

CptS 322 Software Engineering Principles I

Spring 2023

Homework 5

(Due April 23th, 2022 on Canvas)

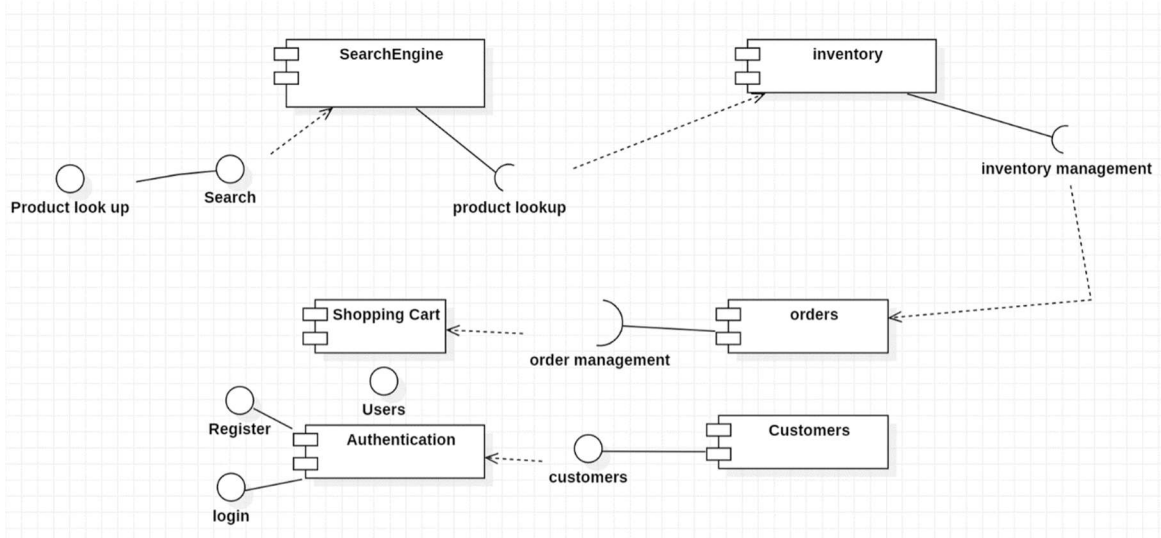
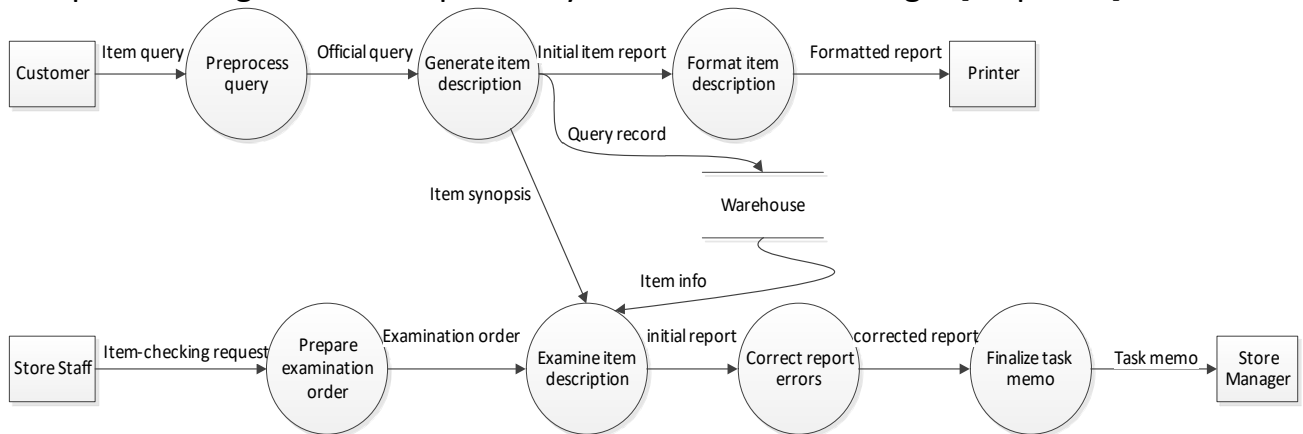
1. We have studied four architectural styles: data-centered, data-flow, call-return, and layered. Now for each style, please give an example project that would be a good fit for the style. Briefly justify your choice. [12 points]

- Data centered -> A good example of a project that would fit a data-centered architecture style is a database management system. In this style, the architecture is centered around the data. The system is designed to store, manipulate and retrieve data efficiently, and the architecture is optimized for data access and manipulation.
- Data-flow -> A project that would be a good example of data-flow is a financial trading system. In such a system, large volumes of data are continuously flowing into the system, which then processes and analyzes the data in real-time to make trading decisions.
- Call-return -> A project that would be a good fit for the call-return architecture style is a web application. In this type of style, the architecture is based on the call and returns of functions, the functions are called in a hierarchical manner to achieve the desired functionality of the system.
- Layered -> An example of a real world project that would be for the layered model is an operating system. In this style, the architecture is based on a layered approach, where different layers are built on top of each other, with each layer providing specific set of services to the layer above it.

2. There are two measures of functional independence in software design: cohesion and coupling. Explain in your words what each of these measures means. Is it better to have higher/lower cohesion? Why? Is it better to have higher/lower coupling? Why? [18 points]

- Cohesion refers to how closely related the functionality of the components within a module is. It measures how well the elements within a module work together to achieve a single purpose or responsibility. A module with high cohesion is where the components are tightly related and work together to accomplish a single task. A module with low cohesion is one where the components are loosely related and perform multiple tasks that are not necessarily related to each other. Coupling refers to the level of interdependence between different modules or components within a system. It measures how much one module relies on another module to perform tasks. A module with high coupling is one that is heavily dependent on other modules. On the other hand, a module with low coupling is one where the components are largely independent of each other. Overall, a well-designed software system should strive for high cohesion and low coupling, high cohesion and low coupling promote functional independence and make the system have a variety of good software traits such as, flexible, maintainable, scalable.

3. The following data flow diagram (DFD) is part of the flow models for the requirements of the ACE store management software. It describes the process of responding to a customer request for querying about an item. Derive the architecture design from this DFD, and draw the (brief) component diagrams that represent your architectural design. [20 points]



Notes:

- Submit a single PDF for these questions to Canvas.
- Use standard UML notations when drawing the diagrams (no hand-drawing is acceptable).