

工作标准

工作标准是为了更好的量化工作者的工作效率, 为绩效考核提供数据依据。

还在为不知道如何评估工作量而烦恼? 为评估纬度不统一而辩解? 为评估结果引争议而伤感?

不必了, 接下来将安排的明明白白, 一个公式让人人, 都是项目经理。

本文将详细介绍货拉拉-卖车及车后大市场业务部工作量化标准。

〇、量化模型

工作量化的模型比较多, 主要有WBS(类比法)、SLOC(代码行)、FP(功能点)较为主流。

本文将结合此三项模型适当调整寻找一个合适的量化标准。

1. 基于WBS的工作量估算

基于WBS的工作量估算方法, 是最常见的一种估算方法, 也是厂商最常用的。

基于WBS的工作量的估算方法, 又称为由底向上法(自下而上法), 通常的估算步骤如下

- 1) 寻找类似的历史项目, 进行项目的类比分析, 根据历史项目的工作量凭经验估计本项目的总工作量;
- 2) 进行WBS分解, 力所能及地将整个项目的任务进行分解;
- 3) 参考类似项目的数据, 采用类比法或专家法, 估计WBS中每类活动的工作量;
- 4) 汇总得到项目的总工作量;
- 5) 与第1步的结果进行印证分析, 根据分析结果, 确定估计结果。

2. 基于代码行的工作量估算

基于代码行(SLOC)的工作量估算, 是从开发者的技术角度出发来度量软件。代码行数是软件开发者最早进行规模测量的主要方法。

进行工作量估算时, 先采用WBS法、类比法等统计出软件项目的代码行数, 然后将代码行数转换为人天数。

其中, 将代码行(SLOC)转换成人天数主要有2种方法。

- 生产率方法: 要求有开发商每人天开发的代码行数, 估算出代码行数后, 直接利用代码行数 \div SLOC/人天, 即得工作量人天数。
- 参数模型法: 利用模型, 将代码行数转换成人天数。常见的模型有:

Putnam模型

Putnam1978 年提出的一种动态多变量模型。估算工作量的公式是： $K = L^3 / (Ck^3 * td^4)$
其中：L 代表源代码行数（以行计），K代表整个开发过程所花费的工作量（以人年计），td 表示开发持续时间（以年计），Ck表示技术状态常数，它反映“妨碍开发进展的限制”，取值因开发环境而异。

COCOMO II 模型

COCOMO II 模型指出，软件开发工作量与软件规模呈指数关系，并且工作量受16个成本驱动因子的影响。COCOMO II 的计算步骤如下：

- 1)估算软件规模size，这里以千代码行（KSLOC）计。
- 2)评估比例因子SF，求指数E。
- 3)求成本驱动因子值EMi。求标称进度工作量PM

3. 基于功能点的工作量估算

基于功能点（FP）的工作量估算，是从用户的角度来度量软件。

进行工作量估算时，先估计出软件项目的功能点数，然后将功能点数（FP）转换为人天数。

其中，估算功能点数的主要方法有3种：IFPUG法、MarkII法、COSMIC FFP法。

这三种方法现在都已经成为国际标准，并有详细的操作手册。

将功能点（FP）转换成人天数主要有2种方法。

- 1) 生产率法：要求有开发商每人天开发的功能点数，估算出功能点数后，直接利用功能点数÷功能点/天，即得工作量人天数。

对于开发商每人天开发的功能点数，SPR有统计，中国的值大约在5.5个功能点/人月。

- 2) 经验模型法，可以依照本企业的历史数据得到关于功能点和工作量的统计方程；也可以采用已有的经验模型，例如：COCOMO II 模型

一、通用标准

1.1 产出物

- 每个阶段必须有产出物, 否则该阶段不参与量化

1.2 跨组织协作

- 如果多个组织协同工作, 其一组织交付工期早于其他组织, 应给予其他组织适当增加资源, 尽可能缩小各组织交付工期差距。

1.3 产品定位评级

1.3.1 评级

项目分为四大类型: 微型项目(也称 **M** 类)、小型项目(也称 **S** 类)、大型项目(也称 **P** 类)、巨型项目(也称 **G** 类)。

所有项目立项会议之后应该对项目有具体的定位和评级; 定位应在《系统建设说明书》中详细说明, 评级介绍如下:

