

# Lab7- 大作业final检查

## 时间、地点

---

7月2日-7月3日 线上检查

## 检查内容

---

### 1. 功能

#### 财务人员

- 账户管理
- 制定收款单
- 制定付款单
- 制定工资单
- 查看销售明细表
- 查看经营历程表
- 查看经营情况表（红冲相关 Bonus功能）
- 期初建账（Bonus功能）

#### 人力资源人员

- 员工管理
- 员工打卡
- 薪酬规则制定

#### 总经理

- 制定促销策略
- 制定年终奖（Bonus功能）

- 查看销售明细表（同财务人员）
- 查看经营历程表（不能使用红冲等功能，其它相同）
- 查看经营情况表（同财务人员）

## 2.代码

编号	检查点	检查点描述
Code01	代码结构	代码的布局是否能够清晰地体现程序的逻辑结构
Code02	代码命名	代码中包、接口、类、方法、变量等的命名是否合理
Code03	代码注释	文档注释和内部注释，复杂类和方法一定要有注释
Code04	代码可靠性	是否利用异常与断言进行防御式编程或契约式设计（可选）
Code05	单元测试代码	检查是否有单元测试代码
Code06	集成测试代码	是否有桩和驱动代码
Code07	系统架构	体系结构设计是否符合某种体系结构风格
Code08	接口设计	接口设计是否合理，是否通过接口进行调用
Code09	控制风格	是否使用合适的控制器风格
Code10	设计模式	是否合理使用设计模式
Code11	其他	架构设计部分是否有其他特点（可选）

### 知识点检查

- 制定工资单
  - 表驱动计算税款
  - 面向接口编程单据操作可抽出接口（比如 approval, makesheet, findsheetByState），具体的单据类实现这些接口
- 制定促销策略
  - 考察策略模式
  - 考察需求变更（之前的促销策略是直接在销售单中填写的数字，现在要求总经理制定策略，可修改相关数据库表以及实现）
- 薪酬规则制定

- 策略模式提供不同工资策略
- 其他
  - 涉及到金额等数字操作时注意边界(防御式编程)
  - 测试的编写也是考察的一环（一些重要的方法和流程按需编写单元测试、集成测试和接口测试；编写测试时需要考虑白盒测试【测到各种代码分支】和黑盒测试【考虑到一些边界条件】）
  - 消除代码中已有的循环依赖，避免引入新依赖

### 3. 过程

编号	检查点	检查点描述
Proc01	版本控制	是否使用版本控制工具，使用工具证明（更新记录截图等）
Proc02	持续集成	是否使用持续集成工具，与git, maven等工具整合（自动构建配置，持续集成记录截图等）
Proc03	团队合作	检查团队合作的方式（SVN,git等）
Proc04	团队交流	检查团队交流方式（会议记录，聊天截图等）
Proc05	开发模式	检查团队开发模式（主程序员，协同开发等）该项与过程得分无关。

注：前来检查过程部分的同学需要说明队员分值（整个大作业，不仅仅是过程）分配，一般来讲为均分，如全部80分，若提出有人工作量多有人工作量少，则按割补法，从工作量少的人扣除分数加到工作量多的人身上，最多10分，如80均分改成70,90,80,80. 如老师发现有队员在团队中贡献过低，将由老师直接打分。

提供相应证据文件。

### 4.文档

编号	检查点	检查点描述
Doc01	文档完整性	包括用例文档、软件需求规格说明文档、体系结构文档、详细设计文档。

Doc02	文档前后一致性	随意抽一个用例，然后要求在用例文档、需求规格说明文档、体系设计文档、详细设计文档中都有对应的匹配，并且各部分要一致（在Lab7中进行选择）
Doc03	文档是否纳入配置管理	各个文档是否有版本号，若有，是否有修订记录
Doc04	是否符合技术文档的规范	是否有清晰的目录结构；是否包含作者、引用文献及引用文档
Doc05	详细设计文档是否和代码匹配	抽一个功能点，检查详细设计中的接口（从界面到逻辑到数据访问）和实际代码中的接口是否匹配
Doc06	需求文档中假设与依赖书写是否正确	查看文档中的依赖和假设书写是否正确
Doc07	需求文档中对于功能性需求是否得当	抽取一个需求场景，查看需求文档的相应功能需求描述是否详尽
Doc08	需求文档中对于非功能需求是否得当	抽取一个需求场景，查看需求文档的相应非功能性需求描述是否详尽
Doc09	需求文档中用例描述是否得当	抽取若干用例，查看用例类图、用例描述等是否符合要求
Doc10	需求文档中非功能需求的约束是否正确	查看文档中非功能需求的定义部分，查看约束是否理解为正确约束
Doc11	需求文档中对于数据需求的定义是否明确	查看文档中关于数据的定义部分，查看是否缺少相应数据库的设定
Doc12	需求文档中假设与依赖书写是否正确	查看文档中的依赖和假设书写是否正确
Doc13	需求文档中对于功能性需求是否得当	抽取一个需求场景，查看需求文档的相应功能需求描述是否详尽
Doc14	架构设计文档中逻辑视图是否规范	抽取其中一个逻辑视图，查看是否满足标准的逻辑过程

	范	
Doc15	架构设计文档中包图是否规范	抽取其中一部分包图，查看类之间的继承关系、组合关系等是否满足相应的设计模式
Doc16	架构设计文档中对应的模块功能是否正确	抽取其中若干模块，查看是否满足相应的模块功能
Doc17	架构设计文档中使用的设计模式	查看文档中使用的架构是否搭配相应的设计模式完成相应的任务
Doc18	架构设计文档中逻辑视图是否规范	抽取其中一个逻辑视图，查看是否满足标准的逻辑过程
Doc19	架构设计文档中逻辑视图是否规范	抽取其中一个逻辑视图，查看是否满足标准的逻辑过程
Doc20	架构设计文档中包图是否规范	抽取其中一部分包图，查看类之间的继承关系、组合关系等是否满足相应的设计模式
Doc21	详细设计文档等需要指出具体设计细节处是否解释得当	抽取一个需求场景，对具体问题进行提问，查看解决方法、异常处理等是否得当
Doc22	详细设计文档接口描述是否规范	抽取设计文档中的若干个接口，查看接口描述和接口命名是否规范
Doc23	详细设计文档中顺序图书写是否规范	抽取一个业务场景，查看顺序图的格式、标号等书写是否满足要求
Doc24	测试用例文档是否完成	是否完成测试用例文档
Doc25	测试用例覆盖种类是否齐全	单元测试，集成测试，系统测试
Doc26	测试用例是否描述测试结果	是否有输入输出，重点关注测试结果的记录

测试文档要求为加分项。文档只涉及Lab7需求。

