

TENDERS & QUOTATIONS MANAGEMENT SYSTEM (TQMS)

DESIGN_DOCUMENT

CS20BTECH11044 – SANDEEP L

ES20BTECH11025 – SAI SIVA ROHITH TIRUMALASETTI

CS20BTECH11046 – SHANTANU PANDEY

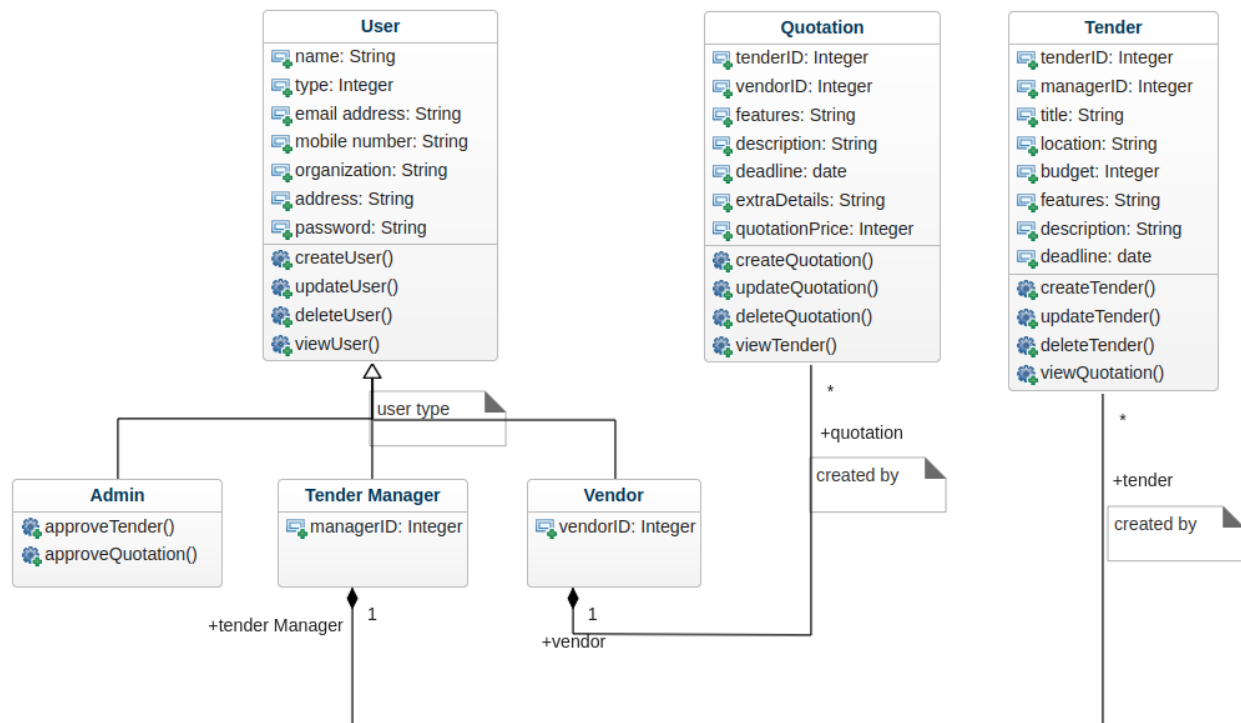
ES20BTECH11005 – ASAMPALLI PRUTHVI RAJ

CS20BTECH11008 – BATHINI ASHWITHA

MODULES

USER	Management of user profiles, authentication, and authorization.
TENDER	Create, edit, and delete tenders requests.
QUOTATION	Create, edit, and delete quotations in response to tender requests and approve/disapprove quotations.

