

ADD Design Process

Highlighted means fixed

ADD Step 2: Establish Iteration Goal by Selecting Drivers

The following are the system's structures to support primary functionality:

- UC-1 Scanning Items
- UC-2 Payment
- CON-1 Cash register contains local database
- CON-2 Cash register is a standalone application
- QA-1 Portability

ADD Step 3: Choose One or More Elements of the System to Refine

The entire cash register system must be refined.

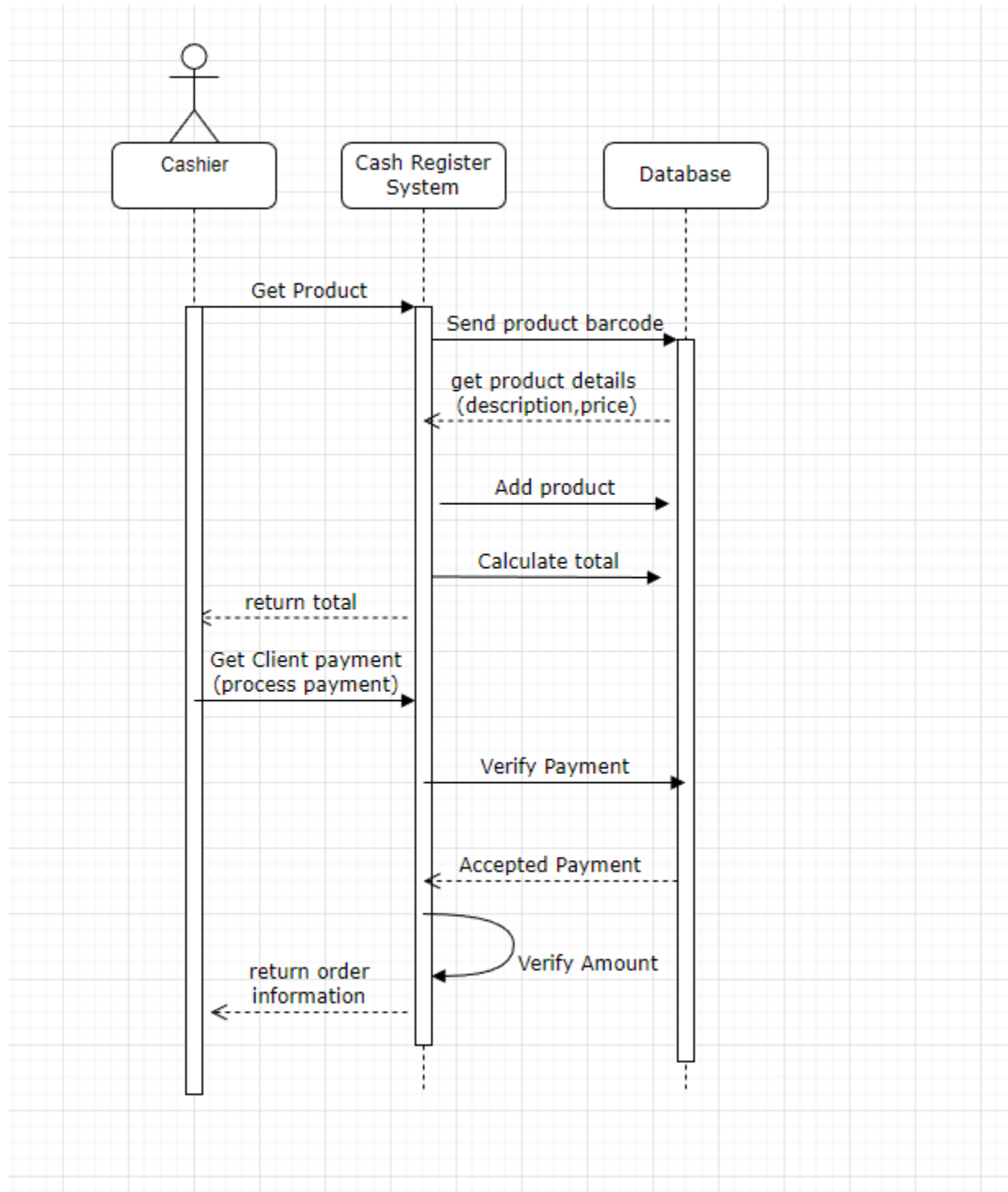
ADD Step 4: Choose One or More Design Concepts that Satisfy the Selected Drivers

