险程度分析法(MES)，是一种半定量的风险评价方法，它是对作业条件危险性分析法(LEC)的改进。其目的是确定每个风险事件的影响程度和发生概率，并据此确定风险的优先级。通过对风险的定量分析和评估，可以帮助组织有效地管理和应对各种风险。该方法也有一定局限性，如可能受主观判断和估计的影响，对不确定性的处理较为简化等。因此，在应用风险程度分析法时，需要结合其他风险评估方法和实践经验，以提高评估的准确性和全面性。

(3) 工作危害分析法(Job Hazard Analysis, JHA)是一种系统性的方法，用于识别和评估工作场所中可能存在的危险和风险。通过工作危害分析，可以帮助组织识别潜在的工作危害，评估其潜在风险，并采取措施控制和减轻这些危害，从而确保员工的安全健康和整体工作环境的安全性。实施步骤上通常包括确定工作活动、识别潜在危害、评估风险和实施和监督，通过工作危害分析法，组织可以全面了解工作场所的潜在危害，及时识别和控制风险，保障员工的安全和健康。这有助于促进良好的工作环境，提高工作效率，减少事故和伤害的发生。工作危害分析法是许多领域中重要的工具，如制造业、建筑业、医疗卫生领域等。定期回顾和更新工作危害分析结果，以适应变化的工作环境和新的危害因素。

本文多方面对比上述方法，选定风险矩阵法对识别出的风险因子进行风险评估。

2.4 贝叶斯网络基础

2.4.1 贝叶斯网络概念

贝叶斯网络(Bayesian Network)是一种概率图模型，用于表示变量之间的概率依赖关系，是一种以图形模式描述随机变量之间依赖关系的理论框架，它将图论和概率论相结合，使用有向无环图(DAG)来描述变量之间的关系，每个节点表示一个变量，边表示变量之间的依赖关系。贝叶斯网络旨在模拟和分析复杂系统中的不确定性和概率关系。结点代表随机变量，结点间的有向边代表了结点间的互相关系(由父结点指向其子结点)，用条件概率进行表达关系强度，没有父结点的用先验概率进行信息表达^[34]。

2.4.2 贝叶斯网络原理

贝叶斯网络(Bayesian Network)在 20 世纪 80 年代中后期被引入人工智能领域,由研究者 Judea Pearl 提出并开展了相关工作,以贝叶斯公式和全概率公式作为其理论基础,由经典概率论中的贝叶斯公式发展而来。

1. 条件概率公式,若 (Ω, F, P) 是一个概率空间, $B \in F$, 若 $P(B) > 0$, 则对于任意的 A 属于 F , 称

$$P(A | B) = \frac{P(AB)}{P(B)} \quad (2.1)$$

2. 全概率公式, 设 S 是试验 E 的样本空间, B_1, B_2, \dots, B_n 为 E 的一组事件, 如果

$$B_i \cap B_j = \Phi, i \neq j, i, j = 1, 2, \dots, n \quad (2.2)$$

$$B_1 \cup B_2 \cup \dots \cup B_n = S \quad (2.3)$$

则称 B_1, B_2, \dots, B_n 为样本空间的一个划分。

假设试验 E 的样本空间为 S , A 为试验 E 的一个事件, B_1, B_2, \dots, B_n 为样本空间 S 的一个划分, 且 $P(B_i) > 0 (i=1, 2, \dots, n)$, 则

$$P(A) = P(A|B_1)P(B_1) + P(A|B_2)P(B_2) + \dots + P(A|B_n)P(B_n) \quad (2.4)$$

就称为全概率公式。

3. 假设试验 B 的样本空间为 S , 试验 E 的一个事件为 A , 样本空间 S 的一个划分为 B_1, B_2, B_n , 且 $P(A) > 0, P(B_i) > 0, (i=1, 2, \dots, n)$, 则

$$P(B_i|A) = \frac{P(A|B_i) P(B_i)}{\sum_{j=1}^n P(A|B_j) P(B_j)} \text{ 称为贝叶斯公式。}$$

4. 概率推理: 用给定变量的概率信息来计算其他变量的概率信息的过程就称为概率推理, 给定变量集合 E 是集合 V 的子集, E 中变量的取值用 e 表示, 即 $E=e$, 若计算条件概率 $P(V=v_i|E=e)$, 即给定证据子集 E 求变量集合 V_i 取值为 v_i 的概率的过程就称为概率推理^[35]。

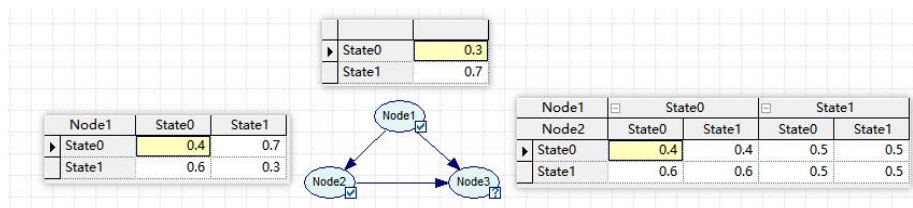


图 2-3 一个简单贝叶斯网络示意图

2.4.3 贝叶斯网络建模

通常,确定贝叶斯网络的结构和参数有以下三种方式:一是由领域专家根据经验知识确定贝叶斯网络的结构和参数;二是由领域专家根据经验知识现确定贝叶斯网络的结构,进而通过机器学习算法从训练数据中学习得到网络参数;三是贝叶斯网络的结构和参数都是通过机器学习算法从大量训练数据中学习得到。本文后续建模将采用第二种方式,集结 F 公司高年资项目经理和技术负责人,结合风险问卷结果确定贝叶斯网络结构和参数。

2.4.4 贝叶斯网络应用

(1) 贝叶斯网络预测

贝叶斯网络是一种概率推理技术,给定部分相关变量的观察数据,利用贝叶斯网络的概率推理算法,可以计算出目标变量的后验概率分布和预测结果。可以应用条件概率、贝叶斯定理、贝叶斯网络的结构和参数等进行推理。贝叶斯网络的预测是指从起因推测一个结果的推理,也称为由顶向下的推理。目的是由原因推导出结果。已知一定的原因(证据),利用贝叶斯网络的推理计算,求出由原因导致的结果发生的概率^[36]。

(2) 贝叶斯网络诊断

贝叶斯网络的诊断是指从结果推测一个起因的推理,也称为由底至上的推理。目的是在已知结果情况下,向上找出导致该结果的原因。已知发生了某些结果,根据贝叶斯网络推理计算造成该结果发生的原因和发生的概率。诊断作用多用于病理诊断、故障诊断中,目的是找到疾病、故障发生的原因^[36]。

(3) 贝叶斯网络学习

贝叶斯网络学习是指从数据中自动推断或构建贝叶斯网络的过程,这个过程可以分为两个主要部分:结构学习和参数学习。本质上,这是一种数据驱动的方法,用于校正和完善初始的信仰结构。贝叶斯网络的持续学习特性意味着,历史学习的成果——即先验网络,将作为下一次学习的基础。这种迭代过程允许用户在每次学习前对先验网络进行微调,以确保每次更新的网络更好地反映数据所蕴含的深层信息。通过这种方式,贝叶斯网络不断适应新的数据和信息,从而提高其对现实世界变化的敏感度和应对能力。

2.5 本章小结

本章主要收集并阐述了支持本文研究的相关基础理论知识,首先总结了敏捷开发的基本定义、特点及其应用过程,然后就项目常用的进度管理内容和方法进行汇总学习,紧接着研究风险的识别和评估技术,并比较选择了适应本文研究的方法,最后研究了贝叶斯网络相关理论知识,为后续进一步研究提供理论和方法准备。

第 3 章 F 公司敏捷开发软件项目管理现状及存在问题

风险管理作为项目管理工作中的重要一环,通过对 F 公司当前敏捷开发软件项目管理的现状从企业、团队、成员的维度进行逐一梳理,并将问题罗列,对其中的问题进行归类分析,只有对相关问题足够重视,才能使软件项目管理水平不断提升。本章共分成两节:第一节介绍 F 公司软件项目的基本情况,包括 F 公司的组织结构、项目总体情况和进度管理现状;第二节描述 F 公司软件项目进度管理存在的问题。通过以上全面的分析总结,找出 F 公司敏捷开发项目进度管理存在的主要问题。

3.1 软件项目概况

3.1.1 F 公司概况

F 公司成立于 2014 年,现在是一家专注于大数据高科技市场的高新技术企业。公司面向政府、央企、外企等各类用户提供应用产品和技术服务。创建之初为保障企业生存选定对日软件外包服务作为主营业务,国内软件开发服务仅为试水阶段。

2015 年,李克强总理在政府工作报告中首次提出“互联网+”行动计划。为顺应“互联网+传统行业”的新业态潮流,F 公司开始全面调整高层战略和业务配比,由单一的对日外包业务改为向国际、国内双向业务线发展。同时,公司资源大比例倾斜国内项目,并在国内项目中推广以需求进化为核心,采用迭代、循序渐进的敏捷开发方法。

目前 F 公司拥有正式员工约 60 人,其中有海外经历的人才 10 余人,美国 PMI 项目管理协会认证 PMP 资格的管理人才 5 名。同时,合作的协力公司员工 10~30 人,整体项目参与人员近百人。资质方面,F 公司已取得 ISO27001、ISO9001、CMMI-3 认证,在项目质量、安全、管理能力方面有相对完整的知识体系和管理标准。自 2022 年开始,F 公司开始谋划组织架构改革,由“一套领导团队统管 n 组开发团队”模式转向梯队建设,采用事业部制管理。现有三个事业部细化对接数智基建、智慧社区、铁路教育等领域的客户、软件开发项目。具体组织架构如图 3-1 所示。

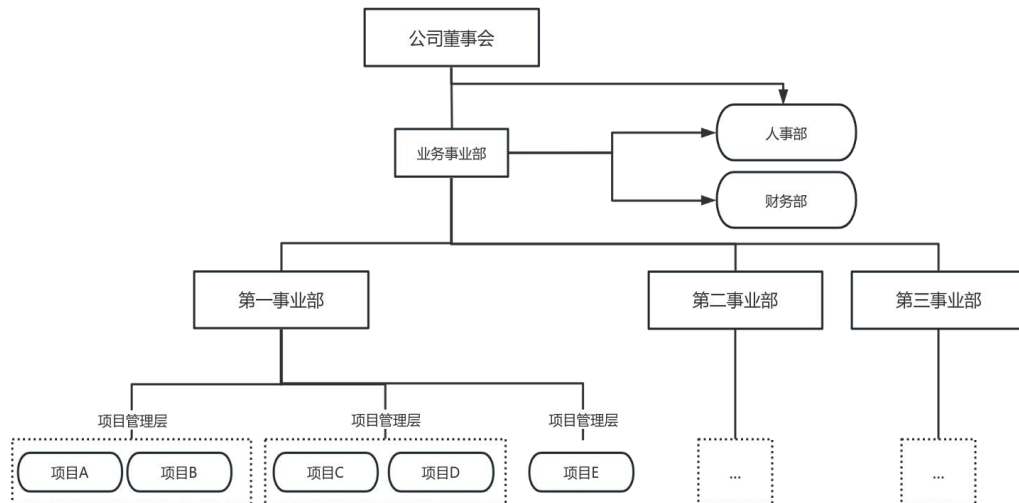


图 3-1 F 公司组织架构示意图

3.1.2 F 公司项目总体概况

F 公司参与的软件开发项目涵盖基建、电商、教育、智慧社区、医疗等多个领域。其中，基建项目背靠国家、地方大型建筑、水利、高速等建设企业开展，参与工程项目管理应用软件研发与运维，为两化融合落地实践提供技术支持；智慧社区则是与项目所在地政府深度合作，深入具体街道社区，在一般社区功能基础上结合社区特性实现客户化需求。铁路教育则是结合铁路领域职工流动特点，以方便学习途径，检验学习成果为目的开展客户化定制需求。服务客户大多是央企、国企、政府等单位，相较于其他客户，他们更加重视项目涉及法律法规、组织化流程及信息安全事宜，同样对于产品交付时间有严格要求。

一般软件项目由确定业务后开始介入，开发周期 3 个月达成初期建设，进而开展项目试点应用。多数试点应用期设定为一个月，期间完成功能验证的同时，收集优化建议和新的功能需求形成进一步的需求清单，作为优化项或下一期新需求的输入。试点应用后根据试点结果和优化情况考虑择机产品推广，推广范围是服务客户下属的公司、分公司的项目运行部门。推广期间也是不断收集改进意见与功能新需求，在一轮又一轮的迭代中不断把产品完善，并完成客户个性化需求。

