# Current Software Architecture

Online exam system is a web-based online application. This project would be very useful for educational institutes. Our system has a difference of user-friendly interface and reliable, than the others.

# Proposed Software Architecture

# Overview

During the system design modeling of online exam, we divided our system into subsystems. This provides us a strong coherence. Our subsystems are ; Account management system, which has account transactions. Student management system, has functions of student actor, Instructor management system which has instructor’s functions and database subsystem to control storage.

## System Decomposition

The decomposition shows the existence of the following subsystems:

* Account management subsystem
* Student management subsystem
* Instructor management subsystem
* Database subsystem

**Account management subsystem**

This subsystem managing user accounts. It offers perform for creating an account, updating an , approve and close an account for admin side. Admin is the only actor who has permission to access close and approve functions. Create and update functions are accessed by instructor and student. This subsystem uses login services of the student management and instructor management subsystems

**The operations provided by this subsystem are:-**

* Login ()
* Change password ().
* Create account ()
* Update account ()
* Close account ()
* Approve account ()

**Student management subsystem**

This subsystem is managing student actor’s function, offers student side to its functions after authenticate. Managing students access to taking exam, answering questions, seeing their results and displaying answers. Displaying and seeing results functions are useable after submitting exam.

**The operations provided by this subsystem are:-**

* Take exam ()
* Answer questions ().
* Submit exam ()
* List exam result ()
* View answers ()

**Instructor management subsystem**

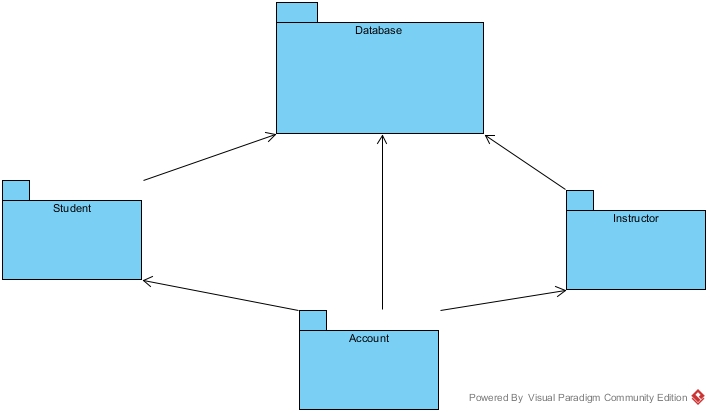
This subsystem is managing instructor actor’s function, offers instructor side to its functions after authenticate. The subsystem managing access of instructor to creating lecture, creating class of the lecture which is connected with database subsystem thus the students which are added class list are available to access exam, add or delete student, also creating and editing exam, adding question, update and delete options.

**The operations provided by this subsystem are:-**

* Create new lecture ()
* Create new class ().
* Add student ()
* Delete student ()
* List class ()
* Create exam ()
* Add question ()
* Delete question()
* Update exam ()
* Remove exam()
* Remove Lecture()

. **Database subsystem**

This subsystem will be implemented by relational database management system used to store the persistent data. All subsystems are related and having service with this subsystem



## Persistant Data Management

Our system will use the SQLSerever database engine to store data. This will allow the database to be easily integrated with and accessed by the rest of the system. The database will retain user information for functions such as login for user, show exams that are created by instructors for students to see etc. Our database structure is seen below with entity fields relations etc.

