

Tech Management

Han Rui

2025 年 9 月 6 日

目录

第一部分 个人发展	5
第一章 他人感知	6
1.1 概述	6
1.2 关键要素	6
1.2.1 观察能力	6
1.2.2 同理心	6
1.2.3 沟通技巧	6
1.3 实践方法	6
1.4 常见挑战	7
1.4.1 主观偏见	7
1.4.2 信息不足	7
1.5 提升建议	7
第二章 学习成长	8
第三章 工程师思维转变	9
第四章 管理技能培养	10
第五章 职业规划	11
第六章 自我认知	12
第二部分 团队管理	13
第七章 人才培养	14
7.1 概述	14
7.2 培养策略	14
7.2.1 个性化发展计划	14
7.2.2 导师制度	14

目录	3
7.2.3 轮岗机制	14
7.3 培养方法	14
7.4 评估体系	15
7.4.1 能力评估	15
7.4.2 绩效跟踪	15
7.5 成功要素	15
第八章 招聘面试	16
第九章 文化建设	17
第十章 绩效管理	18
第三部分 技术体系	19
第十一章 代码审查	20
11.1 概述	20
11.2 审查流程	20
11.2.1 提交前检查	20
11.2.2 同行评审	20
11.2.3 修改完善	20
11.3 审查要点	20
11.4 工具支持	21
11.4.1 代码审查平台	21
11.4.2 静态分析工具	21
11.5 最佳实践	21
第十二章 技术债务	22
第十三章 技术选型	23
第十四章 架构设计	24
第四部分 沟通表达	25
第十五章 会议主持	26
第十六章 倾听技巧	27
第十七章 冲突处理	28

第十八章 反馈收集	29
第十九章 影响力建设	30
第二十章 报告撰写	31
第二十一章 演讲技巧	32
第二十二章 跨部门协作	33
 第五部分 负责人修炼	 34
第二十三章 他人管理	35
第二十四章 决策判断	36
第二十五章 危机处理	37
第二十六章 团队领导	38
第二十七章 自我定位	39
第二十八章 责任担当	40
 第六部分 问题诊断	 41
第二十九章 人际洞察	42
第三十章 情境感知	43
第三十一章 效果评估	44
第三十二章 根因分析	45
第三十三章 解决方案	46
第三十四章 问题识别	47
 第七部分 项目管理	 48

第一部分

个人发展

第一章 他人感知

1.1 概述

他人感知是管理者必备的核心能力之一，涉及如何准确理解和评估他人的想法、情感和行为模式。

1.2 关键要素

1.2.1 观察能力

通过细致观察他人的言行举止，捕捉非语言信息和潜在信号。

1.2.2 同理心

站在他人角度思考问题，理解其动机和需求。

1.2.3 沟通技巧

通过有效沟通验证和深化对他人的理解。

1.3 实践方法

- 主动倾听，关注言外之意
- 观察肢体语言和表情变化
- 定期进行一对一沟通
- 收集多方面反馈信息

1.4 常见挑战

1.4.1 主观偏见

避免用自己的价值观和经验来判断他人。

1.4.2 信息不足

在信息有限的情况下做出准确判断的困难。

1.5 提升建议

持续练习观察和倾听技能，保持开放和好奇的心态，定期反思和调整自己的认知模式。

第二章 学习成长

第三章 工程师思维转变

第四章 管理技能培养

第五章 职业规划

第六章 自我认知

第二部分

团队管理

第七章 人才培养

7.1 概述

人才培养是团队管理的核心职责，通过系统性的培养计划帮助团队成员提升能力和实现职业发展。

7.2 培养策略

7.2.1 个性化发展计划

根据每个人的特点和职业目标制定针对性的培养方案。

7.2.2 导师制度

建立经验丰富的导师与新人的配对机制。

7.2.3 轮岗机制

通过不同岗位的轮换让员工获得全面的经验。

7.3 培养方法

- 在职培训和技能提升
- 项目实战和挑战性任务
- 外部培训和学习机会
- 定期反馈和指导

7.4 评估体系

7.4.1 能力评估

定期评估员工的技能水平和成长进度。

7.4.2 绩效跟踪

通过具体的工作表现衡量培养效果。

7.5 成功要素

建立清晰的职业发展路径，提供充足的学习资源，营造支持成长的团队文化。

第八章 招聘面试

第九章 文化建设

第十章 绩效管理

第三部分

技术体系

第十一章 代码审查

11.1 概述

代码审查是软件开发过程中的重要环节，通过同行评审提高代码质量、分享知识和减少缺陷。

11.2 审查流程

11.2.1 提交前检查

开发者在提交代码前进行自我检查和基本测试。

11.2.2 同行评审

由其他开发者对代码进行详细审查和反馈。

11.2.3 修改完善

根据审查意见进行代码修改和优化。

11.3 审查要点

- 代码逻辑和算法正确性
- 编码规范和风格一致性
- 性能和安全性考虑
- 可读性和可维护性
- 测试覆盖率和质量

11.4 工具支持

11.4.1 代码审查平台

使用 GitHub、GitLab 等平台的 Pull Request 功能。

11.4.2 静态分析工具

集成 SonarQube、ESLint 等自动化检查工具。

11.5 最佳实践

保持审查的及时性和建设性，关注代码质量而非个人，建立团队共同的编码标准。

第十二章 技术债务

第十三章 技术选型

第十四章 架构设计

第四部分

沟通表达

第十五章 会议主持

第十六章 倾听技巧

第十七章 冲突处理

第十八章 反馈收集

第十九章 影响力建设

第二十章 报告撰写

第二十一章 演讲技巧

第二十二章 跨部门协作

第五部分

负责人修炼

第二十三章 他人管理

第二十四章 决策判断

第二十五章 危机处理

第二十六章 团队领导

第二十七章 自我定位

第二十八章 责任担当

第六部分

问题诊断

第二十九章 人际洞察

第三十章 情境感知

第三十一章 效果评估

第三十二章 根因分析

第三十三章 解决方案

第三十四章 问题识别

第七部分

项目管理