F 公司软件开发项目管理中，项目经理作为第一负责人，负责项目管理对内、对外两大方向工作。对外，项目经理作为窗口负责协调客户、开发团队以及利益相关者之间的沟通，确保各方信息同步畅通，为高效协作打通通道。与客户密切合作，梳理需求形成功能清单等成果文档。对内，需要积极推动敏捷实践在团队中的落地，包括

计划编排、每日站会、结对编程、代码审查等，以提高开发团队的效率和质量。同时，需要关注到开发团队的整体氛围和个人情绪，为保障团队的工作效率和满意度，创造一个积极、和谐、开放的工作环境。关注团队的运行状态，收集反馈的基础上持续调整优化。

3.1.3 F 公司项目进度管理现状

为合理规划开发周期和资源投入，保证过程中向客户展示阶段成果，最终向客户交付产品时间节点，F 公司采用的项目进度管理手段包括：里程碑划定、WBS、每日例会、周例会和阶段总结。项目流程阶段划分如图 3-2 所示。启动阶段通过 F 公司统一的功能点估算法进行初始项目估算。开发验证阶段主要利用 WBS 日程表进行管理工作，利用 F 公司通用 WBS 日程管理模板结合项目实际情况进行任务拆分，细化开发阶段每个人每一天的工作。要点包括模块名称、子模块名称、任务描述、负责人、计划开始日期、计划结束日期、计划投入工时（H）、实际开始日期、实际结束日期、每日投入工时（H）、进度（%），可以清晰掌握项目组成员工作推进和时间投入情况。里程碑划分则是从两个维度出发，一是由项目经理与甲方项目负责人商定项目的重要节点作为里程碑节点，并制定出里程碑节点的目标。二是在此基础上项目经理会在项目组内制定出内部里程碑节点，组内节点相对外部会有所提前，提前量作为冗余时间预留给风险发生时的应对。例会和阶段总结会对成本投入、进度推进起到监督作用。

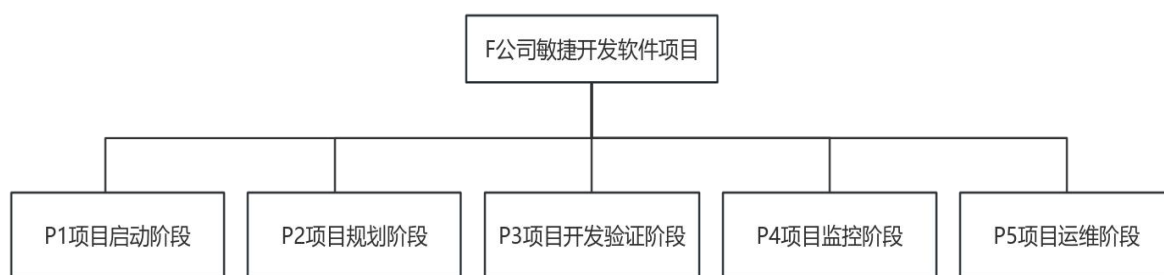


图 3-2 F 公司软件项目阶段分解图

3.2 软件项目进度管理存在问题及成因分析

3.2.1 过多的需求蔓延

需求变更是软件项目中进行功能优化、调整最终呈现效果的常见方法。但是，已

经确认和定义的需求不断增加、修改或变更可能导致项目范围的不确定性和不断扩大,使得开发团队难以按时交付符合要求的软件产品。尽管敏捷开发对于需求变更有一定的适应性,但过多的需求变更和蔓延将导致人力、物力投入到新需求分析、设计、开发、测试等环节,并且变更时点所处阶段越靠后,影响的范围越广泛,大大影响计划的推进实施。为了保证原本计划时间,项目组常会考虑需要增加人员的投入,实际中途追加人力往往伴随着培训和沟通问题,最终不但增加了项目成本,极有可能导致更大的进度风险。例如F公司A项目客户指定了低代码开发平台,项目组人员都没有相关经验,前期参与培训和使用练习投入大量成本。功能进入开发后期时,产品经理对原本的统计需求提出变更,并因此影响到统计数据来源的相关功能。考虑到后续安排能够按照原计划推进,新调整一名前端和一名后端人员支援。人员补充后,实际大量时间都投入到了低代码平台的学习中和业务理解,并且学习过程和实践过程中的问题需要牵扯原项目组人员进行解答,没能按照预想快速推进需求变更的开发。没有人员补充的情况下,只能通过现有团队成员加班来挽回。一方面加班给团队人员造成一定负面情绪,另一方面加班加点情况下给开发质量带来难以预期的后果,进一步加剧进度滞后。

F公司软件开发项目主要有两种服务模式:承包和人力派遣。无论哪种服务模式,客户在初期阶段对产品功能和非功能的需求表达往往不够明确,对同一事物在不同阶段的定义不够清晰,伴随项目的深入而细致化、具体化,极易出现大范围、大数量的需求变更。针对承包项目,客户只重视呈现结果而忽视开发团队的过程投入;大型企业客户由于组织架构复杂、上下层级分明,领导视察、观摩场合提出的意见建议往往第一时间转变为重要且紧急需求,造成原有计划安排被打乱;对于初次合作的客户通常需要一定时间磨合,对同一业务存在理解偏差,从而引发开发过程中需求更改和调整;还有部分偶发情况导致的需求蔓延,例如新的法律法规的出台、市场竞争和新结束解决方案等。而人力派遣项目客户为降本增效,尽可能多的完成开发内容去满足最终客户,对需求范围缺乏边界意识和优先度排序。客户与开发团队间没有建立合理的需求变更流程和规则也是导致需求蔓延的重要原因。例如F公司的J客户提供的需求内容与最终客户存在差距,导致开发团队交付的阶段成果物在演示后发生全部推翻的情况,对于客户提出的修改意见照单全收没有反驳和规则约束,导致开发团队接收到大规模的需求修改,虽然开发团队全力以赴对应仍然无法如期交付客户满意的产品。

3.2.2 沟通问题

有效的沟通是项目过程中的重要环节，对于敏捷开发的软件项目尤为重要。沟通协助项目团队更好地了解彼此的工作职责，及时解决问题和冲突，提高工作透明度、效率和团队协作，保障项目按照既定计划正常推进。F 公司敏捷开发软件项目通常包含两个方向的沟通，开发团队内部的沟通和团队向外的沟通。存在的问题具体如下：

（1）对内沟通意识不足。开发团队的内部沟通应遵循及时、明确、完整、双向和多元化的原则，这就要求团队成员共享、传递信息能够在合适的时间内完成；注意使用明确的描述和术语确保信息的一致性，避免歧义；沟通内容需要全面，确保涵盖到所有相关信息；内部沟通不能限制于单向的信息传达，团队成员接收的同时应提供反馈和建议，以共同参与项目的决策和改进过程；沟通内容要考虑到人员职责、背景的多样性，采用合适的沟通风格和方式。通过访谈和日常观察发现，F 公司敏捷开发团队对沟通重要性的认识不足，在及时、双向、多元化方面均存在明显问题。沟通时效方面，很多问题发生的当下没有及时向上报告或同团队人员沟通，而是等到第二日站立会议时再提出；沟通反馈方面，团队内部对于开发过程中共通问题总结形成“横展开一览表”，特殊情况未按照横展开处理时开发人员没有反馈，导致相关人员依旧参考“横展开一览表”进行后续处理，对接时才发现前后不匹配。多元化方面，因为 F 公司当前敏捷开发软件项目多是采用前后端分离架构，前端、后端开发人员实现过程未考虑对方因素，导致联调联试发现无法对接或缺少逻辑；开发人员接收到需求变更时，仅对应个人职责范围的内容而未沟通相关前端、后端、测试人员，导致功能一致性发生错误的情况也时有发生。

（2）对外沟通不足。对外沟通涉及与项目相关的利益相关者，例如客户、用户、合作伙伴进行沟通和交流。F 公司敏捷开发软件项目中，项目经理作为协调对外沟通的桥梁，其他人辅助在技术、实现方式等方面沟通。例如产品开发过程中，软件开发团队没有与客户充分沟通需求，而是基于自己的理解进行开发，导致最终交付的产品与客户的期望不符，需要花费更多的时间重新进行修改，最终导致项目延期。

3.2.3 人员问题

（1）人员经验不足。人员经验不足情况主要体现在两方面：管理经验不足和开发经验不足。F 公司软件开发项目的第一负责人是项目经理，团队内还会设置一名到多名主管，以便分担项目经理工作。由于公司发展历史原因，项目层管理人员主要从

表现良好、有管理方向发展意向的开发人员中选拔晋升，不一定经过专门的管理学习与培训。提拔后的管理人员在对内、对外的工作过程中依然采用一个开发人员视角看待问题、解决问题，缺乏管理理论和实践经验支持。例如当日程进度延期发生时，对影响范围、是否需要纠正缺乏判断，从而没有采取合理的应对手段，往往会导致更大范围、更大程度的延期风险。F公司整体规模不大，一般将工作经验2年内的员工视为新人，团队组成上来看新人比重较大，每个项目组新人占比一半或以上。这些新人工作年限短，在风险意识、风险识别方面没有经验。接收到一个开发任务急于实现功能，而缺乏行动前的整体规划，例如对实现方式的思考，尽管能够实现功能要求但忽略了性能要求，导致性能测试不过关返工调整实现方式。

(2) 人员技术能力不足。F公司开发团队由于新人比例高，在面对复杂开发时常显被动，任务分配时会考虑新人的生产性，在进度安排上富余1-2天工作时间。实际新人在遇到技术难题时，很难像经验者一样，厘清思路寻求解决方案。大多是通过网络搜索解决办法，逐一尝试。但一个技术问题往往伴随着性能、维护成本等多维度的考量，只是解决当下的一、两行代码并不是最终的方案。新人容易只解决眼前问题，后续再遇到新的问题返回前步骤重新解决，并且容易陷入当前的实现方式不能自拔，导致项目进度缓慢。再如项目前期设计人员一般参考现有市场产品进行界面设计，缺乏结合需求的设计思维，导致呈现效果与用户习惯、应用场景不匹配，易用性差。以上情况均需要为纠正质量问题而投入更多时间。

3.3 本章小结

本章主要介绍F公司的项目基本情况以及进度管理现状，并根据笔者观察从需求、沟通、人员方面梳理当前进度管理中存在的问题。

第 4 章 F 公司敏捷开发软件项目的进度风险识别

本章节风险识别的最终结果是要获得一个影响项目进度风险的风险要素列表，并且使识别出来的风险要素尽量全面无遗漏。本章结合附录 A、B 访谈的统计结果，详细分析 F 公司进度风险因素情况。

4.1 风险初识别

4.1.1 进度风险因子识别过程

基于前章节 2.3.1 几种常见风险因素识别方法的实施效率及优缺点，充分考虑本次研究内容及研究目的，决定采用多种研究方法结合完成风险因素识别过程，如图 4-1 所示。首先，通过查阅历史文献，参考领域内其他学者在敏捷开发、软件项目风险管理相关研究成果，提取典型的进度风险因素作为下一步的输入。其次，与软件行业从业者（其中包含 F 公司事业部门人员）进行个人访谈，进一步收集可能导致进度延迟的风险因子，与前记历史文献研究的风险因子整理汇总形成初步风险清单。最后，在初步风险清单基础上，组织 F 公司项目经理、技术骨干、项目管理层进行 1-2 次头脑风暴，借助团队成员的项目经验补充、过滤、拣选、整理敏捷开发软件项目进度风险因子形成最终的进度风险因素清单，完成进度风险因子的识别。

