SOFE 3200U

Architectural Design Process (Iteration 2)

Cash Register

Activity : Software Quality Attributes
Professor: Dr. Ramiro

2022/11/27

Name	Student ID	Signature
Onosen Aziegbe	100741943	O.A
Lauryne Zachary	100797920	L.Z
Hishan Sayanthan	100786649	H.S

Iteration 2 : Identifying structures to support primary functionalities

Step 2: Establishing Iteration Goal By Selecting Drivers

Primary use cases supporting functionality
Selected Drivers:

- UC-1 Scanning Items
- UC-2 Payment

Step 3: Choose one or more elements of the system to refine

The cash register elements will need to be refined using our defined reference architecture, the Rich Client Architecture following analysis from iteration 1 .

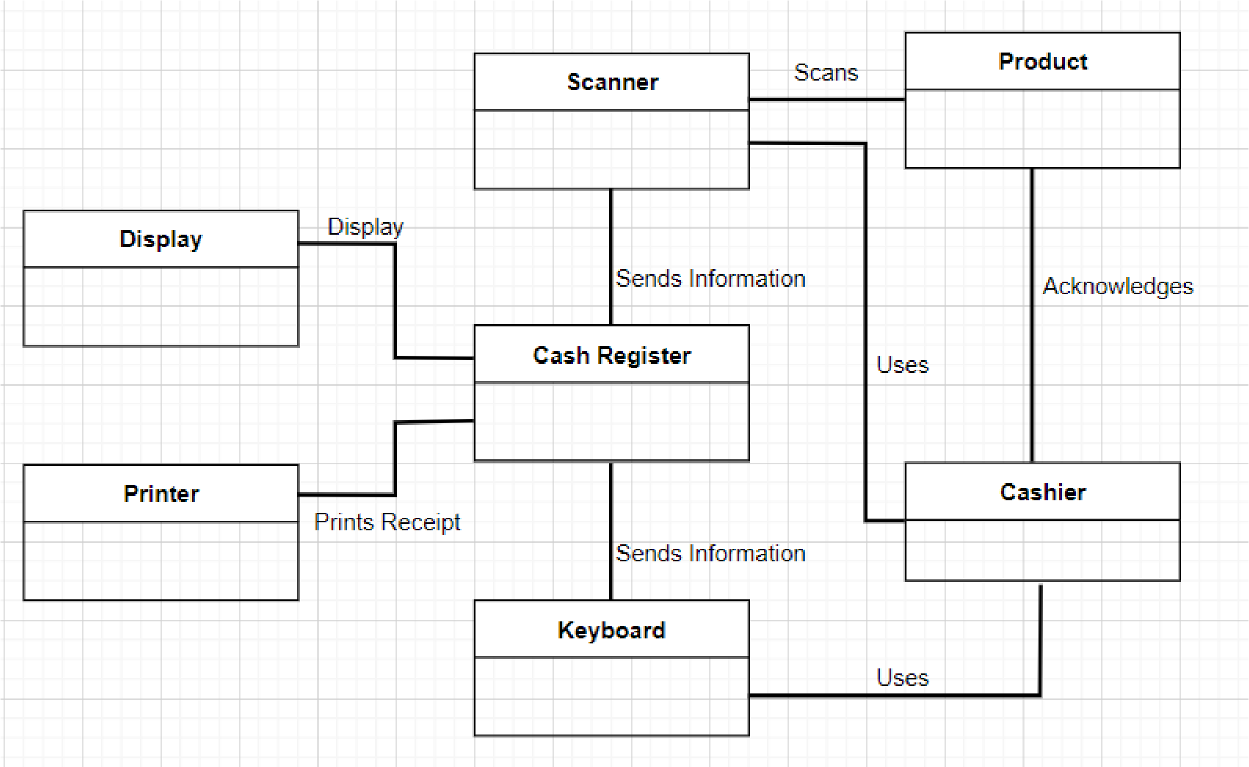
Step 4: Choose one or more design concepts that satisfy the selected drivers

Design decisions and location	**Rationale**
Create a standard UML domain model for the cash register application	The UML will show all the relationships between our key entities and use cases.
Domain objects mapped out to functional requirements	Encapsulating distinct functional application elements into a self-contained building block
Decompose objects into their respective components	Module specialization for layer categorizing and to support functionality
Build the user interface for the client side using a django framework	Using this framework will help optimize the system performance and address the portability quality attributes. (QA-1)