- **M(微型):** 工期极短, 业务量较少, 复杂度较低
 - 微项目(**M2**): 业务量极少 (1-5) 个的单体结构项目
 - 微系统(**M3**): 业务量较少 (6-10) 个的单体结构系统
 - 微服务(**M1**): 基于现有系统架构建设的 单 个微服务
- **S(小型):** 工期较短, 业务量不大, 复杂度不高
 - 小项目(**S1**): 基于现有系统架构建设的 (2-5) 个微服务
 - 小系统(**S2**): 基于现有系统架构建设的 (6-9) 个微服务
- **P(大型):** 工期较长, 业务量大, 复杂度高
 - 小工程(**P1**): 完全独立于现有系统架构之外建设的高扩展性 10w 并发工程
 - 大工程(**P2**): 完全独立于现有系统架构之外建设的高扩展性 百万 并发工程
- **G(巨型):** 工期很长, 业务量极大, 复杂度极高
 - 超级工程(**G1**): 完全独立于现有系统架构之外建设的高扩展性 千万 并发工程
 - 希望工程(**G2**): 完全独立于现有系统架构之外建设的高扩展性 亿级 并发工程

1.3.2 产品因子

产品开发的工期有 扩展性设计、可用性设计等可影响设计难度, 增加工作量.

为了公平合理的计算产品开发工期, 以下讲给所有评级的系统定义产品计算因子 (暂不启用, 需要时间沉淀更真实的数据):

评级	评级别称	产品因子	说明
M1	微服务	0.90	
M2	微项目	0.95	
M3	微系统	0.90	服务的数量增加可能增加交互难度
S1	小项目	1.00	这类项目是我们常见的项目大小
S2	小系统	1.10	服务的数量增加可能增加交互难度
P1	小工程	1.20	
P2	大工程	1.30	
G1	超级工程	1.40	
G2	希望工程	1.50	

1.4 任务难度评级

1.4.1 评级

难度评级分为五大级别: ★(1星)、★★(2星)、★★★(3星)、★★★★(4星)、★★★★★(5星)。

该评级对应市场主流的 A类、B类、C类、D类、E类,下面将对以上各级简要说明(以下列举的 时间 包含单元测试时间):

1. ★

任务极为简单, 不能通过自动生成(否则降1星); 单个任务 30-60分钟 即可完成。比如:

- 增加单条数据
- 删除单条数据
- 修改单条数据
- 查询单条数据
- 搜索多条数据

2. ★★

任务相对简单, 没有现成的模板(否则降1星); 单个任务 1-2小时 即可完成, 比如:

- 单表数据的冗余字段处理
- 单表数据的级联操作处理 (这里的单表指: 只需要级联一张表操作)
- 单表数据的缓存处理
- 单表数据的通用逻辑封装处理 (包含枚举、解析工具、通用接口等)
- 简单的定时任务处理 (该项只有简单和复杂两类)

3. ★★★

任务难度较高, 方便本地测试(否则加1星); 单个任务 2-4小时 才能完成, 比如:

- 多表的冗余字段处理
- 多表的级联操作处理 (这里的单表指: 要级联两张及以上表操作)
- 多表的缓存数据处理
- 多表的事务处理
- 多表数据库的通用逻辑封装处理 (包括工具, 接口, 存储过程, 视图等)
- 复杂的定时任务处理 (该项只有简单和复杂两类)

4. ★★★★

任务难度很高, 能够立刻调试(否则加1星); 单个任务 4-8小时 才能完成, 比如:

- 全局通用工具的设计和封装
- 全局适用的业务封装 (比如: 系统日志存储)
- 攻克新的技术点
- 其他系统的集成
- 第三方平台的交互设计 (比如: 微信, 支付宝、华为云等)
- 简单的文档撰写 (比如: 接口文档, 测试用例等)

5. ★★★★★

任务难度极高, 可以在容忍期限内攻克(否则改方案); 单个任务 1周内 可完成, 比如:

- 平台级的解决方案
- 组织人员系统培训
- 攻克复杂的技术难题
- 难度较高的文档撰写 (比如: 《系统建设说明书》、《部署手册》、《技术规范》等)
- 其他4星以上的任务 (超越5星的任务改方案或者通过招聘符合条件的人才解决)

1.4.2 工期范围

评级	工期范围	说明
★	30 - 60 分钟	
★★	1 - 2 小时	
★★★	2 - 4 小时	
★★★★	4 - 8 小时	
★★★★★	8 - 40 小时	超过这个工期改方案或换人

二、后端标准

本次标准将参考WBS(类比法): 即参考过往需求及其效率进行分类和量化。

2.0 项目申报

- 立项申报及之前的所有阶段没有后端工作, 因此省略..