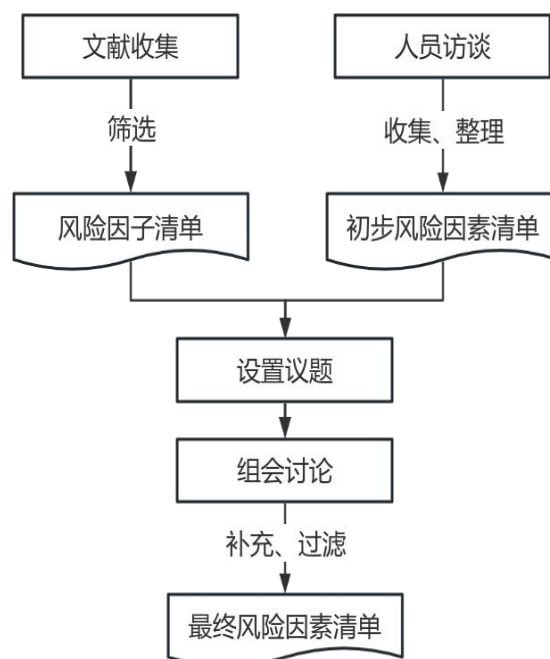


图 4-1 风险因子识别过程

在开展敏捷开发项目进度风险管理研究时,有必要考虑项目完整生命周期内各阶段遇到的各种风险因素。本文首先运用历史文献分析法结合访谈识别出敏捷开发软件项目中可能存在的进度风险,此阶段收集应尽可能覆盖全面。之后通过头脑风暴法进行有效筛选和合理分类,形成进度风险因素清单。通过引入F公司人员的参与使得风险因素清单更加贴合F公司敏捷开发软件项目的实际情况。

4.1.2 进度风险因子识别结果

本节将结合前文所述风险识别方法和F公司敏捷开发软件项目周期将进度风险因子划分进项目阶段。对敏捷开发项目进度风险因子的识别过程中,参考了以往研究经验与结论。并在此基础上,向软件项目管理者定向发放附录A访谈,汇总梳理访谈结果补充风险因子。问卷在A、B两个项目现场发放,A、B现场均为采用敏捷开发或“敏捷+X”混合开发的软件项目现场。A项目现场为F公司自有项目,向现场项目项目经理、项目组长发放问卷;B项目现场为F公司的客户现场,包含F公司在内约有软件服务供应商企业十余家,访谈范围是项目经理、业务经理和各服务供应商驻场负责人。参与访谈人员均为项目或团队管理人员,项目经历丰富,能够给研究结果提供有效支持。

为了接下来研究的风险因子更加贴合F公司的项目情况,从F公司挑选项目经验丰富且担当重要职务如项目经理、负责人等成员,并且工作结构上覆盖相对全面的6人组成本次研究专家组,在研究过程中参与评价、打分、头脑风暴环节给出个人专业意见。具体成员构成参考表4-1,此处以序号代替姓名进行标识。

表4-1 打分组成员简介表

序号	职务	工作年限	主职工作内容
1	产品经理	8	获取客户需求,并进一步分析需求分析,设计原型图
2	设计负责人	6	负责项目涉及的产品UI、图案、logo、占位图等的设计工作
3	项目经理	13	负责项目组内部决策、管理,协调各方资源,规划项目人员、计划,与客户沟通需求等工作
4	后端资深开发 &项目经理	10	解决开发团队遇到的后端技术难题,承担难度高的后端开发工作,配合架构师搭建开发、测试、运行环境等
5	全栈开发&驻 场负责人	10	解决开发团队遇到的前、后端技术难题,承担难度高的前端开发工作;辅助客户协调现场工作
6	测试负责人	8	分析需求并依据项目计划指定测试计划、编写测试用例、执行测试,出具测试报告。

本次访谈收集风险因子,并向受访者了解风险因子发生阶段。经整理并匹配项目

阶段后合并文献收集得出表 4-2 中 59 项风险因子。

表 4-2 进度风险因素初版

序号	项目阶段	风险因子描述	风险因子来源
1	启动阶段	前期规划不合理，具体可行性不落地	徐梓华 ^[37]
2		项目人手不够	访谈
3		项目资金不够	访谈
4		资源不到位	吕景丽 ^[38]
5		项目所需的人力、物力和财力资源是否满足需求	访谈
6		业务是否明确	访谈
7		商定交付时间后，研发时间资源紧缺，导致不能如期交付	访谈
8	规划阶段	范围控制，很难在最初明确完整的范围	访谈
9		计划拆分程度和具体可行性是否合理	访谈
10		需求调研不充分，后期返工	访谈
11		把控风险不足，前期调查不充分	访谈
12		技术架构选型不合理	徐梓华 ^[37]
13		风险评估和规划不足	张涛 ^[39]
14		缺乏合理的风险应对方案	张涛 ^[39]
15		项目分工不明确	徐梓华 ^[37]
16		项目估算不准确	访谈
17		业务复杂度	访谈
18		风险识别不完善	访谈
19		人员体制不能支撑项目按计划推进	访谈
20		过于理想化，团队存在短板人员较多	访谈
21	开发验证阶段/迭代阶段	无法从一开始就充分设计，导致后续功能迭代时可能出现的问题	访谈
22		项目进行中是否面临多次需求变更，是否与客户及时沟通，避免对原计划造成影响。	访谈
23		需求变更频率	访谈
24		对需求没有正确认识	访谈
25		story 拆解太粗	访谈

表 4-2 进度风险因素初版（续表）

26	开发验证阶段/迭代阶段	需求描述模棱两可不明确	访谈
27		需求的复杂程度高	访谈
28		需求频繁变更和范围蔓延	访谈
29		需求理解不到位	访谈
30		人员排期是否合适	访谈
31		项目的任务是否分解明确，并且能够按照合理的顺序进行安排	访谈
32		不确定的系统架构和三方中间件的使用，项目开始时，需做好必要的详细设计	访谈
33		进度控制，很难保证对每个人的任务进度把控	周华等 ^[40]
34		质量（性能+稳定性）控制，最容易在最后失控	访谈
35		功能的难易程度	刘靖 ^[41]
36		解决问题的时间	访谈
37		开发人员的业务理解能力及编码能力高低	访谈
38		开发技术上存在难点	访谈
39		存在不确定的三方中间件的使用	访谈
40		个人能力	访谈
41		开发人员技术能力不足	访谈
42		开发人员对业务的理解程度	吕景丽 ^[38]
43		开发技术上是否有难点	访谈
44		测试是否全面	访谈
45		人员变动	访谈
46		评审会议效果不达标	访谈
47	监控阶段	部署流程不完善	访谈
48		不适当的优先级和资源分配	访谈
49		定期交付过程中，客户不定时修改或新增产品功能，需依据紧急程度对工作内容进行调整	访谈
50		没有及时进行阶段性的功能演示，缺乏对风险的预知	访谈
51		缺乏监测和评估机制	张涛 ^[39]
52	其他	无效的沟通	访谈

表 4-2 进度风险因素初版（续表）

53	其他	跨部门合作	访谈
54		跨团队的沟通	访谈
55		团队内部不和谐	访谈
56		向上传递信息，与客户问题交涉时间长，拖延处理进度	访谈
57		其他的不可控的第三方因素，如金融危机等。	访谈
58		没有预留出解决突发事件的人天	访谈
59		不可控因素，对困难预计准备	访谈

针对表 4-2 的 59 项风险因素，采用头脑风暴会议对因素描述梳理归类，过于模糊的描述进行细化说明，过于细化的风险描述汇总，类似的风险因素归纳，还有 F 公司现有管理策略下考虑不发生的情况删除。例如序号 2 至 5 都提到项目投入资源会对进度产生影响，考虑将这几项合并描述为“资源配置不到位（人力、物力、财力）”；而 52 至 55 都在强调沟通都在提示沟通的重要性，综合描述为“项目内沟通不足（团队内部、团队之间）”；22、23、28 虽然描述存在一定差异实际都是想表达需求范围和变动频率对项目进度的影响较大，结合以往项目经验也有同样结论，故合并成一个风险因素；序号 57 提到不可控的第三方因素，头脑风暴过程中对其有所补充，比如政府政策出台、法律法规实施等；序号 58 提到预留应对突发时间的人天，在 F 公司实际项目中通过内外部计划差异保证有一定量缓冲时间，故此项删除。

通过对 59 项风险因素梳理、合并、归纳、细化后，得到本次研究的进度风险因素 25 项，如下表 4-3 所示：

表 4-3 进度风险因素清单

序号	项目阶段		风险因子
1	启动阶段	/	前期规划不合理，具体可行性不落地
2			资源配置不到位（人力、物力、财力）
3			技术架构选型不合理
4	规划阶段	/	风险评估和规划不足
5			缺乏合理的风险应对方案
6			项目分工不明确
7			项目估算不准确
8			人员体制不合理
9	开发验证阶段	迭代前	Story 拆解不够充分
10			需求描述模棱两可不明确

4-3 进度风险因素清单（续表）

11	开发验证阶段	迭代前	需求的复杂程度高
12			需求频繁变更和范围蔓延
13			需求理解不到位
14		迭代中	开发技术上存在难点
15			存在不确定的三方中间件的使用
16			人员不稳定，频繁变动
17			开发人员技术能力不足
18			评审会议效果不达标
19			测试不够全面
20		迭代后	部署流程不完善
21	监控阶段	/	缺乏有效的问题识别和收集机制
22			不适当的优先级和资源分配
23			缺乏监测和评估机制
24	其他	/	项目内沟通不足（团队内部、团队之间）
25			其他的不可控的第三方因素，如金融危机、国家政策变更等

结合表 4-3 可以发现，25 项进度风险因子中包含软件项目常见的进度风险因子，例如项目分工不明确、需求频繁变更和范围蔓延等，也包含敏捷开发特有的风险因子，例如 Story 拆解不够充分。

4.2 进度风险因素确定与分析

4.2.1 进度风险因子评估标准的制定

在前章节 4.1.2 梳理的进度风险因素清单基础上，为了进一步明确风险因素在 F 公司敏捷开发关键项目中发生概率情况和影响程度，故设计敏捷开发项目进度风险水平评估访谈表^[42]。评分设计采用五级量纲划分，被调查者参考统一的评估标准，对 25 个风险因子逐一进行评分，从进度延迟程度和风险发生概率两个维度给出自己的评估分值。评估标准如表 4-4 所示：

表 4-4 风险因子评估标准

评估分值	0	1	2	3	4
风险发生概率	几乎不发生	极少发生	偶尔发生	经常发生	频繁发生
对项目进度的延迟程度	几乎不延迟	延迟<10%	延迟 10%-20%	延迟 20%-50%	延迟>50%

由表 4-4 知评估分值划分为 5 级，对于进度延迟程度的描述分别对应：几乎不延迟、延迟<10%、延迟 10%-20%、延迟 20%-50%、延迟>50%。发生概率描述分别对

应：几乎不发生、极少发生、偶尔发生、经常发生、频繁发生。

4.2.2 进度风险因子评估分析

本章对汇总的敏捷开发项目进度风险水平评估访谈结果进行梳理分析。本次敏捷开发项目进度风险水平评估访谈共回收问卷结果 89 份，剔除从未参与过敏捷开发项目以及回答明显不合理问卷后有效数量为 83 份，即问卷有效率达 93.3%。

访谈要求被调查人员填写从业年限，用以确认参调者的从业年限分布情况，据此判断统计数据的准确性及可信度。对回收的问卷进行整理统计，得到人员从业年年限分布情况如表 4-5：

表 4-5 被调查人员从业年限分布情况

从业年限	0-5 年	6-10 年	11-15 年	16-20 年	20 年以上
人数	28	38	15	2	0

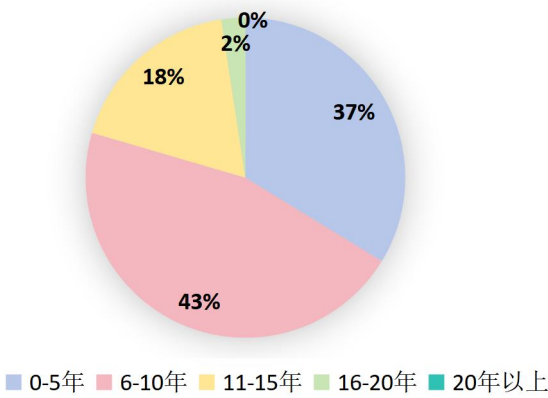


图 4-2 被调查人员从业年限分布情况

由表 4-5 和图 4-2 中可以看出，被调查人员从事软件行业相关工作年限 0-10 年占比 80% 呈现绝对优势，符合软件行业从业者年轻化的现状，而 6-15 年人数占比例 61%，理论上工作能力与工作年限成正比，据此判断本次被调查人员具有较为丰富的项目经验和扎实的专业能力，能够为本次进度风险研究提供有效帮助。