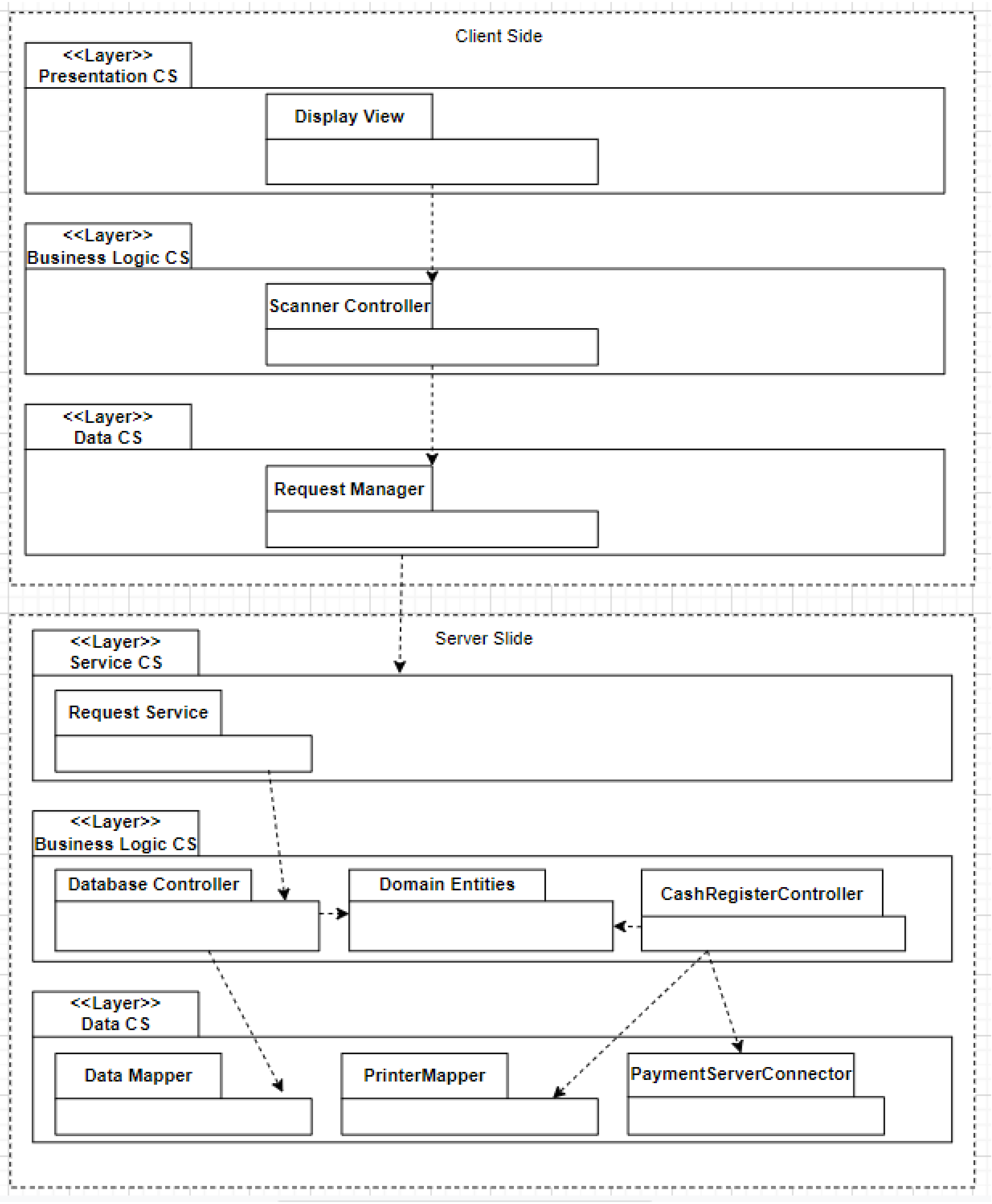
Step 5: Instantiate Architectural Elements, Allocate Responsibilities and Design Interface

Design decisions and location	**Rationale**
Create a *initial* UML domain model for the cash register application	Initial UML model includes the entities that participate in the primary use cases (UC-1 & UC-2)
Use case to domain object mapping	Analysis of objects for use case mapping
Identify layer-specific modules with an explicit interface	Decompose the domain objects to ensure that all key functionalities are met by the modules
Associate django framework with various modules in the data layer	Using the Django ORM mapping to encapsulate the data layer modules.