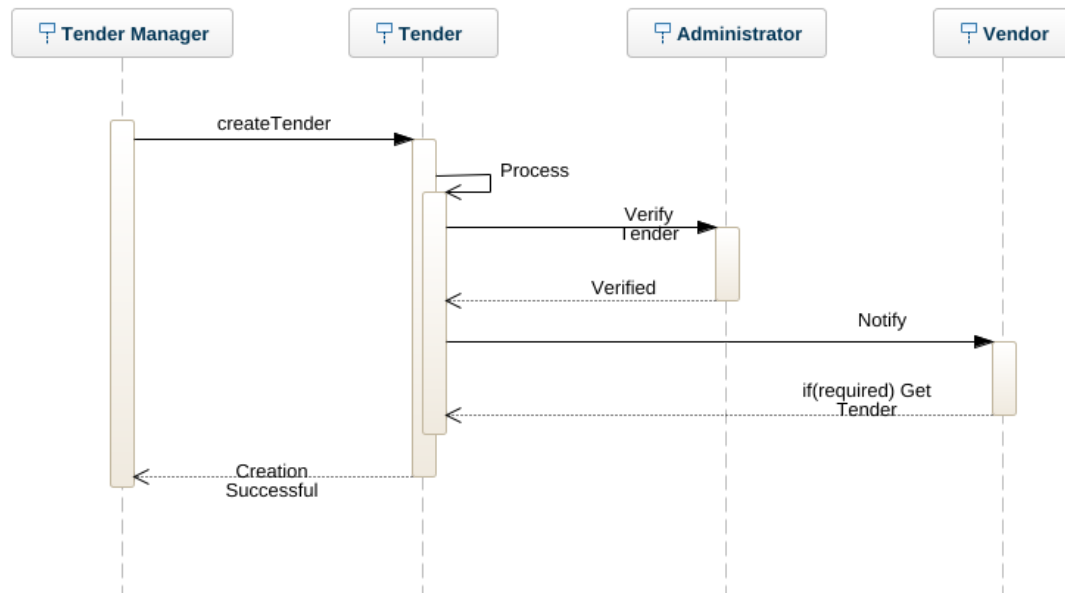
CLASS DIAGRAM



SEQUENCE/ACTIVITY DIAGRAMS

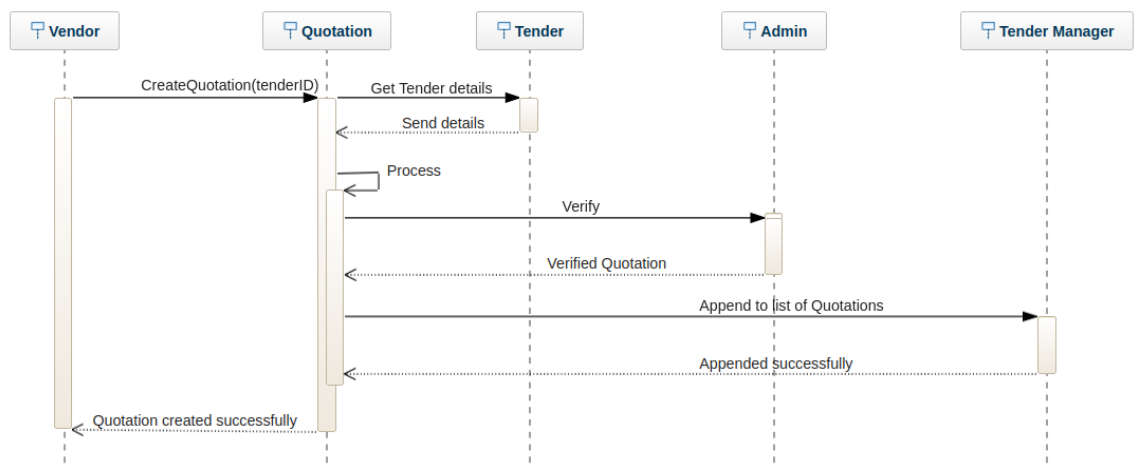
1. Creating Tender

Participating Objects: Tender Manager, tender, administrator, and vendor.



2. Creating Quotation

Participating Objects: Tender Manager, tender, quotation, administrator, and vendor.



