《系统分析与设计》、《软件工程理论与实践》

课程项目说明

（13902295111; issyy@mail.sysu.edu.cn ）

1 要点

- ✧ 项目有意义。需实现一个完整的软件系统；该软件在国家和社会的政治、经济或人民生活领域具有意义；
- ✧ 系统功能合理。该系统功能合理，贴近实际应用。
- ✧ 项目能体现相关的理论（面向对象）。合理运用面向对象方法学、软件架构设计等理论。在软件系统中，建议至少要实现一种设计模式。
- ✧ 应符合当前软件系统分析、设计、实现、测试等方面的主流核心技术规范。
- ✧ 有工作量。每组 1-7 人。
- ✧ 做什么功能及功能大小、采用何种编程平台，不做限定，由各组同学自行决定。

2 提交作业要求

1） 基于内容迭代地提交作业，期末提交电子版+纸质版。

2） 纸质版：仅仅打印“分析与设计文档”即可，其它制品不必打印。

3） 期末提交的电子版材料的内容：每组一个文件夹（命名：学号姓名学号姓名。。。），其包含：

a) XXX 系统.doc

b) **项目中 UML 源文件**

c) **一个子文件夹，命名：CODE，内为代码，需提交所有代码，用于验证**

d) **系统核心部分录像，3-5min**

e) **其他（例如，同学的感悟），这些内容可以侧面反映出团队管理、同学们平时的努力、同学们的技术水平，等等。**

f) **提交时间：**

- **纸质版，请于 2018-06-15，上课时带来**
- **电子版：请于 2018-06-15 之前，按班级分别发给：丘兆丰、孙洪亮、庞景龙三位同学**

3 分析与设计部分的具体内容（供参考）

1	Requirements Overview Produce the following: □ Problem statement □ Use-Case Model main diagram □ Supplementary specification □ Glossary Review the given Requirements artifacts, noting any questions, issues and inconsistencies
2	Architectural Analysis Given the following: □ Some results from the Requirements discipline: □ Problem statement □ Use-Case Model main diagram □ Glossary

	□ Some architectural decisions: □ (textually) The upper-level architectural layers and their dependencies Identify the following: □ The key abstractions Produce the following: □ Class diagram containing the key abstractions □ Class diagram containing the upper-level architectural layers and their dependencies Compare your key abstractions with the rest of the class □ Have the key concepts been identified? □ Does the name of each class reflect the role it plays? Compare your class diagram showing the upper-level layers □ Do the package relationships support the Payroll System architecture?
3	Use-Case Analysis Given the following: □ Use-Case Model, especially the use-case flows of events □ Key abstractions/classes □ The Supplementary Specification □ The possible analysis mechanisms Identify the following for a particular use case: □ The analysis classes, along with their: □ Brief descriptions □ Stereotypes □ Responsibilities □ The collaborations needed to implement the use case □ Analysis class attributes and relationships □ Analysis class analysis mechanisms Produce the following for a particular use case: □ Use-Case Realization Interaction diagram for at least one of the usecase flows of events

	□ VOPC class diagram, containing the analysis classes, their stereotypes, responsibilities, attributes, and relationships □ Analysis class to analysis mechanism map
4	Design Elements（软件工程课程，不必写） Given the following: □ The analysis classes and their relationships □ The layers, packages, and their dependencies Identify the following: □ Design classes, subsystems , their interfaces and their relationships with other design elements □ Mapping from the analysis classes to the design elements □ The location of the design elements (e.g. subsystems and their design classes) in the architecture (i.e., the package/layer that contains the design element) Produce the following: □ For each subsystem, an interface realization class diagram □ Table mapping analysis classes to design elements □ Table listing design elements and their “owning” package Compare your results with the rest of the class □ What subsystem did you find? Is it partitioned logically? Does it realize an interface (s)? □ What analysis classes does it map to? □ Do the package dependencies correspond to the relationships between the contained classes ? □ Are the classes grouped logically? □ Are there classes or collaborations of classes within a package that can be separated into an independent package?
5	Describe the Run-time Architecture （从这里开始，部分内容，系统分析与设计课程需完成，软件工程理论与实践课程不必写） Given the following: □ Design elements (classes and subsystems) and their relationships