Step 6: Sketch Views and Record Design Decisions

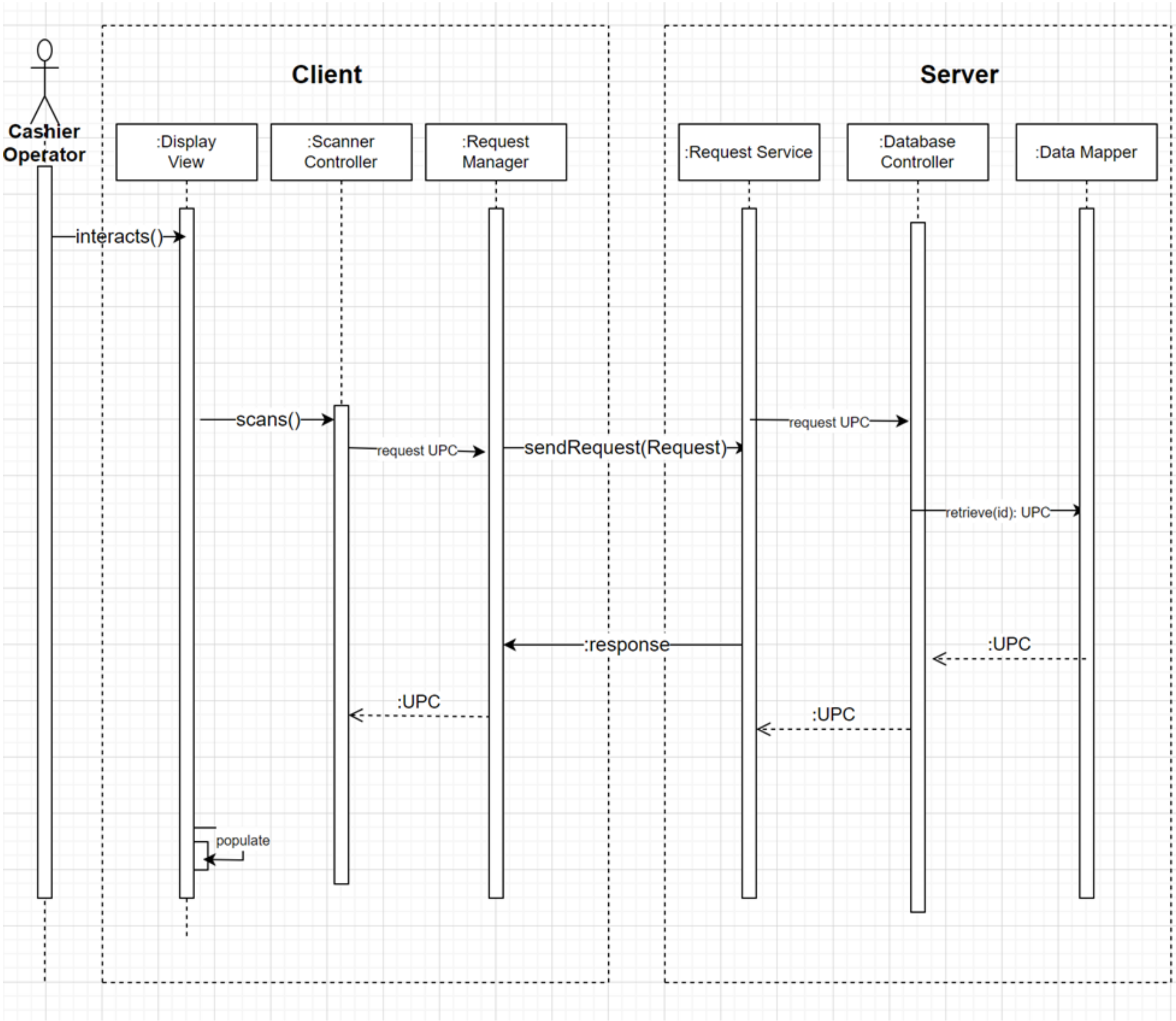


Domain Model



Modules that support the primary use case

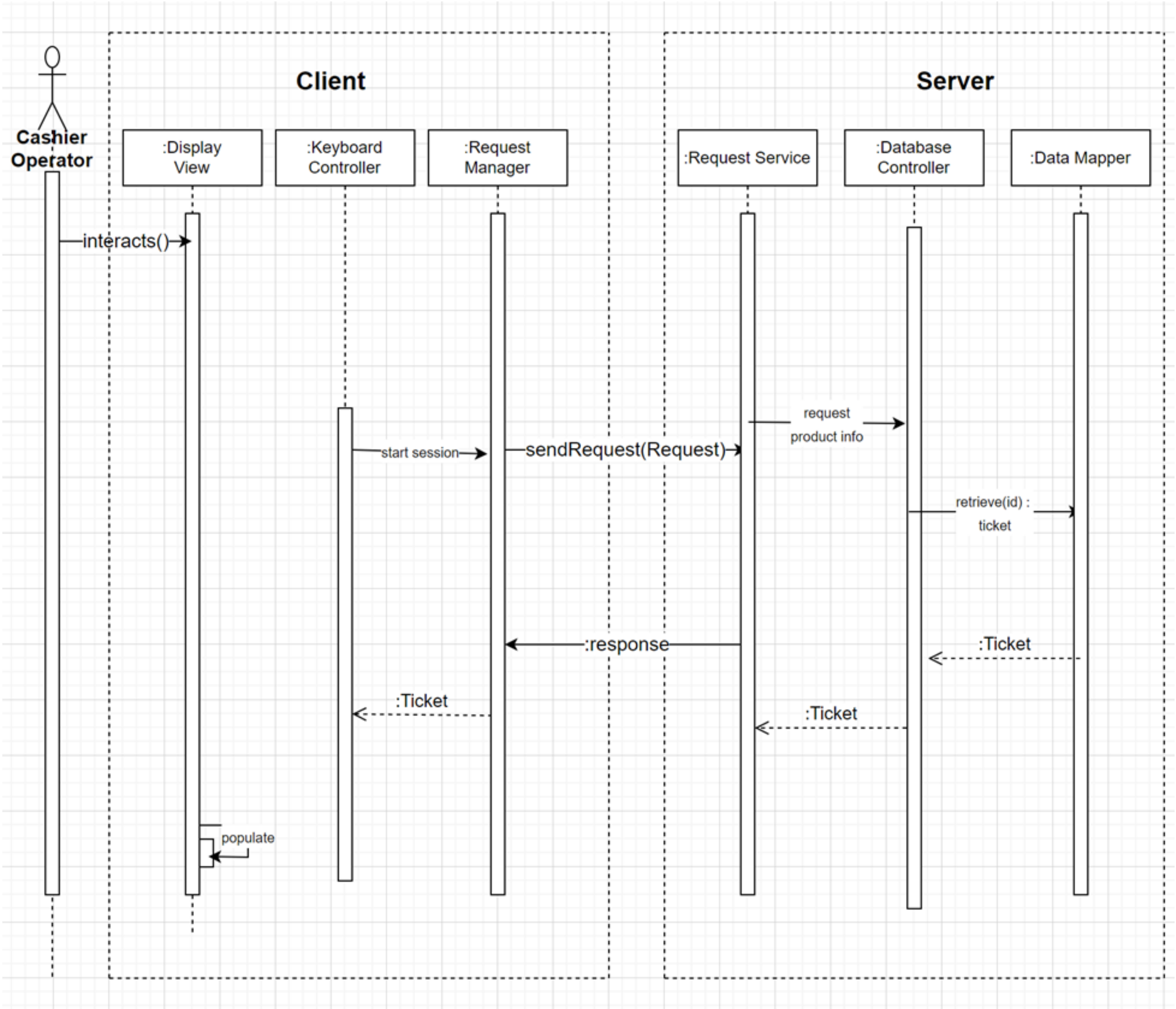
Element	**Responsibility**
Display View	Displays the transaction process and updates when it happens.
Scanner Controller	Responsible for providing the necessary information to the presentation layer
Request Manager	Responsible for communication with the server-side logic
Request Service	Provide a facade that receives requests from the clients.
Database Controller	Contains business logic related to the database information
Domain Entities	Contain entities from the domain model (server side).
CashRegisterController	Contains business logic related to the management of events
DataMapper	Responsible for persistence operations related to the database
PrinterMapper	Responsible for persistence operations related to printing receipt
PaymentServiceConnector	Responsible for communication with 3rd party payment servers for credit/debit payments



UC-1 : Scanning Items

Method Name	**Description**
Element : Scanner Controller	
Requests UPC	**Requests and returns stored product data from its UPC**
Element : Request Manager	
sendRequest(Request)	**Queries server for stored information on database**

Element: Request Service	
Response	**Receives the request**
Element : Database Controller	
Request UPC	**Requests and returns stored product data from its UPC**
Element: Data Mapper	
retrieve(id) :UPC	**Retrieve the product information from indicated barcode**



UC-2 : Payment

Method Name	**Description**
Element : KeyboardController	
Start session	**Opens up the search and retrieve process to secure payment**
Element : Request Manager	
Ticket	**Prints Ticket to the display**
Element: Request Service	
Response	**Receives the request**
Element : Database Controller	
Request product info	**Request to data mapper and returns ticket indicated sale time, items and payment confirmation**
Element: Data Mapper	
retrieve(id) :Ticket	**Retrieve the product information from indicated barcode**

Step 7: Perform Analysis of current design and review iteration

Not addressed	**Partially addressed**	**Completely Addressed**	**Design decisions made during the iteration**
		UC-1	Modules across the layers and Preliminary interfaces to support this use case has been identified.
		UC-2	Modules across the layers and Preliminary interfaces to support this use case has been identified.

	QA-1		No relevant decisions have been made
	CON-1		Modules responsible for collecting storage has been identified.
	CON-2		The framework that support the associated use case(UC-1 & UC-2) have been identified.