2.1 立项会议

2.1.1 参会方

- **构思方:** 产品的构思提供方 (可能是一个组织比如运营部门, 也可能是一个人)
- **产品方:** 产品部门的有效代表 (可能是多个产品参与, 也可能是一个人)
- **技术方:** 技术部门的有效代表 (可能是多个技术参与, 也可能是一个人)
- **测试方:** 测试部门的有效代表 (可能是多个技术参与, 也可能是一个人)
- **运维方:** 运维部门的有效代表 (可能是多个技术参与, 也可能是一个人)
- **验收方:** 可以决定产品走向的项目负责人 (可能是产品方, 也可能是构思方)

2.1.2 产出物

1. 《系统建设说明书》, 至少包含的内容:
 - 产品定位说明
 - 业务边界说明
 - 需求建设说明
 - 验收标准说明

2.2 需求评审

2.2.1 参会方

- 立项会议的所有参会人员
- 具体实施各方工作的职能人员

2.2.2 产出物

1. 产品原型设计稿 *
 1. 符合验收标准
 2. 验收方无异议
2. 关键逻辑说明书
 - 视情况决议是否产出

2.3 系统设计

2.3.1 实施方

- 系统建设工作者
- 接口设计工作者

2.3.2 产出物

- 系统源码, 包含以下内容:
 - 搭建符合项目类型的项目框架
 - 集成项目所需的开发组件
 - 提供常用的代码开发示例
- 数据模型, 包含以下内容:
 - 数据库定义: 库名称, 字符集等
 - 数据表定义: 表名称, 字段名等
- 接口清单, 包含以下内容:
 - 数据格式: xml、form、json、stream等
 - 接口分类: 按服务或者业务分类
 - 接口编号: 自定义不重复的序号
 - 接口名称: 简单描述接口的作用
 - 接口地址: 根据分类和名称命名的请求地址
 - 请求频率: 普通单次请求, 定时触发请求

2.4 工期评审

2.4.1 任务分解

- **开发者** 或者其上级进行任务分解, 要求分解成可进行难度评级的小任务

4.2 评估人

- **开发者** 参考以下 **维度** 进行综合难度评估确定工期范围 (详见1.4):

综合难度是对单个任务整体的难度评估

1. 技术评估: 根据系统源码中的技术熟练度
2. 业务评估: 根据数据模型中的结构展现的业务复杂度
3. 设计评估: 根据接口清单中的要求的设计难度

- **开发者** 根据以下 **公式** 进行工期评估 (暂不启用产品因子):

- 工期计算公式:

根据 [三点估算法](#) 进行实际工期计算;

实际工期 = (工期范围最小值 + 工期范围最大值 + 最可能工期值 * 4) / 6;

- 最可能工期值:

最可能工期值由开发者自主评估, 其值范围必须在相应难度评级的工期范围内 (详见1.4)。

2.4.3 评审人

- 开发者上级 **主管** (或者验收方) 根据以下 **纬度** 进行工作量评审:

1. 工期: 是否在接受的工期内达到预期的目标
2. 成本: 是否在接受的资源(人数, 人天等)范围内协同工作
3. 评级: 是否将项目和分解的任务定义在接受的评级中

- 如果评审人不能接受评估人提交的方案需要当场协商达成一致, 包括但不限于以下内容。

- 人员资源的协调
- 阶段工期的协调

比如: 后端提前一个星期完成阶段工作, 那前端需要加人或其他方式提速, 不能让后端的进度停滞。

- 业务边界的调整

使评估调整至合理范围之后, 仍有不可接受的任何因素, 只能分期进行, 在第一期缩小业务边界。

- 项目评级的调整
- 任务难度的调整

2.4.4 产出物

- 《任务清单》

包含但不限于: 接口设计, 技术学习、业务编码、工具封装..

- 《开发计划表》

包含但不限于: 任务编号, 任务名称, 开始时间, 结束时间, 前置任务, 资源名称..

- 《工作量评估表》

《工作量评估表-模板》将在本文出版之后, 1周内补齐, 敬请期待..

2.5 编码

- 待补充

2.6 后端编码

2.6 测试

- 待补充

2.7 上线

- 待补充

三、前端标准

- 待补充