	□ Processes □ What classes and subsystems are mapped to what processes? Identify the following: □ Process relationships Produce the following: □ Class diagram showing the: □ Processes □ Mapping of classes and subsystems to processes □ Process relationships □ Design element relationships to support process relationships Compare your Process View with those created by the rest of the class □ Are processes and threads stereotyped properly? □ If a thread is defined, is there a composition relationship from the process to the thread? □ Is there a composition relationship from the process elements to the design elements? □ Do the necessary relationships exist between the process elements in order to support the relationships to the design elements mapped to those process elements?
6	Describe Distribution（软件工程课程，不必写） Given the following textual information: □ Network configuration (e.g., nodes and heir connections) □ What processes run on what nodes? Produce the following: □ Deployment diagram depicting: □ Nodes □ Connections ◆ What processes run on what nodes Compare your Deployment Model with those developed by the rest of the class. □ Have nodes and node connections been modeled? □ Have processes been identified and assigned to nodes? □ Do the allocations make sense?

	□ Are the processes listed beneath the nodes in the Deployment diagram?
7	Use-Case Design（软件工程课程，不必写） Given the following: □ Analysis use-case realizations (VOPCs and interaction diagrams) □ The analysis-class-to-design element map Identify the following: □ The design elements that replaced the analysis classes in the analysis usecase realizations □ The design element collaborations needed to implement the usecase □ The relationships between the design elements needed to support the collaborations Produce the following: □ Design use-case realization □ Interaction diagram(s) per use case flow of events that describes the design element collaborations required to implement the usecase □ Class diagram (VOPC) that includes the design elements that must collaborate to perform the use case, and their relationships Compare your use-case realizations □ Have all the main and subflows for this iteration been handled? □ Has all behavior been distributed among the participating design elements? *Continue to the prior page…* □ Has behavior been distributed to the right design elements? □ Are there any messages coming from the interfaces?
8	Class Design （1）（软件工程课程，不必写） Given the following: □ The architectural layers, their packages, and their dependencies □ Design classes for a particular use case Identify the following: □ Attributes, operations, and their complete attribute signatures □ Attribute and operation scope and visibility

	□ Any additional relationships and/or classes to support the defined attributes and attribute signatures □ Class(es) with significant state-controlled behavior □ The important states and transitions for the identified class Produce the following: □ Design Use-Case Realization □ Statechart for one of the classes that exhibits state-controlled behavior □ Class diagram (VOPC) that includes all operations, operation signatures, attributes, and attribute signatures Compare your results □ Is the name of each operation descriptive and understandable? Does the name of the operation indicate its outcome? □ Does each attribute represent a single conceptual thing? □ Is the name of each attribute descriptive and does it correctly convey the information it stores? □ Is the state machine understandable? □ Do state names and transitions reflect the context of the domain of the system? Does the state machine contain any superfluous states or transitions?
9	Class Design （2）（软件工程课程，不必写） Given the following: □ The Use-Case Realization for a use case and/or the detailed design of a subsystem □ The design of all participating design elements Identify the following: □ The required navigability for each relationship □ Any additional classes to support the relationship design □ Any associations refined into dependencies □ Any associations refined into aggregations or compositions □ Any refinements to multiplicity □ Any refinements to existing generalizations □ Any new applications of generalization considered □ Make sure any metamorphosis is

	Produce the following: □ An updated VOPC, including the relationship refinements (generalization, dependency, association) *Continue…* Compare your results □ Do your dependencies represent context independent relationships? □ Are the multiplicities on the relationships correct? □ Does the inheritance structure capture common design abstractions, and not implementation considerations? □ Is the obvious commonality reflected in the inheritance hierarchy?
10	Testing Case Design （都要写） For a particular Usecase, by using Equivalence Partitioning and Boundary Value Analysis, design the Testing Cases

4 实验成绩占总成绩 40%。以最终的成绩这次为准。并且，必须提交纸质版本的“XXX 系统.doc”

5 评分标准。

软件工程理论、技术运用正确、合理	UML（图制品恰如其分、详略得当、结构清晰）	CODE（与设计匹配）	类的设计正确合理	类的机制（设计合理并实现）	文档规范	测试案例设计	其它（工作量、技术先进性、综合质量、作业的创造性）	项目管理和文档质量
15%	**15%**	**10%**	**10%**	**10%**	**10%**	**10%**	**10%**	**10%**

6 欢迎同学的任何疑问、建议，请 EMAIL（issyy@mail.sysu.edu.cn）。