为了进一步建立风险等级矩阵，规划风险等级划分标准，首先需要明确风险等级，并且指定改进建议，在此标准之上优化风险等级矩阵。

表 4-6 风险等级划分标准

风险等级	描述	需要的行动	改进建议
R3	高度风险	必须通过管理、技术、工具等手段进行纠正	需要并制定专门的管理方案予以削减

表 4-6 风险等级划分标准（续表）

R2	中度风险	应当通过管理、技术、工具等手段进行纠正	需要并制定专门的管理方案予以削减
R1	低度风险	持续关注，设立时间节点确认风险范围、程度是否扩大	个案评估。

为进一步考量各风险因子的风险水平，采用风险等级矩阵对各风险因子进行等级划分，统计数据也可作为后续模型参数提供参考。风险矩阵如表 4-6 所示，其中 R1、R2、R3 分别表示风险因子的风险大小低、中、高三档：

表 4-7 风险等级矩阵

延迟 发生 概率 等级	0	R1	R1	R1	R2	R2
	1	R1	R1	R2	R2	R2
	2	R1	R2	R2	R2	R3
	3	R2	R2	R2	R3	R3
	4	R2	R2	R3	R3	R3
风险矩阵		0	1	2	3	4
		延迟影响程度				

在表 4-7 中，以 0-4 表示风险因子对敏捷开发项目延迟造成的影响程度及其发生概率的 5 个等级，结合表 4-4 评估标准对回收到的 83 份问卷进行风险等级转换，即延迟发生概率与延迟影响程度等级对应值确定各风险因子的 R1~R3 等级，统计结果如表 4-8 所示：

表 4-8 风险评估统计数据

序号	风险因子	统计数据		
		R1	R2	R3
1	前期规划不合理，具体可行性不落地	0.20	0.35	0.45
2	资源配置不到位（人力、物力、财力）	0.20	0.60	0.20
3	架构选型不合理	0.36	0.41	0.23
4	story 拆解不够充分	0.37	0.46	0.17
5	项目分工不明确	0.36	0.49	0.15
6	项目估算不准确	0.2	0.46	0.34
7	需求模棱两可不明确	0.23	0.43	0.34

表 4-8 风险评估统计数据（续表）

8	需求的复杂程度高	0.27	0.53	0.20
9	需求频繁变更和范围蔓延	0.11	0.40	0.49
10	需求理解不到位	0.29	0.45	0.26
11	开发技术上存在难点	0.31	0.47	0.22
12	存在不确定的三方中间件的使用	0.39	0.41	0.20
13	人员体制不合理	0.39	0.40	0.21
14	人员不稳定，频繁变动	0.34	0.37	0.29
15	开发人员技术能力不足	0.32	0.46	0.22
16	评审会议效果不达标	0.33	0.45	0.22
17	部署流程不完善	0.43	0.40	0.17
18	测试不够全面	0.30	0.45	0.25
19	缺乏有效的问题识别和收集机制	0.31	0.46	0.23
20	对于反馈采取了不适当的优先级和资源分配	0.34	0.43	0.23
21	缺乏监测和评估机制	0.29	0.43	0.28
22	项目内沟通不足（团队内部、团队之间）	0.35	0.33	0.32
23	风险评估和规划不足	0.28	0.42	0.30
24	没有合理的风险应对方案	0.33	0.36	0.31
25	其他的不可控的第三方因素，如金融危机、国家政策变更等。	0.58	0.27	0.15

由于当前风险因子较多，全部纳入建模势必增加复杂度，并且其中对敏捷开发项目进度影响程度较低的风险因子会影响建模后聚焦关键因子，故采用加权评分法对现有因素进行加权计算并排序，最终保留前 10 项风险因子用于后续的贝叶斯网络构建。专家组对 25 项进度风险因子进行打分，打分计算结果作为当前进度风险因子的权重，再结合访谈风险影响程度结果平均数乘积作为最终评分，具体打分和计算结果如表 4-9 所示：

表 4-9 专家打分结果统计表

序号	风险因子	专家 1	专家 2	专家 3	专家 4	专家 5	专家 6	平均分	权重
----	------	---------	---------	---------	---------	---------	---------	-----	----

表 4-9 专家打分结果统计表（续表）

1	前期规划不合理，具体可行性不落地	7	4	3	4	4	5	4.50	4.5%
2	资源配置不到位（人力、物力、财力）	7	6	5	5	6	5	5.67	5.67%
3	技术架构选型不合理	3	4	3	4	5	4	3.83	3.83%
4	风险评估和规划不足	4	5	5	3	3	3	3.83	3.83%
5	缺乏合理的风险应对方案	3	4	3	4	4	3	3.50	3.5%
6	项目分工不明确	2	4	4	4	4	3	3.67	3.67%
7	项目估算不准确	6	5	6	5	2	5	4.84	4.84%
8	人员体制不合理	5	2	5	3	2	4	3.83	3.83%
9	Story 拆解不够充分	6	5	5	5	6	5	5.33	5.33%
10	需求描述模棱两可不明确	3	3	2	4	4	2	3.33	3.33%
11	需求的复杂程度高	4	3	3	3	4	2	3.33	3.33%
12	需求频繁变更和范围蔓延	8	6	7	5	6	7	6.50	6.5%
13	需求理解不到位	2	4	3	3	4	4	3.33	3.33%
14	开发技术上存在难点	4	4	3	5	4	2	3.67	3.67%
15	存在不确定的三方中间件的使用	3	4	3	3	3	3	3.17	3.17%
16	人员不稳定，频繁变动	4	3	4	3	5	4	4.00	4%
17	开发人员技术能力不足	6	5	6	5	4	5	5.17	5.17%
18	评审会议效果不达标	2	4	4	4	3	5	3.67	3.67%
19	测试不够全面	5	6	4	7	7	6	5.83	5.83%
20	部署流程不完善	2	2	3	2	4	3	2.83	2.83%
21	缺乏有效的问题识别和收集机制	2	2	3	3	3	3	2.67	2.67%
22	不适当的优先级和资源分配	2	2	3	3	3	4	2.83	2.83%
23	缺乏监测和评估机制	3	4	4	5	4	4	2.50	2.5%

表 4-9 专家打分结果统计表（续表）

24	项目内沟通不足（团队内部、团队之间）	5	6	7	5	5	6	5.67	5.67%
25	其他的不可控的第三方因素，如金融危机、国家政策变更等	2	3	2	3	1	3	2.50	2.5%
	合计	100	100	100	100	100	100	100.00	100%

依据权重倒序排列，风险因子影响力较大的 10 项分别是：需求频繁变更和范围蔓延、测试不够全面、项目内沟通不足（团队内部、团队之间）、资源配置不到位（人力、物力、财力）、前期规划不合理，具体可行性不落地、开发人员技术能力不足、Story 拆解不够充分、项目估算不准确、风险评估和规划不足、人员不稳定，频繁变动。结合上表所得各进度风险因子权重，乘以访谈所得影响程度平均数得到加权平均数，倒序排列后结果如表 4-10 所示：

表 4-10 进度风险因子排序表

序号	风险因子	加权平均数	排序	编号
1	需求频繁变更和范围蔓延	15.9	1	F1
2	测试不够全面	11.52	2	F2
3	项目内沟通不足（团队内部、团队之间）	11.48	3	F3
4	资源配置不到位（人力、物力、财力）	10.86	4	F4
5	前期规划不合理，具体可行性不落地	10.73	5	F5
6	开发人员技术能力不足	9.53	6	F6
7	Story 拆解不够充分	8.93	7	F7
8	项目估算不准确	8.46	8	F8
9	风险评估和规划不足	8.12	9	F9
10	人员不稳定，频繁变动	7.66	10	F10
11	缺乏合理的风险应对方案	7.21	11	-
12	需求描述模棱两可不明确	7.14	12	-
13	人员体制不合理	7.01	13	-
14	技术架构选型不合理	6.69	14	-
15	开发技术上存在难点	6.63	15	-
16	需求的复杂程度高	6.54	16	-

表 4-10 进度风险因子排序表（续表）

17	项目分工不明确	6.46	17	-
18	需求理解不到位	6.42	18	-
19	评审会议效果不达标	6.32	19	-
20	存在不确定的三方中间件的使用	5.5	20	-
21	不适当的优先级和资源分配	5.39	21	-
22	缺乏有效的问题识别和收集机制	5.11	22	-
23	缺乏监测和评估机制	4.67	23	-
24	部署流程不完善	4.57	24	-
25	其他的不可控的第三方因素，如金融危机、国家政策变更等	3.37	25	-

依照上表排序结果得到排名前十的风险因子分别是：需求频繁变更和范围蔓延、测试不够全面、项目内沟通不足（团队内部、团队之间）、资源配置不到位（人力、物力、财力）、前期规划不合理，具体可行性不落地、开发人员技术能力不足、Story拆解不够充分、项目估算不准确、风险评估和规划不足、人员不稳定，频繁变动，作为裁剪后进度风险因子，与权重结果对比，风险因素前期规划不合理，具体可行性不落地、风险评估和规划不足对进度影响程度排序上升，Story拆解不够充分、人员不稳定，频繁变动、项目估算不准确的排序则是发生后移。对比章节 3.2 发现日常暴漏的问题与统计数据得出的结果相匹配，一定程度说明筛选有效。

4.3 本章小结

本章主要采用文献分析法、访谈法、头脑风暴法多方法结合对敏捷开发项目进度风险因子进行了识别，并据识别结果设计调查问卷，通过有效的回收问卷结果结合风险矩阵获得各项风险因子的风险评估等级，继而利用加权平均裁剪出对 F 公司敏捷开发项目进度影响较大的 10 项风险因子，用于后续研究使用。

第 5 章 F 公司敏捷开发项目进度风险推理与管理建议

基于第四章进度风险因子的研究成果作为贝叶斯网络模型构建的输入，借助 GeNIe2.0 软件最终建立贝叶斯网络模型，并利用软件的可视化界面及推理能力确定 F 公司敏捷开发软件项目的重大风险因子，进而有针对性的提出改善策略。

5.1 贝叶斯网络风险评价模型构建

构建贝叶斯网络模型主要包括以下步骤：明确问题定义、构建网络结构、定义条件概率模型、推断、评估和优化。在上一节的研究叙述中已经确定建模所需的 10 项敏捷开发软件项目进度风险因子，如表 4-10 的 F1~F10 所示，并完成条件概率数据的收集统计。本章节采用机器学习和专家知识结合的方式确定网络拓扑结构，并在此基础上参考表 4-8 代入风险因子的条件概率，建立敏捷开发软件项目进度风险贝叶斯网络模型^[42]。具体建模流程如图 5-1 所示

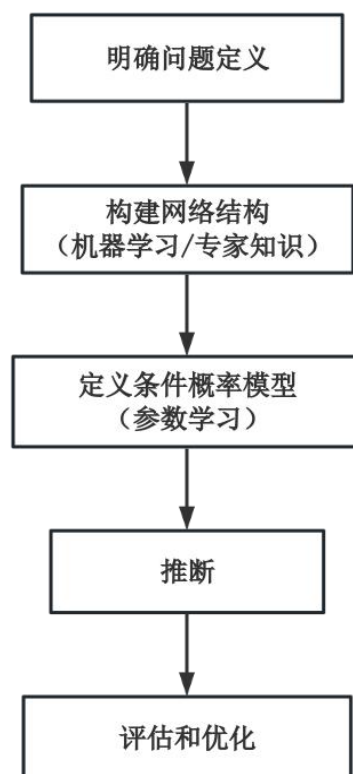


图 5-1 贝叶斯网络模型构建流程

5.1.1 贝叶斯网络结构

构建贝叶斯网络拓扑结构首先要进行结构学习，确定各节点间逻辑关系。本环节

借助 GeNIe2.0 软件，打开训练数据集文件，初始网络结构学习得到图 5-2。因此处显示节点标号，与进度风险因子对应关系参考表 5-1。

表 5-1 编号-进度风险因子对照表

编号	风险因子
F1	需求频繁变更和范围蔓延
F2	测试不够全面
F3	项目内沟通不足（团队内部、团队之间）
F4	资源配置不到位（人力、物力、财力）
F5	前期规划不合理，具体可行性不落地
F6	开发人员技术能力不足
F7	Story 拆解不够充分
F8	项目估算不准确
F9	风险评估和规划不足
F10	人员不稳定，频繁变动