DESIGN PATTERNS

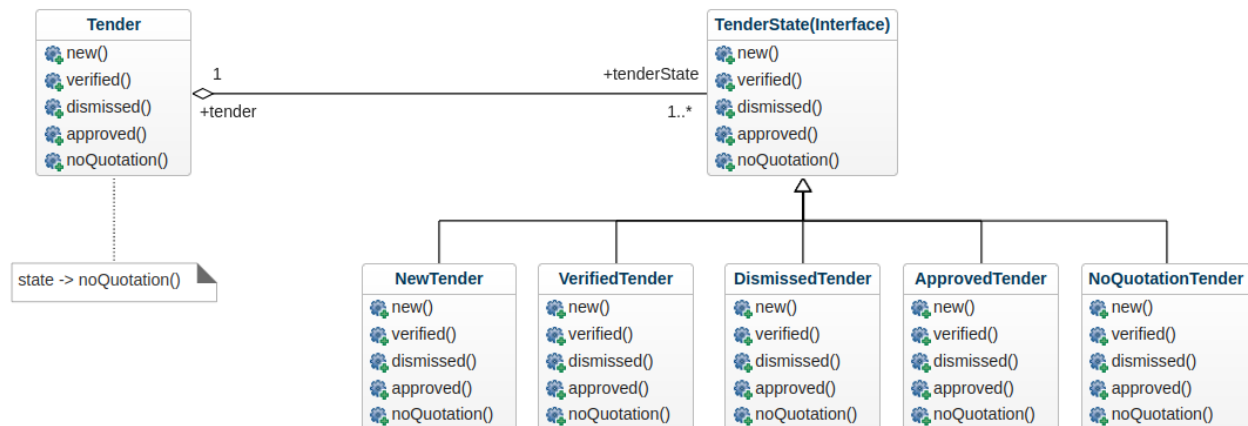
❖ **STATE PATTERN**

➤ TENDER

The Tender object can have multiple states and must change its behavior at the run-time depending on that state. We use State Pattern in which the following describes the structure:

- ➔ Context - Tender
- ➔ State - TenderState
- ➔ ConcreteState(s) - NewTender, VerifiedTender, DismissedTender, ApprovedTender, and NoQuotationTender.

The State pattern will be used to dynamically change the state of the Tender by the clients during the run-time.



➤ QUOTATION

A quotation can be in one of the following states depending on when the `viewQuotation` function is called:

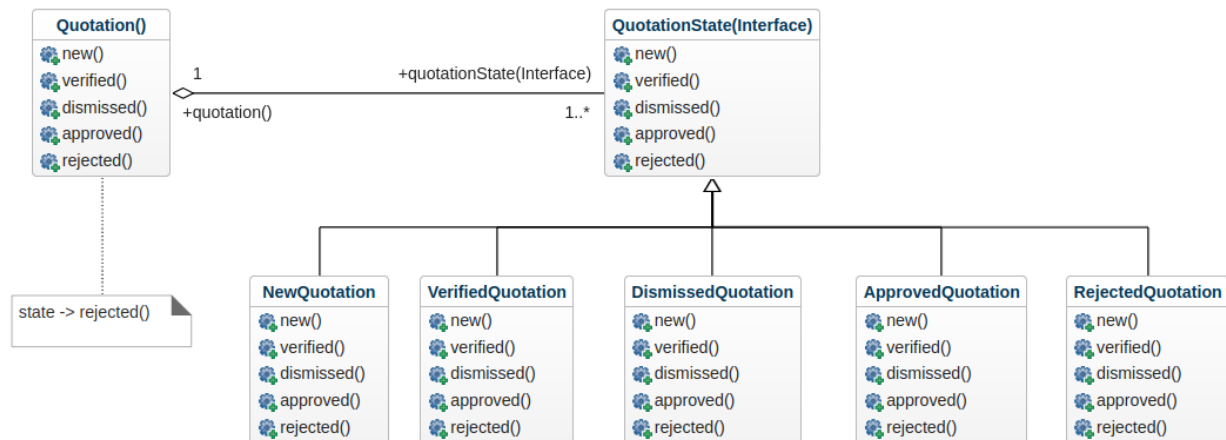
1. When a Quotation is created and submitted

against a tender.

2. Accepted/rejected by the Admin as a valid Quotation.

3. Accepted/rejected by the Tender Manager as the selected Quotation for a Tender.

So we can use the state pattern to represent the state of the quotation dynamically during run-time.



❖ **OBSERVER PATTERN**

When a ConcreteTenderManager creates a new Tender, we must notify all the vendors who have subscribed to that particular ConcreteTenderManager. Here, We have used the Observer pattern to leverage its property of broadcasting information to observers of a subject.

There can be many subjects {ConcreteTenderManager(s)}, and observers {concreteVendor(s)} can subscribe to as many subjects as they wish.

Below is the attached collaboration diagram after a new Tender is created by a ConcreteTenderManager(for uniformity, we tried to use the representations used in course slides). As per a rule, we do not allow observers(vendors) to set the state of the subject(Tender Managers).