Design Decisions and Location	Rationale
Designing a two-tier deployment pattern for the registration system	As addressed in CON-1 and CON-2 the system must be a standalone application and have its own database. This method can satisfy these concerns
Developing on Distributed Deployment architectural reference	This architectural structure can allow for components to be implemented on different tiers, and allow for scalability. Although the downside for this method is the cost, complexity and increased deployment effort.
Using Java Swing Framework to build a User Interface	This way the code can satisfy QA-1. The development team has experience using this technology to deploy the system. This however would encompass more functionality than needed, and more effort to develop.

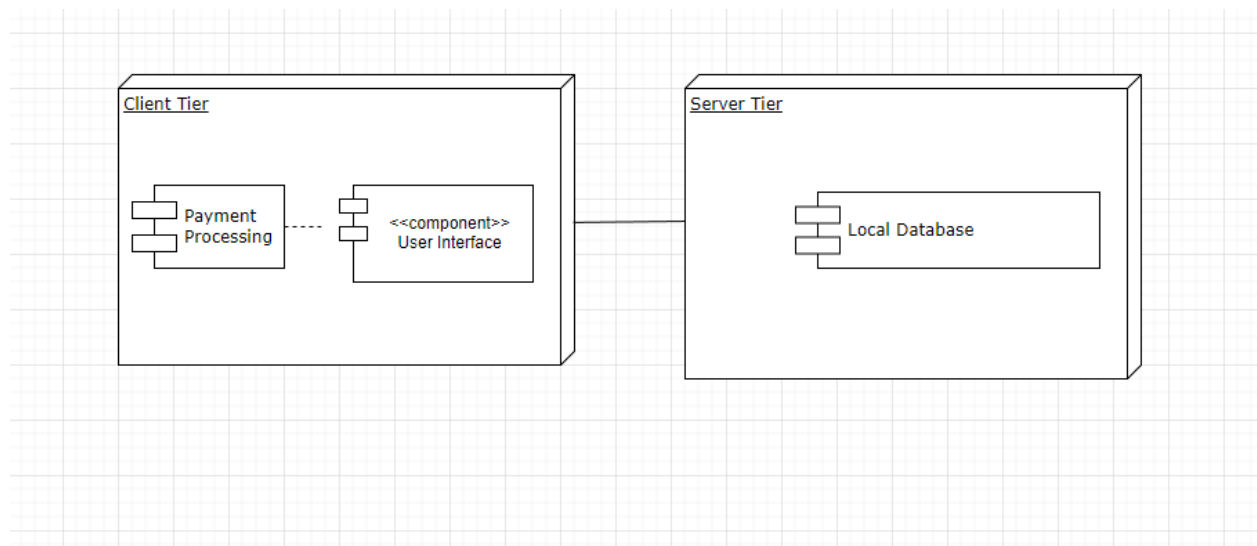
ADD Step 5: Instantiate Architectural Elements, Allocate Responsibilities, and Define Interfaces

Design Decision and Location	Rationale
For server tier use MySQL Workbench	Knowledge can be leveraged from the team. Use of a relational database is a good fit for a grocery store, since all items will need to include all specifications within the table.
Use Java for backend	Including the use of Java Swing Framework to handle UI elements, Java will be used to handle backend for the application to communicate with Workbench.

ADD Step 6: Sketch of Design View



Two-Tier Deployment (Client-Server)



Component Name	Responsibility
Local Database	Using mysql to handle item barcodes, pricing and users information
Payment Processing	Responsible for verifying clients payment, via credit or debit
User Interface	Displays users prompts, allowing them to pay in differe

Step 7: Perform Analysis of Current Design and Review Iteration

Not Addressed	Partially Addressed	Completely Addressed	Design Decisions Made During the Iteration
		UC-1	Using Java, UPC barcode scanning.
		UC-2	All transactions are handled with Java and MySQL.
		QA-1	Java and MySQL will allow platform changes.
		CON-1	MySQL Workbench to handle products.

		CON-2	Cash Register does not depend on any external application.
--	--	-------	--