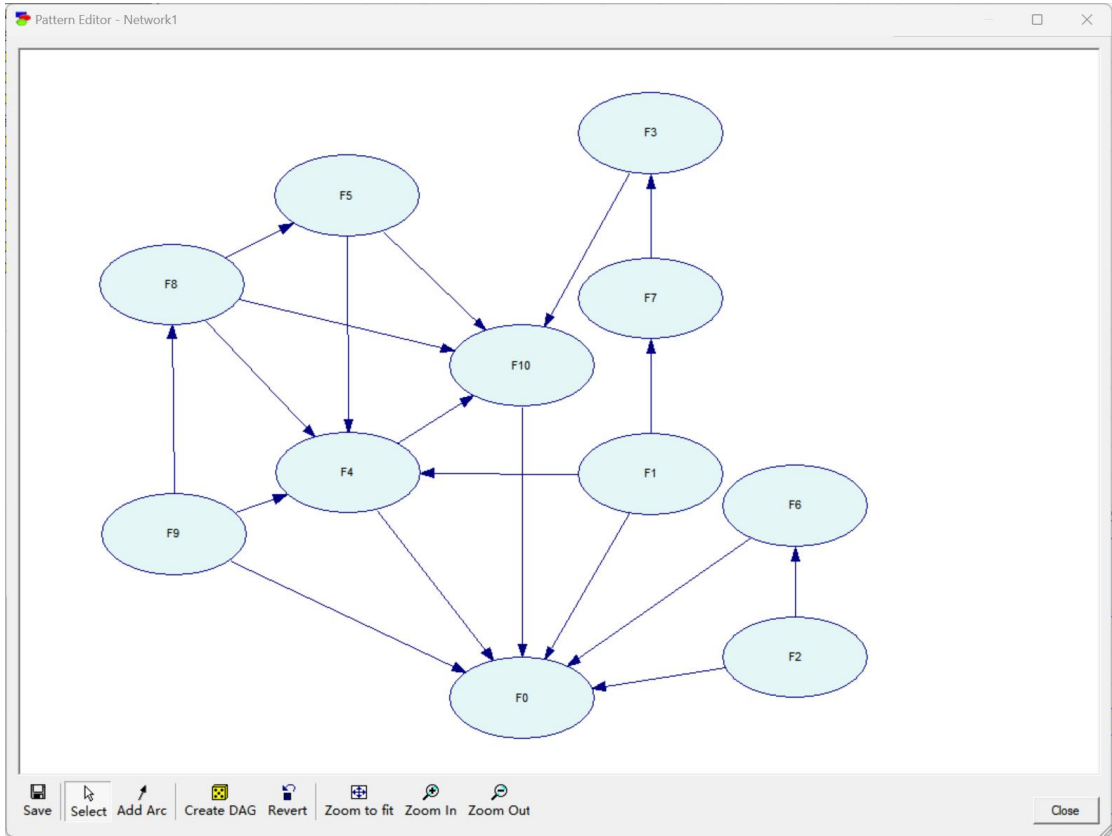


图 5-2 进度风险贝叶斯网络机器学习初始结构

如图可见机器学习成果下的初始网络结构，但其中存在明显不合理的逻辑关系，例如：“风险评估和规划不足”到“资源配置不到位”、“开发人员技术能力不足”到“项目内沟通不足”等不存在逻辑关系，“风险评估和规划不足”到“项目估算不准确”、“项目估算不准确”到“前期规划不合理，具体可行性不落地”存在反向逻辑关系。因此，需要参考专家意见对各节点间关系继续调整优化。

此处引入因果关系分析，对机器学习出的网络结构进行调优，使网络结构更加贴近实际。梳理图 5-2 的网络结构制定因果关系的评估调查表，该调查表由评估标准和评估表构成，评估标准如表 5-2 所示：

表 5-2 评估标准

分值	0	2	3	4
关系	无关系	有一定关系	较强关系	很强关系

依据上表的评估标准，对初始网络结构中存在因果逻辑风险因子两两进行判定，如“开发人员技术能力不足”到“项目内沟通不足”不存在逻辑关系则评估结果判定 0。对专家组评分汇总分析，选取因果关系评估均值大于等于 2 风险因子组合，判定其风险因子之间具有直接因果关系，统计结果如表 5-3 所示：

表 5-3 风险因子因果关系分析

原因	结果	平均估值	排序
需求频繁变更和范围蔓延	资源配置不到位	3.06	1
需求频繁变更和范围蔓延	进度延迟	3	2
项目估算不准确	资源配置不到位	3	2
项目内沟通不足	story 拆解不够充分	2.7	3
前期规划不合理，具体可行性不落地	项目估算不准确	2.7	3
story 拆解不够充分	需求频繁变更和范围蔓延	2.7	3
风险评估和规划不足	进度延迟	2.63	4
前期规划不合理，具体可行性不落地	资源配置不到位	2.52	5
人员不稳定，频繁变动	进度延迟	2.52	5
项目估算不准确	风险评估和规划不足	2.52	5
测试不够全面	进度延迟	2.21	6

表 5-3 风险因子因果关系分析（续表）

资源配置不到位	人员不稳定，频繁变动	2.21	6
风险评估和规划不足	进度延迟	2.21	6
人员不稳定，频繁变动	进度延迟	2.13	7
学校基础设施不足	进度延迟	2	8

根据专家意见修正优化后的贝叶斯网络结构如图 5-3 所示，其中“前期规划不合理”会导致“项目估算不准确”和“资源配置不到位”；“项目内沟通不足”-“story 拆解不够充分”-“需求频繁变更和范围蔓延”则是影响进度的一整条链路；“风险评估和规划不足”、“资源配置不到位”、“人员不稳定”、“测试不够全面”均可直接作用于进度结果导致延迟发生。各节点右下位置显示“?”表示参数还未标定。

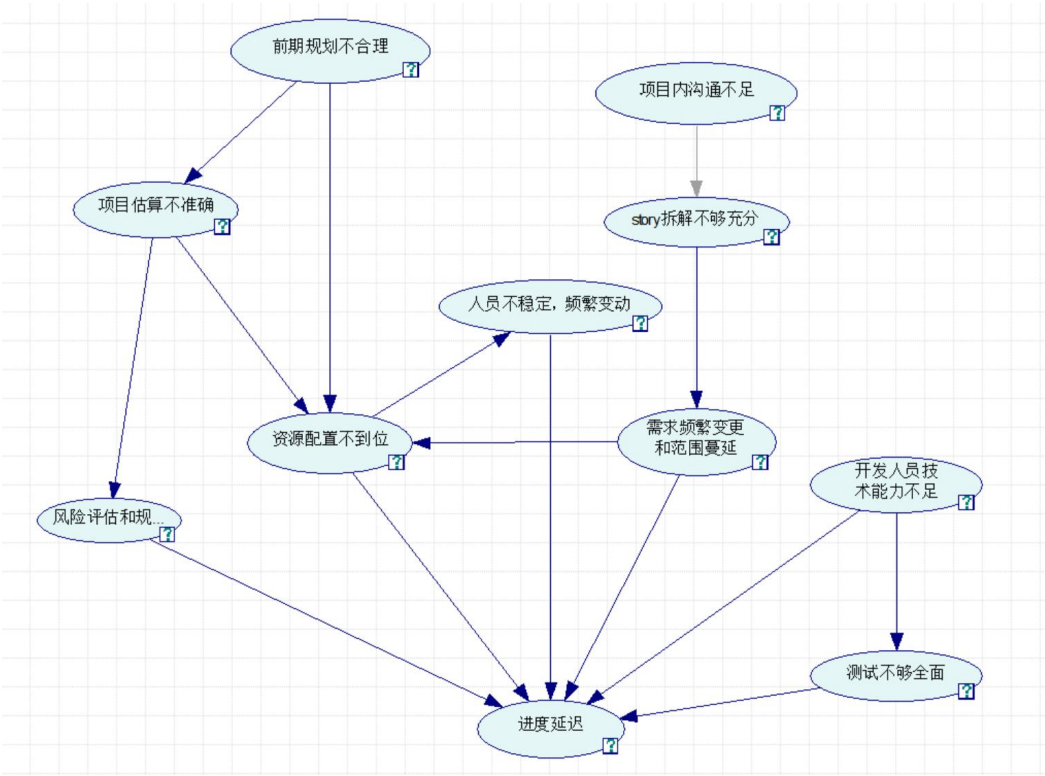


图 5-3 进度风险贝叶斯网络结构

参考既往项目总结数据，删除项目评估不准确与人员不稳定之间的关联。频繁的需求变更可能导致项目团队需要不断调整工作计划和资源分配，项目团队需要额外的人力、物力和时间资源来满足变更需求，而资源配备不及时，进而影响项目的进度，因此增加需求频繁变更和范围蔓延到资源配置不到位间的关联。同时，频繁的需求变更会导致项目范围不断扩大，而项目团队可能无法有效控制和管理变更范围，增加项目进度风险。因此，控制需求变更的范围与频度是控制项目进度的首要条件。在需求

阶段需要与利益相关者进行充分的沟通和理解,确保对项目需求的理解一致。在项目启动时,进行全面的需求分析,明确和记录各个利益相关者的需求和期望,尽可能详细地描述功能和非功能需求。强调需求稳定性的同时,建立一个明确的变更管理流程,要求所有需求变更都必须经过评估、批准和追踪。变更必须合理且有价值,不能随意发生。

项目不合理的前期规划可能造成项目过程中人力、物力等资源配置不到位和项目范围边界模糊不清,从而使项目阶段性成果无法把控,最终导致项目进度延误或目标无法实现。除此之外,人员的技术能力是影响时间规划的一个重要因素,正确评估人员技术能力匹配生产性规划推进节奏是保障进度的有效手段。

最后,风险评估和规划不足、资源配置不到位、需求频繁变更和范围蔓延、开发人员技术能力不足及测试不够全面都直接作用于进度结果,造成延迟结果。因此项目推进的过程中,要充结合项目实际情况,充分考虑各方面风险,实时监控进度推进情况以及减少不必要的风险。

5.1.2 参数学习

上一节阐述了贝叶斯网络结构构建、优化过程,并确定最终网络拓扑结构,本节将结合项目实例进行参数学习辅助专家意见确定各风险因子的条件概率分布。根据根据第四章调查问卷结果数据制作 csv 文件作为参数学习输入文件获取到原始概率数据,然后结合专家意见对概率数据进行调整^[43]。

(1) 节点变量的初始化设置

首先,对个节点变量进行值域编辑,结合前章节 4.2.2 中统计出的 R1~R3 风险值,分别对应风险的 L、M、H 状态,三种状态对应关系分别是 L-Low、M-Medium、H-High。导入项目实例 csv 文件并执行参数学习操作,借助专家意见再对个别节点的条件概率进行单独配置实现最终条件概率的定版。选定一个网络节点,通过“Definition”页签定义三种状态标识符和值字段,节点初始默认 state0 和 state1,可自行增加配置符合研究的标识符名称和概率值。设置界面如图 5-4 所示:

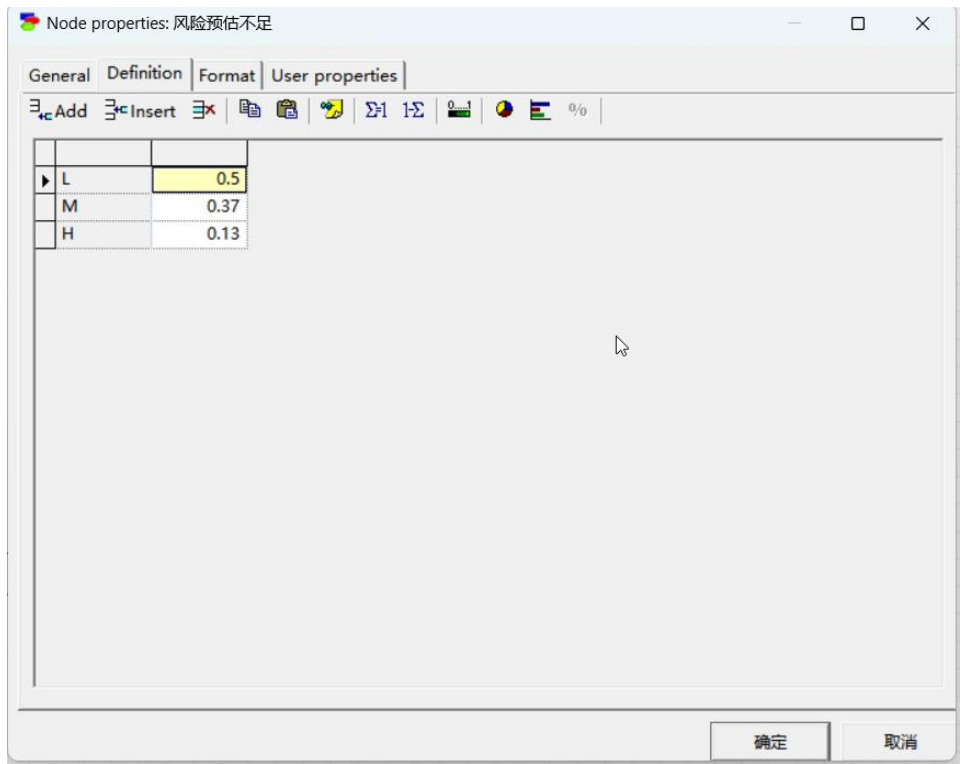


图 5-4 节点概率值设置示意图

(2) 参数学习结果分析

将模型展示模式设置为“Bar Chart”实现每个节点重点信息可视化展示，包括各等级概率以横柱显示，并颜色加以区分。以便参数学习后观察变化。贝叶斯网络的参数学习结果如图 5-5 所示。参考前表 4-6 对风险等级的划分与行动建议，与贝叶斯网络模型中的节点等级进行匹配对应：低度风险 R1 对应 L 状态，是对项目进度影响较小的等级，需要关注但不必采取防控措施；中度风险 R2 对应 M 状态，该等级风险对项目有明显影响，需要采取有效措施进行纠正避免影响进一步恶化，相对安全；高度风险等级 R3 对应 H 状态，该等级风险对项目进度影响显著，随时可能出现进度延迟，必须立即采取措施进行干预防控。对比表 4-8 风险评估统计数据 and 图 5-5 可见，“story 拆解不够充分”的三个风险等级发生了明显变化，高度风险概率从 17%变为 47%，相对安全状态概率从 46%变为 19%，安全状态概率从 37%变为 34%，实际项目数据的引入使得“story 拆解不够充分”这项进度风险因子安全以外等级概率升高，说明在项目中实际发生可能性比预想要高。

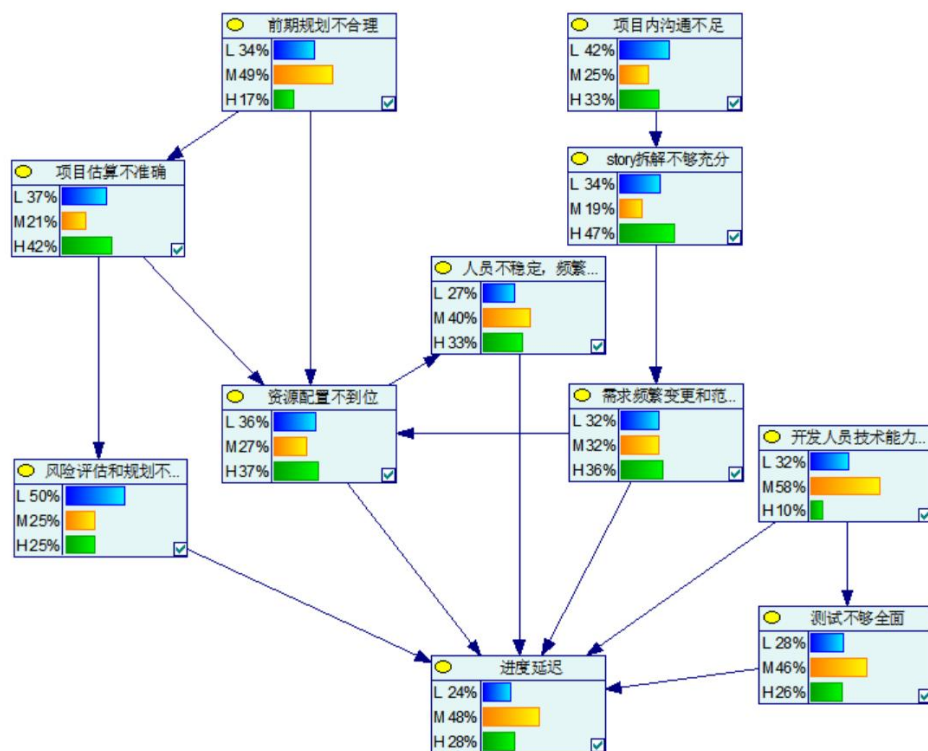


图 5-5 参数学习结果

目标节点“进度延迟”的风险状态为危险的概率是 28%，相对安全的概率是 48%，安全的概率是 24%。为有效把控风险，需要将更多关注投放到高风险，优先处理风险等级处于 H 且概率比较高的风险因子。由图 5-5 网络模型显示可知，在 10 项风险因素中，story 拆解不够充分处于 H 的概率最高为 47%，紧随其后的是项目估算不准确 42%，其他还有需求频繁变更和范围蔓延、资源配置不到位对应 H 的概率相近，均大于 35%，需要在项目推进过程中对这些因素加强防范与控制。

5.2 敏捷开发贝叶斯网络推理

前文已成功建立起针对 F 公司敏捷开发软件项目的进度风险贝叶斯网络模型，并得到目标节点“进度延迟”三个风险等级的概率分布，及各风险节点的条件概率分布，接下来借助工具软件 GeNIe2.0 实现贝叶斯网络的推理功能，深入挖掘影响敏捷开发软件项目进度的关键风险因子。

5.2.1 逆向推理

贝叶斯网络逆向推理是从目标变量出发，逐步推导出与之相关的其他变量的取值概率分布，从而更好地理解数据和做出决策。逆向推理需要将目标节点“进度延迟”

的 H 状态（高度风险）概率设置为 100%，然后更新节点概率，刷新展示各节点不同风险等级的概率值，从中比较出对目标节点影响较大的节点，作为导致进度延迟的关键风险因子，逆向推理结果如图 5-6 所示：

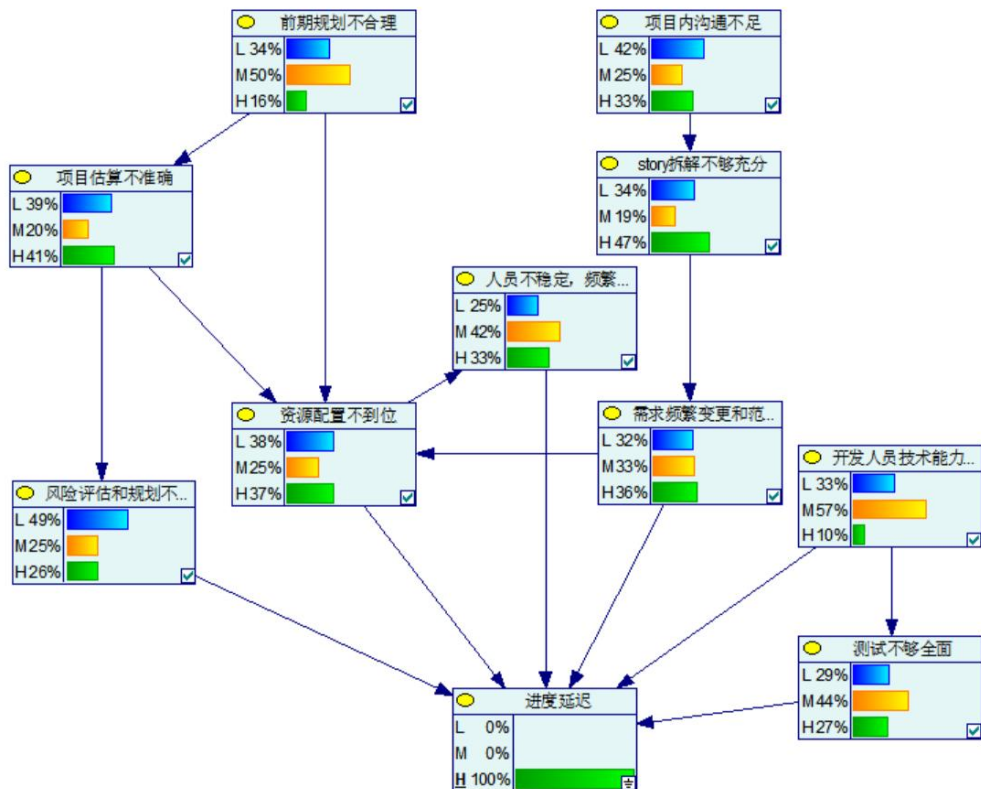


图 5-6 逆向推理结果

设置目标节点“进度延迟”的 H 状态（高度风险）的概率为 100%，进行逆向推理得出需求频繁变更和范围蔓延、资源配置不到位、项目估算不准确、story 拆解不够充分这 4 项风险因子处于 H（高度风险）的概率较高，说明这几个风险因子引起项目进度延迟的概率比其他风险因子大。当这 3 个风险因子中的一个或者多个同时发生，项目进度容易出现延迟的状况。值得注意的是高风险因子 story 拆解不够充分和需求频繁变更和范围蔓延存在逻辑关系，需求来源于用户故事列表，通过将用户故事列表拆解成一个一个 story 最终形成项目需求，拆解过程存在没有深入挖掘或分解的需求，后续工作的开展都是建立在不完备需求上进行的。此种情况大多要到测试或应用阶段才能发现需求的不完备而不得不开展需求变更补充功能和流程上的缺陷。因为问题发生的阶段相对处于上游而暴露问题的阶段滞后，极可能推翻原有功能导致修改范围超出预想发生蔓延。

另外，借助贝叶斯网络分析工具 GeNIe2.0 可以查看关键风险因子的条件概率的分布情况，选中“项目估算不准确”节点双击显示其条件概率分布，可见该因子在“前期规划不合理”条件下各风险等级的条件概率值，如图 5-7 所示：

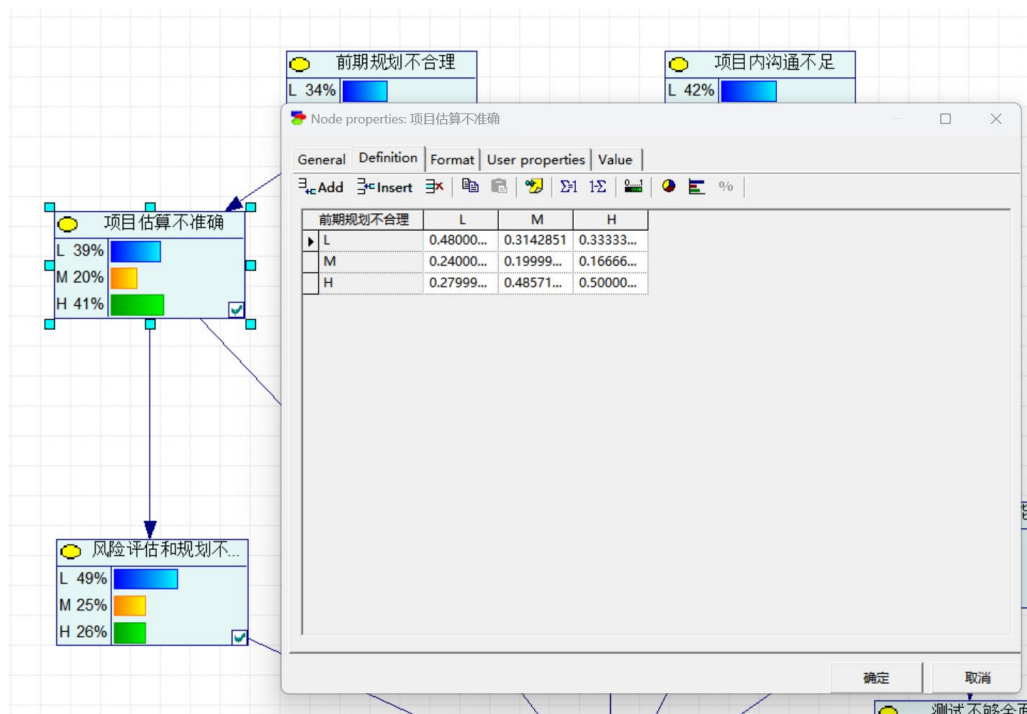


图 5-7 某风险因子条件概率分布

5.2.2 最大致因链分析

贝叶斯网络的最大致因链分析能够协助找出导致目标变量取值概率分布最大链路中的关键因素。在 GeNIe2.0 中，通过对模型中有向边加粗显示来表明父节点对子节点具有较强的影响力，识别出关键路径和主导因素，结果如图 5-7 所示。

分析结果可知，导致进度延迟的最大致因链，得到结果分别是：“前期规划不合理，具体可行性不落地” - “资源配置不到位（人力、物力、财力）”、“项目估算不准确” - “资源配置不到位（人力、物力、财力）”、“开发人员技术能力不足” - “测试不够全面”三条路径。

总体来看，导致进度延迟的主要因素大部分来自项目启动和规划阶段，处于整个项目生命周期的上游，少部分来自迭代阶段。F 公司虽然建有完整的规划知识体系和文件模板，但知识内容复杂，模板文档数量大，同时缺少全面完整的文件说明，学习过程依赖相关人员口口相传。纷繁复杂的文件加之项目经理的相关理论不足导致实际项目启动和规划开展中很多需要在前期达成的工作没有做到位，进而在后续工作中暴

露。并且项目团队体制的成立大多依据稼动空闲原则，能够满足人数要求组建团队，但从人员技术配比、工作经验年西安的维度看无法满足；虽然项目中提供开发用的电脑，但相当一部分数量的设备来自公司成立之初集中购进的二手机器，仅是在原有基础上增加了内存条进行提升。使用这部分电脑的人员常会出现操作卡顿、蓝屏等情况，需要花费额外时间处理硬件问题。另外，开发人员技术能力作为软件项目的核心要素之一，也是关联进度情况重要因素。开发人员技术能力不足主要从两个维度可见，其一是人员具备相关知识，但因为缺乏实践经验熟练度不够。这种情况下依靠开发人员本身也能够完成任务，只是生产效率上略低，需要经验者提示或简单协助，计划编排时需要多分配些计划工时。其二是人员能力完全无法胜任一项任务的基本要求，能力不足时往往会把注意力集中到个人能够完成的部分，投入大量的时间精力实际完成率只是总体极小部分，为了避免过多的偏差项目经验者常会出手解决问题，造成一个任务实际投入过多的人力成本。在新人较多的团队中，此项风险极易发生。沟通是贯穿项目始末的行为，而项目前期的沟通尤为重要。前期项目干系人通过共通确定项目最终目标和各重点环节的小目标、时间节点，此时沟通不足则是自上而下的影响。

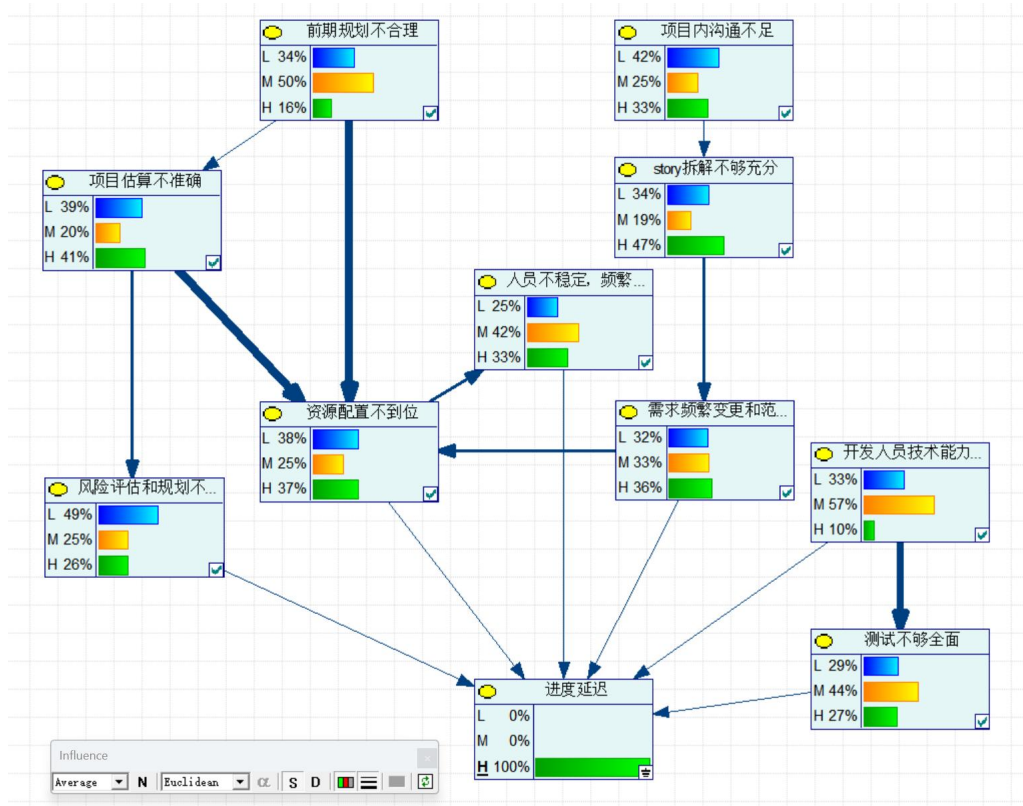


图 5-8 最大致因链分析

5.3 进度风险管理建议

针对前章研究结果,本章依据最大致因链分析结果从项目、团队、客户的维度分别提出具体的管理意见,以增加管理建议的可操作性。风险应对策略可以从风险发生的概率和风险影响程度等方面提出多种策略,主要包括规避风险、减轻风险、风险预留、接受风险和转移风险^[44]。规避风险是指项目风险发生的概率很高,对项目造成的影响程度较大,没有其他折中措施的情况下,放弃会造成高风险的方案;减轻风险指的是在风险发生之前采取一些措施降低风险发生的可能性或减少风险可能造成的损失;风险预留是指事先为项目风险预留一部分后备资源,一旦风险发生,后备资源能够应对风险;转移风险是借用合同或协议,在风险发生时将责任的全部或一部分转移到项目以外的有能力承受或控制风险的其他方;接受风险是指风险不可避免情况下,项目团队来承担风险后果^[44]。

5.3.1 项目管理

通过贝叶斯网络的最大致因链分析发现,导致项目进度延迟风险发生的因素中有三项来自于项目的启动与规划阶段,也就是说延迟在项目早期可能就埋下隐患,故主要采取风险策略是规避。项目启动与规划阶段多是管理人员介入,与客户商定项目里程碑,制定项目计划,侧重点更多的放在开发和交付阶段的计划上,对开展研发工作前置的环节关注不多,也是造成前期规划不合理、估算不准确的原因,因此针对项目要做预先的安排:

(1) 加强评审。F 公司为了辅助项目经理估算整体工作量,提供了工作量转换模板,可以将一个模块根据增删改查、预计组件数量、特殊性 etc 对标到难度等级,根据难度等级对应的人天合计整个项目预计投入,但是一个对标过程依赖项目经理的管理水平和经验,个人的主观理解也会产生一些偏差。为了减少这些偏差,除了项目经理参与工时评估外,还需引入一个高年资项目经理一起参与工作量评审,作为双重检查,当两人评估差异超过 15%要考虑重新评估。对于前期规划阶段产出的风险管理计划、沟通计划、质量计划等要上会评审,以往这些计划内容都是拷贝上一个项目加以修改,存在项目经理偷懒的情况,计划内容并没有根据新项目做调整,实际项目开展可能并不向下传达执行。通过评审会议确认计划是否包含什么时间、执行内容、执行目标、执行结果评价几个重要信息,也能够避免不合理的计划无法落地执行。对开发

过程成果的不定期评审，也能及时发现问题，例如需求故事拆解不充分问题能够在早期纠正。

(2) 资源梳理。资源包括人力资源、物资资源、财务资源等，F公司以往项目多是在人力资源和物资资源方面出现问题，进而引发财务资源过多消耗。项目前期应尽早梳理项目要求配置的管理人员、开发人员、测试人员，对于固定体制的团队，根据与项目的匹配度，提前培训、自学弥补技术短板；没有固定体制的前提则要按照技术需要向部门申请人员，明确加入时间，人员无法调配的向部门汇报。另外，硬件和软件也要早做规划，开发用的电脑能否支撑项目启动、运行，性能是否达标。对于现有人员不定期更新办公电脑配置信息，收集使用问题，及时改善，避免因运行慢、机器卡导致开发推进不顺利。

(3) 项目沟通记录。建立良好的沟通机制，确定沟通频率、渠道和对象，确保项目各方之间及时有效地沟通和信息共享。在沟通计划的基础上，建立沟通记录在线文档，全员编辑权限。在项目推进过程中，需求、实现方式等与客户确认事项应记录在档，明确时间、对接双方，沟通内容。需要与客户确认业务前可以先查阅沟通记录，确认是否已有人完成沟通并记录结论，避免反复向客户确认相同内容，同时作为沟通证据避免业务反复而发生扯皮。另外，项目组长应与组员定期进行沟通，了解在项目过程中是否存在困难，哪些方面需要改善，也可以向组员收集项目管理过程的意见建议，通过双向沟通互相促进。

5.3.2 团队管理

团队管理对于保障项目进度的重要性不言而喻。一个团结、高效的团队可以在很大程度上提高项目进度的执行效率和顺利完成项目目标。软件项目作为复杂度高、技术性强的综合性活动，对人的依赖和要求比较高，而人作为不确定因素需要加强管理的手段和力度达成约束和提升人的行为，因此团队人员风险采取的策略以减轻为主。通常F公司对于新进员工有相对完备的入职培训，但缺少针对性的技术培训，可以结合公司整体项目特点对新人安排技术培训以达到未来参与项目快速上手快速融入的目的，其次是对项目人员加强项目内业务和技术培训，有些项目使用的技术框架、开发语言具有独特性，人员在开发语言转换的过程中投入大量学习成本，通过培训的方式提升效率压缩学习成本。业务培训则是让全员了解自己要做什么、在做什么的过程，在业务理解的基础上投入工作对项目质量、成本、时间多维度提升都有重要意义。理

解业务能够了解功能间关联关系，能够在更早期发现不闭环的功能设计问题；明确的业务理解能够避免反复的确认过程，帮助识别潜在的瓶颈和优化机会，从而提高效率并降低成本；理解业务流程和需求帮助团队更快地明确任务的优先级和执行方式，减少浪费时间在错误的方向。针对 F 公司人员技术能力不足的情况，还可适当提升开发人员进入门槛，针对实习生增加入职专业技能培训，开展老带新等形式的协助，帮助新人尽快适应并进入高效的工作状态。对于项目经理职责担当的上游工作，还需开展专项培训，帮助技术转管理的中层人员首先补充理论知识，并通过在项目过程中的进一步实践完成理论到经验的打通。项目进行过程中，管理人员应注意加班、工作不饱和等特殊场景下人员的心态及情绪变化，及时跟进调节，避免因一时的情绪问题导致的进度崩盘。

5.3.3 客户管理

客户对于项目进度的影响是非常显著的。客户的需求、期望、参与度和反馈都会对项目的进度产生直接或间接的影响。具体从下面三个方面着手：

（1）与客户统一思想。在项目前期接触中，对于项目的目标、计划、需求等内容达成一致的认识。客户需求的频繁变更或不确定性可能导致项目范围蔓延，在软件项目中此种情况比较常见，可与客户共通制定需求变更流程和管理方法，对于需求变更不能全盘接收。设置变更量阈值，偏离阈值及时联络客户商讨对应策略，是调整当前计划以满足优先度要求，还是将非必要需求放入下一次迭代升级。

（2）与客户明确权利与责任。客户与开发团队抱有同一个项目目标，即保质保量交付项目，顺利上线。由于客户作为甲方，多是采用结果导向思维，在项目过程中注重成果的呈现，不时提出需求变更，而忽视了因此给开发团队带来的成本、工作量的增加。即便是甲方也需明确权责，避免对项目的期望过高或不切实际，导致项目团队面临巨大的压力和挑战，影响项目进度和质量。项目明确客户方责任人对接人，避免遇见问题不知道找谁、找不到人的情况，能够及时应对业务问题、确认事项、关键决策无法做出。

（3）与客户保持密切合作。通过开发团队的沟通与互动增加客户在项目中的参与感，项目团队不仅仅是作为工具人去完成交代事项，也要站在客户的角度思考更有利的项目推进方式和方法。与客户保持良好的合作关系，及时回款从财力方面起到补充项目资源的作用。

5.4 本章小结

本章在第四章进度风险因子识别和初步分析的基础上,采用专家知识和机器学习实现 F 公司敏捷开发软件项目进度风险模型构建,并借助贝叶斯网络分析软件 GeNIe2.0 实现逆向推理和最大致因链分析,确定出导致进度延迟的主要风险因子并就结果提出对应的管理改善措施,结合风险应对策略从项目管理、团队管理、客户管理三个方面提出了改善措施,进一步监管风险要素。

第 6 章 总结与展望

6.1 总结

F 公司软件项目进度延迟的情况时有发生,反映其在进度风险管理方面存在改进空间。本文从 F 公司基于敏捷开发方式的软件项目入手,通过企业情况、项目情况和进度管理现状结合笔者观察发现项目进度存在的风险问题,进而借助文献分析和问卷调查、头脑风暴等科学研究方法识别出项目进度风险因子 25 项,再次设计发放访谈统计进度风险因子发生概率与影响程度,据此结果配合风险矩阵对各风险因子进行阶段划分和风险等级概率统计,并通过加权平均数计算裁剪出影响较大的 10 个进度风险因子为后续贝叶斯网络建模做准备。结合项目实例根据专家知识和机器学习构建贝叶斯网络拓扑结构并确定出各条件概率,借助贝叶斯网络分析软件找出 F 公司敏捷开发软件项目导致进度延迟的最大致因链,并就此提出针对性改善策略。

开展本文写作前,笔者对风险管理、进度管理相关的理论、方法、工具进行学习,并在 F 公司实际参与项目过程,收集历史项目数据,为本次研究从理论、现象、数据、对策等方面打下基础。考虑到风险的一般管理方法直接应用到 F 公司具有一定作用,但不能很好的适应其特殊性和个性化,引入贝叶斯网络,结合实际情况使得进度风险模型更加贴合企业情况。也可以更加有针对性的制定应对措施,让进度风险管理更见成效。

在软件行业内卷和激烈市场竞争下,按时交付是赢得市场先机和客户满意度的重要一环,通过进度风险管理提早制定管理计划、尽早识别风险、执行对应策略并实施,保证风险不发生、不恶化、最终消亡的目的。

6.2 展望

本文聚焦 F 公司敏捷开发软件项目,对进度风险识别、分析、控制进行研究并提出改善建议。考虑到研究的不足和局限性需要未来进一步完善:

其一,当前模型训练收集的实例数据较少,考虑到专家知识在网络建模过程中占比较大,存在一定主观性影响,未来可以更多有目的地收集项目过程资料、经验总结作为模型的输入数据。

其二,当前包括 PMP 认证、CMMI 认证、ISO90001 认证等的企业、个人认证体

系为项目进度管理、进度风险管理提供了理论依据,结合贝叶斯网络模型提出的进度风险管理策略在实际运用和管理实践中的与项目的匹配程度和实施有效性是不确定的,需要应用过程中结合持续改进和调整,以建立能够有效管理项目进度的框架策略。

参考文献

- [1] 张越.瀑布模型、快速原型模型和增量模型的对比[J].电子技术与软件工程, 2019(03):32.
- [2] 周榕.A 公司软件开发项目进度管理研究[D].北京邮电大学,2019
- [3] 范俊辉.软件开发项目进度管理[J].数字通信世界,2018,(02):229
- [4] ZHANG Yu,and DU Hui.Improved Critical Chain Based R D Project Schedule Management Optimization[A].Conference Proceedings of the 8th International Symposium on Project Management, China (ISPM2020).Ed.School of Economics and Management, Beijing Jiaotong University;Product Planning Director in ZTE Corporation, 2020
- [5] 王坤琦.项目进度管理的动态调整研究与应用[D].北京交通大学,2021
- [6] 王一然.机场休息室管理系统软件项目进度管理研究[D].浙江大学,2021
- [7] 陈祥.管理信息系统建设中的进度管理研究[J].科技资讯,2021,19(27):101-103
- [8] 刘心语.D 公司 A 软件项目延期交付案例研究[D].大连理工大学,2022
- [9] 邵俊,倪枫,刘姜,吴霞,尹思淼.基于熵权法的软件开发项目进度风险影响因素分析研究[J].项目管理技术, 2022, 20(10): 7-10
- [10]刘华清.敏捷开发中进度管理的策略[J].数字通信, 2013, 40(04): 78-80+94
- [11]武红兰.敏捷开发在 A 公司软件项目进度管理中的应用研究[D].北京交通大学, 2019
- [12]陈建.探析敏捷开发在软件项目进度管理中的应用[J].计算机产品与流通, 2020(03): 11+171
- [13]宋戈.基于敏捷模式的 F 公司 A 软件项目进度管理研究[D].山东大学, 2021
- [14]赵剑冬,林健.敏捷方法在软件项目开发中的实践[J].计算机工程与设计,2007,(12):2772-2774
- [15]许孟杰,徐昶.敏捷开发在高职软件技术课程教学中的应用探究[J].计算机教育,2022,(08):138-141
- [16]王晓龙.基于敏捷开发的持续交付系统的研究[J].信息技术,2023, (05):131-136
- [17]Alyahya S ,Ivins K W ,Gray A W .Raising the Awareness of Development Progress in Distributed Agile Projects[J].Journal of Software,2013,8(12):3066-3081.

- [18]吴风景. Scrum 敏捷开发在初创型企业软件项目进度管理中的应用研究[D].浙江大学,2019
- [19]张立业. 软件项目敏捷开发的进度风险管理[D].中国石油大学(北京), 2019
- [20]Vasudeva S S ,Urvashi R .A Goal-driven Risk management approach for Distributed Agile Development Projects[J].Australasian Journal of Information Systems,2019,23
- [21]Srivastava A ,Mehrotra D ,Kapur K P , et al.Analytical evaluation of agile success factors influencing quality in software industry[J].International Journal of System Assurance Engineering and Management,2020,11(2s):1-11
- [22]Corey B ,Rudy H .Issues, challenges, and a proposed theoretical core of agile software development research[J].Information Systems Journal,2021,32(1):103-129
- [23]李义杰, 蒋靖, 程政.基于贝叶斯网络的软件项目进度管理模型[J].计算机工程与科学,2011,33(11): 140-143
- [24]严莉敏.基于贝叶斯网络的项目风险管理应用[J].企业技术开发, 2018, 37(01): 108-110
- [25]胡姗. 基于贝叶斯网络的 ERP 项目风险管理研究[D].中南财经政法大学, 2019.
- [26]尤建新, 秦云.基于贝叶斯网络模型的关键链项目进度管理[J].同济大学学报(自然科学版), 2015, 43(10): 1606-1612
- [27]杜元伟, 石方园, 杨娜.基于证据理论/层次分析法的贝叶斯网络建模方法[J].计算机应用, 2015, 35(01): 140-146+151
- [28]Chen Zhaorong, Deng Xiaoji, Cai Zhili, Zeng Changluo, Zhang Guoshen. Progress Risk Identification of Suspended High-Rise Buildings Renewal Based on Dynamic Bayesian Network[J]. Science Discovery,2022,10(6)
- [29]Hong Suk S ,Chulsoon P .Project Schedule Risk Assessment Based on Bayesian Nets[J].Journal of Society of Korea Industrial and Systems Engineering,2016,39(1):9-16
- [30]Agile Manifesto. Agilemanifesto. org. [EB/OL],
<https://agilemanifesto.org/iso/zhchs/manifesto.html>,2023-8-29
- [31]傅云霞.L 公司敏捷开发项目管理绩效评价研究[D].青岛科技大学,2018
- [32]马国丰, 陈强.项目进度管理的研究现状及其展望[J].上海管理科学, 2006(04): 70-74

- [33]陈德强,潘高.几种项目风险评估方法比较[J].合作经济与科技,2011,(18):50-51
- [34]董春芳.基于贝叶斯网络的超高层项目全过程成本风险管理研究[D].武汉大学,2019
- [35]余娅婷.基于贝叶斯网的学位论文进度风险管理及开发[D].昆明理工大学,2020
- [36]海丽切木·阿布来提.浅谈数据挖掘技术中常用的分类器构造方法[J].福建电脑,2012,28(02):98-100
- [37]徐梓华. S 公司敏捷开发互联网软件项目的风险管理[D]. 电子科技大学, 2022.
- [38]吕景丽. 软件项目管理中的进度控制问题及对策分析 [J]. 无线互联科技,2019,16(12):45-46
- [39]张涛.软件项目管理问题与优化策略研究[J].中国管理信息化,2023,26(18):150-152
- [40]周华,俞时权.敏捷环境下软件开发的风险管理[J].科技管理研究,2007,(07):162-164
- [41]刘婧.A 公司智慧图书馆软件产品敏捷开发风险管理研究[D].浙江大学,2021
- [42]丁蓓妍. 基于贝叶斯网络的公路养护作业风险评价与控制研究[D].长安大学, 2018
- [43]鹿浩,张鸿洲,钟寒,等.融合专家综合知识的贝叶斯网络参数学习方法[J].科学技术与工程,2021,21(21):8944-8950.
- [44]张莉.H 公司软件开发项目进度风险管理研究[D].山东大学,2021

附录 A 关于敏捷开发项目进度影响因素的问卷调查

尊敬的女士/先生：

您好，十分感谢您于百忙之中参与本问卷调查！本问卷旨在对采用敏捷开发软件项目的进度风险因素影响程度进行评估调查，请依据您的真实意见填写。本问卷结果仅用于课题研究，并承诺绝不向第三方透露您的个人信息，感谢您的配合与支持！

第一部分：基本情况

1. 您参与敏捷开发项目的时点？

- ☐ 从未参与
- ☐ 正在参与
- ☐ 以前参与过

2. 您在敏捷开发项目中担当的角色？

- ☐ 产品负责人
- ☐ 敏捷教练
- ☐ 技术顾问
- ☐ 项目经理
- ☐ 设计人员
- ☐ 开发人员
- ☐ 测试人员
- ☐ 用户代表
- ☐ 其他

3. 您参与的敏捷开发项目，进度延迟发生的频率情况如何？

- ☐ 极少
- ☐ 偶尔
- ☐ 有时
- ☐ 经常
- ☐ 频繁

4. 您从事软件行业的年限是多少？

- ☐ 1-3 年
☐ 3-5 年
☐ 5-10 年
☐ 10-15 年
☐ 15-20 年
☐ 20 年以上

第二部分：项目进度风险因素识别

本问卷所调查的敏捷开发项目进度风险因子是指对项目进度造成延迟的风险因素。请您按照下表的评分标准分别评估 25 个风险因子对项目进度的影响程度及风险发生的概率，并选择对应的评估分值。

评估分值	0	1	2	3	4
风险发生概率	几乎不发生	极少发生	偶尔发生	经常发生	频繁发生
对项目进度的延迟程度	几乎不延迟	延迟<10%	延迟 10%-20%	延迟 20%-50%	延迟>50%

5. 本题列举了 25 项可能导致项目进度延迟的风险因子，请您结合自身项目经历评估风险发生的概率情况。

进度风险因子	0	1	2	3	4
前期规划不合理，具体可行性不落地	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
资源配置不到位（人力、物力、财力）	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
技术架构选型不合理	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
风险评估和规划不足	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏合理的风险应对方案	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目分工不明确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目估算不准确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
人员体制不合理	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Story 拆解不够充分	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求描述模棱两可不明确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

需求的复杂程度高	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求频繁变更和范围蔓延	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求理解不到位	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
开发技术上存在难点	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
存在不确定的三方中间件的使用	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
人员不稳定，频繁变动	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
开发人员技术能力不足	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
评审会议效果不达标	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
测试不够全面	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
部署流程不完善	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏有效的问题识别和收集机制	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
不适当的优先级和资源分配	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏监测和评估机制	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目内沟通不足（团队内部、团队之间）	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
其他的不可控的第三方因素，如金融危机、国家政策变更等	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

6. 本题列举了 25 项可能导致项目进度延迟的风险因子，请您结合自身项目经历评估每项因子对项目进度的影响程度。

进度风险因子	0	1	2	3	4
前期规划不合理，具体可行性不落地	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
资源配置不到位（人力、物力、财力）	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
技术架构选型不合理	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
风险评估和规划不足	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏合理的风险应对方案	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目分工不明确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目估算不准确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
人员体制不合理	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Story 拆解不够充分	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

需求描述模棱两可不明确	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求的复杂程度高	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求频繁变更和范围蔓延	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
需求理解不到位	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
开发技术上存在难点	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
存在不确定的三方中间件的使用	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
人员不稳定，频繁变动	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
开发人员技术能力不足	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
评审会议效果不达标	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
测试不够全面	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
部署流程不完善	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏有效的问题识别和收集机制	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
不适当的优先级和资源分配	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
缺乏监测和评估机制	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
项目内沟通不足（团队内部、团队之间）	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
其他的不可控的第三方因素，如金融危机、国家政策变更等	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

7. 除已列举的可能造成项目进度延迟的风险因子外，您是否还有其他风险因子需要补充，请列举。